A One-step Approach to Covariate Shift Adaptation

Tianyi Zhang ZHANG@MS.K.U-TOKYO.AC.JP
The University of Tokyo, Japan.

Ikko Yamane YAMANE@MS.K.U-TOKYO.AC.JP
The University of Tokyo, Japan.

Nan Lu LU@MS.K.U-TOKYO.AC.JP
The University of Tokyo/RIKEN, Japan.

Masashi Sugiyama SUGI@K.U-TOKYO.AC.JP
RIKEN/The University of Tokyo, Japan.

Editors: Sinno Jialin Pan and Masashi Sugiyama

Abstract

A default assumption in many machine learning scenarios is that the training and test samples are drawn from the *same* probability distribution. However, such an assumption is often violated in the real world due to non-stationarity of the environment or bias in sample selection. In this work, we consider a prevalent setting called *covariate shift*, where the input distribution differs between the training and test stages while the conditional distribution of the output given the input remains unchanged. Most of the existing methods for covariate shift adaptation are two-step approaches, which first calculate the *importance* weights and then conduct *importance-weighted empirical risk minimization*. In this paper, we propose a novel *one-step approach* that jointly learns the predictive model and the associated weights in one optimization by minimizing an upper bound of the test risk. We theoretically analyze the proposed method and provide a generalization error bound. We also empirically demonstrate the effectiveness of the proposed method.

Keywords: covariate shift adaptation, empirical risk minimization, alternating optimization

1. Introduction

When developing algorithms of supervised learning, it is commonly assumed that samples used for training and samples used for testing follow the same probability distribution (Bishop, 1995; Duda et al., 2012; Hastie et al., 2009; Schölkopf and Smola, 2001; Vapnik, 1998; Wahba, 1990). However, this common assumption may not be fulfilled in many real-world applications due to sample selection bias or non-stationarity of environments (Huang et al., 2007; Quionero-Candela et al., 2009; Sugiyama and Kawanabe, 2012; Zadrozny, 2004).

Covariate shift, which was first introduced by Shimodaira (2000), is a prevalent setting for supervised learning in the wild, where the input distribution is different in the training and test phases but the conditional distribution of the output variable given the input variable remains unchanged. Covariate shift is conceivable in many real-world applications such as brain-computer interfacing (Li et al., 2010), emotion recognition (Jirayucharoensak et al., 2014), human activity recognition (Hachiya et al., 2012), spam filtering (Bickel and Scheffer, 2007), or speaker indentification (Yamada et al., 2010).

Due to the difference between the training and test distributions, the model trained by employing standard machine learning techniques such as *empirical risk minimization* (Schölkopf and Smola, 2001; Vapnik, 1998) may not generalize well to the test data. However, as shown by Shimodaira (2000), Sugiyama and Müller (2005), Sugiyama et al. (2007), and Zadrozny (2004), this problem can be mitigated by *importance sampling* (Cochran, 2007; Fishman, 2013; Kahn and Marshall, 1953): weighting the training loss terms according to the *importance*, which is the ratio of the test and training inut densities. As a consequence, most previous work (Huang et al., 2007; Kanamori et al., 2009; Sugiyama et al., 2008) mainly focused on accurately estimating the importance. Then the estimated importance is used to train a predictive model in the training phase. Thus, most of the existing methods of covariate shift adaptation are two-step approaches.

However, according to *Vapnik's principle* (Vapnik, 1998), which advocates that one should avoid solving a more general problem as an intermediate step when the amount of information is restricted, directly solving the covariate shift problem may be preferable to two-step approaches when the amount of covariate shift is substantial and the number of training data is not large. Moreover, Yamada et al. (2011) argued that density ratio estimation, the intermediate step for covariate shift adaptation, is indeed rather hard, suggesting that the importance approximation could be unreliable and thus deteriorate the performance of learning in practice.

In this paper, we propose a novel one-step approach to covariate shift adaptation, without the intermediate step of estimating the ratio of the training and test input densities. We jointly learn the predictive model and the associated weights by minimizing an upper bound of the test risk. Furthermore, we establish a generalization error bound based on the Rademacher complexity to give a theoretical guarantee for the proposed method. Experiments on synthetic and benchmark datasets highlight the advantage of our method over the existing two-step approaches.

2. Preliminaries

In this section, we briefly introduce the problem setup of covariate shift adaptation and relevant previous methods.

2.1. Problem Formulation

Let us start from the setup of supervised learning. Let $\mathcal{X} \subset \mathbb{R}^d$ be the input space(d is a positive integer), $\mathcal{Y} \subset \mathbb{R}$ (regression) or $\mathcal{Y} = \{-1, +1\}$ (binary classification) be the output space, and $\left\{\left(\boldsymbol{x}_i^{\mathrm{tr}}, y_i^{\mathrm{tr}}\right)\right\}_{i=1}^{n_{\mathrm{tr}}}$ be the training samples drawn independently from a training distribution with density $p_{\mathrm{tr}}(\boldsymbol{x}, y)$, which can be decomposed into the marginal distribution and the conditional probability distribution, i.e., $p_{\mathrm{tr}}(\boldsymbol{x}, y) = p_{\mathrm{tr}}(\boldsymbol{x})p_{\mathrm{tr}}(y|\boldsymbol{x})$. Let $(\boldsymbol{x}^{\mathrm{te}}, y^{\mathrm{te}})$ be a test sample drawn from a test distribution with density $p_{\mathrm{te}}(\boldsymbol{x}, y) = p_{\mathrm{te}}(\boldsymbol{x})p_{\mathrm{te}}(y|\boldsymbol{x})$.

Formally, the goal of supervised learning is to obtain a model $f \colon \mathcal{X} \to \mathbb{R}$ with the training samples that minimizes the expected loss over the test distribution (which is also called the *test risk*):

$$R(f) \coloneqq \mathbb{E}_{(\boldsymbol{x}^{\mathrm{te}}, y^{\mathrm{te}}) \sim p_{\mathrm{te}}(\boldsymbol{x}, y)} \left[\ell(f(\boldsymbol{x}^{\mathrm{te}}), y^{\mathrm{te}}) \right], \tag{1}$$

where $\ell \colon \mathbb{R} \times \mathcal{Y} \to \mathbb{R}$ denotes the *loss function* that measures the discrepancy between the true output value y and the predicted value $\widehat{y} \coloneqq f(\boldsymbol{x})$. In this paper, we assume that ℓ is bounded from above. We will discuss the practical choice of loss function in Section 3.

