LQ2021@CIKM2021

Nov. 1, 2021

# Mixture Proportion Estimation in Weakly Supervised Learning


Masashi Sugiyama

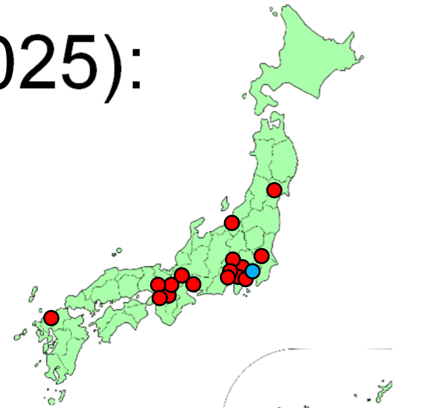
RIKEN Center for Advanced Intelligence Project

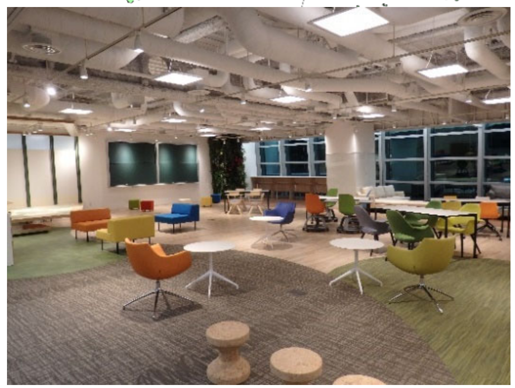
The University of Tokyo

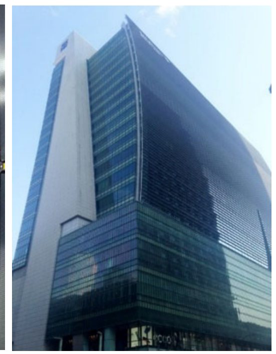
http://www.ms.k.u-tokyo.ac.jp/sugi/

# RIKEN Center for Advanced Intelligence Project (AIP)

- 10-year national project in Japan (2016-2025):
  - Develop next-generation AI technology (learning and optimization theory, etc.)
  - Accelerate scientific research (material, cancer, stem cells, genomics, etc.)
  - Solve socially critical problems (natural disaster, elderly healthcare, etc.)
  - Study of ethical, legal and social issues of AI (ethical guideline, privacy protection, etc.)
  - Human resource development (150+ researchers, 200+ students, 150+ interns, 300+ visiting scientists, 40+ industry projects)



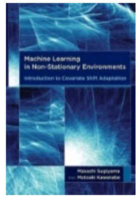




# My Research Interests

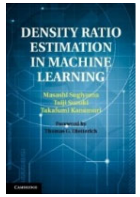
- **Transfer learning**:
  - Adaptive importance weighting

Sugiyama & Kawanabe, Machine Learning in Non-Stationary Environments, MIT Press, 2012



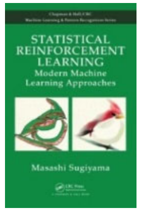
- **Density ratio estimation**:
  - Versatile statistical tool, where GAN is a special case.

Sugiyama, Suzuki & Kanamori, Density Ratio Estimation in Machine Learning, Cambridge University Press, 2012



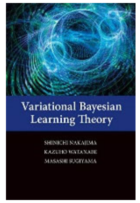

- **Reinforcement learning**:
  - Sample reuse

Sugiyama, Statistical Reinforcement Learning, Chapman and Hall/CRC, 2015




- **Variational Bayes**:
  - Implicit regularization

Nakajima, Watanabe & Sugiyama, Variational Bayesian Learning Theory, Cambridge University Press, 2019




- **Weakly supervised learning**:
  - Empirical risk minimization approach

Sugiyama, Bao, Ishida, Lu, Sakai & Niu, Machine Learning from Weak Supervision, MIT Press, in Press.

Coming soon

- **Noise-robust learning**:
  - Going beyond robust statistics and regularization

In the near future…

# Today's Topic: Mixture Proportion Estimation

- Goal: Find a mixture proportion of unknown probability distributions.
  - From some data, find $\theta_1, \ldots, \theta_c$ such that

$$p_0 = \sum_{y=1}^{c} \theta_y p_y \qquad \sum_{y=1}^{c} \theta_y = 1 \qquad \theta_1, \ldots, \theta_c \geq 0$$

$p_0, p_1, \ldots, p_c$ : Unknown probability distributions

- Various applications in machine learning:
  - Class-prior shift adaptation: Importance weight estimation
  - Positive-unlabeled classification: Class-prior estimation
  - Noisy label classification: Noise transition estimation

# Contents





# Semi-Supervised Classification with Class-Prior Shift



- Given: Labeled training data and unlabeled test data:

$$\{(\boldsymbol{x}_i^{\mathrm{tr}}, y_i^{\mathrm{tr}})\}_{i=1}^{n_{\mathrm{tr}}} \overset{\mathrm{i.i.d.}}{\sim} p_{\mathrm{tr}}(\boldsymbol{x}, y)$$

$$\{\boldsymbol{x}_j^{\mathrm{te}}\}_{j=1}^{n_{\mathrm{te}}} \overset{\mathrm{i.i.d.}}{\sim} p_{\mathrm{te}}(\boldsymbol{x})$$

$\boldsymbol{x} \in \mathcal{X} \subset \mathbb{R}^d$ : Input pattern

$y \in \mathcal{Y} = \{1, \ldots, c\}$ : Class label

- Goal: Train a classifier $y = f(\boldsymbol{x})$ that works well in the test domain.

$$\min_f R(f) \qquad R(f) = \mathbb{E}_{p_{\mathrm{te}}(\boldsymbol{x},y)}[\ell(f(\boldsymbol{x}), y)]$$

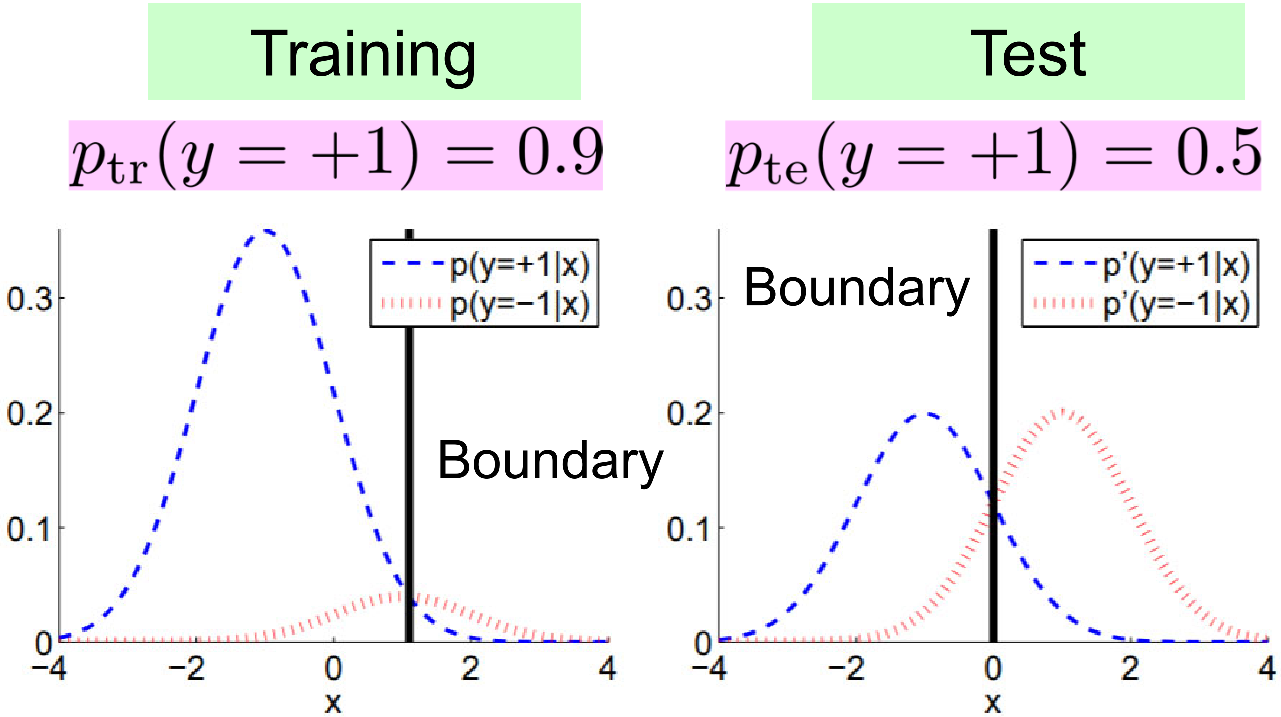
$\ell$ : loss function

- Challenge: Overcome the class-prior shift!

