Positive-Unlabeled Classification under Class Prior Shift and Asymmetric Error

Nontawat Charoenphakdee\textsuperscript{1,2} and Masashi Sugiyama\textsuperscript{2,1}

The University of Tokyo\textsuperscript{1}
RIKEN AIP\textsuperscript{2}
Supervised binary classification (PN classification)

Positive and Negative data are given.

Data collection

Features (input)  Labels (output)

Binary Classifier

Machine learning
Positive-unlabeled classification (PU classification)

Positive and Unlabeled data are given.

Data collection

Features (input)  Labels (output)

Machine learning

Binary Classifier

Features (input)

Labels (output)
Why PU classification?

Unlabeled data are cheaper to obtain.
Sometimes, negative data are hard to describe.
In some real-world applications, collecting negative data is impossible.

Applications:
• Bioinformatics (Yang+, 2012, Singh-Blom+ 2013, Ren+, 2015)
• Text classification (Li+, 2003)
• Time series classification (Nguyen+, 2011)
• Medical diagnosis (Zuluaga+, 2011)
• Remote-sensing classification (Li+, 2011)
Class prior shift

The ratio of positive-negative in the training and test data are different.

Examples:
- Collect unlabeled data from the internet.
- Collect unlabeled data from all users/patients/etc. for personalized application.
Existing **PU classification** work assumes class prior of **training** and **test** data are the same (du Plessis+, 2014 2015, Kiryo+, 2017).

Existing class prior shift work is not applicable since they require **positive-negative** data (Saerens, 2002, du Plessis+, 2012).
# PU classification under class prior shift

Given: Two sets of data

| Observed          | Unobserved       | Test            |
|-------------------|------------------|-----------------|
| **Positive**      |                  |                 |
| $X_P := \{x_i^p\}_{i=1}^{n_P}$ i.i.d. $\sim \text{pos}(x)$ |                  | $X_{\text{te}} := \{x_k^{\text{te}}\}_{k=1}^{n_{\text{te}}}$ i.i.d. $\sim \pi_{\text{te}} \text{pos}(x) + (1 - \pi_{\text{te}}) \text{neg}(x)$ |
| **Unlabeled**     |                  |                 |
| $X_U := \{x_j^U\}_{j=1}^{n_U}$ i.i.d. $\sim \pi_{\text{tr}} \text{pos}(x) + (1 - \pi_{\text{tr}}) \text{neg}(x)$ |                  |                 |

$\pi : p(y = 1)$
$\text{pos}(x) : p(x|y = 1)$
$\text{neg}(x) : p(x|y = -1)$

Q: Does class prior shift heavily degrade the performance?
Classifier may fail miserably under class prior shift...

Accuracy reported in mean and std. error of 10 trials with density ratio method.

| Dataset | Accuracy (no shift) | Accuracy (shifted) | Accuracy (shifted) |
|---------|---------------------|--------------------|--------------------|
| banana  | 90.1 (0.6)          | 82.3 (0.5)         | 87.9 (0.3)         |
| ijcnn1  | 72.9 (0.4)          | 37.8 (0.7)         | 71.7 (0.3)         |
| MNIST   | 86.0 (0.4)          | 69.8 (0.7)         | 82.5 (0.6)         |
| susy    | 79.5 (0.5)          | 57.5 (0.9)         | 75.9 (0.5)         |
| cod-rna | 87.4 (0.6)          | 78.5 (0.6)         | 84.7 (0.4)         |
| magic   | 76.7 (0.5)          | 60.6 (1.4)         | 79.0 (0.5)         |

No shift: $\pi_{tr} = \pi_{te} = 0.3$

Shift! $\pi_{tr} = 0.7$, $\pi_{te} = 0.3$
Problem setting

• Given: Two sets of data and test class prior $\pi_{te}$

Positive

$X_P := \{x_i^P\}_{i=1}^{n_P} \overset{i.i.d.}{\sim} pos(x)$

Unlabeled

$X_U := \{x_j^U\}_{j=1}^{n_U} \overset{i.i.d.}{\sim} \pi_{tr} pos(x) + (1 - \pi_{tr}) neg(x)$