Since the assumption that the joint distributions are unchanged (i.e., $p_{\mathrm{tr}}(\boldsymbol{x}, y) = p_{\mathrm{te}}(\boldsymbol{x}, y)$) does not hold under covariate shift (i.e., $p_{\mathrm{tr}}(\boldsymbol{x}) \neq p_{\mathrm{te}}(\boldsymbol{x})$, $\mathrm{supp}(p_{\mathrm{tr}}) = \mathrm{supp}(p_{\mathrm{te}})$, and $p_{\mathrm{tr}}(y|\boldsymbol{x}) = p_{\mathrm{te}}(y|\boldsymbol{x})$), we utilize unlabeled test samples $\{\boldsymbol{x}_i^{\mathrm{te}}\}_{i=1}^{n_{\mathrm{te}}}$ besides the labeled training samples $\left\{ \left(\boldsymbol{x}_i^{\mathrm{tr}}, y_i^{\mathrm{tr}} \right) \right\}_{i=1}^{n_{\mathrm{tr}}}$ to compensate the difference of distributions. The goal of covariate shift adaptation is still to obtain a model that minimizes the test risk (1).

2.2. Previous Work

Empirical risk minimization (ERM) (Schölkopf and Smola, 2001; Vapnik, 1998), a standard technique in supervised learning, may fail under covariate shift due to the difference between the training and test distributions.

Importance sampling was used to mitigate the influence of covariate shift (Shimodaira, 2000; Sugiyama and Müller, 2005; Sugiyama et al., 2007; Zadrozny, 2004):

$$\mathbb{E}_{(\boldsymbol{x}^{\mathrm{te}}, y^{\mathrm{te}}) \sim p_{\mathrm{te}}(\boldsymbol{x}, y)} \left[\ell(f(\boldsymbol{x}^{\mathrm{te}}), y^{\mathrm{te}}) \right] = \mathbb{E}_{(\boldsymbol{x}^{\mathrm{tr}}, y^{\mathrm{tr}}) \sim p_{\mathrm{tr}}(\boldsymbol{x}, y)} \left[\ell(f(\boldsymbol{x}^{\mathrm{tr}}), y^{\mathrm{tr}}) r(\boldsymbol{x}^{\mathrm{tr}}) \right],$$

where $r(\boldsymbol{x}) = p_{\mathrm{te}}(\boldsymbol{x}) / p_{\mathrm{tr}}(\boldsymbol{x})$ is referred to as the importance, and this leads to the *importance weighted ERM* (IWERM): $\min_{f \in \mathcal{F}} \frac{1}{n_{\mathrm{tr}}} \sum_{i=1}^{n_{\mathrm{tr}}} \ell(f(\boldsymbol{x}_i^{\mathrm{tr}}), y_i^{\mathrm{tr}}) r(\boldsymbol{x}_i^{\mathrm{tr}})$, where $\mathcal{F}$ is a hypothesis set. For any fixed $f \in \mathcal{F}$, the importance weighted empirical risk is an unbiased estimator of the test risk.

However, IWERM tends to produce an estimator with high variance making the resulting test risk large (Shimodaira, 2000; Sugiyama and Kawanabe, 2012). Reducing the variance by slightly flattening the importance weights is practically useful, which results in *exponentially-flattened importance weighted ERM* (EIWERM) proposed by Shimodaira (2000): $\min_{f \in \mathcal{F}} \frac{1}{n_{\mathrm{tr}}} \sum_{i=1}^{n_{\mathrm{tr}}} \ell(f(\boldsymbol{x}_i^{\mathrm{tr}}), y_i^{\mathrm{tr}}) r(\boldsymbol{x}_i^{\mathrm{tr}})^{\gamma}$, where $\gamma \in [0, 1]$ is called the flattening parameter.

Therefore, how to estimate the importance accurately becomes the key to success of covariate shift adaptation. Unconstrained Least-Squares Importance Fitting (uLSIF) (Kanamori et al., 2009) is one of the commonly used density ratio estimation methods which is computationally efficient and comparable to other methods (Huang et al., 2007; Sugiyama et al., 2008) in terms of performance.

Yamada et al. (2011) argued that estimation of the density ratio is rather hard, which weakens the effectiveness of EIWERM. Then they proposed a method that directly estimates a flattened version of the importance weights, called *relative importance weighted ERM* (RIWERM): $\min_{f \in \mathcal{F}} \frac{1}{n_{\mathrm{tr}}} \sum_{i=1}^{n_{\mathrm{tr}}} \ell(f(\boldsymbol{x}_i^{\mathrm{tr}}), y_i^{\mathrm{tr}}) r_{\alpha}(\boldsymbol{x}_i^{\mathrm{tr}})$, where $r_{\alpha}(\boldsymbol{x}) \coloneqq \frac{p_{\mathrm{te}}(\boldsymbol{x})}{(1-\alpha) p_{\mathrm{te}}(\boldsymbol{x}) + \alpha p_{\mathrm{tr}}(\boldsymbol{x})}$ is called the α-relative importance ($\alpha \in [0, 1]$). The relative importance $r_{\alpha}(\boldsymbol{x})$ can be estimated by relative uLSIF (RuLSIF) as presented by Yamada et al. (2011).

Hyper-parameters such as the flattening parameter γ or α need to be appropriately chosen in order to obtain a good generalization capability. However, cross validation (CV), a standard technique for model selection, does not work well under covariate shift. To cope with this problem, a variant of CV called importance-weighted CV (IWCV) has been proposed by Sugiyama et al. (2007), which is based on the importance sampling technique to give an almost unbiased estimate of the generalization error with finite samples. However, the importance used in IWCV still needs to be estimated from samples.

3. Proposed Method

In this section, in order to overcome the drawbacks of the existing two-step approaches, we propose a one-step approach which integrates the importance estimation step and the importance-weighted empirical risk minimization step by upper-bounding the test risk. Moreover, we provide a theoretical analysis of the proposed method.

3.1. One-step Approach

First, we derive an upper bound of the test risk, which is the key of our one-step approach.

