Aug. 21, 2021

Rethinking Importance Weighting for Transfer Learning

Masashi Sugiyama

RIKEN Center for Advanced Intelligence Project/
The University of Tokyo

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

東京大学
THE UNIVERSITY OF TOKYO

Problem of Transfer Learning

- **Given:** Training data

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

$\boldsymbol{x}$: Input

y : Output

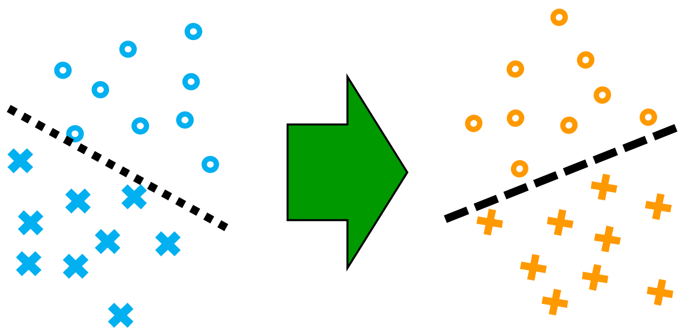
- **Goal**: Train a predictor $y = f(\boldsymbol{x})$ that works well in the test domain.

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

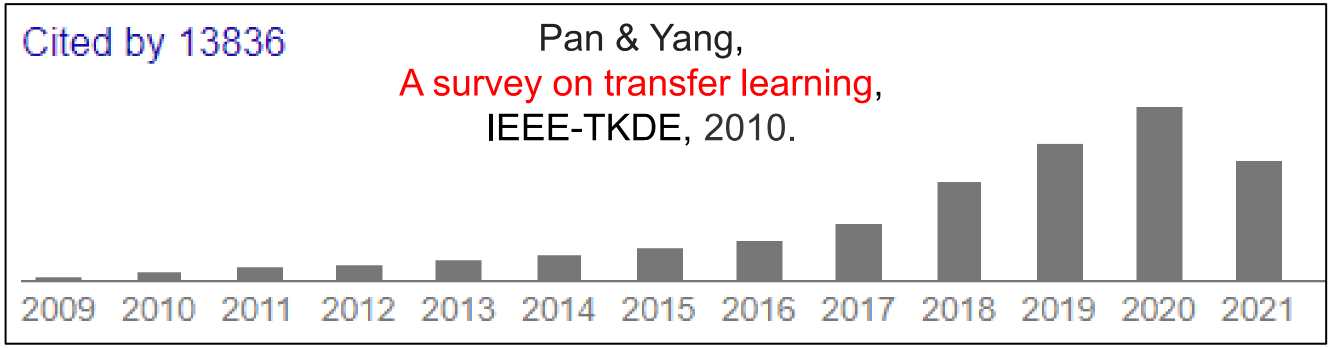
ℓ : loss function

- **Challenge**: Overcome changing distributions!

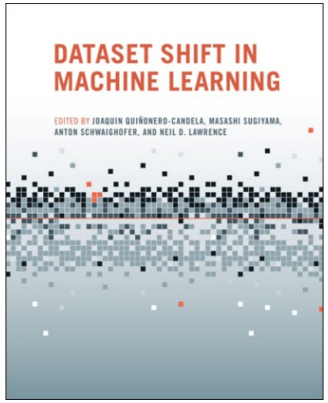
$$p_{\mathrm{tr}}(\boldsymbol{x}, y) \neq p_{\mathrm{te}}(\boldsymbol{x}, y)$$



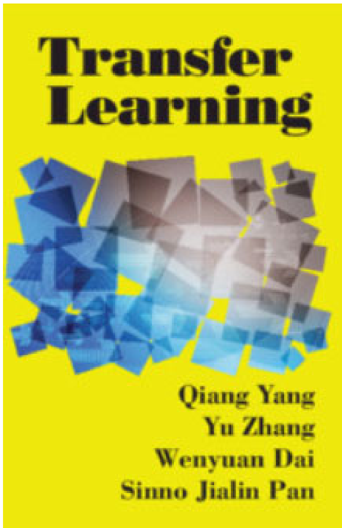
Transfer Learning Has been a Hot Topic for Many Years!



Pan & Yang,
A survey on transfer learning,
IEEE-TKDE, 2010.

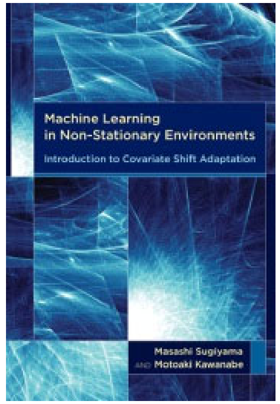


Quiñonero-Candela, Sugiyama,
Schwaighofe & Lawrence (Eds.),
Dataset Shift in Machine Learning,
MIT Press, 2009.