• Goal: Find a prediction function $g$ that minimizes

$$R_{\text{Shift}}^{\ell_{0-1}}(g) = \pi_{te} \mathbb{E}_P[\ell_{0-1}(g(x))] + (1 - \pi_{te}) \mathbb{E}_N[\ell_{0-1}(-g(x))]$$
Proposed methods

We proposed two approaches for **PU classification** under **class prior shift**:

- **Risk minimization approach:**
  
  Learn a classifier based on **empirical risk minimization** principle (Vapnik, 1998).

- **Density ratio approach:**
  
  1. Estimate a **density ratio** of positive and unlabeled densities.
  2. Use an appropriate threshold to classify.

Later, we will show that our methods are also applicable for **PU classification** with **asymmetric error**.
Risk minimization approach

Consider the following classification risk:

\[ R_{\text{Shift}}^{\ell_{0-1}}(g) = \pi_{\text{te}} \mathbb{E}_P [\ell_{0-1}(g(x))] + (1 - \pi_{\text{te}}) \mathbb{E}_N [\ell_{0-1}(-g(x))] \]

With \( \mathbb{E}_u[\cdot] = \pi_{\text{tr}} \mathbb{E}_P [\cdot] + (1 - \pi_{\text{tr}}) \mathbb{E}_N [\cdot] \), we can rewrite \( R_{\text{Shift}}^{\ell_{0-1}}(g) \) as

\[ R_{\text{Shift}}^{\ell_{0-1}}(g) = \mathbb{E}_P \left[ \pi_{\text{te}} \ell_{0-1}(g(x)) - \frac{\pi_{\text{tr}}(1 - \pi_{\text{te}})}{1 - \pi_{\text{tr}}} \ell_{0-1}(-g(x)) \right] + \frac{1 - \pi_{\text{te}}}{1 - \pi_{\text{tr}}} \mathbb{E}_u [\ell_{0-1}(-g(x))] \]

Equivalent to existing methods (du Plessis+, 2015) if \( \pi_{\text{tr}} = \pi_{\text{te}} \).

No access to distribution: we minimize empirical error (Vapnik, 1998):

\[ \hat{R}_{\text{PU-shift}}^{\ell_{0-1}}(g) = \frac{1}{n_P} \sum_{i=1}^{n_P} \left[ \pi_{\text{te}} \ell_{0-1}(g(x_i^P)) - \frac{\pi_{\text{tr}}(1 - \pi_{\text{te}})}{1 - \pi_{\text{tr}}} \ell_{0-1}(-g(x_i^P)) \right] + \frac{1}{n_U} \frac{1 - \pi_{\text{te}}}{1 - \pi_{\text{tr}}} \sum_{j=1}^{n_U} \ell_{0-1}(-g(x_j^u)) \]
Surrogate losses for binary classification

Directly minimize 0-1 loss is difficult.

• NP-Hard, discontinuous, not differentiable (Ben-david+, 2003, Feldman+, 2012)

In practice, minimize a surrogate loss (regularization can also be added):

\[
\hat{R}_{\text{PU-shift}}^\ell(g) = \frac{1}{n_P} \sum_{i=1}^{n_P} \left[ \pi_{te} \ell(g(x_i^p)) - \frac{\pi_{tr}(1 - \pi_{te})}{1 - \pi_{tr}} \ell(-g(x_i^p)) \right] + \frac{1}{n_U} \frac{1 - \pi_{te}}{1 - \pi_{tr}} \sum_{j=1}^{n_U} \ell(-g(x_j^u))
\]
Density ratio estimation