Theorem 1 *Let $r(\boldsymbol{x})$ be the importance $p_{\mathrm{te}}(\boldsymbol{x})/p_{\mathrm{tr}}(\boldsymbol{x})$ and $\mathcal{F} \subseteq \{f\colon \mathcal{X} \to \mathbb{R}\}$ be a given hypothesis set. Suppose that there is a constant $m \in \mathbb{R}$ such that $\ell(f(\boldsymbol{x}), y) \leq m$ holds for every $f \in \mathcal{F}$ and every $(\boldsymbol{x}, y) \in \mathcal{X} \times \mathcal{Y}$. Then, for any $f \in \mathcal{F}$ and any measurable function $g\colon \mathcal{X} \to \mathbb{R}$, the test risk is bounded as*

$$\frac{1}{2}R^2(f) \leq J(f,g) \coloneqq \left(\mathbb{E}_{(\boldsymbol{x}^{\mathrm{tr}},y^{\mathrm{tr}})\sim p_{\mathrm{tr}}(\boldsymbol{x},y)}\left[\ell(f(\boldsymbol{x}^{\mathrm{tr}}), y^{\mathrm{tr}})g(\boldsymbol{x}^{\mathrm{tr}})\right]\right)^2 + m^2\mathbb{E}_{\boldsymbol{x}^{\mathrm{tr}}\sim p_{\mathrm{tr}}(\boldsymbol{x})}\left[\left(g(\boldsymbol{x}^{\mathrm{tr}}) - r(\boldsymbol{x}^{\mathrm{tr}})\right)^2\right]. \tag{2}$$

Furthermore, if g is non-negative and ℓ_{UB} bounds ℓ from above, we have

$$J(f,g) \leq J_{\mathrm{UB}}(f,g) \coloneqq \left(\mathbb{E}_{(\boldsymbol{x}^{\mathrm{tr}},y^{\mathrm{tr}})\sim p_{\mathrm{tr}}(\boldsymbol{x},y)}\left[\ell_{\mathrm{UB}}(f(\boldsymbol{x}^{\mathrm{tr}}), y^{\mathrm{tr}})g(\boldsymbol{x}^{\mathrm{tr}})\right]\right)^2 + m^2\mathbb{E}_{\boldsymbol{x}^{\mathrm{tr}}\sim p_{\mathrm{tr}}(\boldsymbol{x})}\left[\left(g(\boldsymbol{x}^{\mathrm{tr}}) - r(\boldsymbol{x}^{\mathrm{tr}})\right)^2\right]. \tag{3}$$

Proof According to the Cauchy-Schwarz inequality, we have

$$\begin{aligned}
\frac{1}{2}R^2(f) &= \frac{1}{2}\left(\mathbb{E}_{(\boldsymbol{x}^{\mathrm{tr}},y^{\mathrm{tr}})}\left[\ell(f(\boldsymbol{x}^{\mathrm{tr}}), y^{\mathrm{tr}})r(\boldsymbol{x}^{\mathrm{tr}})\right]\right)^2 \\
&\leq \left(\mathbb{E}_{(\boldsymbol{x}^{\mathrm{tr}},y^{\mathrm{tr}})}\left[\ell(f(\boldsymbol{x}^{\mathrm{tr}}), y^{\mathrm{tr}})g(\boldsymbol{x}^{\mathrm{tr}})\right]\right)^2 + \left(\mathbb{E}_{(\boldsymbol{x}^{\mathrm{tr}},y^{\mathrm{tr}})}\left[\ell(f(\boldsymbol{x}^{\mathrm{tr}}), y^{\mathrm{tr}})\left(r(\boldsymbol{x}^{\mathrm{tr}}) - g(\boldsymbol{x}^{\mathrm{tr}})\right)\right]\right)^2 \\
&\leq \left(\mathbb{E}_{(\boldsymbol{x}^{\mathrm{tr}},y^{\mathrm{tr}})}\left[\ell(f(\boldsymbol{x}^{\mathrm{tr}}), y^{\mathrm{tr}})g(\boldsymbol{x}^{\mathrm{tr}})\right]\right)^2 \\
&\quad + \mathbb{E}_{(\boldsymbol{x}^{\mathrm{tr}},y^{\mathrm{tr}})}\left[\ell^2(f(\boldsymbol{x}^{\mathrm{tr}}), y^{\mathrm{tr}})\right]\mathbb{E}_{\boldsymbol{x}^{\mathrm{tr}}}\left[\left(g(\boldsymbol{x}^{\mathrm{tr}}) - r(\boldsymbol{x}^{\mathrm{tr}})\right)^2\right] \\
&\leq \left(\mathbb{E}_{(\boldsymbol{x}^{\mathrm{tr}},y^{\mathrm{tr}})}\left[\ell(f(\boldsymbol{x}^{\mathrm{tr}}), y^{\mathrm{tr}})g(\boldsymbol{x}^{\mathrm{tr}})\right]\right)^2 + m^2\mathbb{E}_{\boldsymbol{x}^{\mathrm{tr}}}\left[\left(g(\boldsymbol{x}^{\mathrm{tr}}) - r(\boldsymbol{x}^{\mathrm{tr}})\right)^2\right],
\end{aligned}$$

where $(\boldsymbol{x}^{\mathrm{tr}}, y^{\mathrm{tr}}) \sim p(\boldsymbol{x}^{\mathrm{tr}}, y^{\mathrm{tr}})$. This proves (2), and based on this, (3) is obvious. ■

For classification problems, $R(f)$ is typically defined by the zero-one loss $\ell(\widehat{y}, y) = I(\widehat{y}y \leq 0)$, where I is the indicator function, and thus the boundedness assumption of the loss function in Theorem 1 holds with $m = 1$. For regression problems, The squared loss

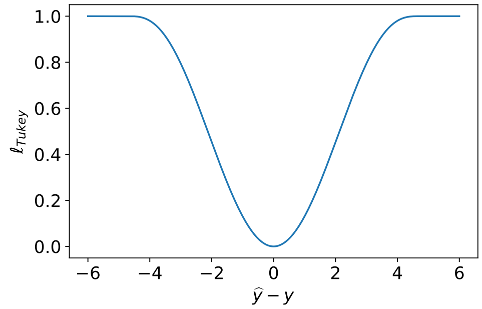

Figure 1: Tukey's loss defined as $\ell_{\text{Tukey}}(\widehat{y}, y) := \min\left(1 - \left[1 - (\widehat{y} - y)^2/\rho^2\right]^3, 1\right) \leq 1$. It is widely used in the context of robust statistics. The hyperparameter $\rho > 0$ is usually set to 4.685 for this loss function, and it provides an asymptotic efficiency 95% of that of least squares for Gaussian noise (Andersen, 2008). Here, we scale the standard Tukey's bisquare loss for convenience, which does not change the minimizer of the test risk.