$$p_{\mathrm{tr}}(y) \neq p_{\mathrm{te}}(y) \qquad p_{\mathrm{tr}}(\boldsymbol{x}|y) = p_{\mathrm{te}}(\boldsymbol{x}|y) = p(\boldsymbol{x}|y)$$

# Illustration of Class-Prior Shift






- Class-prior shift changes the optimal boundary.
- Adaptation is needed!

# Contents





# Empirical Risk Minimization (ERM)



$$\operatorname*{argmin}_{f\in\mathcal{F}}\left[\sum_{i=1}^{n_{\mathrm{tr}}}\ell(f(\boldsymbol{x}_i^{\mathrm{tr}}),y_i^{\mathrm{tr}})\right] \qquad \{(\boldsymbol{x}_i^{\mathrm{tr}},y_i^{\mathrm{tr}})\}_{i=1}^{n_{\mathrm{tr}}}\overset{\mathrm{i.i.d.}}{\sim}p_{\mathrm{tr}}(\boldsymbol{x},y)$$

- Generally, ERM is consistent:
  - Learned function converges to the optimal solution when $n_{\mathrm{tr}}\to\infty$.
- However, class-prior shift makes ERM inconsistent:

$$\operatorname*{argmin}_{f\in\mathcal{F}}\left[\frac{1}{n_{\mathrm{tr}}}\sum_{i=1}^{n_{\mathrm{tr}}}\ell(f(\boldsymbol{x}_i^{\mathrm{tr}}),y_i^{\mathrm{tr}})\right]$$

$$\overset{n_{\mathrm{tr}}\to\infty}{\longrightarrow}\operatorname*{argmin}_{f\in\mathcal{F}}\Big[\overbrace{\mathbb{E}_{p(\boldsymbol{x}|y)p_{\mathrm{tr}}(y)}[\ell(f(\boldsymbol{x}),y)]}^{\neq R(f)}\Big]$$

$$R(f)=\mathbb{E}_{p(\boldsymbol{x}|y)p_{\mathrm{te}}(y)}[\ell(f(\boldsymbol{x}),y)]$$

$$p_{\mathrm{tr}}(y)\neq p_{\mathrm{te}}(y)$$

# Importance-Weighted ERM (IWERM) 

$$\operatorname*{argmin}_{f\in\mathcal{F}}\left[\sum_{i=1}^{n_{\mathrm{tr}}}\underbrace{\frac{p_{\mathrm{te}}(y_i^{\mathrm{tr}})}{p_{\mathrm{tr}}(y_i^{\mathrm{tr}})}}_{\text{Importance}}\ell(f(\boldsymbol{x}_i^{\mathrm{tr}}),y_i^{\mathrm{tr}})\right]$$

- IWERM is consistent even under class-prior shift.

$$\operatorname*{argmin}_{f\in\mathcal{F}}\left[\frac{1}{n_{\mathrm{tr}}}\sum_{i=1}^{n_{\mathrm{tr}}}\frac{p_{\mathrm{te}}(y_i^{\mathrm{tr}})}{p_{\mathrm{tr}}(y_i^{\mathrm{tr}})}\ell(f(\boldsymbol{x}_i^{\mathrm{tr}}),y_i^{\mathrm{tr}})\right]$$

$$\overset{n_{\mathrm{tr}}\to\infty}{\longrightarrow}\operatorname*{argmin}_{f\in\mathcal{F}}\left[\mathbb{E}_{p(\boldsymbol{x}|y)p_{\mathrm{tr}}(y)}\left[\frac{p_{\mathrm{te}}(y)}{p_{\mathrm{tr}}(y)}\ell(f(\boldsymbol{x}),y)\right]\right]$$

$$=\operatorname*{argmin}_{f\in\mathcal{F}}\left[\underbrace{\mathbb{E}_{p(\boldsymbol{x}|y)p_{\mathrm{te}}(y)}[\ell(f(\boldsymbol{x}),y)]}_{=R(f)}\right]$$

- How can we know the importance weight?

# Class-Prior Estimation by the EM Algorithm

Saerens et al. (NeCo2001)

1. Obtain a training class-posterior estimator $\hat{p}_{\mathrm{tr}}(y|\boldsymbol{x})$ from $\{(\boldsymbol{x}_i^{\mathrm{tr}}, y_i^{\mathrm{tr}})\}_{i=1}^{n_{\mathrm{tr}}} \overset{\mathrm{i.i.d.}}{\sim} p_{\mathrm{tr}}(\boldsymbol{x}, y)$.
2. Estimate the training class-prior by $\hat{p}_{\mathrm{tr}}(y) \propto n_y$.

   $n_y$ : Number of training samples in class $y$
3. Set $\hat{p}_{\mathrm{te}}(y|\boldsymbol{x}) = \hat{p}_{\mathrm{tr}}(y|\boldsymbol{x})$ and $\hat{p}_{\mathrm{te}}(y) = \hat{p}_{\mathrm{tr}}(y)$ .
4. Repeat until convergence:
   i. Update the test class-posterior as $\hat{p}_{\mathrm{te}}(y|\boldsymbol{x}) \propto \dfrac{\hat{p}_{\mathrm{te}}(y)}{\hat{p}_{\mathrm{tr}}(y)} \hat{p}_{\mathrm{tr}}(y|\boldsymbol{x})$.
   ii. Update the test class-prior as $\hat{p}_{\mathrm{te}}(y) = \dfrac{1}{n_{\mathrm{te}}} \displaystyle\sum_{j=1}^{n_{\mathrm{te}}} \hat{p}_{\mathrm{te}}(y|\boldsymbol{x}_j^{\mathrm{te}})$ .

$\{\boldsymbol{x}_j^{\mathrm{te}}\}_{j=1}^{n_{\mathrm{te}}} \overset{\mathrm{i.i.d.}}{\sim} p_{\mathrm{te}}(\boldsymbol{x})$

- Can we avoid using $\hat{p}_{\mathrm{tr}}(y|\boldsymbol{x})$?

# Contents





# EM Method as Distribution Matching under KL Divergence

du Plessis et al. (NN2014)

- Let $q(\boldsymbol{x}) = \sum_{y=1}^{c} \theta_y p_{\mathrm{tr}}(\boldsymbol{x}|y)$ . $\theta_1, \ldots, \theta_c \geq 0$ $\sum_{y=1}^{c} \theta_y = 1$
- Fit $q(\boldsymbol{x})$ to $p_{\mathrm{te}}(\boldsymbol{x})$ under KL divergence:

$$\underset{\theta_1,\ldots,\theta_c}{\operatorname{argmin}} \mathrm{KL}[p_{\mathrm{te}}(\boldsymbol{x}) \| q(\boldsymbol{x})] = \underset{\theta_1,\ldots,\theta_c}{\operatorname{argmin}} \int p_{\mathrm{te}}(\boldsymbol{x}) \log \frac{p_{\mathrm{te}}(\boldsymbol{x})}{q(\boldsymbol{x})} \mathrm{d}\boldsymbol{x}$$

$$\approx \underset{\theta_1,\ldots,\theta_c}{\operatorname{argmin}} \frac{1}{n_{\mathrm{te}}} \sum_{j=1}^{n_{\mathrm{te}}} \log \frac{p_{\mathrm{te}}(\boldsymbol{x}_j^{\mathrm{te}})}{q(\boldsymbol{x}_j^{\mathrm{te}})} \qquad \{\boldsymbol{x}_j^{\mathrm{te}}\}_{j=1}^{n_{\mathrm{te}}} \overset{\mathrm{i.i.d.}}{\sim} p_{\mathrm{te}}(\boldsymbol{x})$$

- Fixed-point iteration to solve the KKT condition recovers the EM approach!
- Without estimating $\hat{p}_{\mathrm{tr}}(y|\boldsymbol{x})$, can we directly minimize the KL divergence?

