경북대 기계학습 연구실
손정우

Coping with distribution difference between training and test corpus

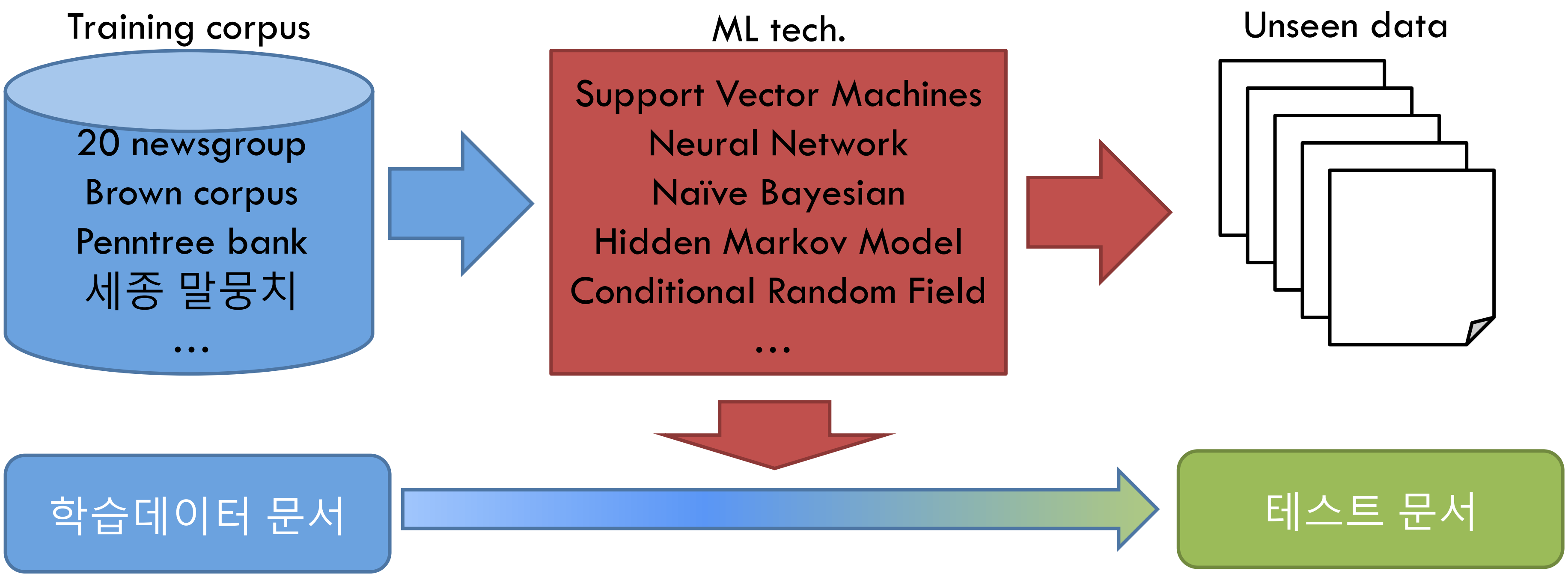
Contents

- 문제 소개
- 기존 연구들
- Learning with Local Importance Weight
- 실험
- 결론

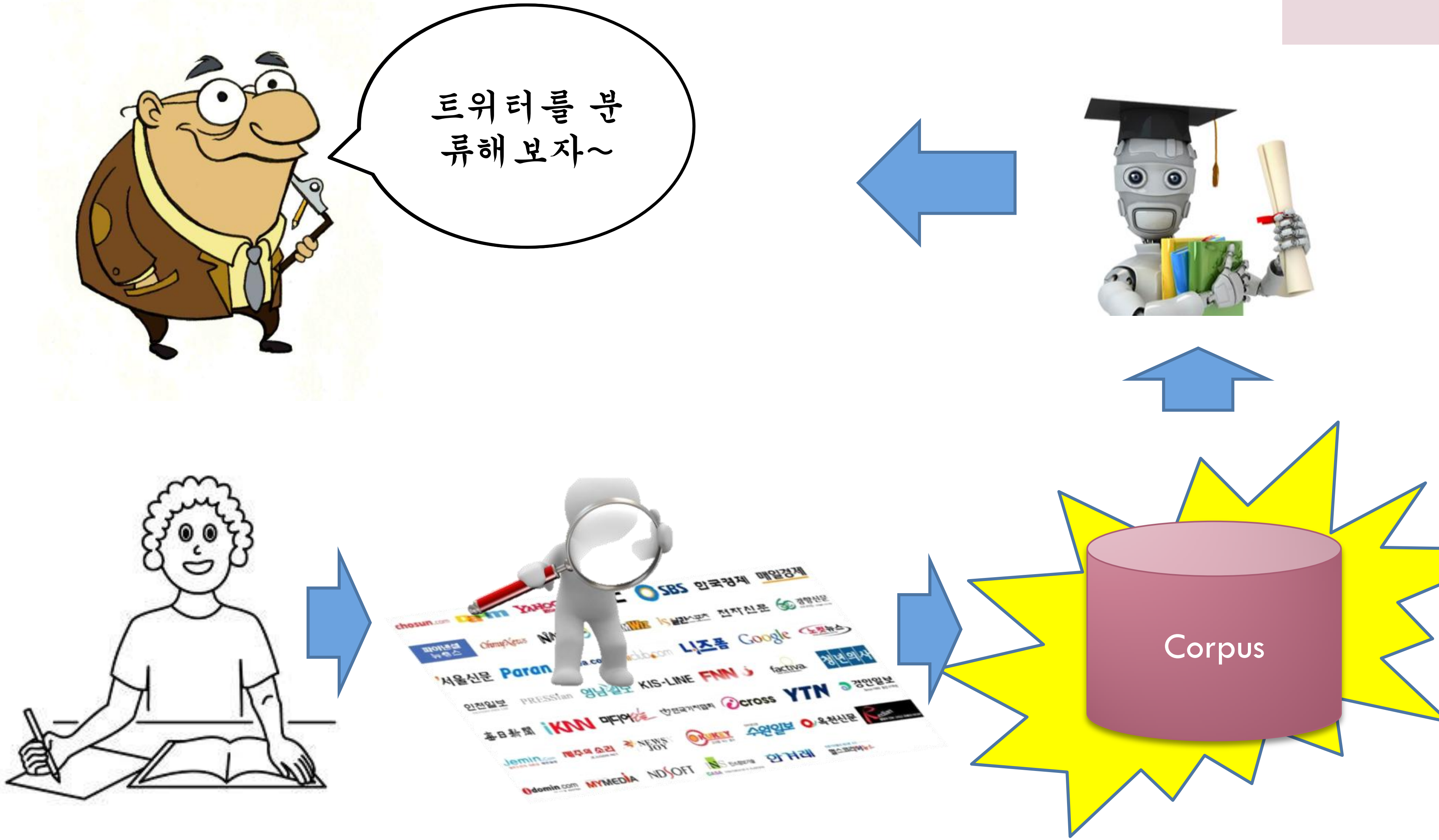
문제 소개

Distribution difference?