$\ell(\widehat{y}, y) = (\widehat{y} - y)^2$ is a typical choice, but it violates the boundedness assumption. Instead, we define $R(f)$ using Tukey's bisquare loss (Beaton and Tukey, 1974) (see Fig. 1).[1]

Remark 2 *The two-step approach that first applies uLSIF to estimate the importance weights and then employs IWERM is equivalent to minimizing the second term of the above upper bounds first and then minimizing the first term, which leads to a sub-optimal solution to the upper-bound minimization.*

Instead of estimating the unknown $r(\boldsymbol{x})$ for minimizing $R(f)$ as in the previous two-step approaches, we propose a one-step approach that minimizes the upper bound $J(f, g)$ or $J_{\text{UB}}(f, g)$ based on Theorem 1.

For classification problems, $J(f, g)$ is defined using the zero-one loss, with which training will not be tractable (Ben-David et al., 2003). Fortunately, the latter part of Theorem 1 allows us to minimize $J_{\text{UB}}(f, g)$ instead, with ℓ_{UB} being any (sub-)differentiable approximation that bounds the zero-one loss from above so that we can apply any gradient method such as stochastic gradient descent (Robbins and Monro, 1951). Examples of such ℓ_{UB} include the hinge loss $\ell(\widehat{y}, y) = \max(0, 1 - \widehat{y}y)$ and the squared loss. For regression problems, Tukey's loss is already differentiable, but we can use the squared loss that bounds Tukey's loss which makes the optimization problem simpler as described later. This is again justified by Theorem 1 with the squared loss used for the upper-bound loss ℓ_{UB}.

Although the second expectation in $J_{\text{UB}}(f, g)$ contains an unknown term $r(\boldsymbol{x})$, it can be estimated from the samples on hand up to addition by a constant due to the fact that

$$\mathbb{E}_{\boldsymbol{x}^{\text{tr}} \sim p_{\text{tr}}(\boldsymbol{x})}\left[\left(g(\boldsymbol{x}^{\text{tr}}) - r(\boldsymbol{x}^{\text{tr}})\right)^2\right] = \mathbb{E}_{\boldsymbol{x}^{\text{tr}} \sim p_{\text{tr}}(\boldsymbol{x})}\left[g^2(\boldsymbol{x}^{\text{tr}})\right] - 2\mathbb{E}_{\boldsymbol{x}^{\text{te}} \sim p_{\text{te}}(\boldsymbol{x})}\left[g(\boldsymbol{x}^{\text{te}})\right] + C,$$

where C is a constant that does not depend on the function f nor g.

[1] There is another bounded loss called Welsch loss (Ke et al., 2020) which has a similar shape to that of Tukey's bisquare loss. In this paper, we focus on Tukey's bisquare loss.

Since we cannot directly evaluate $J_{\mathrm{UB}}(f,g)$, we minimize its empirical version $\widehat{J}_{\mathrm{UB}}(f,g;S)$ with respect to f and non-negative g in some given hypothesis sets $\mathcal{F}$ and $\mathcal{G}_+$:

$$\begin{aligned}\widehat{J}_{\mathrm{UB}}(f,g;S) \coloneqq & \left(\frac{1}{n_{\mathrm{tr}}}\sum_{i=1}^{n_{\mathrm{tr}}}\ell_{\mathrm{UB}}(f(\boldsymbol{x}_i^{\mathrm{tr}}),y_i^{\mathrm{tr}})g(\boldsymbol{x}_i^{\mathrm{tr}})\right)^2 \\ & + m^2\left(\frac{1}{n_{\mathrm{tr}}}\sum_{i=1}^{n_{\mathrm{tr}}}g^2(\boldsymbol{x}_i^{\mathrm{tr}}) - \frac{2}{n_{\mathrm{te}}}\sum_{i=1}^{n_{\mathrm{te}}}g(\boldsymbol{x}_i^{\mathrm{te}}) + C\right), \end{aligned} \tag{4}$$

where $S = \left\{\left(\boldsymbol{x}_i^{\mathrm{tr}}, y_i^{\mathrm{tr}}\right)\right\}_{i=1}^{n_{\mathrm{tr}}} \cup \left\{\boldsymbol{x}_i^{\mathrm{te}}\right\}_{i=1}^{n_{\mathrm{te}}}$ is the set of sample points. Notice that constant C can be safely ignored in the minimization.

Below, we present an efficient alternating minimization algorithm described in Algorithm 1 that can be employed when $f(\boldsymbol{x})$ and $g(\boldsymbol{x})$ are linear-in-parameter models, i.e.,

$$f(\boldsymbol{x}) = \boldsymbol{\alpha}^\top\boldsymbol{\phi}(\boldsymbol{x}) \quad \text{and} \quad g(\boldsymbol{x}) = \boldsymbol{\beta}^\top\boldsymbol{\psi}(\boldsymbol{x}), \tag{5}$$

where $\boldsymbol{\alpha} \in \mathbb{R}^{b_f}$ and $\boldsymbol{\beta} \in \mathbb{R}^{b_g}$ are parameters, and $\boldsymbol{\phi}$ and $\boldsymbol{\psi}$ are b_f-dimensional and b_g-dimensional vectors of basis functions.

First, we minimize the objective (4) with respect to g while fixing f. This step has an analytic solution as shown in Algorithm 1, Line 6, where $\boldsymbol{\Phi}_{\mathrm{tr}} = (\boldsymbol{\phi}(\boldsymbol{x}_1^{\mathrm{tr}}),\ldots,\boldsymbol{\phi}(\boldsymbol{x}_{n_{\mathrm{tr}}}^{\mathrm{tr}}))^\top$, $\boldsymbol{\Psi}_{\mathrm{tr}} = (\boldsymbol{\psi}(\boldsymbol{x}_1^{\mathrm{tr}}),\ldots,\boldsymbol{\psi}(\boldsymbol{x}_{n_{\mathrm{tr}}}^{\mathrm{tr}}))^\top$, $\boldsymbol{\Psi}_{\mathrm{te}} = (\boldsymbol{\psi}(\boldsymbol{x}_1^{\mathrm{te}}),\ldots,\boldsymbol{\psi}(\boldsymbol{x}_{n_{\mathrm{te}}}^{\mathrm{te}}))^\top$, $\boldsymbol{1} = (1,\ldots,1)^\top$, and $\boldsymbol{I}$ is the identity matrix.