# Direct KL-Divergence Approximation by Density Ratio Estimation



Keziou (2003), Nguyen et al. (NIPS2007), Sugiyama et al. (NIPS2007)

- Identity (from Fenchel's inequality):

$$\mathrm{KL}[p_{\mathrm{te}}(\boldsymbol{x}) \| q(\boldsymbol{x})] = \sup_{s} \left\{ -\int p_{\mathrm{te}}(\boldsymbol{x}) s(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} + \int q(\boldsymbol{x}) \log s(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} \right\} + 1$$

  - Maximizer is $s(\boldsymbol{x}) = q(\boldsymbol{x}) / p_{\mathrm{te}}(\boldsymbol{x})$.

$$q(\boldsymbol{x}) = \sum_{y=1}^{c} \theta_y p_{\mathrm{tr}}(\boldsymbol{x}|y)$$

- Empirical approximation:

$$\widehat{\mathrm{KL}}[p_{\mathrm{te}}(\boldsymbol{x}) \| q(\boldsymbol{x})] = \max_{s} \left\{ -\frac{1}{n_{\mathrm{te}}} \sum_{j=1}^{n_{\mathrm{te}}} s(\boldsymbol{x}_j^{\mathrm{te}}) + \sum_{y=1}^{c} \frac{\theta_y}{n_y} \sum_{i: y_i = y} \log s(\boldsymbol{x}_i^{\mathrm{tr}}) \right\} + 1$$

  - Maximization corresponds to estimating density ratio $s(\boldsymbol{x})$.

- Then we can directly estimate the test class-prior as

$$\operatorname*{argmin}_{\theta_1, \ldots, \theta_c} \mathrm{KL}[p_{\mathrm{te}}(\boldsymbol{x}) \| q(\boldsymbol{x})] \approx \operatorname*{argmin}_{\theta_1, \ldots, \theta_c} \widehat{\mathrm{KL}}[p_{\mathrm{te}}(\boldsymbol{x}) \| q(\boldsymbol{x})]$$

# Distribution Matching under the $f$-Divergence

du Plessis et al. (NN2014)

- We don't have to stick to the KL divergence.
  - We can use any divergence such as the $f$-divergence:

For convex $f$ such that $f(1)=0$,

Ali & Slivey (1966), Csiszár (1967)

$$\mathrm{Div}_f[p_{\mathrm{te}}(\boldsymbol{x})\|q(\boldsymbol{x})] = \int p_{\mathrm{te}}(\boldsymbol{x}) f\left(\frac{q(\boldsymbol{x})}{p_{\mathrm{te}}(\boldsymbol{x})}\right) \mathrm{d}\boldsymbol{x}$$

$$q(\boldsymbol{x}) = \sum_{y=1}^{c} \theta_y p_{\mathrm{tr}}(\boldsymbol{x}|y)$$

- Directly estimate the $f$-divergence from data:

$$\widehat{\mathrm{Div}}_f[p_{\mathrm{te}}(\boldsymbol{x})\|q(\boldsymbol{x})] \quad \{\boldsymbol{x}_i^{\mathrm{tr}}\}_{i=1}^{n_{\mathrm{tr}}} \overset{\mathrm{i.i.d.}}{\sim} p_{\mathrm{tr}}(\boldsymbol{x}) \quad \{\boldsymbol{x}_j^{\mathrm{te}}\}_{j=1}^{n_{\mathrm{te}}} \overset{\mathrm{i.i.d.}}{\sim} p_{\mathrm{te}}(\boldsymbol{x})$$

- Estimate the test class-prior as

$$\underset{\theta_1,\ldots,\theta_c}{\mathrm{argmin}}\, \mathrm{Div}_f[p_{\mathrm{te}}(\boldsymbol{x})\|q(\boldsymbol{x})] \approx \underset{\theta_1,\ldots,\theta_c}{\mathrm{argmin}}\, \widehat{\mathrm{Div}}_f[p_{\mathrm{te}}(\boldsymbol{x})\|q(\boldsymbol{x})]$$

- How do we estimate the $f$-divergence from data?

# Direct $f$-Divergence Approximation by Density Ratio Estimation

Keziou (2003), Nguyen et al. (NIPS2007), Sugiyama et al. (AISM2012)

- Identity (from Fenchel's inequality):

$$\mathrm{Div}_f[p_{\mathrm{te}}(\boldsymbol{x})\|q(\boldsymbol{x})]$$

$$= -\inf_s \left\{ \int p_{\mathrm{te}}(\boldsymbol{x})\Big(\partial f\big(s(\boldsymbol{x})\big)s(\boldsymbol{x}) - f\big(s(\boldsymbol{x})\big)\Big)\mathrm{d}\boldsymbol{x} - \int q(\boldsymbol{x})\partial f\big(s(\boldsymbol{x})\big)\mathrm{d}\boldsymbol{x} \right\}$$

  - Equality holds when $s(\boldsymbol{x}) = q(\boldsymbol{x})/p_{\mathrm{te}}(\boldsymbol{x})$ .

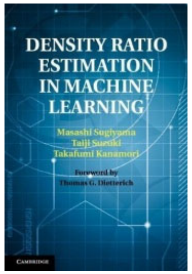
- Empirical approximation:

$$\widehat{\mathrm{Div}}_f[p_{\mathrm{te}}(\boldsymbol{x})\|q(\boldsymbol{x})]$$

$$= -\min_s \frac{1}{n_{\mathrm{te}}} \sum_{j=1}^{n_{\mathrm{te}}} \left( \partial f\big(s(\boldsymbol{x}_j^{\mathrm{te}})\big)s(\boldsymbol{x}_j^{\mathrm{te}}) - f\big(s(\boldsymbol{x}_j^{\mathrm{te}})\big) \right) - \sum_{y=1}^{c} \frac{\theta_y}{n_y} \sum_{i:y_i=y} f'\big(s(\boldsymbol{x}_i^{\mathrm{tr}})\big)$$

  - Minimization corresponds to density ratio matching under the Bregman divergence.

Sugiyama, Suzuki & Kanamori, Density Ratio Estimation in Machine Learning (Cambridge University Press, 2012)

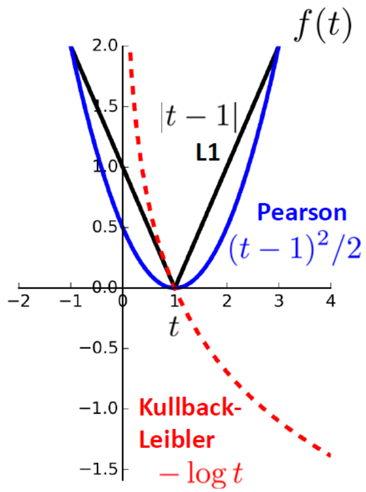


# Various Choices of Function $f$

For convex $f$ such that $f(1) = 0$,

$$\mathrm{Div}_f[p_{\mathrm{te}}(\boldsymbol{x}) \| q(\boldsymbol{x})] = \int p_{\mathrm{te}}(\boldsymbol{x}) f\left(\frac{q(\boldsymbol{x})}{p_{\mathrm{te}}(\boldsymbol{x})}\right) \mathrm{d}\boldsymbol{x}$$

- Kullback-Leibler (KL) divergence: $f(t) = -\log t$
  - Popular choice, but sensitive to outliers.
  - Optimization is convex if $s(\boldsymbol{x})$ is a linear model.
- Pearson (PE) divergence: $f(t) = (t-1)^2/2$
  - Robust to outliers.
  - Optimization is analytic if $s(\boldsymbol{x})$ is a linear model.
- Power divergence: $f(t) = (t^\alpha - 1)t/\alpha$ for $\alpha > 0$
  - Generalization of KL ($\alpha \to 0$) and PE ($\alpha = 1$).
  - More robust for $\alpha > 1$, but optimization becomes non-convex.