- NLP + machine learning
  - 학습 데이터 → 패턴 학습
  - 학습된 패턴 → 새로운 데이터



Simple scenario



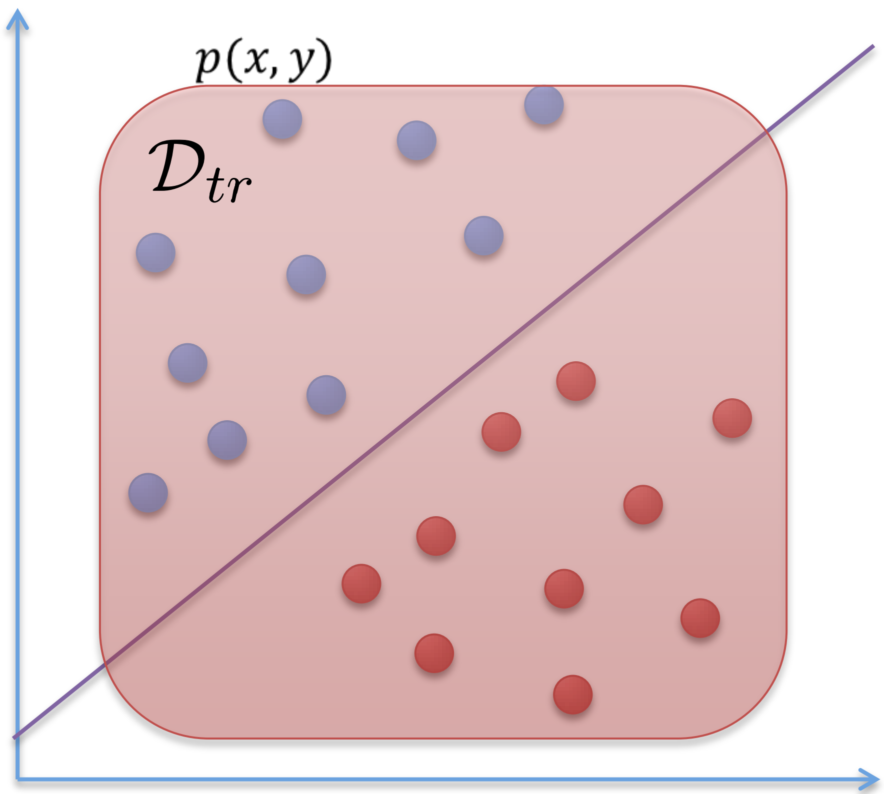
The 1st Problem & solution

- (Martinez & Agirre, EMNLP 2000), (Esucudero et al., EMNLP 2000)
  - Training corpus ≠ test corpus → performance drop (10%)

- 해결책
  - 다르면, 같도록 하자.

Machine learning

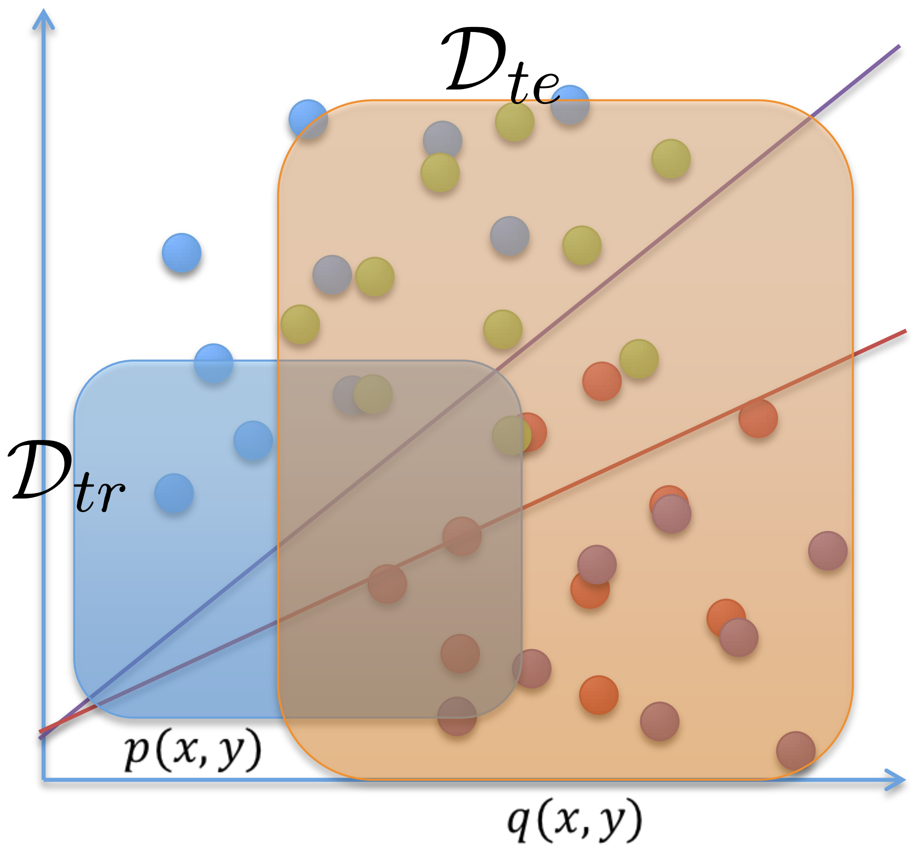
- **Goal:** $f(x;q) \to y$



$$
\begin{aligned}
\theta^* &= \arg\min_{\theta\in\Theta} \mathbf{E}\left[L(x,y,\theta)\right] \\
&= \arg\min_{\theta\in\Theta} \sum_{(x,y)\in X\times Y} p(x,y)L(x,y,\theta) \\
&\approx \arg\min_{\theta\in\Theta} \sum_{(x,y)\in X\times Y} \hat{p}(x,y)L(x,y,\theta) \\
&= \arg\min_{\theta\in\Theta} \sum_{i=1}^{|\mathcal{D}_{tr}|} L(x_i,y_i,\theta).
\end{aligned}
$$