Next, we minimize the objective (4) with respect to f while fixing g. In this step, we can safely ignore the second term and remove the square operation of the first term in the objective (4) to reduce the problem into weighted empirical risk minimization (cf. Algorithm 1, Line 12) by forcing g to be non-negative with a rounding up technique (Kanamori et al., 2009) as shown in Algorithm 1, Line 7. For regression problems, the method of iteratively reweighted least squares (IRLS) (Beaton and Tukey, 1974) can be used for optimizing the Tukey's bisquare loss. In practice, we suggest using the squared loss as a convex approximation of the Tukey's loss to obtain a closed-form solution as shown in Algorithm 1, Line 10 for reducing computation time, and we compare their performance in the experiments. For classification with linear-in-parameter models using the hinge loss, then the weighted support vector machine (Yang et al., 2007) can be used. After this step, we go back to the step for updating g and repeat the procedure.

3.2. Theoretical Analysis

In what follows, we establish a generalization error bound for the proposed method in terms of the *Rademacher complexity* (Koltchinskii, 2001).

Lemma 3 *Assume that (a) there exist some constants $M \geq m$ and $L > 0$ such that $\ell_{\mathrm{UB}}(f(\boldsymbol{x}),y) \leq M$ holds for every $f \in \mathcal{F}$ and every $(\boldsymbol{x},y) \in \mathcal{X}\times\mathcal{Y}$ and $y \mapsto \ell_{\mathrm{UB}}(y,y')$ is L-Lipschitz for every fixed $y' \in \mathcal{Y}$;*[2] *(b) there exists some constant $G \geq 1$ such that $g(\boldsymbol{x}) \leq G$*

[2] This assumption is valid when $\sup_{f\in\mathcal{F}}\|f\|_\infty$ and $\sup_{y\in\mathcal{Y}}|y|$ are bounded.

Algorithm 1 Alternating Minimization

```
$\boldsymbol{\alpha}_0 \leftarrow$ an arbitrary $b_f$-dimensional vector
$\lambda_g \leftarrow$ a positive $\ell_2$-regularization parameter
$\lambda_f \leftarrow$ a positive $\ell_2$-regularization parameter
for $t = 0, 1, \ldots, T-1$ do
    $\boldsymbol{l}_t \leftarrow (\ell_{\mathrm{UB}}(\boldsymbol{\alpha}_t^\top \boldsymbol{\phi}(\boldsymbol{x}_1^{\mathrm{tr}}), y_1^{\mathrm{tr}}), \ldots, \ell_{\mathrm{UB}}(\boldsymbol{\alpha}_t^\top \boldsymbol{\phi}(\boldsymbol{x}_{n_{\mathrm{tr}}}^{\mathrm{tr}}), y_{n_{\mathrm{tr}}}^{\mathrm{tr}}))^\top$
    $\boldsymbol{\beta}_{t+1} \leftarrow \left(\frac{1}{n_{\mathrm{tr}}}\boldsymbol{\Psi}_{\mathrm{tr}}^\top \boldsymbol{\Psi}_{\mathrm{tr}} + \frac{1}{m^2 n_{\mathrm{tr}}^2}\boldsymbol{\Psi}_{\mathrm{tr}}^\top \boldsymbol{l}_t \boldsymbol{l}_t^\top \boldsymbol{\Psi}_{\mathrm{tr}} + \frac{1}{m^2}\lambda_g \boldsymbol{I}\right)^{-1} \frac{1}{n_{\mathrm{te}}}\boldsymbol{\Psi}_{\mathrm{te}}^\top \mathbf{1}$
    $\boldsymbol{\beta}_{t+1} \leftarrow \max(\boldsymbol{\beta}_{t+1}, \mathbf{0})$
    $w_i^{t+1} \leftarrow \boldsymbol{\beta}_{t+1}^\top \boldsymbol{\psi}(\boldsymbol{x}_i^{\mathrm{tr}}),\ i = 1, \ldots, n_{\mathrm{tr}}$
    if $\ell_{\mathrm{UB}}$ is the squared loss then
        $\boldsymbol{\alpha}_{t+1} \leftarrow \left(\boldsymbol{\Phi}_{\mathrm{tr}}^\top \boldsymbol{W}_{t+1} \boldsymbol{\Phi}_{\mathrm{tr}} + \lambda_f n_{\mathrm{tr}} \boldsymbol{I}\right)^{-1} \boldsymbol{\Phi}_{\mathrm{tr}}^\top \boldsymbol{W}_{t+1} \boldsymbol{y}_{\mathrm{tr}}$,
        where $\boldsymbol{W}_{t+1} = \mathrm{diag}(w_1^{t+1}, \ldots, w_{n_{\mathrm{tr}}}^{t+1})$ and $\boldsymbol{y}_{\mathrm{tr}} = (y_1^{\mathrm{tr}}, \ldots, y_{n_{\mathrm{tr}}}^{\mathrm{tr}})^\top$
    else
        $\boldsymbol{\alpha}_{t+1} \leftarrow \arg\min_{\boldsymbol{\alpha}} \frac{1}{n_{\mathrm{tr}}} \sum_{i=1}^{n_{\mathrm{tr}}} w_i^{t+1} \ell_{\mathrm{UB}}(\boldsymbol{\alpha}_t^\top \boldsymbol{\phi}(\boldsymbol{x}_i^{\mathrm{tr}}), y_i^{\mathrm{tr}}) + \lambda_f \boldsymbol{\alpha}^\top \boldsymbol{\alpha}$
    end if
end for
```

for every $g \in \mathcal{G}_+$ and every $\boldsymbol{x} \in \mathcal{X}$. Let $\mathcal{G} = \mathcal{G}_+ \cup -\mathcal{G}_+$ Then for any $\delta > 0$, with probability at least $1 - \delta$ over the draw of S, the following holds for all $f \in \mathcal{F}, g \in \mathcal{G}_+$:

$$\begin{aligned} J_{\mathrm{UB}}(f, g) \leq & \widehat{J}_{\mathrm{UB}}(f, g; S) + 8MG\,(M + G)\left(L\mathfrak{R}_{n_{\mathrm{tr}}}^{\mathrm{tr}}(\mathcal{F}) + \mathfrak{R}_{n_{\mathrm{tr}}}^{\mathrm{tr}}(\mathcal{G})\right) \\ & + 4M^2 \mathfrak{R}_{n_{\mathrm{te}}}^{\mathrm{te}}(\mathcal{G}) + 5M^2G^2\sqrt{\frac{\log\frac{1}{\delta}}{2}}\left(\frac{1}{\sqrt{n_{\mathrm{tr}}}} + \frac{1}{\sqrt{n_{\mathrm{te}}}}\right), \end{aligned} \tag{6}$$

where $\mathfrak{R}_{n_{\mathrm{tr}}}^{\mathrm{tr}}(\mathcal{F})$ and $\mathfrak{R}_{n_{\mathrm{tr}}}^{\mathrm{tr}}(\mathcal{G})$ are the Rademacher complexities of $\mathcal{F}$ and $\mathcal{G}$, respectively, for the sampling of size n_{tr} from $p_{\mathrm{tr}}(\boldsymbol{x})$, and $\mathfrak{R}_{n_{\mathrm{te}}}^{\mathrm{te}}(\mathcal{G})$ is the Rademacher complexity of $\mathcal{G}$ for the sampling of size n_{te} from $p_{\mathrm{te}}(\boldsymbol{x})$.

We provide a proof of Lemma 3 in Appendix A. Combining (2), (3) and (6), we obtain the following theorem.

Theorem 4 *Suppose that the assumptions in Lemma 3 hold. Then, for any $\delta > 0$, with probability at least $1 - \delta$ over the draw of S, the test risk can be bounded as follows for all $f \in \mathcal{F}$ uniformly:*

$$\begin{aligned} \frac{1}{2}R^2(f) \leq & \min_{g \in \mathcal{G}_+} \widehat{J}_{\mathrm{UB}}(f, g; S) + 8MG\,(M + G)\left(L\mathfrak{R}_{n_{\mathrm{tr}}}^{\mathrm{tr}}(\mathcal{F}) + \mathfrak{R}_{n_{\mathrm{tr}}}^{\mathrm{tr}}(\mathcal{G})\right) \\ & + 4M^2 \mathfrak{R}_{n_{\mathrm{te}}}^{\mathrm{te}}(\mathcal{G}) + 5M^2G^2\sqrt{\frac{\log\frac{1}{\delta}}{2}}\left(\frac{1}{\sqrt{n_{\mathrm{tr}}}} + \frac{1}{\sqrt{n_{\mathrm{te}}}}\right). \end{aligned} \tag{7}$$

Theorem 4 implies that minimizing $\widehat{J}_{\mathrm{UB}}(f, g; S)$, as the proposed method does, amounts to minimizing an upper bound of the test risk. Furthermore, the following theorem shows a generalization error bound for the minimizer obtained by the proposed method.

Theorem 5 *Let* $(\widehat{f},\widehat{g}) = \arg\min_{(f,g)\in\mathcal{F}\times\mathcal{G}_+} \widehat{J}_{\mathrm{UB}}(f,g;S)$*. Then, under the assumptions of Lemma 3, for any* $\delta > 0$*, it holds with probability at least* $1-\delta$ *over the draw of* S *that*

$$\frac{1}{2}R^2(\widehat{f}) \leq \min_{f\in\mathcal{F},g\in\mathcal{G}_+} J_{\mathrm{UB}}(f,g) + 8MG\,(M+G)\left(L\mathfrak{R}^{\mathrm{tr}}_{n_{\mathrm{tr}}}(\mathcal{F}) + \mathfrak{R}^{\mathrm{tr}}_{n_{\mathrm{tr}}}(\mathcal{G})\right) + 4M^2\mathfrak{R}^{\mathrm{te}}_{n_{\mathrm{te}}}(\mathcal{G}) + 10M^2G^2\sqrt{\frac{\log\frac{1}{\delta}}{2}}\left(\frac{1}{\sqrt{n_{\mathrm{tr}}}} + \frac{1}{\sqrt{n_{\mathrm{te}}}}\right) + M^2G^2\frac{1}{n_{\mathrm{tr}}}. \tag{8}$$

A proof of Theorem 5 is presented in Appendix B. If we use linear-in-parameter models with bounded norms, then $\mathfrak{R}^{\mathrm{tr}}_{n_{\mathrm{tr}}}(\mathcal{F}) = O(1/\sqrt{n_{\mathrm{tr}}})$, $\mathfrak{R}^{\mathrm{tr}}_{n_{\mathrm{tr}}}(\mathcal{G}) = O(1/\sqrt{n_{\mathrm{tr}}})$, and $\mathfrak{R}^{\mathrm{te}}_{n_{\mathrm{te}}}(\mathcal{G}) = O(1/\sqrt{n_{\mathrm{te}}})$. Furthermore, if we assume that the approximation error of $\mathcal{G}_+$ is zero, i.e., $r \in \mathcal{G}_+$, then $\min_{f\in\mathcal{F},g\in\mathcal{G}_+} J_{\mathrm{UB}}(f,g) \leq J_{\mathrm{UB}}(f^*,r) = R^2_{\mathrm{UB}}(f^*)$, where R_{UB} is the test risk defined with ℓ_{UB} and $f^* = \arg\min_{f\in\mathcal{F}} R_{\mathrm{UB}}(f)$. Thus,

$$R(\widehat{f}) \leq \sqrt{2}R_{\mathrm{UB}}(f^*) + O_p(1/\sqrt[4]{n_{\mathrm{tr}}} + 1/\sqrt[4]{n_{\mathrm{te}}}).$$

When the best-in-class test risk $R_{\mathrm{UB}}(f^*)$ is small, this bound would theoretically guarantee a good performance of the proposed method.

4. Experiments

In this section, we examine the effectiveness of the proposed method via experiments on toy and benchmark datasets.

4.1. Illustration with Toy Datasets

First, we conduct experiments on a toy regression dataset.