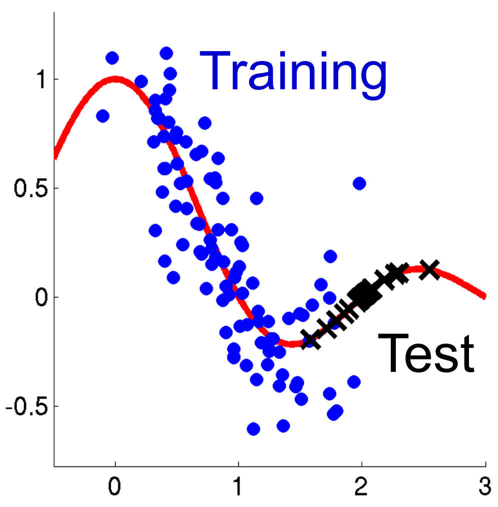
Yang, Zhang, Dai & Pan,
Transfer Learning,
Cambridge University Press, 2020

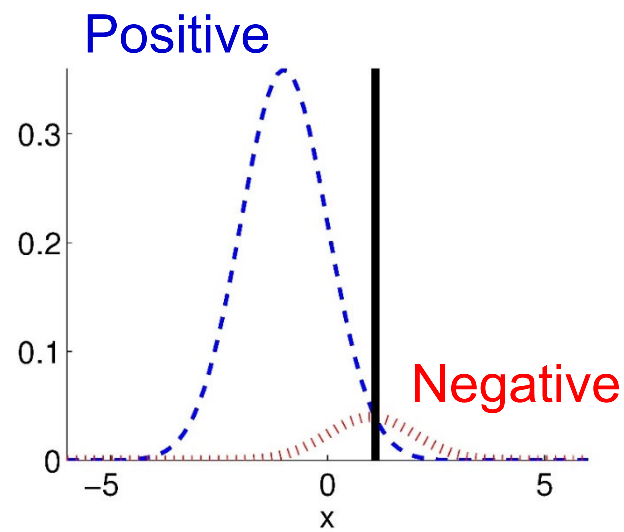
Sugiyama & Kawanabe,
Machine Learning
in Non-Stationary Environments,
MIT Press, 2012

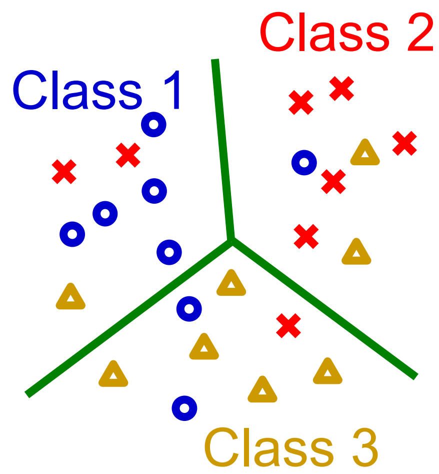


Various Scenarios

- Full-distribution shift: $p_{\mathrm{tr}}(\boldsymbol{x}, y) \neq p_{\mathrm{te}}(\boldsymbol{x}, y)$
- Covariate shift: $p_{\mathrm{tr}}(\boldsymbol{x}) \neq p_{\mathrm{te}}(\boldsymbol{x})$
- Class-prior/target shift: $p_{\mathrm{tr}}(y) \neq p_{\mathrm{te}}(y)$
- Output noise: $p_{\mathrm{tr}}(y|\boldsymbol{x}) \neq p_{\mathrm{te}}(y|\boldsymbol{x})$
- Class-conditional shift: $p_{\mathrm{tr}}(\boldsymbol{x}|y) \neq p_{\mathrm{te}}(\boldsymbol{x}|y)$







Organization of My Talk

1. Introduction
2. Classical results
   A) Importance weighting
   B) Adaptive importance weighting
3. Recent results
   A) Joint upper-bound minimization
   B) Dynamic importance weighting

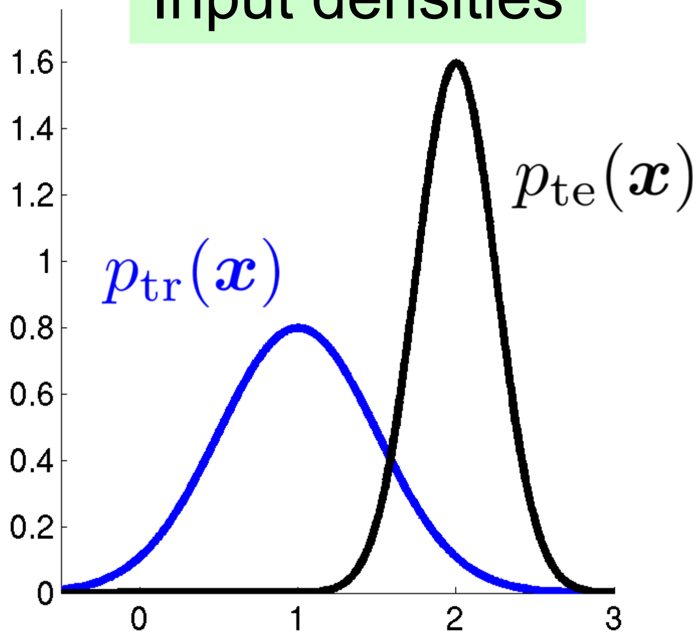
Regression under Covariate Shift

- Covariate shift: Shimodaira (JSPI2000)
  - Training and test input distributions are different:

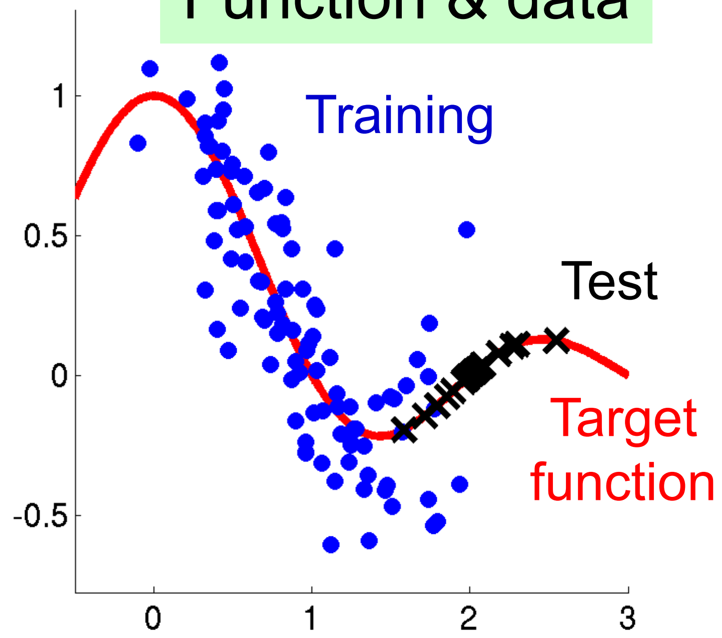
$$p_{\mathrm{tr}}(\boldsymbol{x}) \neq p_{\mathrm{te}}(\boldsymbol{x})$$

  - But the output-given-input distribution remains unchanged: $p_{\mathrm{tr}}(y|\boldsymbol{x}) = p_{\mathrm{te}}(y|\boldsymbol{x}) = p(y|\boldsymbol{x})$

Input densities



Function & data

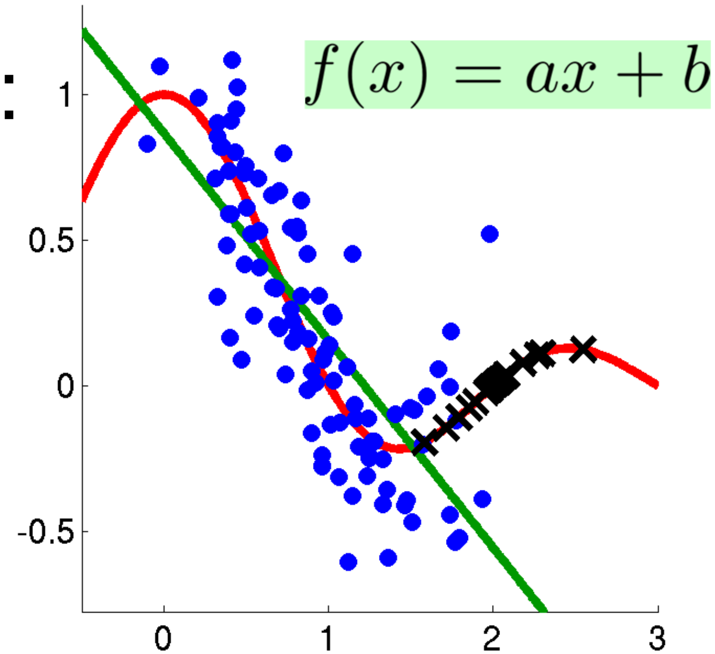


Empirical Risk Minimization (ERM)

$$\min_f \left[\sum_{i=1}^{n_{\mathrm{tr}}} \ell(f(\boldsymbol{x}_i^{\mathrm{tr}}), y_i^{\mathrm{tr}}) \right]$$

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

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



$$\frac{1}{n_{\mathrm{tr}}} \sum_{i=1}^{n_{\mathrm{tr}}} \ell(f(\boldsymbol{x}_i^{\mathrm{tr}}), y_i^{\mathrm{tr}}) \overset{n_{\mathrm{tr}} \rightarrow \infty}{\longrightarrow} \mathbb{E}_{p_{\mathrm{tr}}(\boldsymbol{x}, y)}[\ell(f(\boldsymbol{x}), y)] \neq R(f)$$

$$p_{\mathrm{tr}}(\boldsymbol{x}) \neq p_{\mathrm{te}}(\boldsymbol{x})$$

Organization of My Talk

1. Introduction
2. Classical results
   A) Importance weighting
   B) Adaptive importance weighting
3. Recent results
   A) Joint upper-bound minimization
   B) Dynamic importance weighting

Importance-Weighted ERM (IWERM)

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

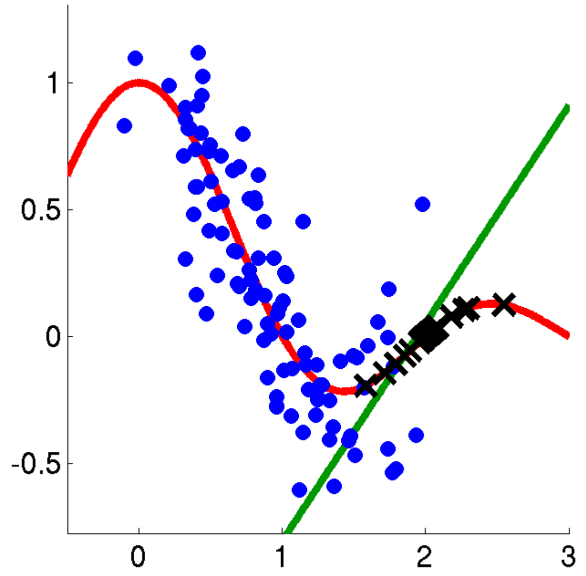
- IWERM is consistent even under covariate shift.