# Contents





# Summary: Semi-Supervised Classification with Class-Prior Shift

- Importance-weighted empirical risk minimization.
  - Estimation of the test class-prior $p_{\mathrm{te}}(y)$ is needed.
- EM is seminal, but requires $\widehat{p}_{\mathrm{tr}}(y|\boldsymbol{x})$.
  - EM is KL-div minimization with fix-point iteration.
  - Can we directly minimize KL-div without $\widehat{p}_{\mathrm{tr}}(y|\boldsymbol{x})$?
- KL-div approximation with density ratio estimation.
  - Can we use another divergence?
- Various divergences/distances can be used.
  - $f$-div approximation by density ratio estimation.
  - $L_2$-distance approximation by density difference estimation.

Sugiyama et al. (NIPS2012, NeCo2013)

# Contents





# Positive-Unlabeled (PU) Classification



- **Given:** Positive and unlabeled samples

$$\{\boldsymbol{x}_i^{\mathrm{P}}\}_{i=1}^{n_{\mathrm{P}}} \overset{\mathrm{i.i.d.}}{\sim} p(\boldsymbol{x}|y=+1)$$

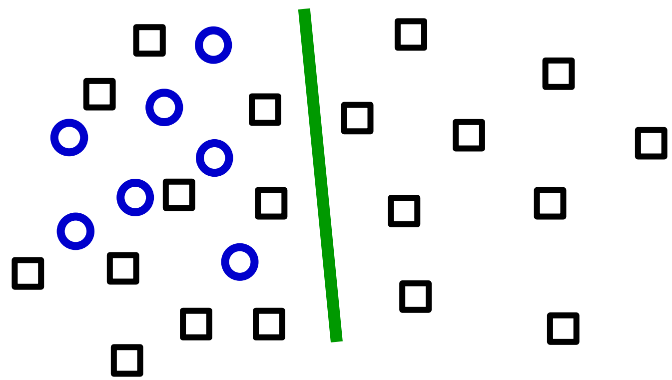
$$\{\boldsymbol{x}_j^{\mathrm{U}}\}_{j=1}^{n_{\mathrm{U}}} \overset{\mathrm{i.i.d.}}{\sim} p(\boldsymbol{x})$$

$\boldsymbol{x} \in \mathcal{X} \subset \mathbb{R}^d$ : Input pattern

$y \in \mathcal{Y} = \{+1, -1\}$ : Class label

  - No negative data

- **Goal:** Obtain a positive-negative (PN) classifier




- Example: Ad-click prediction
  - Clicked ad: User likes it → P
  - Unclicked ad: User dislikes it or User likes it but doesn't have time to click it → U (=P or N)

# Contents





# PN Risk Decomposition



- Risk of classifier $f$ :

$$R(f) = \mathbb{E}_{p(\boldsymbol{x},y)}\Big[\ell\Big(yf(\boldsymbol{x})\Big)\Big]$$

$\ell$ : loss function

$$= \underbrace{\pi\mathbb{E}_{p(\boldsymbol{x}|y=+1)}\Big[\ell\Big(f(\boldsymbol{x})\Big)\Big]}_{\text{Risk for P data}} + \underbrace{(1-\pi)\mathbb{E}_{p(\boldsymbol{x}|y=-1)}\Big[\ell\Big(-f(\boldsymbol{x})\Big)\Big]}_{\text{Risk for N data}}$$

$\pi = p(y=+1)$ : Class-prior probability
(for the moment, assume it is known)

- Since we do not have N data in the PU setting, the risk cannot be directly estimated.
  - How can we overcome this problem?

# PU Risk Estimation



du Plessis et al. (ICML2015)

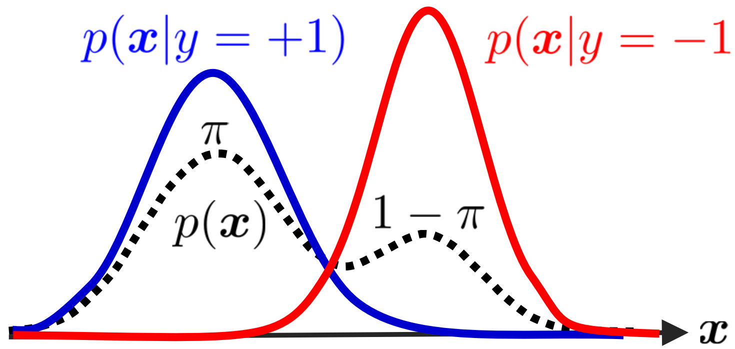
$$R(f) = \pi \mathbb{E}_{p(\boldsymbol{x}|y=+1)}\Big[\ell\Big(f(\boldsymbol{x})\Big)\Big] + (1-\pi)\mathbb{E}_{p(\boldsymbol{x}|y=-1)}\Big[\ell\Big(-f(\boldsymbol{x})\Big)\Big]$$

- U-density is a mixture of P- and N-densities:

$$p(\boldsymbol{x}) = \pi p(\boldsymbol{x}|y=+1) + (1-\pi)p(\boldsymbol{x}|y=-1)$$

- This allows us to eliminate the N-density as

$$R(f) = \pi \mathbb{E}_{p(\boldsymbol{x}|y=+1)}\Big[\ell\Big(f(\boldsymbol{x})\Big)\Big] + \mathbb{E}_{p(\boldsymbol{x})}\Big[\ell\Big(-f(\boldsymbol{x})\Big)\Big] - \pi \mathbb{E}_{p(\boldsymbol{x}|y=+1)}\Big[\ell\Big(-f(\boldsymbol{x})\Big)\Big]$$



# PU Empirical Risk Minimization



$$R(f) = \pi \mathbb{E}_{p(\boldsymbol{x}|y=+1)}\Big[\ell\Big(f(\boldsymbol{x})\Big)\Big] + \mathbb{E}_{p(\boldsymbol{x})}\Big[\ell\Big(-f(\boldsymbol{x})\Big)\Big] - \pi \mathbb{E}_{p(\boldsymbol{x}|y=+1)}\Big[\ell\Big(-f(\boldsymbol{x})\Big)\Big]$$

- Replacing expectations by sample averages gives an empirical risk:

$$\widehat{R}_{\mathrm{PU}}(f) = \frac{\pi}{n_{\mathrm{P}}} \sum_{i=1}^{n_{\mathrm{P}}} \ell\Big(f(\boldsymbol{x}_i^{\mathrm{P}})\Big) + \frac{1}{n_{\mathrm{U}}} \sum_{j=1}^{n_{\mathrm{U}}} \ell\Big(-f(\boldsymbol{x}_j^{\mathrm{U}})\Big) - \frac{\pi}{n_{\mathrm{P}}} \sum_{i=1}^{n_{\mathrm{P}}} \ell\Big(-f(\boldsymbol{x}_i^{\mathrm{P}})\Big)$$

$$\{\boldsymbol{x}_i^{\mathrm{P}}\}_{i=1}^{n_{\mathrm{P}}} \overset{\text{i.i.d.}}{\sim} p(\boldsymbol{x}|y=+1) \qquad \{\boldsymbol{x}_i^{\mathrm{U}}\}_{i=1}^{n_{\mathrm{U}}} \overset{\text{i.i.d.}}{\sim} p(\boldsymbol{x})$$

- Optimal convergence rate is attained: Niu et al. (NIPS2016)

$$R(\widehat{f}_{\mathrm{PU}}) - R(f^*) \leq C(\delta)\left(\frac{2\pi}{\sqrt{n_{\mathrm{P}}}} + \frac{1}{\sqrt{n_{\mathrm{U}}}}\right)$$

with probability $1-\delta$

$$\widehat{f}_{\mathrm{PU}} = \mathrm{argmin}_f \, \widehat{R}_{\mathrm{PU}}(f)$$

$$f^* = \mathrm{argmin}_f \, R(f)$$

$n_{\mathrm{P}}, n_{\mathrm{U}}$ : # of P, U samples

- But, in practice, $\pi = p(y=+1)$ is unknown!