Let us consider a one-dimensional regression problem. Let the training and test input densities be

$$p_{\mathrm{tr}}(x) = N(x;1,(0.5)^2) \quad\text{and}\quad p_{\mathrm{te}}(x) = N(x;2,(0.25)^2),$$

where $N(x;\mu,\sigma^2)$ denotes the Gaussian density with mean μ and variance σ^2. Consider the output labels of examples are generated by

$$y = f^*(x) + \epsilon \quad\text{with}\quad f^*(x) = \mathrm{sinc}(x),$$

and the noise ϵ following $N\left(0,(0.1)^2\right)$ is independent of x. As illustrated in Fig. 2, the training input points are distributed on the left-hand side of the input domain and the test input points are distributed on the right-hand side. We sample $n_{\mathrm{tr}} = 150$ labeled i.i.d. training samples $\left\{\left(x_i^{\mathrm{tr}}, y_i^{\mathrm{tr}}\right)\right\}_{i=1}^{n_{\mathrm{tr}}}$ with each x_i^{tr} following $p_{\mathrm{tr}}(x)$ and $n_{\mathrm{te}} = 150$ unlabeled i.i.d. test samples $\{x_i^{\mathrm{te}}\}_{i=1}^{n_{\mathrm{te}}}$ following $p_{\mathrm{te}}(x)$ for learning the target function $f^*(x)$ in the experiment. In addition, we sample 10000 labeled i.i.d. test samples $\left\{(x_i^{\mathrm{eval}}, y_i^{\mathrm{eval}})\right\}_{i=1}^{n_{\mathrm{eval}}}$ with each $(x_i^{\mathrm{eval}}, y_i^{\mathrm{eval}})$ following $p_{\mathrm{te}}(x,y)$ for evaluating the performance of the learned function.

We compare our one-step approach with three baseline methods, which are the ordinary ERM, EIWERM with uLSIF, and RIWERM. We use the linear-in-parameter models (5) with the following Gaussian kernels as basis functions for learning the input-output relation

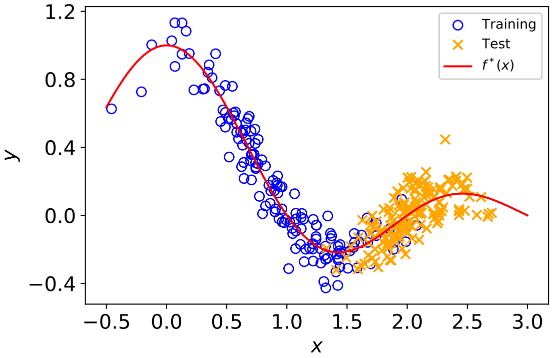

Figure 2: A toy regression example. The training input points (blue) are distributed on the left-hand side of the input domain and the test input points (orange) are distributed on the right-hand side. The two distributions share the same regression function f^* (the red curve).

and the importance (or the relative importance) in all the experiments including those in Section 4.2:

$$\phi_i(\boldsymbol{x}) = \exp\left\{-\frac{\|\boldsymbol{x}-\boldsymbol{c}_i^f\|^2}{2\sigma_f^2}\right\} \quad \text{and} \quad \psi_i(\boldsymbol{x}) = \exp\left\{-\frac{\|\boldsymbol{x}-\boldsymbol{c}_i^g\|^2}{2\sigma_g^2}\right\},$$

where σ_f and σ_g are the bandwidths of the Gaussian kernels, and $\boldsymbol{c}_i^f$ and $\boldsymbol{c}_i^g$ are the kernel centers randomly chosen from $\{\boldsymbol{x}_i^{\mathrm{te}}\}_{i=1}^{n_{\mathrm{te}}}$ (Kanamori et al., 2009; Sugiyama et al., 2008). We set $b_f = b_g = 50$ in all the experiments. Moreover, we use ℓ_2 regularization in all the experiments, which introduces two more hyperparameters λ_f and λ_g associated with models f and g respectively.

Let us clarify the hyperparameter tuning procedure for each method. For the ordinary ERM, the standard cross validation is applied for tuning σ_f and λ_f. For the EIWERM with uLSIF, the hyperparameter tuning of σ_g and λ_g in the importance estimation step uses the cross validation naturally based on its learning objective (cf. Kanamori et al. (2009)), and we apply IWCV in the training step for selecting σ_f, λ_f and flattening parameter γ. For RIWERM, the built-in cross validation with its learning objective is used for tuning σ_g and λ_g (cf. Yamada et al. (2011)), and the selection of σ_f, λ_f and parameter α in the training step is achieved by IWCV (the importance is obtained by uLSIF). Finally, for our one-step approach, we can naturally do cross validation based on the proposed learning objective (4). To reduce computation time in the one-step approach, we set the bandwidths to the median distances between samples and kernel centers, which is a popular heuristic in practice (Schölkopf and Smola, 2001), but we tune the regularization parameters by cross validation. For a fair comparison, we also report the results of the baseline methods using the median heuristic.

As suggested in Section 3.1, we use the squared loss in the one-step approach for obtaining a more efficient solution. We also employ the IRLS algorithm for optimizing Tukey's bisquare loss in the one-step approach. For better comparison, we report the results of the baseline methods using both the squared loss and Tukey's bisquare loss.

The experimental results of the toy regression problem are summarized in Table 1. Note that when the target function f^* is perfectly learned, the mean squared error is the variance of ϵ, i.e., 0.01. Therefore, our method significantly mitigates the influence of covariate shift. Since the IRLS algorithm is needed when using Tukey's bisquare loss, the training should

Table 1: Mean squared errors averaged over 100 trails on the toy dataset. The numbers in the brackets are the standard deviations. The best method and comparable ones based on the *Wilcoxon signed-rank test* (Wilcoxon, 1945) at the significance level 5% are described in bold face. "Unweighted" denotes the ordinary ERM, "squared" denotes the squared loss, "Tukey" denotes Tukey's bisquare loss, and "median" means that the bandwidths of the kernel models are determined by the median heuristic (other hyperparameters are still chosen by cross validation).

Methods	MSE(SD)	Computation time (sec)
unweighted (squared)	0.0517 (0.0300)	18.15
unweighted (squared, median)	0.1453 (0.1812)	3.22
unweighted (Tukey)	0.0511 (0.0455)	59.36
unweighted (Tukey, median)	0.0760 (0.0733)	8.45
uLSIF (squared)	0.0259 (0.0345)	70.67
uLSIF (squared, median)	0.0198 (0.0151)	12.92
uLSIF (Tukey)	0.0253 (0.0433)	586.54
uLSIF (Tukey, median)	0.0161 (0.0106)	71.99
RuLSIF (squared)	0.0226 (0.0261)	115.00
RuLSIF (squared, median)	0.0142 (0.0071)	27.65
RuLSIF (Tukey)	0.0205 (0.0167)	594.11
RuLSIF (Tukey, median)	0.0140 (0.0064)	83.04
one-step (squared)	0.0140 (0.0058)	89.05
one-step (Tukey)	**0.0125** (**0.0021**)	157.55

take longer time than that when using the squared loss, and we confirm it according to the results in Table 1.