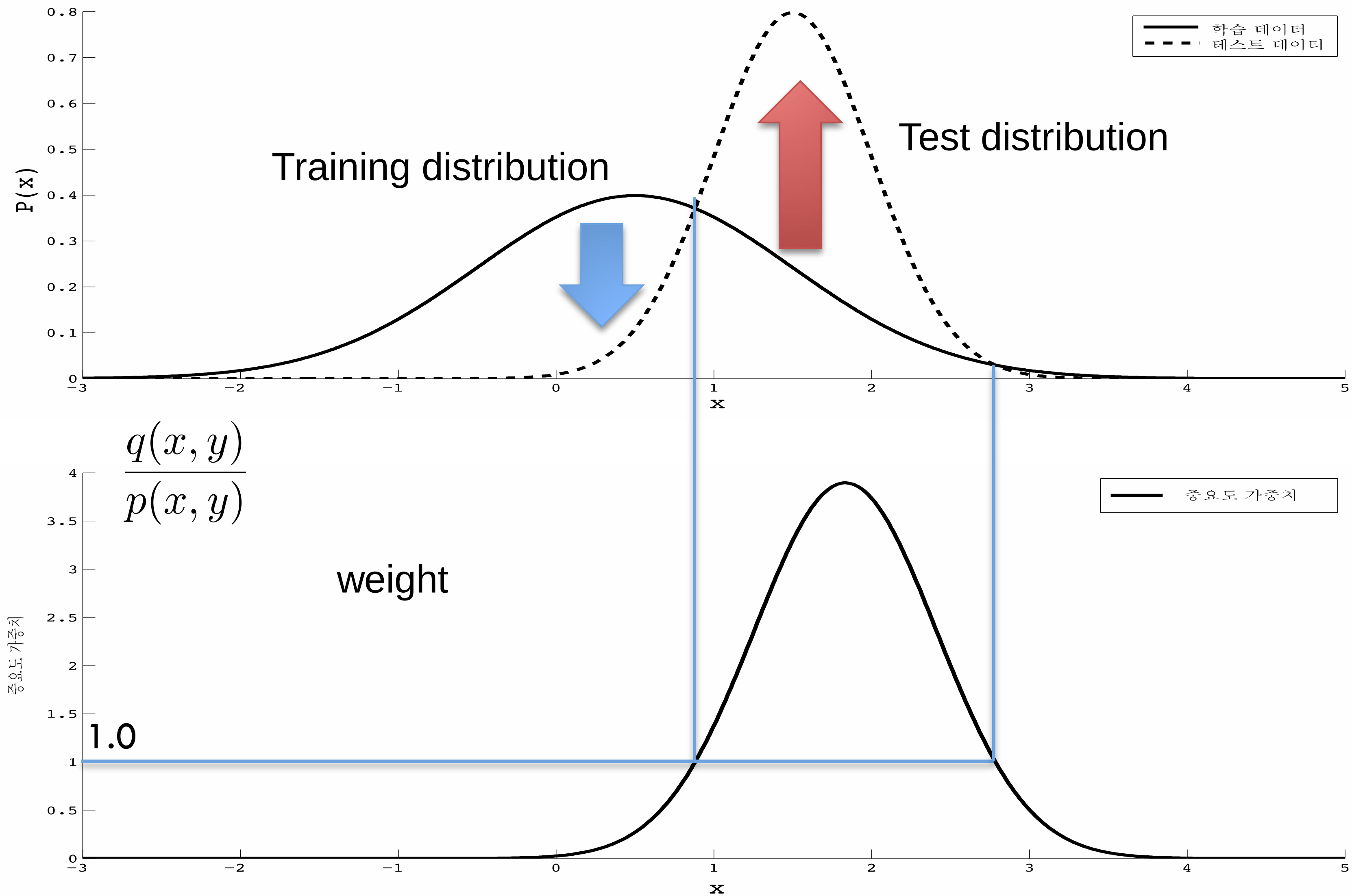
ML without I.I.D

- Problem of machine learning tech. with I.I.D



$$
\begin{aligned}
\theta^* &= \arg\min_{\theta\in\Theta} \sum_{(x,y)\in X\times Y} q(x,y)L(x,y,\theta) \\
&= \arg\min_{\theta\in\Theta} \sum_{(x,y)\in X\times Y} \frac{q(x,y)}{p(x,y)} p(x,y)\cdot L(x,y,\theta) \\
&\approx \arg\min_{\theta\in\Theta} \sum_{(x,y)\in X\times Y} \frac{q(x,y)}{p(x,y)} \hat{p}(x,y)\cdot L(x,y,\theta) \\
&= \arg\min_{\theta\in\Theta} \sum_{i=1}^{|\mathcal{D}_{tr}|} \underline{\frac{q(x,y)}{p(x,y)}} L(x_i,y_i,\theta).
\end{aligned}
$$

Example

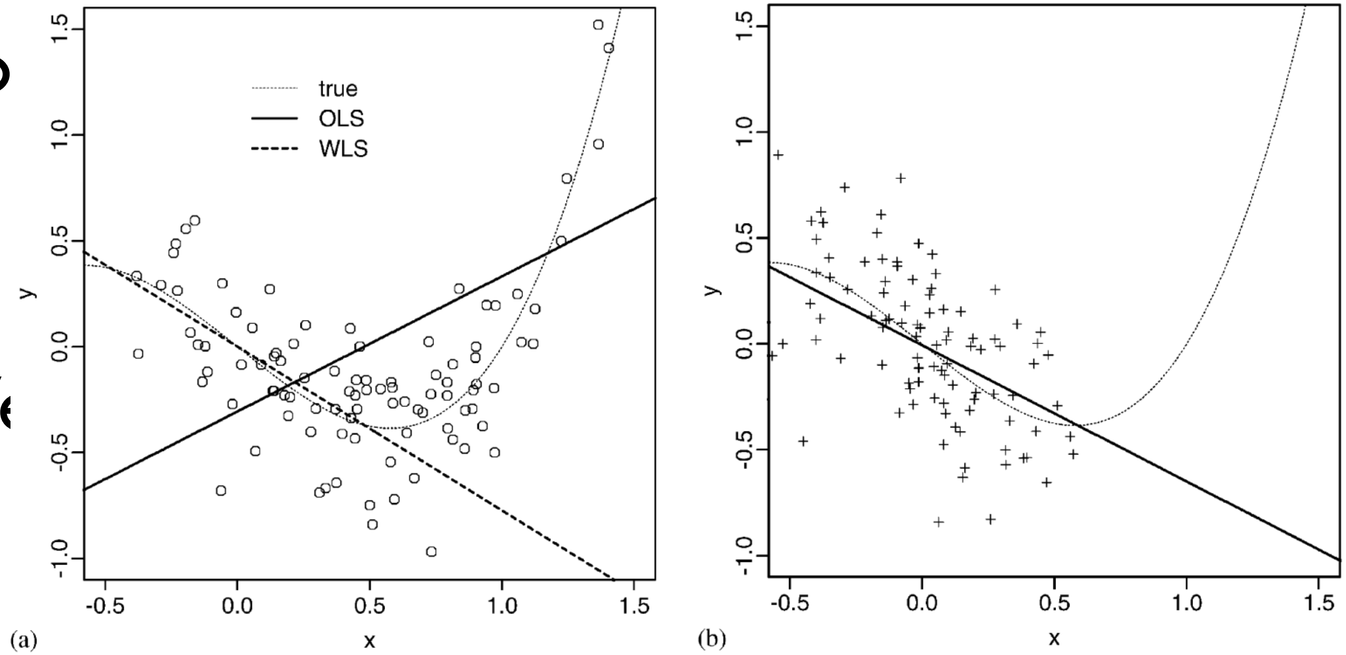


P(x)
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
−3
−2
−1
0
1
2
3
4
5
x
학습 데이터
테스트 데이터
Training distribution
Test distribution
$\frac{q(x,y)}{p(x,y)}$
weight
중요도 가중치
4
3.5
3
2.5
2
1.5
1
0.5
0
1.0
−3
−2
−1
0
1
2
3
4
5
x
중요도 가중치

기존 연구들

Covariate Shift adaptation

- Co
- We



- H. Shimodaira (JSPI 2000)
  - Showing that the weight works on synthetic data

Covariate Shift adaptation

- Weight estimation = density estimation
  - The most difficult task in machine learning
- Alternative way
  - Estimate the weight directly
  - Object?

$$\frac{q(x)}{p(x)} \cdot p(x) = q(x) \qquad w(x;\theta) \approx \frac{q(x)}{p(x)} \qquad w(x;\theta) \cdot p(x) \approx q(x)$$

$$\underset{\theta \in \emptyset}{\text{minimize}} \; dist\{w(x;\theta) \cdot p(x) - q(x)\}$$

Kernel Mean Matching (Huang et al. NIPS 2006)