# Class-Prior Estimation with Non-Traditional Classification

Elkan & Noto (KDD2008)

- Consider PU label $s \in \{0, 1\}$:
  - If $\boldsymbol{x}$ is P (or U), $s = 1$ (or $s = 0$ ).

$$\pi = \frac{p(s=1)}{p(s=1|y=+1)}$$

- Train a "non-traditional" classifier $\hat{p}(s|\boldsymbol{x})$ from PU data.
  - Usual supervised classification from $\{\boldsymbol{x}_i^{\mathrm{P}}\}_{i=1}^{n_{\mathrm{P}}}, \{\boldsymbol{x}_i^{\mathrm{U}}\}_{i=1}^{n_{\mathrm{U}}}$
    (Assume P is labeled from U when $s = 1$ .)

- Obtain $\hat{\pi}$ with

$\mathcal{P}$ : Set of validation P data

$$\hat{p}(s=1) = \frac{n_{\mathrm{P}}}{n_{\mathrm{P}} + n_{\mathrm{U}}} \qquad \hat{p}(s=1|y=+1) = \frac{1}{|\mathcal{P}|} \sum_{\boldsymbol{x} \in \mathcal{P}} \hat{p}(s=1|\boldsymbol{x})$$

- Can we avoid training a non-traditional classifier?

cf. Original paper solves PU classification by

$$\hat{p}(y=+1|\boldsymbol{x}) = \frac{\hat{p}(s=1|\boldsymbol{x})}{\hat{p}(s=1|y=+1)}$$

# Contents



# Non-Traditional Classification as Partial Distribution Matching

$$p(\boldsymbol{x}) = \pi p(\boldsymbol{x}|y=+1) + (1-\pi)p(\boldsymbol{x}|y=-1)$$

- PN classification: (Full) distribution matching

$$\min_{\theta\in[0,1]} \mathrm{Div}[p(\boldsymbol{x})\|q(\boldsymbol{x})] \qquad q(\boldsymbol{x}) = \theta p(\boldsymbol{x}|y=+1) + (1-\theta)p(\boldsymbol{x}|y=-1)$$

- PU classification: Partial distribution matching

$$\min_{\theta\in[0,1]} \mathrm{Div}[p(\boldsymbol{x})\|q'(\boldsymbol{x})] \qquad q'(\boldsymbol{x}) = \theta p(\boldsymbol{x}|y=+1)$$

du Plessis & Sugiyama (IEICE2014)

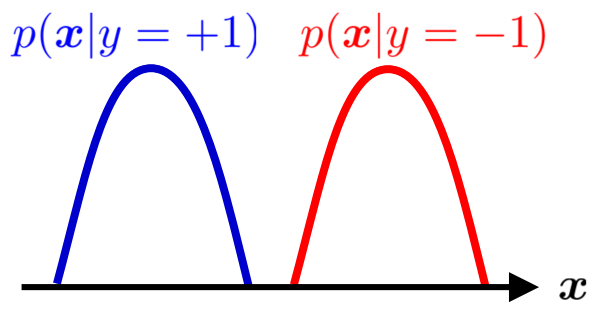
- Class-prior estimation by non-traditional classification can be interpreted as partial matching with Pearson divergence.

$$\frac{1}{2}\int p(\boldsymbol{x})\left(\frac{q'(\boldsymbol{x})}{p(\boldsymbol{x})} - 1\right)^2 \mathrm{d}\boldsymbol{x}$$

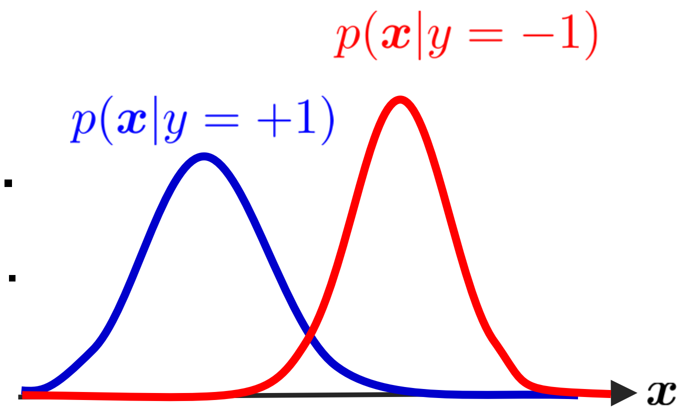
# Behaviors of Partial Matching

$$\min_{\theta\in[0,1]} \mathrm{Div}_f[p(\boldsymbol{x})\|q'(\boldsymbol{x})] \qquad q'(\boldsymbol{x}) = \theta p(\boldsymbol{x}|y=+1)$$

- If two classes have no overlap, naïve partial matching works.
  - Just fitting $p(\boldsymbol{x}|y=+1)$ is sufficient.




- If two classes are overlapped, partial matching generally over-estimates the true class-prior.
  - Tails of $p(\boldsymbol{x}|y=-1)$ affect the solution.



# Non-Identifiability of the Class-Prior



Blanchard et al. (JMLR2010)

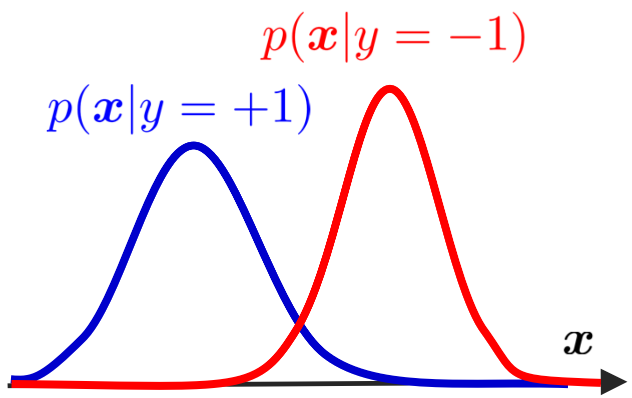
$$p(\boldsymbol{x}) = \pi p(\boldsymbol{x}|y=+1) + (1-\pi)\underbrace{p(\boldsymbol{x}|y=-1)}_{\text{Non-estimable}}$$

$$\{\boldsymbol{x}_i^{\mathrm{P}}\}_{i=1}^{n_{\mathrm{P}}} \overset{\text{i.i.d.}}{\sim} p(\boldsymbol{x}|y=+1)$$

$$\{\boldsymbol{x}_j^{\mathrm{U}}\}_{j=1}^{n_{\mathrm{U}}} \overset{\text{i.i.d.}}{\sim} p(\boldsymbol{x})$$

- When $p(\boldsymbol{x}|y=+1)$ and $p(\boldsymbol{x}|y=-1)$ are overlapped, they may share some common component.
  - Its proportion can be arbitrarily changed.
  - Indeed, any $\theta \in \left\{\exists p'(\boldsymbol{x}),\ p(\boldsymbol{x}) = \theta p(\boldsymbol{x}|y=+1) + (1-\theta)p'(\boldsymbol{x})\right\}$ can be a valid solution.

- We need a reasonable assumption to obtain a meaningful solution!