**Goal:** Estimate the density ratio:

\[ r(x) = \frac{p_{\text{nu}}(x)}{p_{\text{de}}(x)} \]

from two sets of data

\[ X_{\text{nu}} := \{ x_{i}^{\text{nu}} \}_{i=1}^{n_{\text{nu}}} \overset{\text{i.i.d.}}{\sim} p_{\text{nu}}(x) \]
\[ X_{\text{de}} := \{ x_{j}^{\text{de}} \}_{j=1}^{n_{\text{de}}} \overset{\text{i.i.d.}}{\sim} p_{\text{de}}(x) \]

**Applications:** outlier detection \((Hido+, 2011)\), change-point detection \((Liu+, 2013)\), robot control \((Hachiya+, 2009)\), event detection in images/movies/text \((Yamanaka, 2011, Matsugu, 2011, Liu, 2012)\), etc.

**Naïve approach:** estimate \( \hat{p}_{\text{nu}}(x) \), \( \hat{p}_{\text{de}}(x) \) separately then perform division \( \frac{\hat{p}_{\text{nu}}(x)}{\hat{p}_{\text{de}}(x)} \).  
**Does not work well** (estimation error is amplified from division operation).  

Please check this book to learn more about density ratio estimation \((Sugiyama+, 2012)\).
**Unconstrained least-squares important fitting (uLSIF)**

**Goal**: Estimate the density ratio:

\[ r(x) = \frac{p_{nu}(x)}{p_{de}(x)} \]

*(Kanamori+, 2012)*

**How**: estimate \( \hat{r} \) by minimizing squared loss objective:

\[
\text{SQ}(\hat{r}) = \int \left( \hat{r}(x) - r(x) \right)^2 p_{de}(x) dx
\]

Squared loss decomposition:

\[
\text{SQ}(\hat{r}) = \int \left( \hat{r}(x) \right)^2 p_{de}(x) dx - 2 \int \hat{r}(x)p_{nu}(x) dx + \text{Constant}
\]

Empirical minimization (constant can be safely ignored):

\[
\hat{\text{SQ}}(\hat{r}) = \frac{1}{n_{de}} \sum_{j=1}^{n_{de}} \left( \hat{r}(x_{j}^{de}) \right)^2 - \frac{2}{n_{nu}} \sum_{i=1}^{n_{nu}} \hat{r}(x_{i}^{nu})
\]
Unconstrained least-squares important fitting (cont.)

Model: linear-in parameter model

\[ \hat{r}(\mathbf{x}) = \sum_b \theta_b \phi_b(\mathbf{x}) = \theta^\top \phi(\mathbf{x}) \]

Objective:

\[ \min_{\theta} \left[ \frac{1}{2} \theta^\top \hat{H} \theta - \hat{h}^\top \theta + \frac{\lambda}{2} \theta^\top \theta \right] \]

Global solution can be computed **analytically**: \( \hat{\theta} = (\hat{H} + \lambda I)^{-1} \hat{h} \)

Parameter tuning (regularization, basis) can be done by **cross-validation**.

\( \phi_b(\mathbf{x}) \): basis function (e.g., Gaussian kernel)

\[ \hat{H} = \frac{1}{n_{de}} \sum_{j=1}^{n_{de}} \phi(\mathbf{x}_j^{de}) \phi(\mathbf{x}_j^{de})^\top \]

\[ \hat{h} = \frac{1}{n_{nu}} \sum_{i=1}^{n_{nu}} \phi(\mathbf{x}_i^{nu}) \]

\( \lambda \): regularization parameter

\( I \): identity matrix
Density ratio approach

Consider Bayes-optimal classifier of binary classification (no prior shift)

\[
\begin{align*}
\text{pos}(\mathbf{x}) & : p(\mathbf{x}|y = 1) \\
\text{neg}(\mathbf{x}) & : p(\mathbf{x}|y = -1) \\
\text{unl}(\mathbf{x}) & = \pi_{tr}\text{pos}(\mathbf{x}) + (1 - \pi_{tr})\text{neg}(\mathbf{x})
\end{align*}
\]

\[
f_{\text{Bayes}}^*(\mathbf{x}) = \text{sign}
\left[
\frac{p(y = +1|\mathbf{x})}{2}
\right]
\]

We can rewrite it as

\[
f_{\text{Bayes}}^*(\mathbf{x}) = \text{sign}
\left[
\frac{\pi_{tr}\text{pos}(\mathbf{x})}{\text{unl}(\mathbf{x})} - \frac{1}{2}
\right]
\]

Density ratio!