$$\frac{1}{n_{\mathrm{tr}}} \sum_{i=1}^{n_{\mathrm{tr}}} \frac{p_{\mathrm{te}}(\boldsymbol{x}_i^{\mathrm{tr}})}{p_{\mathrm{tr}}(\boldsymbol{x}_i^{\mathrm{tr}})} \ell(f(\boldsymbol{x}_i^{\mathrm{tr}}), y_i^{\mathrm{tr}})$$

$$\overset{n_{\mathrm{tr}} \to \infty}{\longrightarrow} \mathbb{E}_{p_{\mathrm{tr}}(\boldsymbol{x}, y)} \left[\frac{p_{\mathrm{te}}(\boldsymbol{x})}{p_{\mathrm{tr}}(\boldsymbol{x})} \ell(f(\boldsymbol{x}), y) \right]$$

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

$$f(x) = ax + b$$

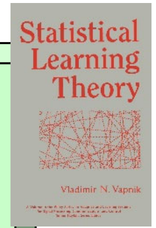


- How can we know the importance weight?

Importance Weight Estimation

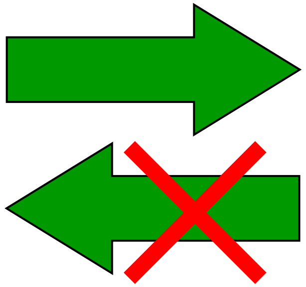
Vapnik's principle: Vapnik (Wiley, 1998)

When solving a problem of interest, one should not solve a more general problem as an intermediate step



Knowing densities

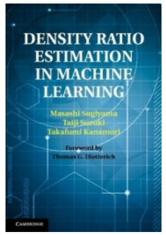
$$p_{\mathrm{te}}(\boldsymbol{x}), p_{\mathrm{tr}}(\boldsymbol{x})$$



Knowing ratio

$$r^*(\boldsymbol{x}) = \frac{p_{\mathrm{te}}(\boldsymbol{x})}{p_{\mathrm{tr}}(\boldsymbol{x})}$$

- Estimating the density ratio is substantially easier than estimating both the densities!
- Various direct density-ratio estimators were developed.



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

Least-Squares Importance Fitting (LSIF)

Kanamori, Hido & Sugiyama (JMLR2009)

- Given training and test input data:

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

- Directly fit a model r to $r^*(\boldsymbol{x}) = \dfrac{p_{\mathrm{te}}(\boldsymbol{x})}{p_{\mathrm{tr}}(\boldsymbol{x})}$ by LS:

$$\min_r Q(r) \qquad Q(r) = \int \Big(r(\boldsymbol{x}) - r^*(\boldsymbol{x})\Big)^2 p_{\mathrm{tr}}(\boldsymbol{x})\mathrm{d}\boldsymbol{x}$$

  - Empirical approximation:

$$Q(r) = \int r(\boldsymbol{x})^2 p_{\mathrm{tr}}(\boldsymbol{x})\mathrm{d}\boldsymbol{x} - 2\int r(\boldsymbol{x}) p_{\mathrm{te}}(\boldsymbol{x})\mathrm{d}\boldsymbol{x} + C$$

$$\approx \frac{1}{n_{\mathrm{tr}}}\sum_{i=1}^{n_{\mathrm{tr}}} r(\boldsymbol{x}_i^{\mathrm{tr}})^2 - \frac{2}{n_{\mathrm{te}}}\sum_{j=1}^{n_{\mathrm{te}}} r(\boldsymbol{x}_j^{\mathrm{te}}) + C$$

Organization of My Talk

1. Introduction
2. Classical results
   A) Importance weighting
   B) Adaptive importance weighting
3. Recent results
   A) Joint upper-bound minimization
   B) Dynamic importance weighting

Bias-Variance Trade-Off

- Importance-weighted empirical risk estimator

$$\frac{1}{n_{\mathrm{tr}}}\sum_{i=1}^{n_{\mathrm{tr}}}\frac{p_{\mathrm{te}}(\boldsymbol{x}_i^{\mathrm{tr}})}{p_{\mathrm{tr}}(\boldsymbol{x}_i^{\mathrm{tr}})}\ell(f(\boldsymbol{x}_i^{\mathrm{tr}}),y_i^{\mathrm{tr}})$$

  has no bias, but has large variance.

- The ordinary empirical risk estimator

$$\frac{1}{n_{\mathrm{tr}}}\sum_{i=1}^{n_{\mathrm{tr}}}\ell(f(\boldsymbol{x}_i^{\mathrm{tr}}),y_i^{\mathrm{tr}})$$

  has small variance (statistically efficient), but has large bias.

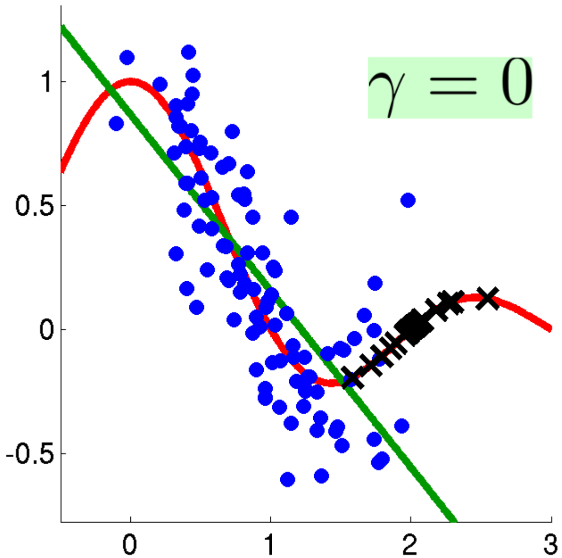
- How can we control the bias-variance trade-off?