- Define the distance between training and test distribution as

$$\left\|\mu(\mathrm{P}_{\mathrm{te}}) - \mathbf{E}_{x \sim \mathrm{P}_{\mathrm{tr}}(x)}\left[\beta(x)\Phi(x)\right]\right\|$$

Mean of test data on the kernel space

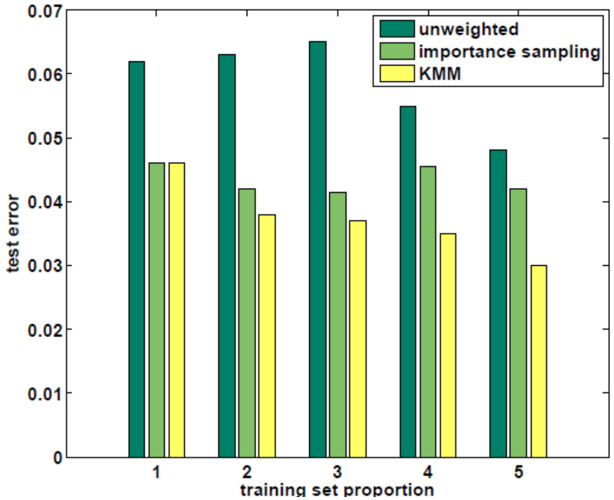
Mean of weighted training data on the kernel space (RKHS)

- After estimating weights

$$\underset{\theta,\xi}{\text{minimize}} \ \frac{1}{2}\|\theta\|^2 + C\sum_{i=1}^{n_{\mathrm{tr}}} \beta_i \xi_i$$

$$\text{subject to } \langle \Phi(x_i^{\mathrm{tr}}, y_i^{\mathrm{tr}}) - \Phi(x_i^{\mathrm{tr}}, y), \theta \rangle \geq 1 - \xi_i / \Delta(y_i^{\mathrm{tr}}, y)$$

$$\text{for all } \ y \in \mathcal{Y}, \text{ and } \xi_i \geq 0.$$



Kullback-Leibler Importance Estimation Procedure (Sugiyama et al. NIPS 2007)

- KLIEP
  - Use Kullback-Leibler divergence to measure the distance

$$\mathrm{KL}[p_{\mathrm{te}}(\boldsymbol{x}) \| \widehat{p}_{\mathrm{te}}(\boldsymbol{x})] = \int_{\mathcal{D}} p_{\mathrm{te}}(\boldsymbol{x}) \log \frac{p_{\mathrm{te}}(\boldsymbol{x})}{\widehat{w}(\boldsymbol{x}) p_{\mathrm{tr}}(\boldsymbol{x})} d\boldsymbol{x}$$

$$\widehat{w}(\boldsymbol{x}) = \sum_{\ell=1}^{b} \alpha_\ell \varphi_\ell(\boldsymbol{x})$$

	1991/3			1991/6			1991/9		
Time	IWKLR $(1.4, 10^{-2})$	KLR $(1.0, 10^{-2})$	GMM (16)	IWKLR $(1.3, 10^{-4})$	KLR $(1.0, 10^{-2})$	GMM (16)	IWKLR $(1.2, 10^{-4})$	KLR $(1.0, 10^{-2})$	GMM (16)
1.5s	**91.0**	88.2	89.7	**91.0**	87.7	90.2	**94.8**	91.7	92.1
3.0s	**95.0**	92.9	94.4	**95.3**	91.1	94.0	**97.9**	96.3	95.0
4.5s	**97.7**	96.1	94.6	**97.4**	93.4	96.1	**98.8**	98.3	95.8
Std	0.34	n/a	n/a	0.37	n/a	n/a	0.35	n/a	n/a

Signal processing 2010

$$\approx \frac{1}{n_{\mathrm{te}}} \sum_{j=1} \log \widehat{w}(\boldsymbol{x}_j) = \frac{1}{n_{\mathrm{te}}} \sum_{j=1} \log \left(\sum_{\ell=1} \alpha_\ell \varphi_\ell(\boldsymbol{x}_j) \right)$$

Surrogate Kernel
(Zhang et al. ICML 2013)

- Consider covariate shift in Hilbert space
- Reproducing kernel Hilbert space
  - Input space for kernel-based methods

- Matching two kernel matrices from training and test data

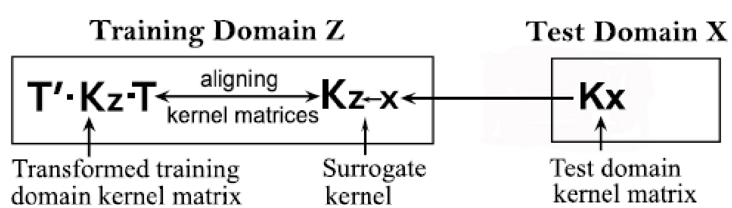
$$K_{tr} \longleftrightarrow K_{te}$$

Impossible!!!
Why?

Surrogate Kernel

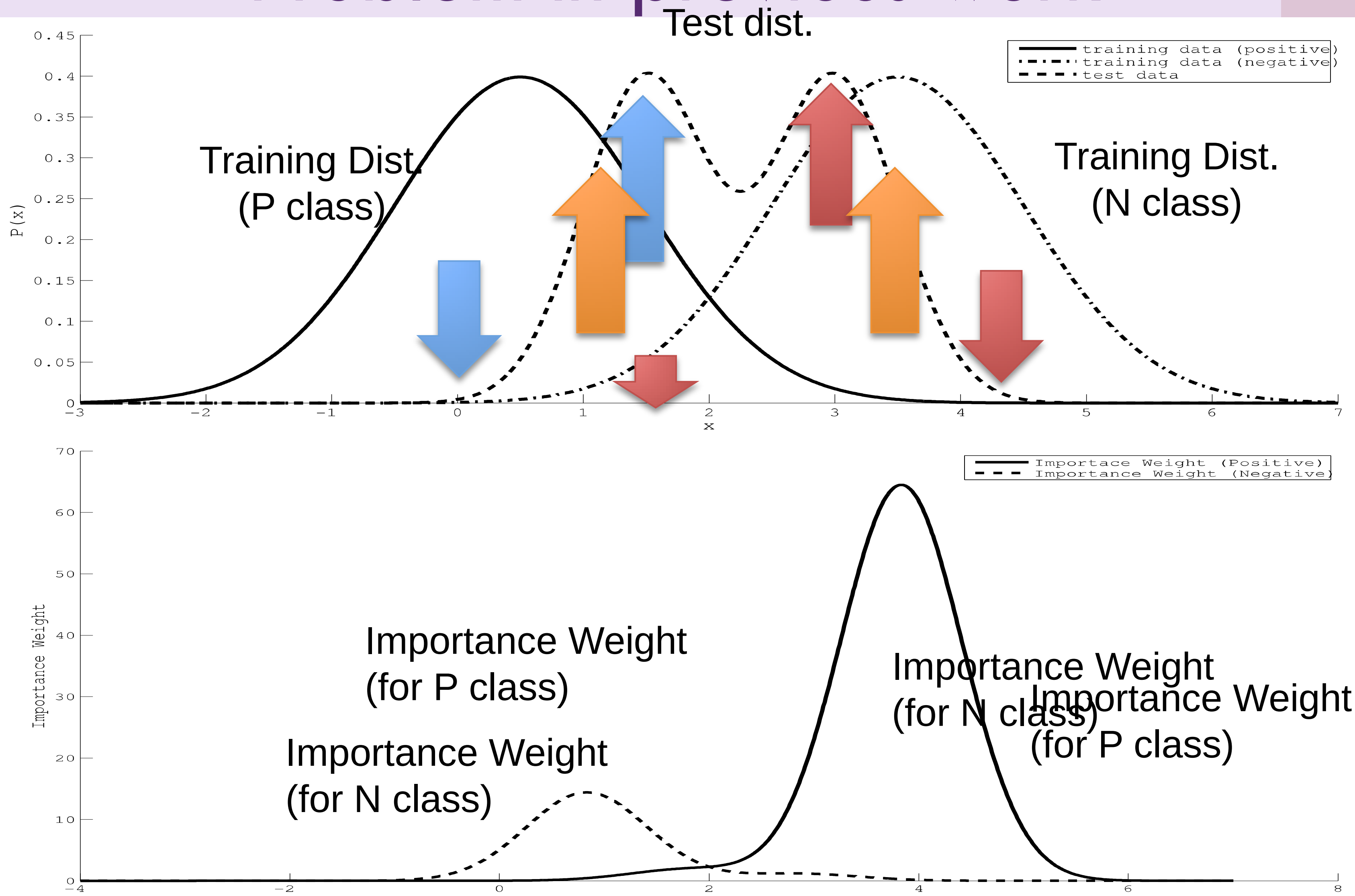
- Surrogate Kernel $K_{tr \leftarrow te}$
  - Mapping test data onto the kernel space spanned by training data
    - Construct the kernel matrix of test data with the key structures of training data (eigenvalues & eigenfunctions)