Another formulation is

\[
f_{\text{Bayes}}^*(\mathbf{x}) = \text{sign}
\left[
\pi_{tr} - \frac{1}{2}\frac{\text{unl}(\mathbf{x})}{\text{pos}(\mathbf{x})}
\right]
\]

Q1: How to modify when class prior shift occurs?
Q2: Which formulation is preferable?
Q1: Density ratio approach (shift)

Consider Bayes-optimal classifier of binary classification

\[ f_{\text{Bayes}}(\mathbf{x}) = \text{sign} \left[ p(y = +1|\mathbf{x}) - \frac{1}{2} \right] \]

\[ \text{pos}(\mathbf{x}) : p(\mathbf{x}|y = 1) \]
\[ \text{neg}(\mathbf{x}) : p(\mathbf{x}|y = -1) \]
\[ \text{unl}(\mathbf{x}) = \pi_{\text{tr}} \text{pos}(\mathbf{x}) + (1 - \pi_{\text{tr}}) \text{neg}(\mathbf{x}) \]

We can rewrite it as

\[ f_{\text{Bayes}}(\mathbf{x}) = \text{sign} \left[ \frac{\pi_{\text{tr}} \text{pos}(\mathbf{x})}{\pi_{\text{tr}} \text{unl}(\mathbf{x})} - \frac{\pi_{\text{tr}} (1 - \pi_{\text{te}})}{\pi_{\text{te}} + \pi_{\text{tr}} - 2\pi_{\text{tr}} \pi_{\text{te}}} \right] \]

Another formulation is

\[ f_{\text{Bayes}}(\mathbf{x}) = \text{sign} \left[ \frac{\pi_{\text{te}} + \pi_{\text{tr}} - 2\pi_{\text{tr}} \pi_{\text{te}}}{(1 - \pi_{\text{te}})} - \frac{\text{unl}(\mathbf{x})}{\pi_{\text{tr}} \text{pos}(\mathbf{x})} \right] \]

Density ratio!

Simply modifying the threshold can solve this problem!
Q2: Difficulty of density ratio estimation

In general, density ratio is unbounded. 😞

$$ r(x) = \frac{p_{nu}(x)}{p_{de}(x)} $$

$r(x)$ is unbounded when $p_{de}(x) = 0$. This raises issues of robustness and stability.

We show that the density ratio $\frac{pos(x)}{unl(x)}$ is bounded in PU classification. 😊
Q2: Density ratio in PU

In **PU classification**, density ratio $\frac{\text{pos}(\mathbf{x})}{\text{unl}(\mathbf{x})}$ is bounded.

$$\begin{align*}
\text{pos}(\mathbf{x}) & : p(\mathbf{x} | y = 1) \\
\text{neg}(\mathbf{x}) & : p(\mathbf{x} | y = -1) \\
\text{unl}(\mathbf{x}) & = \pi_{tr}\text{pos}(\mathbf{x}) + (1 - \pi_{tr})\text{neg}(\mathbf{x})
\end{align*}$$

0 \leq \frac{\text{pos}(\mathbf{x})}{\text{unl}(\mathbf{x})} \leq \frac{1}{\pi_{tr}} \quad \text{Lower and upper bounded 😊}

\pi_{tr} \leq \frac{\text{unl}(\mathbf{x})}{\text{pos}(\mathbf{x})} \quad \text{Unbounded from above 😞}

Insight: estimate $\frac{\text{pos}(\mathbf{x})}{\text{unl}(\mathbf{x})}$ is preferable.