# Contents

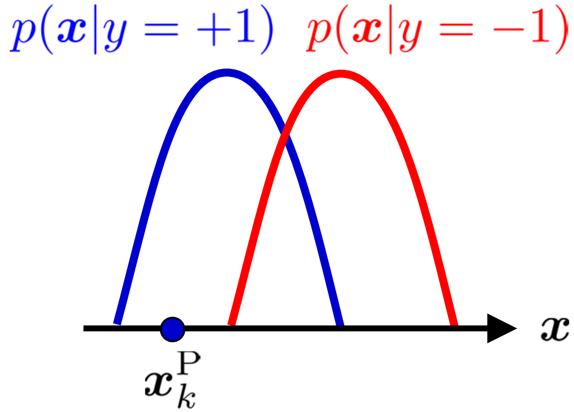


# Class-Prior Estimation under Anchor Point Assumption

Sugiyama et al. (MIT Press, in press)

- Assume there exists an anchor point in $\{\boldsymbol{x}_i^{\mathrm{P}}\}_{i=1}^{n_{\mathrm{P}}}$:
  - For some $k \in \{1, \ldots, n_{\mathrm{P}}\}$, $p(\boldsymbol{x}_k^{\mathrm{P}}|y=+1) > 0$ and $p(\boldsymbol{x}_k^{\mathrm{P}}|y=-1) = 0$.

$$\pi = \frac{p(\boldsymbol{x}_k^{\mathrm{P}})}{p(\boldsymbol{x}_k^{\mathrm{P}}|y=+1)}$$




- Density ratio estimation gives

$$\hat{\pi} = \min_{i \in \{1,\ldots,n_{\mathrm{P}}\}} \hat{r}^{-1}(\boldsymbol{x}_i^{\mathrm{P}}) \qquad \hat{r}(\boldsymbol{x}) \approx \frac{p(\boldsymbol{x}|y=+1)}{p(\boldsymbol{x})}$$

- Simple and nice!
  - But the anchor point assumption may be too strong.

# Contents



# Partial Matching with Differentiable $f$-Divergence

du Plessis et al. (ACML2015, MLJ2017)

$$\mathrm{div}(\theta) = \mathrm{Div}_f[p(\boldsymbol{x}) \| q'(\boldsymbol{x})] = \int p(\boldsymbol{x}) f\left(\frac{q'(\boldsymbol{x})}{p(\boldsymbol{x})}\right) \mathrm{d}\boldsymbol{x}$$

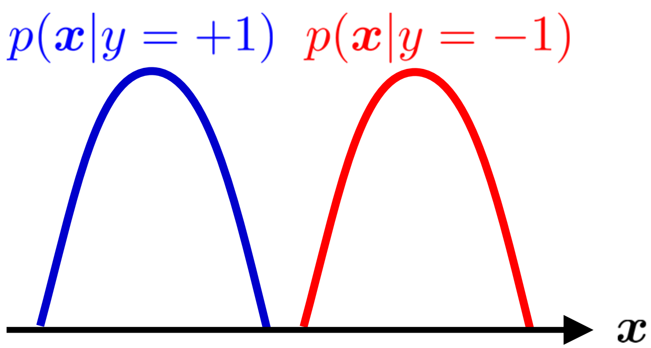
$q'(\boldsymbol{x}) = \theta p(\boldsymbol{x}|y=+1)$

Suppose $f(t)$ has the minimum at $t \geq 1$.

- When $f(t)$ is differentiable,
  $\mathrm{div}'(\pi) = 0$ is necessary for $\pi = \underset{\theta \in [0,1]}{\mathrm{argmin}}\, \mathrm{div}(\theta)$.

$$\mathrm{div}'(\pi) = \int f'\big(p(y=+1|\boldsymbol{x})\big) p(\boldsymbol{x}|y=+1) \mathrm{d}\boldsymbol{x}$$

- $\mathrm{div}'(\pi) = 0$ if and only if
  - Two classes are non-overlapped,
  - and $f'(1) = 0$ (e.g., Pearson div).



# With Non-Differentiable $f$-Divergence

$$\mathrm{div}(\theta) = \mathrm{Div}_f[p(\boldsymbol{x})\|q'(\boldsymbol{x})] = \int p(\boldsymbol{x}) f\left(\frac{q'(\boldsymbol{x})}{p(\boldsymbol{x})}\right) \mathrm{d}\boldsymbol{x}$$

Suppose $f(t)$ has the minimum at $t \geq 1$.

- When $f(t)$ is non-differentiable at $t = 1$, $0 \in \partial \mathrm{div}(\pi)$ is necessary for $\pi = \underset{\theta \in [0,1]}{\mathrm{argmin}}\, \mathrm{div}(\theta)$.

$\partial$ : subdifferential

$$\partial \mathrm{div}(\pi) = \int \partial f\big(p(y=+1|\boldsymbol{x})\big) p(\boldsymbol{x}|y=+1) \mathrm{d}\boldsymbol{x}$$

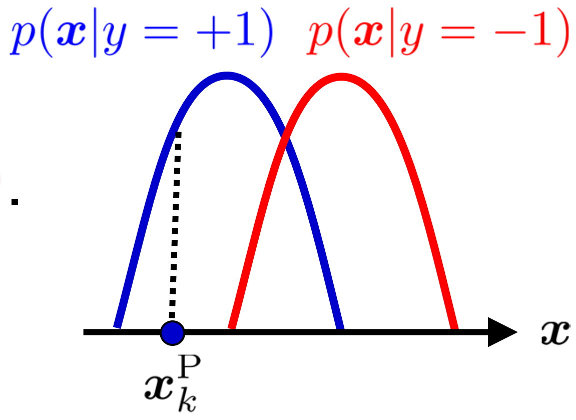
- $0 \in \partial \mathrm{div}(\pi)$ even when two classes are overlapped, if
  - $f(t)$ is penalized as $f(t) \leftarrow \begin{cases} f(t) & (t \leq 1) \\ \infty & (t > 1) \end{cases}$,
  - and the irreducibility assumption holds: Blanchard et al. (JMLR2010)
    - $p(\boldsymbol{x}|y=+1)$ is not a component of $p(\boldsymbol{x}|y=-1)$.

# Irreducibility and Anchor Points

- **Irreducibility**: Blanchard et al. (JMLR2010)
  - $p(\boldsymbol{x}|y=+1)$ is not a component of $p(\boldsymbol{x}|y=-1)$ .

$$\pi = \sup\left\{\pi' \;\middle|\; \exists p'(\boldsymbol{x}),\; p(\boldsymbol{x}) = \pi' p(\boldsymbol{x}|y=+1) + (1-\pi')p'(\boldsymbol{x})\right\}$$

- **Anchor points**: Liu & Tao (IEEE-TPAMI2015)
  - For some $k \in \{1, \ldots, n_{\mathrm{P}}\}$, $p(\boldsymbol{x}_k^{\mathrm{P}}|y=+1) > 0$ and $p(\boldsymbol{x}_k^{\mathrm{P}}|y=-1) = 0$.




- Irreducibility holds if and only if at least one anchor point exists:
  - Density ratio based method uses the anchor point explicitly.
  - Partial matching only assumes its existence implicitly.
- Therefore, the required assumption is weaker!