	SVM	KMM	KLIEP	TSVM	TCA	ours
comp-vs-sci	63.96±1.69	60.96±6.03	64.00±1.66	63.38±5.81	**65.50±6.75**	**65.05±3.19**
comp-vs-talk	64.48±2.08	64.95±2.01	64.92±1.87	68.03±1.72	71.98±4.12	**76.08±1.53**
rec-vs-sci	57.91±3.35	54.75±2.03	58.43±3.52	**62.03±2.39**	56.31±4.62	**62.10±2.32**
rec-vs-talk	62.83±2.52	63.73±3.09	62.51±1.18	65.63±2.64	63.40±3.02	**66.17±2.26**
sci-vs-talk	60.43±2.35	60.15±2.82	59.83±1.63	61.80±1.52	56.51±1.64	**66.00±2.15**



Learning with Local Importance Weight

Problem in previous work



Local Importance Weight

- Intuition behind Local Importance Weight
  - 1. Components in data distributions are corresponding to classes
  - 2. For each class, importance weights for training data are estimated.

$$\theta^* = \arg\min_{\theta \in \Theta} \sum_{i=1}^{|\mathcal{D}_{tr}|} \frac{q(x,y)}{p(x,y)} L(x_i, y_i, \theta).$$

$$\mathbf{W}_x = \{w_{x1}, w_{x2}, \ldots, w_{x|Y|}\}$$

$$\frac{q(x,y)}{p(x,y)} = \frac{q(y|x)}{p(y|x)} \cdot \frac{q(x)}{p(x)}$$

$$w_{iy} = \frac{q(x_i,y)}{p(x_i,y)} = \frac{q(x_i|y)}{p(x_i|y)} \cdot \frac{q(y)}{p(y)}$$

$$\frac{q(x,y)}{p(x,y)} = \frac{q(x|y)}{p(x|y)} \cdot \frac{q(y)}{p(y)}$$

$$= w_{i|y} \cdot w_y.$$

Class labels of test data are needed.

Learning with Local Importance Weight

Model $f(x;\theta_M)$

Class estimation

LIW estimation

The model is optimized under the training dist.

$$w_{i|y} = \sum_{x_j \in \mathcal{D}_{te}, y = y_j = y_i} \alpha_{yj} K(x_i, x_j)$$

$$w_{i|y} = \frac{q(x_i|y)}{p(x_i|y)} \qquad q(x_k|y) = w_{k|y} \cdot p(x_k|y) = \hat{q}(x_k|y)$$

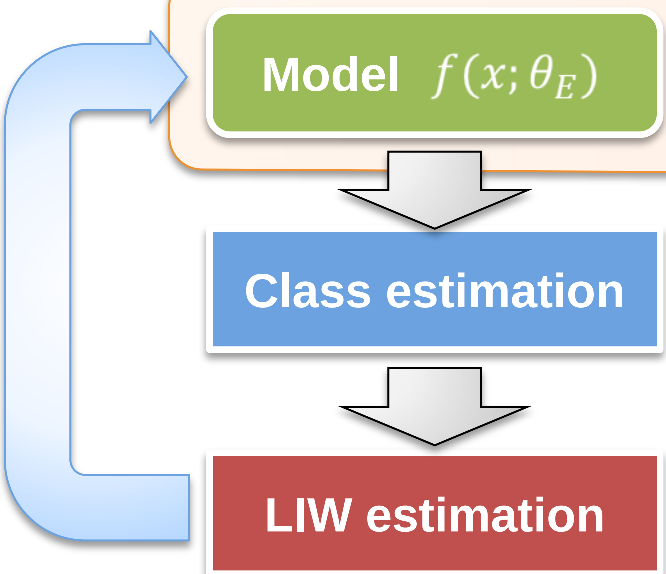
$$KL(q(x_k|y) \| \hat{q}(x_k|y)) = \int_{\mathcal{D}(y)} q(x_k|y) \log \frac{q(x_k|y)}{\hat{q}(x_k|y)} \mathrm{d}x_k$$

$$\operatorname*{arg\,max}_{\boldsymbol{\alpha}_y} \sum_{x_i \in \mathcal{D}_{te}(y)} \frac{1}{|\mathcal{D}_{te}(y)|} \log \left(\sum_{x_j \in \mathcal{D}_{te}(y)} \alpha_{yj} K(x_i, x_j) \right)$$

$$\text{subject to} \sum_{x_i \in \mathcal{D}_{tr}(y)} w_{i|y} = |\mathcal{D}_{tr}(y)|,$$

$$\text{and } \alpha_{yj} \geq 0, \text{ for all } x_j \in \mathcal{D}_{te}(y).$$