Our experimental results agree with this observation.
Experiments: class prior shift  train 0.7 -> test 0.3

Datasets: banana, ijcnn1, MNIST, susy, cod-rna, magic

Methods:

- Density ratio \( \frac{\text{pos}(x)}{\text{unl}(x)} \) (\( \frac{p}{u} \)uLSIF)
- Density ratio \( \frac{\text{unl}(x)}{\text{pos}(x)} \) (\( \frac{u}{p} \)uLSIF)
- Linear-in input model (Lin): Double hinge loss (DH-Lin), squared loss (Sq-Lin)
- Kernel model (Ker): Double hinge loss (DH-Ker), squared loss (Sq-Ker)

Parameter selection: (regularization, kernel width) 5-fold cross-validation.

We also investigated when wrong test class prior is given.

Results reported in mean and std. error of accuracy of 10 trials.
Outperforming methods are bolded based on one-sided t-test with significance level 5%.
Dataset information and more experiments and can be found in the paper.
### Results: class prior shift \( \pi_{tr} = 0.7, \pi_{te} = 0.3 \)

| Dataset | \( \pi' \) | \( \frac{\pi}{\mu} \) uLSIF | \( \frac{\mu}{\pi} \) uLSIF | DH-Lin | DH-Ker | Sq-Lin | Sq-Ker |
|---------|-----------|----------------|----------------|-------|-------|-------|-------|
| banana  | 83.0 (1.0)| 86.4 (0.5)    | 70.2 (0.5)     | 78.3 (1.0) | 70.0 (0.0) | 83.4 (0.4) |
| ijcnn1  | 70.8 (0.6)| 74.2 (0.7)    | 70.0 (0.1)     | 69.8 (0.2) | 71.5 (0.3) | 69.2 (0.5) |
| MNIST   | 79.3 (0.5)| 81.7 (0.5)    | 74.0 (1.1)     | 82.4 (1.0) | 52.3 (1.4) | 83.4 (0.9) |
| susy    | 74.3 (0.5)| 76.0 (0.3)    | 72.7 (0.6)     | 70.0 (0.0) | 75.5 (1.4) | 74.7 (0.7) |
| cod-rna | 82.1 (1.0)| 82.8 (0.8)    | 87.3 (0.7)     | 77.3 (0.8) | 85.2 (1.1) | 80.2 (1.0) |
| magic   | 71.5 (0.7)| 75.8 (0.6)    | 72.7 (1.1)     | 70.8 (0.4) | 75.0 (1.0) | 72.9 (0.7) |

| Dataset | \( \pi' \) | \( \frac{\pi}{\mu} \) uLSIF | \( \frac{\mu}{\pi} \) uLSIF | DH-Lin | DH-Ker | Sq-Lin | Sq-Ker |
|---------|-----------|----------------|----------------|-------|-------|-------|-------|
| banana  | 84.7 (1.1)| 88.7 (0.7)    | 54.9 (1.4)     | 81.7 (1.6) | 53.6 (1.2) | 83.8 (1.3) |
| ijcnn1  | 64.9 (1.4)| 66.6 (1.0)    | 60.4 (1.4)     | 51.6 (3.0) | 62.2 (1.2) | 48.2 (2.8) |
| MNIST   | 81.9 (0.4)| 84.1 (0.6)    | 72.5 (1.0)     | 82.5 (0.7) | 52.9 (1.1) | 81.9 (0.9) |
| susy    | 75.9 (1.1)| 77.0 (0.6)    | 67.5 (1.4)     | 75.5 (0.6) | 71.6 (1.0) | 72.8 (1.1) |
| cod-rna | 85.3 (0.7)| 85.4 (0.5)    | 86.2 (0.7)     | 80.1 (1.1) | 86.5 (0.9) | 81.2 (1.2) |
| magic   | 67.6 (0.8)| 73.6 (0.9)    | 72.6 (0.7)     | 62.4 (1.9) | 71.8 (0.7) | 68.9 (0.8) |