# Practical Choice of $f$: Penalized L1-Distance



du Plessis et al. (ACML2015, MLJ2017)

$$\mathrm{pen}L_1(\theta) = \int p(\boldsymbol{x}) f\left(\frac{q'(\boldsymbol{x})}{p(\boldsymbol{x})}\right) \mathrm{d}\boldsymbol{x}$$

$$f(t) = \begin{cases} 1-t & (t \leq 1) \\ \infty & (t > 1) \end{cases}$$

- Regularized least-squares density ratio estimation gives a divergence approximator analytically as

$$\widehat{\mathrm{pen}L_1}(\theta) = \frac{1}{\lambda} \sum_{b=1}^{B} \max(0, \beta_b)\beta_b - \theta + 1$$

$\lambda > 0$ : Regularization parameter

$$\beta_b = \frac{\theta}{n_{\mathrm{P}}} \sum_{i=1}^{n_{\mathrm{P}}} \varphi_b(\boldsymbol{x}_i^{\mathrm{P}}) - \frac{1}{n_{\mathrm{U}}} \sum_{j=1}^{n_{\mathrm{U}}} \varphi_b(\boldsymbol{x}_j^{\mathrm{P}})$$

$$\{\boldsymbol{x}_i^{\mathrm{P}}\}_{i=1}^{n_{\mathrm{P}}} \overset{\mathrm{i.i.d.}}{\sim} p(\boldsymbol{x}|y=+1)$$

$$\{\boldsymbol{x}_j^{\mathrm{U}}\}_{j=1}^{n_{\mathrm{U}}} \overset{\mathrm{i.i.d.}}{\sim} p(\boldsymbol{x})$$

  - Model of density ratio $q'(\boldsymbol{x})/p(\boldsymbol{x})$:

$$s(\boldsymbol{x}) = \sum_{b=1}^{B} \alpha_b \varphi_b(\boldsymbol{x}) + 1$$

$\varphi_b(\boldsymbol{x}) \geq 0$ : Basis function

$\alpha_b \geq 0$ : Parameter

# Implementation and Analysis

$$\widehat{\mathrm{pen}L_1}(\theta) = \frac{1}{\lambda}\sum_{b=1}^{B} \max(0, \beta_b)\beta_b - \theta + 1$$

- Algorithm: Find a minimizer w.r.t. $\theta \in [0, 1]$.
  - Computationally very efficient!

- Optimal convergence rate is achieved!

$$\mathrm{pen}L_1(\hat{\pi}) - \mathrm{pen}L_1(\pi) = \mathcal{O}_p(1/\sqrt{n_\mathrm{P}} + 1/\sqrt{n_\mathrm{U}})$$

$$\hat{\pi} = \mathop{\mathrm{argmin}}_{0 \le \theta \le 1} \widehat{\mathrm{pen}L_1}(\theta)$$

$$\beta_b = \frac{\theta}{n_\mathrm{P}}\sum_{i=1}^{n_\mathrm{P}} \varphi_b(\boldsymbol{x}_i^\mathrm{P}) - \frac{1}{n_\mathrm{U}}\sum_{j=1}^{n_\mathrm{U}} \varphi_b(\boldsymbol{x}_j^\mathrm{P})$$

$$\{\boldsymbol{x}_i^\mathrm{P}\}_{i=1}^{n_\mathrm{P}} \overset{\text{i.i.d.}}{\sim} p(\boldsymbol{x}|y=+1)$$

$$\{\boldsymbol{x}_j^\mathrm{U}\}_{j=1}^{n_\mathrm{U}} \overset{\text{i.i.d.}}{\sim} p(\boldsymbol{x})$$

$$\mathrm{pen}L_1(\theta) = \int p(\boldsymbol{x}) f\left(\frac{q'(\boldsymbol{x})}{p(\boldsymbol{x})}\right) \mathrm{d}\boldsymbol{x}$$

$$f(t) = \begin{cases} 1-t & (t \le 1) \\ \infty & (t > 1) \end{cases}$$

$$p(\boldsymbol{x}) = \pi p(\boldsymbol{x}|y=+1) + (1-\pi)p(\boldsymbol{x}|y=-1$$

$$q'(\boldsymbol{x}) = \theta p(\boldsymbol{x}|y=+1)$$

- However, there is no way to assess irreducibility in practice.

# Contents





# Class-Prior Estimation without Irreducibility

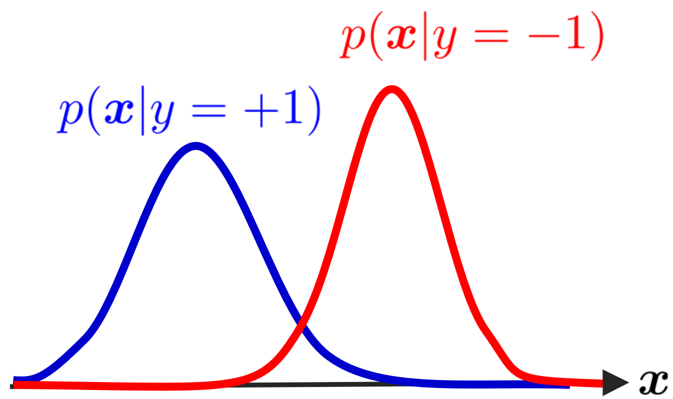
$$p(\boldsymbol{x}) = \pi p(\boldsymbol{x}|y=+1) + (1-\pi)p(\boldsymbol{x}|y=-1)$$

- Without irreducibility, any

$$\theta \in \left\{ \exists p'(\boldsymbol{x}),\ p(\boldsymbol{x}) = \theta p(\boldsymbol{x}|y=+1) + (1-\theta)p'(\boldsymbol{x}) \right\}$$

  can be a valid solution, due to common components.
  - Partial matching actually gives its maximum value.

- Can we mitigate the positive bias in the absence of irreducibility?



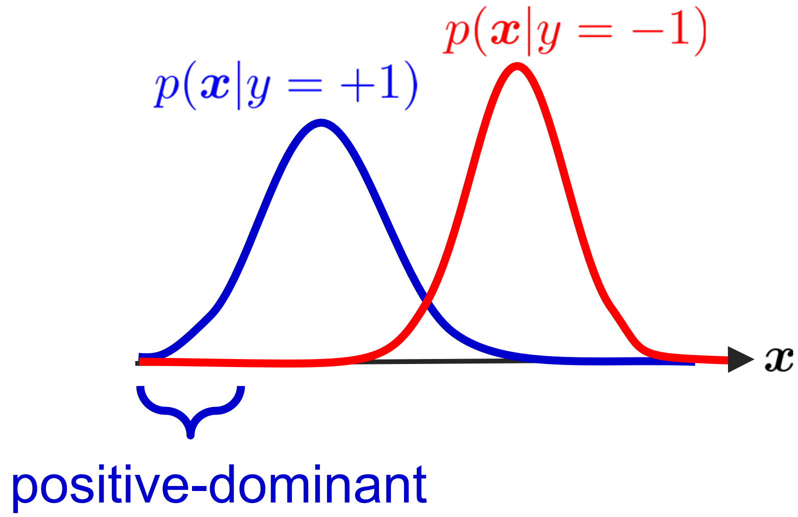
# Regrouping

Yao et al. (arXiv2020)



$$p(\boldsymbol{x}) = \pi p(\boldsymbol{x}|y=+1) + (1-\pi)p(\boldsymbol{x}|y=-1)$$

- Idea: Regroup a small positive-dominant region to be fully positive.




- By this regrouping,
  - $\pi$ is slightly increased,
  - but irreducibility is satisfied!

- How can we find a positive-dominant region?

# Implementation



- Consider PU label $s \in \{0, 1\}$:
  - If $\boldsymbol{x}$ is P (or U), $s = 1$ (or $s = 0$ ).

- Train a "non-traditional" classifier $\hat{p}(s|\boldsymbol{x})$:
  - Usual supervised classification from $\{\boldsymbol{x}_i^{\mathrm{P}}\}_{i=1}^{n_{\mathrm{P}}}, \{\boldsymbol{x}_i^{\mathrm{U}}\}_{i=1}^{n_{\mathrm{U}}}$.

- Select some unlabeled samples that have the highest positive-confidence: $\hat{p}(s = 1|\boldsymbol{x}_j^{\mathrm{U}})$

- Copy them and give positive labels.

- Solve the converted class-prior estimation problem.