Flattened Importance Weighting

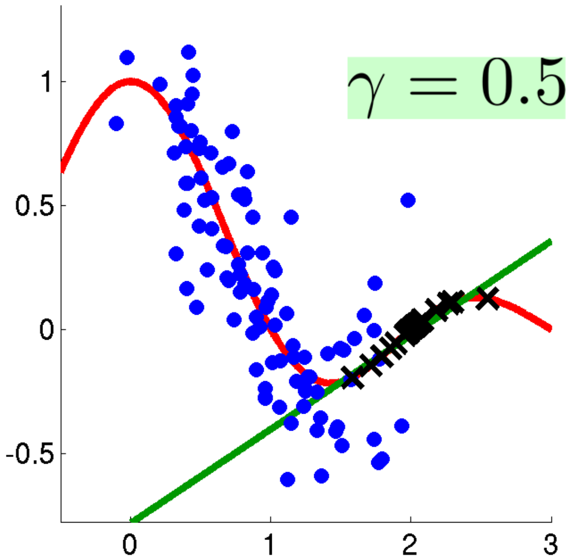
$$\min_f \left[\sum_{i=1}^{n_{\mathrm{tr}}} \left(\frac{p_{\mathrm{te}}(\boldsymbol{x}_i^{\mathrm{tr}})}{p_{\mathrm{tr}}(\boldsymbol{x}_i^{\mathrm{tr}})} \right)^{\gamma} \ell(f(\boldsymbol{x}_i^{\mathrm{tr}}), y_i^{\mathrm{tr}}) \right]$$

Shimodaira (JSPI2000)

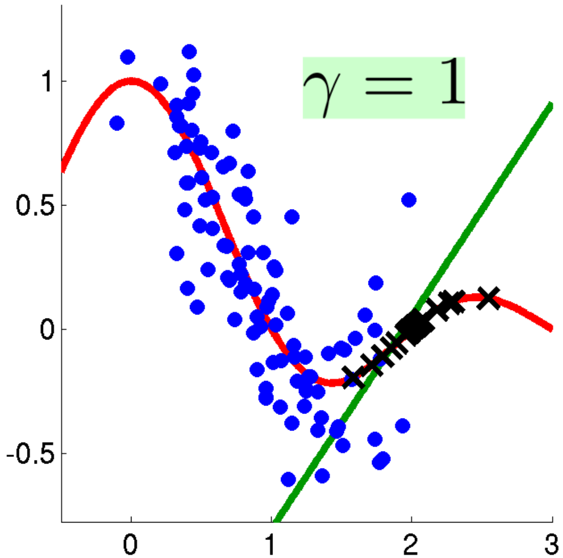
$0 \leq \gamma \leq 1$



Large bias, small variance



(Intermediate)



Small bias, large variance

- Flattening factor γ may be chosen by
  - Importance-weighted Akaike information criterion — Shimodaira (JSPI2000)
  - Importance-weighted cross-validation — Sugiyama, Krauledat & Müller (JMLR2007)

Relative Importance Weighting

- Even with direct methods, reliably estimating the importance weight is hard:
  - $r^*(\boldsymbol{x})$ could be highly fluctuated.

$$r^*(\boldsymbol{x}) = \frac{p_{\mathrm{te}}(\boldsymbol{x})}{p_{\mathrm{tr}}(\boldsymbol{x})}$$

- Thus, flattening unreliable importance estimator $\widehat{r}(\boldsymbol{x})$ by power factor γ is also unreliable.

$$\min_f \left[\sum_{i=1}^{n_{\mathrm{tr}}} \widehat{r}(\boldsymbol{x}_i^{\mathrm{tr}})^{\gamma} \ell(f(\boldsymbol{x}_i^{\mathrm{tr}}), y_i^{\mathrm{tr}}) \right]$$

- Let's use relative importance weight:

Yamada, Suzuki, Kanamori, Hachiya & Sugiyama (NIPS2011, NeCo2013)

$$r_{\beta}(\boldsymbol{x}) = \frac{p_{\mathrm{te}}(\boldsymbol{x})}{\beta p_{\mathrm{tr}}(\boldsymbol{x}) + (1-\beta) p_{\mathrm{te}}(\boldsymbol{x})} \qquad 0 \le \beta \le 1$$

  - Directly estimable for each β by relative LSIF.

Organization of My Talk

1. Introduction
2. Classical results
   - A) Importance weighting
   - B) Adaptive importance weighting
3. Recent results
   - A) Joint upper-bound minimization
   - B) Dynamic importance weighting

One-Step Adaptation

- The classical approaches are two steps:
  1. Weight estimation (e.g., LSIF):

     $$\widehat{r} = \underset{r}{\operatorname{argmin}}\, \mathbb{E}_{p_{\mathrm{tr}}(\boldsymbol{x})}[(r(\boldsymbol{x}) - r^*(\boldsymbol{x}))^2]$$

  2. Weighted predictor training (e.g., IWERM):

     $$\widehat{f} = \underset{f}{\operatorname{argmin}}\, \mathbb{E}_{p_{\mathrm{tr}}(\boldsymbol{x},y)}[\widehat{r}(\boldsymbol{x})\ell(f(\boldsymbol{x}), y)]$$

- Can we integrate these two steps?