Learning with Local Importance Weight



$$\theta_E = \{\theta_M, \theta_W\}$$

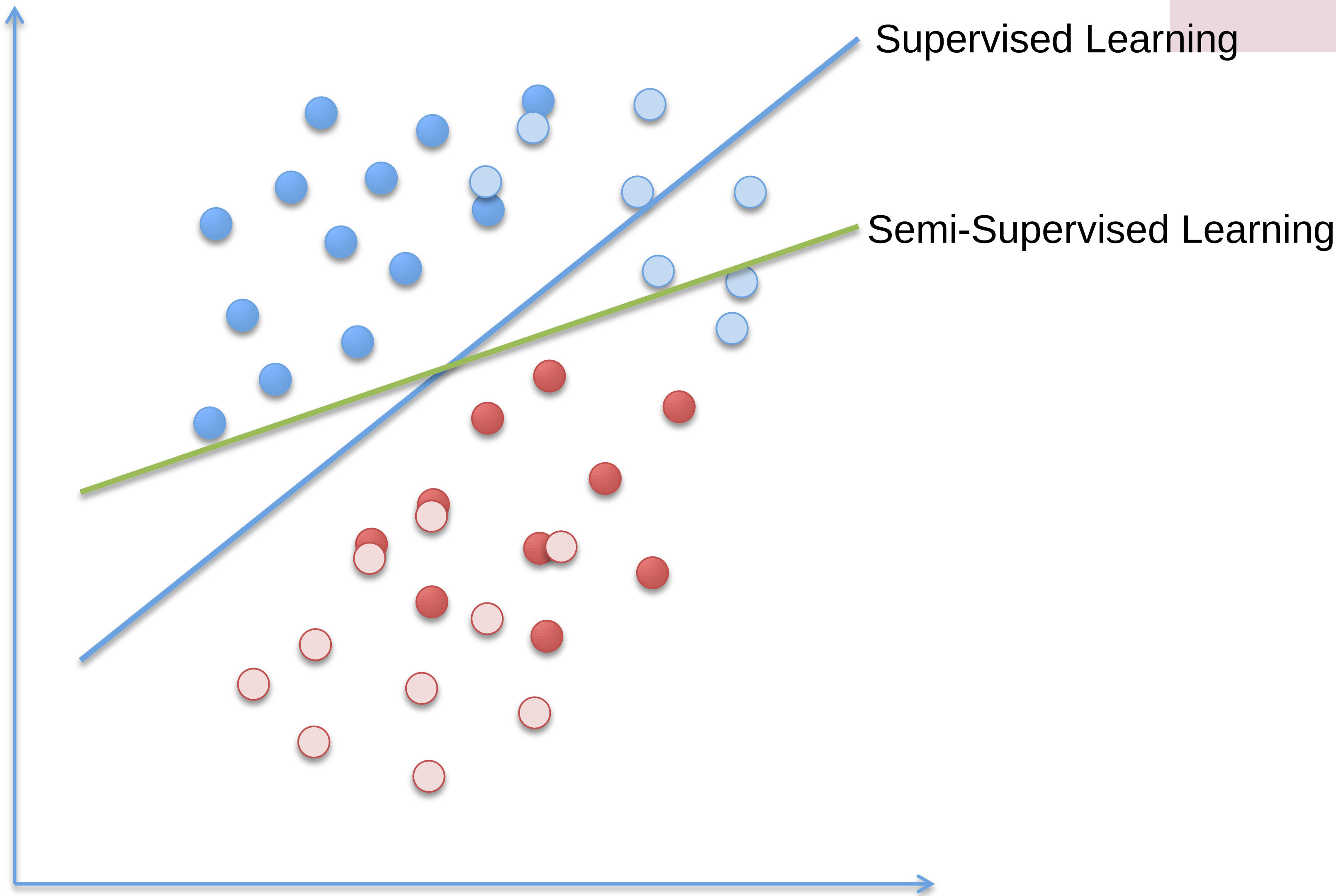
$$p(y_{tr} = y_i | \theta_M, \theta_W) \approx q(y_{tr} = y_i | \theta_M),$$

$$p(x_{tr} | y_{tr} = y_i; \theta_M, \theta_W) \approx q(x_{tr} | y_{tr} = y_i; \theta_M)$$

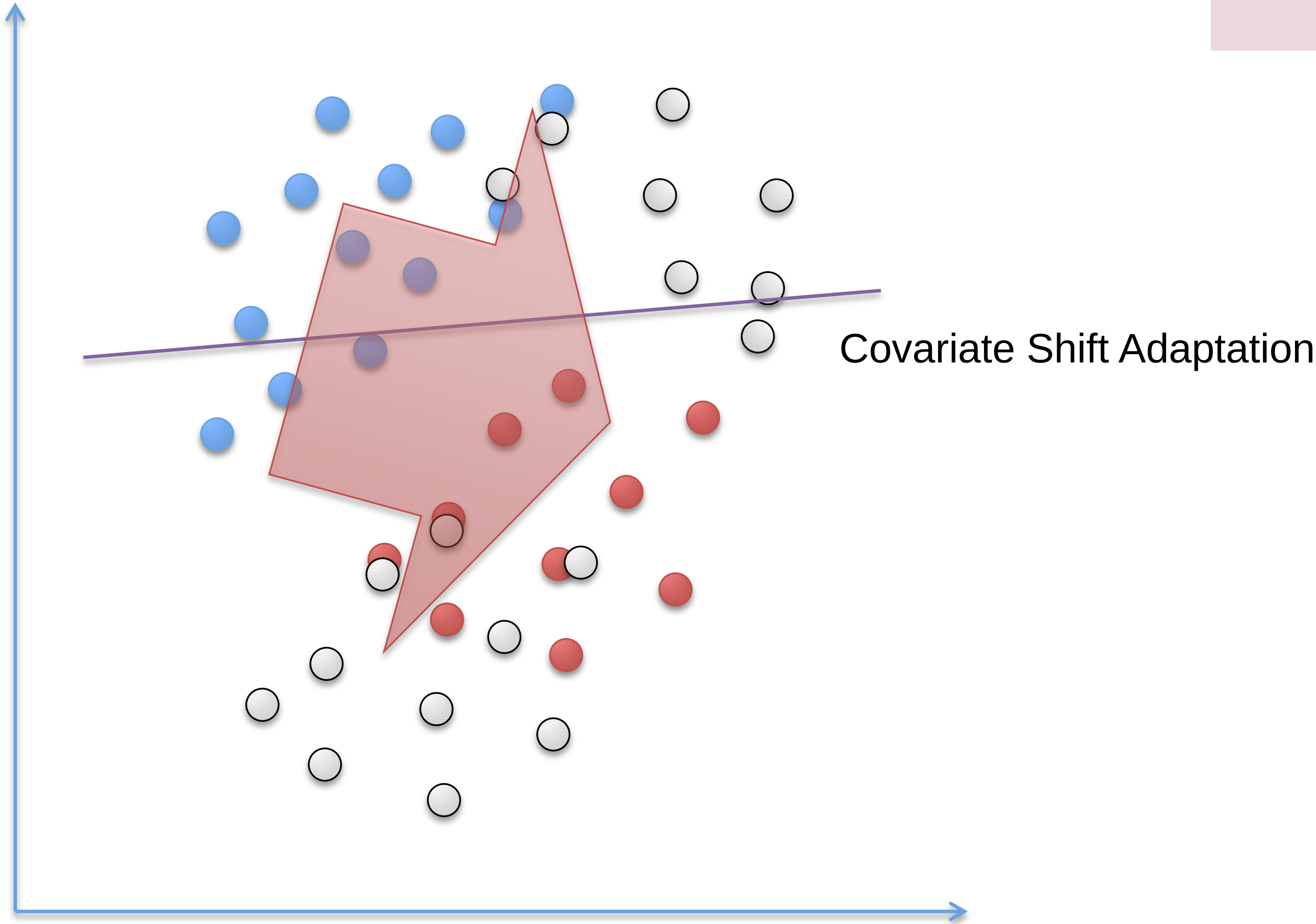
$$\sum_{x_i \in (\mathcal{D}_{tr} \cup \mathcal{D}_{te})} \sum_{j=1}^{|Y|} z_{i_j} \log(q(y_j | \theta_M) \cdot q(x_i | y_j; \theta_M))$$