| Dataset | \( \pi \) | \( \frac{\pi}{\mu} \) uLSIF | \( \frac{\mu}{\pi} \) uLSIF | DH-Lin | DH-Ker | Sq-Lin | Sq-Ker |
|---------|-----------|----------------|----------------|-------|-------|-------|-------|
| banana  | 80.6 (1.3)| 82.1 (1.1)    | 31.8 (0.9)     | 48.9 (1.5) | 30(0.0) | 69.9 (1.1) |
| ijcnn1  | 35.2 (1.4)| 42.4 (0.9)    | 30.0 (0.0)     | 30.0 (0.0) | 32.4 (0.5) | 30.9 (0.4) |
| MNIST   | 79.9 (0.7)| 72.6 (0.6)    | 71.1 (1.1)     | 64.8 (1.1) | 64.0 (0.6) | 74.2 (1.0) |
| susy    | 35.6 (3.1)| 44.2 (2.9)    | 30.0 (0.0)     | 30.0 (0.0) | 42.0 (1.5) | 36.8 (1.3) |
| cod-rna | 77.7 (2.2)| 77.8 (2.1)    | 79.6 (0.7)     | 67.8 (0.8) | 78.2 (0.5) | 68.3 (1.0) |
| magic   | 51.6 (0.3)| 60.3 (1.5)    | 56.2 (2.7)     | 32.8 (0.7) | 58.7 (1.4) | 50.1 (1.6) |

- **Correct test prior is given**
- **Wrong test prior is given**
- **Traditional PU**

**Preferable method in our experiments** (density ratio \( \frac{\mu}{\pi} \) uLSIF)
PU classification with asymmetric error

• **Given:** Given two sets of sample:

  Positive \( X_P := \{ x_i^P \}_{i=1}^n \) i.i.d. \( \sim \) pos(\( \mathbf{x} \))

  Unlabeled \( X_U := \{ x_i^U \}_{i=1}^{n'} \) i.i.d. \( \sim \) \( \pi_tr \) pos(\( \mathbf{x} \)) + \( (1 - \pi_tr) \) neg(\( \mathbf{x} \))

• **Goal:** Find a prediction function \( g \) that minimizes

\[
R_{Asym}^\ell(g) = (1 - \alpha)\pi_tr \mathbb{E}_P [\ell(g(x_P))] + \alpha(1 - \pi_tr) \mathbb{E}_N [\ell(-g(x_N))]
\]

Reduce to symmetric error when \( \alpha = 0.5 \)
The equivalence of prior shift and asymmetric error

\[ \alpha = \frac{\pi_{tr}(1 - \pi_{te})}{\pi_{te} + \pi_{tr} - 2\pi_{tr}\pi_{te}} \]

\[ \pi_{te}' = \frac{\pi_{te} - \alpha\pi_{te}}{\pi_{te} + \alpha - 2\alpha\pi_{te}} \]

\[ \alpha' = \frac{\pi_{tr}(1 - \pi_{te}')}{\pi_{te}' + \pi_{tr} - 2\pi_{tr}\pi_{te}'} \]

PU prior shift \quad \rightarrow \quad PU prior shift and asymmetric error \quad \rightarrow \quad PU asymmetric error

\[ \pi_{te} = \frac{\pi_{tr} - \alpha\pi_{tr}}{\pi_{tr} + \alpha - 2\alpha\pi_{tr}} \]

We can relate these problems based on the analysis of Bayes-optimal classifier.
Conclusion

Class prior shift may heavily degrade the performance of positive-unlabeled classification (PU classification).

- Proposed two approaches for handling this problem effectively:
  - Risk minimization approach
  - Density ratio approach
- Showed the equivalence of class prior shift and asymmetric error problems in PU classification.
  - Our methods are applicable for both problems.
  - Also applicable when considering both problems simultaneously.
- Poster: #31: May 2\textsuperscript{nd} from 7:00-9:00PM