Organization of My Talk

1. Introduction
2. Classical results
   - A) Importance weighting
   - B) Adaptive importance weighting
3. Recent results
   - A) Joint upper-bound minimization
   - B) Dynamic importance weighting

Risk Upper-Bounding

Zhang, Yamane, Lu & Sugiyama (ACML2020, SNCS2021)

- For $\ell \leq 1, \ell' \geq \ell, r \geq 0$, $\frac{1}{2} R_\ell(f)^2 \leq J_{\ell'}(r, f)$:

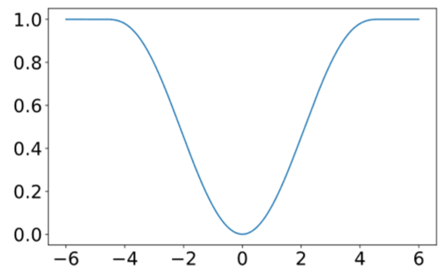
$$R_\ell(f) = \mathbb{E}_{p_{\mathrm{te}}(\boldsymbol{x},y)}[\ell(f(\boldsymbol{x}), y)]$$

$$J_{\ell'}(r, f) = \left(\mathbb{E}_{p_{\mathrm{tr}}(\boldsymbol{x},y)}[r(\boldsymbol{x})\ell'(f(\boldsymbol{x}), y)]\right)^2 \quad \leftarrow \text{IWERM}$$

$$+\mathbb{E}_{p_{\mathrm{tr}}(\boldsymbol{x})}[(r(\boldsymbol{x}) - r^*(\boldsymbol{x}))^2] \quad \leftarrow \text{LSIF}$$

- In terms of this upper-bound minimization, LSIF followed by IWERM is sub-optimal:
  - Let's directly minimize the upper bound w.r.t. r, f !
- $\ell \leq 1, \ell' \geq \ell$ is satisfied by
  - ℓ : 0/1, ℓ' :hinge/softmax cross-entropy (classification)
  - ℓ : Tukey, ℓ' : squared (regression)

Tukey loss



Theoretical Analysis

- Let $\widehat{f} = \mathop{\mathrm{argmin}}_{f} \min_{r} \widehat{J}_{\ell'}(r, f)$ be an empirical solution.

$$\widehat{J}_{\ell'}(r, f) = \left(\frac{1}{n_{\mathrm{tr}}}\sum_{i=1}^{n_{\mathrm{tr}}} r(\boldsymbol{x}_i^{\mathrm{tr}})\ell'(f(\boldsymbol{x}_i^{\mathrm{tr}}), y_i^{\mathrm{tr}})\right)^2 + \left(\frac{1}{n_{\mathrm{tr}}}\sum_{i=1}^{n_{\mathrm{tr}}} r(\boldsymbol{x}_i^{\mathrm{tr}})^2 - \frac{2}{n_{\mathrm{te}}}\sum_{j=1}^{n_{\mathrm{te}}} r(\boldsymbol{x}_j^{\mathrm{tr}}) + C\right)$$

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

- Under some mild conditions, the risk of the empirical solution is upper-bounded as

$$R_\ell(\widehat{f}) \leq \sqrt{2}\min_{f} R_{\ell'}(f) + \mathcal{O}_p(n_{\mathrm{tr}}^{-1/4} + n_{\mathrm{te}}^{-1/4})$$

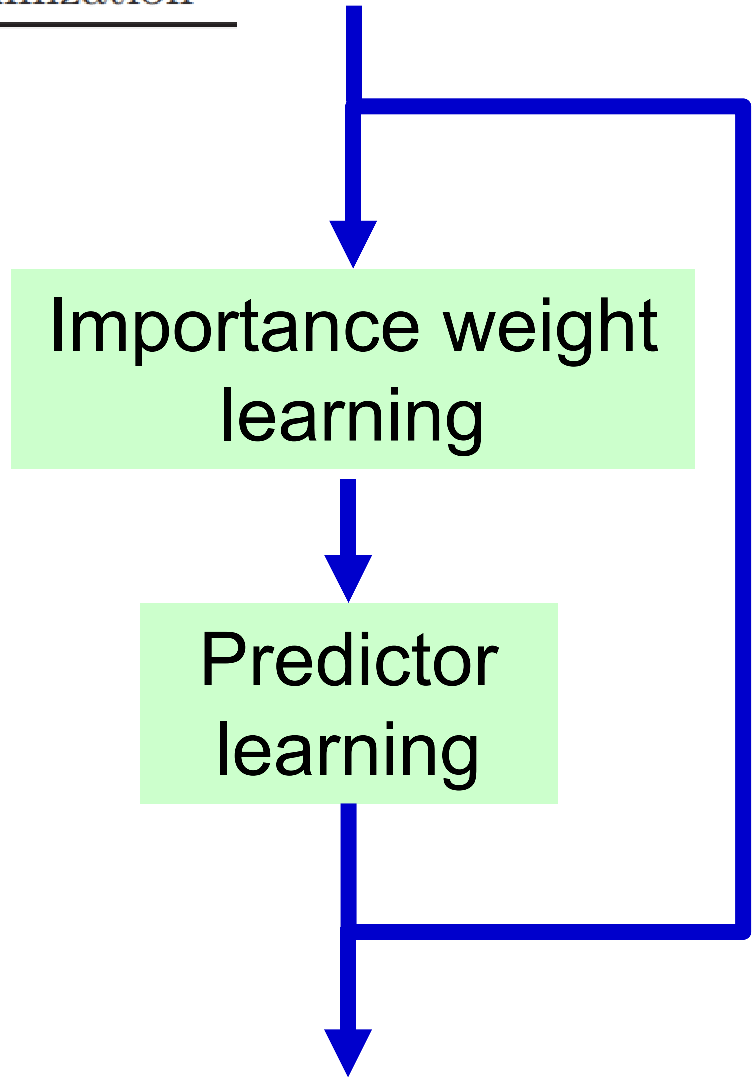
$$R_\ell(\widehat{f}) = \mathbb{E}_{p_{\mathrm{te}}(\boldsymbol{x},y)}[\ell(\widehat{f}(\boldsymbol{x}), y)]$$