- Expectation Maximization with Local Importance Weight
  - Maximizes a weighted log likelihood
  - Converged on a local optimum

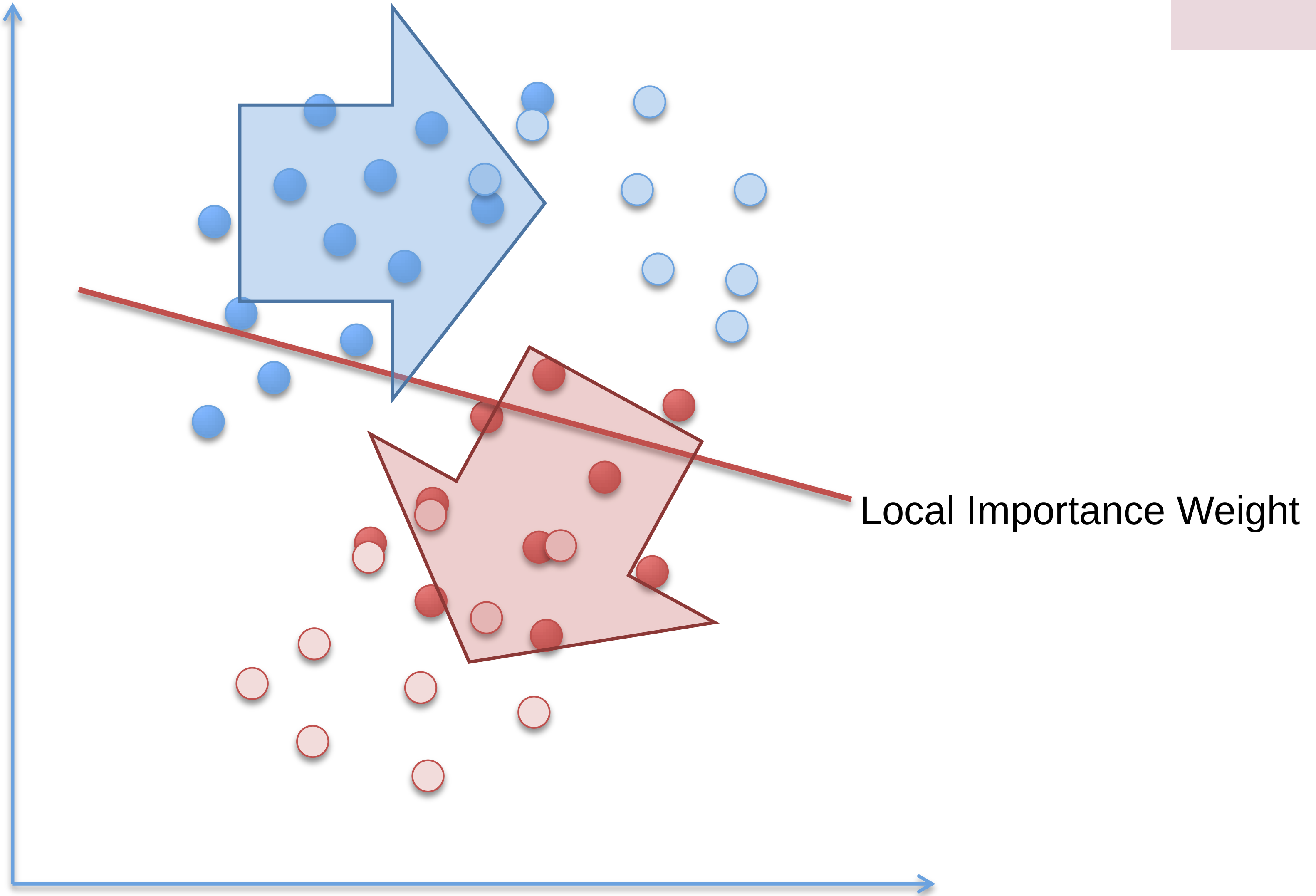
Comparison with a simple example



Comparison with a simple example

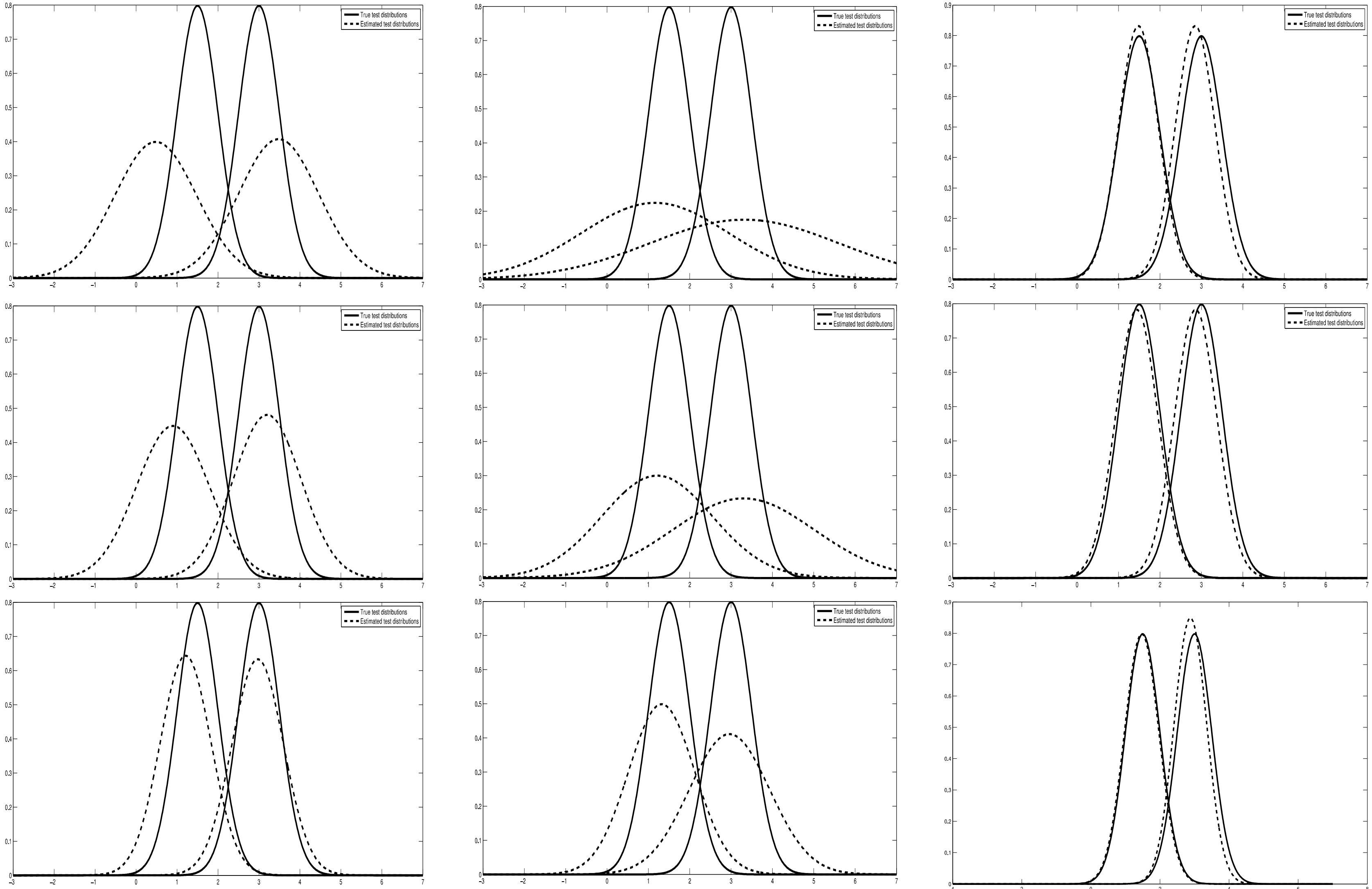


Comparison with a simple example



Experimental results

GM | IW-GM | LIW-GM



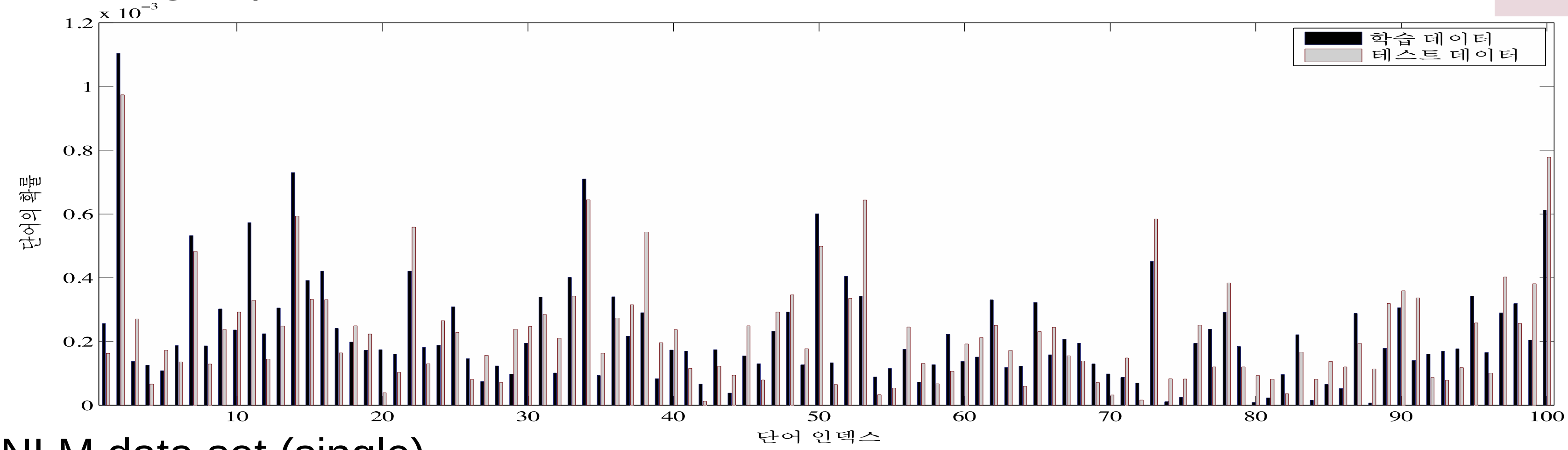
The 2nd Solution & Problem

- Constructing training corpus
  - Training data distribution = test one
- Target: twitter → training corpus(tweets)
- 완벽히 해결?

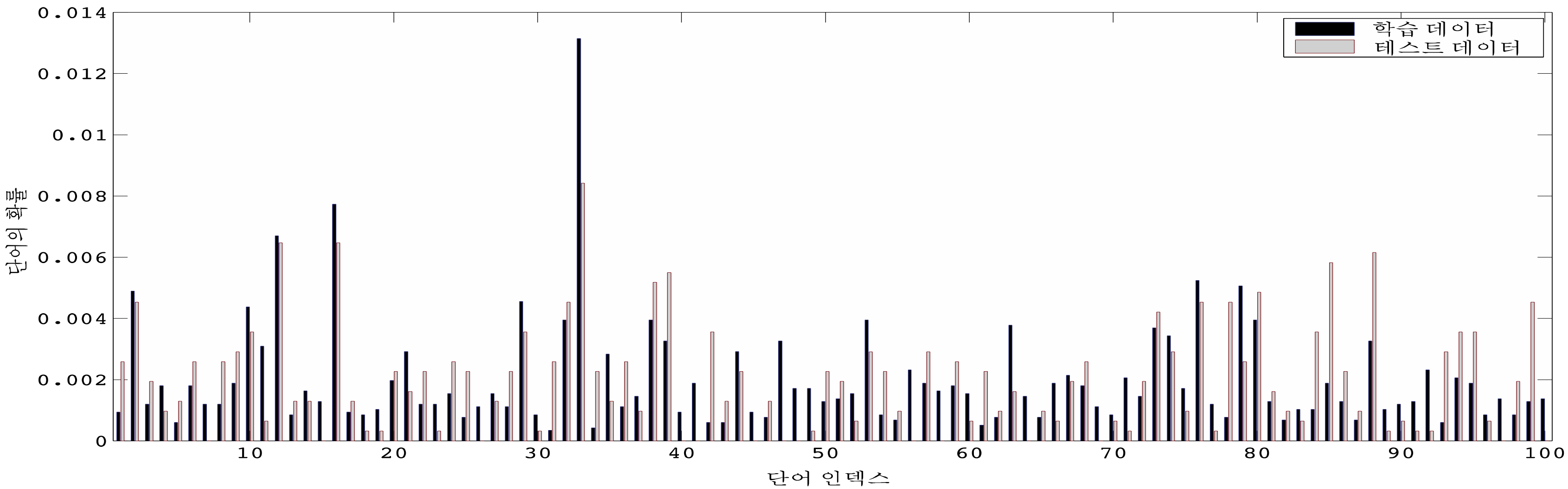
자연어의 특성 상, 완벽히 같은 분포를 가지는 corpus는 불가능
→ 고차원의 단어 공간
→ 불용어, 신조어, … 의 변화

Data distribution

20Newsgroup



NLM data set (single)



Performances

- Classification

실험 방법	인위적인 20newsgroups	20newsgroups
NB	0.72	0.86