# Contents



1. Semi-supervised class-prior shift adaptation
2. Positive-unlabeled classification
   - A) Basic solution
   - B) Identifiability
   - C) Density ratio estimation with anchor points
   - D) Partial distribution matching with irreducibility
   - E) Regrouping without irreducibility
   - F) Summary
3. Conclusions

# Summary: PU Classification

- There is a nice empirical risk minimization method, given class-prior $\pi = p(y = +1)$ can be estimated.

- However, the class-prior is not identifiable in general.

- Simple density ratio estimation solution:
  - Use anchor points (i.e., 100% positive), which may be strong.

- Computationally efficient penL$_1$-div partial matching.
  - Without irreducibility (P-density is not part of N-density), its solution is positively biased.
  - Existence of anchor points is sufficient, but not assessable.

- Regrouping:
  - By preprocessing of data, the positive bias can be reduced.

# Contents



# Summary: Mixture Proportion Estimation

- Many applications in machine learning:
  - Class-prior shift adaptation: Importance weight $p_{\mathrm{te}}(y)$
    - Identifiability allows naïve distribution matching to solve.
  - Positive-unlabeled (PU) classification: Class-prior $p(y)$
    - Non-identifiability posed significant challenges.
  - Noisy label classification: Noise transition $p(\bar{y}|y)$

$$\underbrace{p(\bar{y}|\boldsymbol{x})}_{\text{Observed}} = \sum_y \underbrace{p(\bar{y}|y)\,p(y|\boldsymbol{x})}_{\text{Non-observed}}$$

$y$ : Clean class label

$\bar{y}$ : Noisy class label

    - Multiple non-identifiability is even more challenging!

# Challenge: Overcoming Non-Identifiability

- Identifiability conditions have been investigated:
  - Irreducibility, anchor set, anchor points…

    Blanchard et al. (JMLR2010) Liu & Tao (IEEE-TPAMI2015)

- However, these identifiability conditions may not be satisfied in practice.

- Even without identifiability, it is promising to
  - Reduce estimation bias by regrouping (in PU classification). Yao et al. (arXiv2020)
  - Use a weaker "sufficiently scattered" assumption (in noisy-label classification). Li et al. (ICML2021)

# Challenge: Towards Better Machine Learning (ML)

- The estimated proportion is later used in ML tasks.
- Current approach is two-step:
  - Estimate the mixture proportion. $\hat{\pi} = \operatorname{argmin}_{\pi} \mathrm{MPE}(\pi)$
  - Use the estimated proportion to solve the target ML problem. $\hat{f} = \operatorname{argmin}_{f} \mathrm{ML}(f, \hat{\pi})$
- 1st step is preformed without regards to 2nd step.
- Combining them into one-step is more promising:

$$\hat{f} = \operatorname*{argmin}_{f} \min_{\pi} \mathrm{ML\&MPE}(f, \pi)$$

  - Alternate optimization. Kato et al. (arXiv2018)
  - Joint upper-bound optimization. Zhang et al. (ACML2020, LNCS2021)
  - Dynamic stochastic optimization. Xia et al. (NeurIPS2019) Fang et al. (NeurIPS2020) Zhang et al. (ICML2021)

# Grateful to Great Collaborators!

RIKEN

AIP

| | |
|---|---|
| Team leader PAGE > <br> Masashi Sugiyama | Research scientist <br> Gang Niu |
| Research scientist <br> Fumiko Kawasaki | Research scientist <br> Takahiro Mimori |
| Postdoctoral researcher <br> Voot Tangkaratt | Postdoctoral researcher <br> Jingfeng Zhang |
| Postdoctoral researcher <br> Jiaqi Lyu | Postdoctoral researcher <br> Shuo Chen |
| Technical scientist <br> Yuka Mori | Technical staff I <br> Masashi Ugawa |
| Senior visiting scientist <br> Shinichi Nakajima | Visiting scientist <br> Yoshihiro Nagano |
| Visiting scientist <br> Florian Yger | Visiting scientist <br> Takashi Ishida |
| Visiting scientist <br> Miao Xu | Visiting scientist <br> Takayuki Osa |
| Visiting scientist <br> Bo Han | Visiting scientist <br> Daichi Noborio |
| Visiting scientist <br> Yuko Kuroki | Visiting scientist <br> Hisashi Yoshida |
| Visiting scientist <br> Ryohei Kasai | Visiting scientist <br> Feng Liu |
| Visiting scientist <br> Lei Feng | Visiting scientist <br> Tongliang Liu |
| Junior research associate <br> Takeshi Teshima | Junior research associate <br> Yifan Zhang |
| Part-time worker I <br> Zhenghang Cui | Part-time worker I <br> Han Bao |
| Part-time worker I <br> Masahiro Fujisawa | |

and many interns
over the world!

- Professor
  - Masashi Sugiyama (Complexity, Computer, Information, RIKEN)
- Lecturer
  - Naoto Yokoya (Complexity, Computer, Information, RIKEN)
  - Takashi Ishida (Complexity, Computer, Information)
- Project Lecturer
  - Nobutaka Ito (Complexity)
- Associate professor (to Sato Lab from April 2020)
  - Issei Sato (Computer, Information, Complexity, RIKEN)
- Project Assistant Professor
  - Chao-Kai Chiang (Complexity)
- Project Researcher (Postdoctoral Researcher)
  - Yoshihiro Nagano (Complexity)
  - Dongxian Wu (Complexity)
- Project Specialist
  - Yuko Kawashima (Complexity)
  - Soma Yokoi (Complexity)
  - Fumi Sato (Complexity)
- Doctor Student
  - Seiya Tokui (Computer)* Sato lab.
  - Shinji Nakadai (Computer)
  - Kento Nozawa (Complexity)* Sato lab.
  - Kento Suzuki (Complexity)
  - Han Bao (Computer)
  - Zhenghang Cui (Computer)* Sato lab.
  - Liyuan Xu (Computer)
  - Takeshi Teshima (Complexity)
  - Ryuichi Kiryo (Computer)
  - Masahiro Fujisawa (Complexity)* Sato lab.
  - Jongyeong Lee (Computer)
  - Tianyi Zhang (Complexity)
  - Yivan Zhang (Computer)
  - Riou Charles (Computer)
  - Valliappa Chockalingam (Computer)
  - Tongtong Fang (Complexity)
  - Boyo Chen (Complexity)
  - Xiaoyu Dong (Complexity)
  - Yujie Zhang (Complexity)
  - Xinqiang Cai (Complexity)
  - Jian Song (Complexity)
  - Wanshui Gan (Complexity)
- Master Student
  - Atsushi Ito (Complexity)
  - Shinji Kawakami (Complexity)* Sato lab.
  - Wataru Ohtori (Computer)
  - Tokio Kajitsuka (Computer)
  - Shota Nakajima (Computer)
  - Takahiro Suzuki (Computer)* Sato lab.
  - Kei Mukaiyama (Computer)* Sato lab.
  - Mingcheng Hou (Computer)* Sato lab.
  - Hyunggyu Park (Complexity)* Sato lab.
  - Yuting Tang (Complexity)
  - Shintaro Nakamura (Complexity)
  - Xujie Wang (Complexity)
  - Toshiki Kodera (Computer)
  - Yuma Aoki (Computer)
  - Jiahuan Li (Computer)
  - Chengwei Liang (Computer)
  - Huanjian Zhou (Complexity)
  - Kun Yang (Complexity)
  - Takanori Shirasaka (Complexity)
  - Reo Iizuka (Complexity)
  - Xiaomou Hou (Complexity)
- Bachelor Student
  - Kanma Noda (Information Science)
  - Kento Yamamoto (Information Science)
  - Kazuki Ota (Information Science)
  - Kazutaka Yahiro (Information Science)
  - Hikaru Fujita (Information Science)
- Research Student
  - Kenny Song (Computer)
  - Or Raveh (Computer)
  - Johannes Ackermann (Computer)
  - Anan Methasate (Computer)
  - Cemal Erat (Computer)

東京大学
THE UNIVERSITY OF TOKYO

ist