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

Practical Implementation

Algorithm 2 Gradient-based Alternating Minimization

$\mathcal{Z}^{\mathrm{tr}}, \mathcal{X}^{\mathrm{te}} \leftarrow \{(\boldsymbol{x}_i^{\mathrm{tr}}, y_i^{\mathrm{tr}})\}_{i=1}^{n_{\mathrm{tr}}}, \{\boldsymbol{x}_i^{\mathrm{te}}\}_{i=1}^{n_{\mathrm{te}}}$
$\mathcal{A} \leftarrow$ a gradient-based optimizer
$\boldsymbol{f} \leftarrow$ an arbitrary classifier
for round $= 0, 1, \ldots, \mathrm{numOfRounds} - 1$ **do**
  for epoch $= 0, 1, \ldots, \mathrm{numOfEpochsForG} - 1$ **do**
    for $i = 0, 1, \ldots, \mathrm{numOfMiniBatches} - 1$ **do**
      $\mathcal{Z}_i^{\mathrm{tr}}, \mathcal{X}_i^{\mathrm{te}} \leftarrow \mathrm{sampleMiniBatch}(\mathcal{Z}^{\mathrm{tr}}, \mathcal{X}^{\mathrm{te}})$
      $g \leftarrow \mathcal{A}(g, \nabla_g \widehat{J}_{\mathrm{UB}}(\boldsymbol{f}, g; \mathcal{Z}_i^{\mathrm{tr}} \cup \mathcal{X}_i^{\mathrm{te}}))$
    end for
  end for
  for epoch $= 0, 1, \ldots, \mathrm{numOfEpochsForF} - 1$ **do**
    for $i = 0, 1, \ldots, \mathrm{numOfMiniBatches} - 1$ **do**
      $\mathcal{Z}_i^{\mathrm{tr}} \leftarrow \mathrm{sampleMiniBatch}(\mathcal{Z}^{\mathrm{tr}})$
      $w_j \leftarrow \max(g(\boldsymbol{x}_j), 0), \ \forall(\boldsymbol{x}_j, \cdot) \in \mathcal{Z}_i^{\mathrm{tr}}$
      $w_j \leftarrow w_j / \sum_j w_j, \ \forall j$
      $L_i \leftarrow \sum_{(\boldsymbol{x}_j, y_j) \in \mathcal{Z}_i^{\mathrm{tr}}} w_j \ell_{\mathrm{UB}}(\boldsymbol{f}(\boldsymbol{x}_j), y_j)$
      $\boldsymbol{f} \leftarrow \mathcal{A}(\boldsymbol{f}, \nabla_{\boldsymbol{f}} L_i)$
    end for
  end for
end for



Experimental Evaluation

Table 3 Mean test classification accuracy averaged over 5 trials on image datasets with neural networks. The numbers in the brackets are the standard deviations. For each dataset, the best method and comparable ones based on the *paired t-test* at the significance level 5% are described in bold face.

Dataset	Shift Level (a, b)	ERM	EIWERM	RIWERM	one-step
Fashion-MNIST	(2, 4)	81.71(0.17)	84.02(0.18)	84.12(0.06)	**85.07(0.08)**
	(2, 5)	72.52(0.54)	76.68(0.27)	77.43(0.29)	**78.83(0.20)**
	(2, 6)	60.10(0.34)	65.73(0.34)	66.73(0.55)	**69.23(0.25)**
Kuzushiji-MNIST	(2, 4)	77.09(0.18)	80.92(0.32)	81.17(0.24)	**82.45(0.12)**
	(2, 5)	65.06(0.26)	71.02(0.50)	72.16(0.19)	**74.03(0.16)**
	(2, 6)	51.24(0.30)	58.78(0.38)	60.14(0.93)	**62.70(0.55)**

Shimodaira (JSPI2000)

Yamada, Suzuki, Kanamori, Hachiya & Sugiyama (NIPS2011, NeCo2013)

Organization of My Talk

1. Introduction
2. Classical results
   - A) Importance weighting
   - B) Adaptive importance weighting
3. Recent results
   - A) Joint upper-bound minimization
   - B) Dynamic importance weighting

Dynamic Importance Weighting

Fang, Lu, Niu & Sugiyama (NeurIPS2020)

- Deep learning adopts iterative optimization.

$$f \leftarrow f - \eta \nabla \widehat{R}(f)$$

$\eta > 0$: Learning rate

- Let's learn
  - Importance weight r
  - predictor f

dynamically in the mini-batch-wise manner.