EMNB	0.71	**0.88**
NBTC	0.73	0.85
LIW-NB	**0.75**	**0.88**

- Word sense disambiguation

Method	PubMed	NLM
Baseline	0.53	0.62
SVM	0.88	0.79
TSVM	0.88	0.81
IW-SVM	0.87	0.81
LIW-SVM	**0.90**	**0.83**

결론

Distribution difference?

- 현실적으로 피할 수 없는 문제
  - 기계학습 기술의 성능 저하를 야기
- $p(x,y) \neq q(x,y)$
  - Use the weight $q(x,y)/p(x,y)$
- 연구 소재가 많다!

Transfer learning

$$\frac{q(x,y)}{p(x,y)} = \frac{q(x|y)}{p(x|y)} \cdot \frac{q(y)}{p(y)} = \frac{q(y|x)}{p(y|x)} \cdot \frac{q(x)}{p(x)}$$

Class imbalance

Covariate shift
Domain adaptation

Supplement

- 도움이 될 만한 것들
  - Site
    - Sugiyama's Homepage (KLIEP)
      - http://sugiyama-www.cs.titech.ac.jp/~sugi/
    - Transfer learning resources ☆
      - http://www.cse.ust.hk/TL/
  - Book
    - Dataset Shift in Machine Learning
    - Machine Learning in Non-Stationary Environments