Mini-Batch-Wise Loss Matching

- Suppose we are given
  - (Large) training data: $\{(\boldsymbol{x}_i^{\mathrm{tr}}, y_i^{\mathrm{tr}})\}_{i=1}^{n_{\mathrm{tr}}} \overset{\mathrm{i.i.d.}}{\sim} p_{\mathrm{tr}}(\boldsymbol{x}, y)$
  - (Small) test data: $\{(\boldsymbol{x}_i^{\mathrm{te}}, y_i^{\mathrm{te}})\}_{i=1}^{n_{\mathrm{te}}} \overset{\mathrm{i.i.d.}}{\sim} p_{\mathrm{te}}(\boldsymbol{x}, y)$

- For each mini-batch $\{(\bar{\boldsymbol{x}}_i^{\mathrm{tr}}, \bar{y}_i^{\mathrm{tr}})\}_{i=1}^{\bar{n}_{\mathrm{tr}}}, \{(\bar{\boldsymbol{x}}_i^{\mathrm{te}}, \bar{y}_i^{\mathrm{te}})\}_{i=1}^{\bar{n}_{\mathrm{te}}}$, importance weights are estimated by matching loss values by kernel mean matching:

Huang, Gretton, Borgwardt, Schölkopf & Smola (NeurIPS2007)

$$\frac{1}{\bar{n}_{\mathrm{tr}}} \sum_{i=1}^{\bar{n}_{\mathrm{tr}}} r_i \ell(f(\bar{\boldsymbol{x}}_i^{\mathrm{tr}}), \bar{y}_i^{\mathrm{tr}}) \approx \frac{1}{\bar{n}_{\mathrm{te}}} \sum_{j=1}^{\bar{n}_{\mathrm{te}}} \ell(f(\bar{\boldsymbol{x}}_j^{\mathrm{te}}), \bar{y}_j^{\mathrm{te}})$$

- No covariate shift assumption is needed!

Practical Implementation

Algorithm 1 Dynamic importance weighting (in a mini-batch).

Require: a training mini-batch $\mathcal{S}^{\mathrm{tr}}$, a validation mini-batch $\mathcal{S}^{\mathrm{v}}$, the current model $\boldsymbol{f}_{\theta_t}$

1: `forward` the input parts of $\mathcal{S}^{\mathrm{tr}}$ & $\mathcal{S}^{\mathrm{v}}$
2: `compute` the loss values as $\mathcal{L}^{\mathrm{tr}}$ & $\mathcal{L}^{\mathrm{v}}$
3: `match` $\mathcal{L}^{\mathrm{tr}}$ & $\mathcal{L}^{\mathrm{v}}$ to obtain $\mathcal{W}$
4: `weight` the empirical risk $\widehat{R}(\boldsymbol{f}_\theta)$ by $\mathcal{W}$
5: `backward` $\widehat{R}(\boldsymbol{f}_\theta)$ and update θ

Experimental Evaluation

Table 4: Mean accuracy (standard deviation) in percentage on Fashion-MNIST (F-MNIST for short), CIFAR-10/100 under label noise (5 trials). Best and comparable methods (paired *t*-test at significance level 5%) are highlighted in bold. p/s is short for pair/symmetric flip.

	Noise	Clean	Uniform	Random	IW	Reweight	DIW
F-MNIST	0.3 p	71.05 (1.03)	76.89 (1.06)	84.62 (0.68)	82.69 (0.38)	**88.74 (0.19)**	88.19 (0.43)
	0.4 s	73.55 (0.80)	77.13 (2.21)	84.58 (0.76)	80.54 (0.66)	85.94 (0.51)	**88.29 (0.18)**
	0.5 s	73.55 (0.80)	73.70 (1.83)	82.49 (1.29)	78.90 (0.97)	84.05 (0.51)	**87.67 (0.57)**
CIFAR-10	0.3 p	45.62 (1.66)	77.75 (3.27)	83.20 (0.62)	45.02 (2.25)	82.44 (1.00)	**84.44 (0.70)**
	0.4 s	45.61 (1.89)	69.59 (1.83)	76.90 (0.43)	44.31 (2.14)	76.69 (0.57)	**80.40 (0.69)**
	0.5 s	46.35 (1.24)	65.23 (1.11)	71.56 (1.31)	42.84 (2.35)	72.62 (0.74)	**76.26 (0.73)**
CIFAR-100	0.3 p	10.82 (0.44)	50.20 (0.53)	48.65 (1.16)	10.85 (0.59)	48.48 (1.52)	**53.94 (0.29)**
	0.4 s	10.82 (0.44)	46.34 (0.88)	42.17 (1.05)	10.61 (0.53)	42.15 (0.96)	**53.66 (0.28)**
	0.5 s	10.82 (0.44)	41.35 (0.59)	34.99 (1.19)	10.58 (0.17)	36.17 (1.74)	**49.13 (0.98)**

Organization of My Talk

1. Introduction
2. Classical results
   - A) Importance weighting
   - B) Adaptive importance weighting
3. Recent results
   - A) Joint upper-bound minimization
   - B) Dynamic importance weighting

Conclusions

- In transfer learning, combining importance estimation and predictor training is promising.
- What should we do if the training and test distributions are very different?
  - Mechanism transfer!

Teshima, Sato & Sugiyama (ICML2020)