4.2. Experiments on Benchmark Datasets

Finally, we conduct experiments on classification benchmark datasets from LIBSVM.[3]

We introduce covariate shift in the following way similarly to Cortes et al. (2008). First, we use Z-score normalization to preprocess all the input samples. Then, an example $(\boldsymbol{x}, y)$ is assigned to the training dataset with probability $\exp(v)/(1+\exp(v))$ and to the test dataset with probability $1/(1+\exp(v))$, where $v = 16\boldsymbol{w}^\top\boldsymbol{x}/\sigma$, σ is the standard deviation of $\boldsymbol{w}^\top\boldsymbol{x}$, and $\boldsymbol{w} \in \mathbb{R}^d$ is some given projection vector. To ensure that the methods are tested in challenging covariate shift situations, we randomly sample projection directions and choose one such that the classifier trained on the training dataset generalizes worst to the test dataset.

By following the procedure, we obtain one projection vector for each benchmark dataset, which is used to separate the dataset into a training dataset and a test dataset with some randomness. Then we sample a certain number (depending on the size of the dataset) of training samples and test input samples for training. We use the rest of test samples for evaluating the performance. We run 100 trails for each dataset.

[3] https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/

Table 2: Mean misclassification rates averaged over 100 trails on benchmark datasets. The numbers in the brackets are the standard deviations. For each dataset, the best method and comparable ones based on the *Wilcoxon signed-rank test* at the significance level 5% are described in bold face.

Dataset	Dim	unweighted	unweighted	uLSIF	uLSIF	RuLSIF	RuLSIF	one-step
			(median)		(median)		(median)	
australian	14	32.02	31.62	30.65	31.00	30.33	32.54	**25.57**
		(16.88)	(17.88)	(16.37)	(17.47)	(15.06)	(18.05)	**(12.74)**
breast-cancer	10	21.65	23.03	20.03	20.90	21.31	20.65	**12.57**
		(13.48)	(12.84)	(12.44)	(11.95)	(12.96)	(11.57)	**(10.17)**
diabetes	8	45.78	43.35	42.42	41.67	44.26	46.72	**38.57**
		(8.88)	(9.56)	(7.66)	(8.66)	(8.63)	(10.03)	**(6.36)**
heart	13	**34.72**	36.31	**35.86**	**35.22**	36.80	36.53	**33.84**
		(9.91)	(12.19)	**(11.43)**	**(11.89)**	(11.70)	(13.49)	**(10.94)**
sonar	60	38.06	34.41	35.36	**33.06**	36.27	**32.72**	**32.35**
		(12.96)	(11.90)	(13.24)	**(11.80)**	(13.50)	**(11.76)**	**(12.45)**

The models and the hyperparameter tuning procedure follow what we discussed in Section 4.1. In addition, as discussed in Section 3.1, we use the squared loss as the surrogate loss function for all the methods including the one-step approach in the experiments.

The experimental results on benchmark datasets are summarized in Table 2. The table shows the proposed one-step approach outperforms or is comparable to the baseline methods with the best performance, which suggests that it is a promising method for covariate shift adaptation.

5. Conclusion

In this work, we studied the problem of covariate shift adaptation. Unlike the dominating two-step approaches in the literature, we proposed a one-step approach that learns the predictive model and the associated weights simultaneously by following Vapnik's principle. Our experiments highlighted the advantage of our method over previous two-step approaches, suggesting that the proposed one-step approach is a promising method for covariate shift adaptation.

Acknowledgments

NL was supported by MEXT scholarship No. 171536. IY and MS were supported by JST CREST Grant Number JPMJCR18A2.

References

Robert Andersen. *Modern Methods for Robust Regression*, volume 152. SAGE, 2008.

Albert E. Beaton and John W. Tukey. The fitting of power series, meaning polynomials, illustrated on band-spectroscopic data. *Technometrics*, 16(2):147–185, 1974.

Shai Ben-David, Nadav Eiron, and Philip M. Longc. On the difficulty of approximately maximizing agreements. *Journal of Computer and System Sciences*, 66(3):496–514, 2003.

Steffen Bickel and Tobias Scheffer. Dirichlet-enhanced spam filtering based on biased samples. In *Advances in Neural Information Processing Systems*, pages 161–168, 2007.

Christopher M. Bishop. *Neural Networks for Pattern Recognition.* Oxford university press, 1995.

William G. Cochran. *Sampling Techniques.* John Wiley & Sons, 2007.

Corinna Cortes, Mehryar Mohri, Michael Riley, and Afshin Rostamizadeh. Sample selection bias correction theory. In *International Conference on Algorithmic Learning Theory*, pages 38–53. Springer, 2008.

Richard O. Duda, Peter E. Hart, and David G. Stork. *Pattern Classification.* John Wiley & Sons, 2012.

George Fishman. *Monte Carlo: Concepts, Algorithms, and Applications.* Springer Science & Business Media, 2013.

Hirotaka Hachiya, Masashi Sugiyama, and Naonori Ueda. Importance-weighted least-squares probabilistic classifier for covariate shift adaptation with application to human activity recognition. *Neurocomputing*, 80:93–101, 2012.

Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning: Data mining, Inference, and Prediction.* Springer Science & Business Media, 2009.

Jiayuan Huang, Arthur Gretton, Karsten Borgwardt, Bernhard Schölkopf, and Alex J Smola. Correcting sample selection bias by unlabeled data. In *Advances in Neural Information Processing Systems*, pages 601–608, 2007.

Suwicha Jirayucharoensak, Setha Pan-Ngum, and Pasin Israsena. EEG-based emotion recognition using deep learning network with principal component based covariate shift adaptation. *The Scientific World Journal*, 2014, 2014.

Herman Kahn and Andy W Marshall. Methods of reducing sample size in Monte Carlo computations. *Journal of the Operations Research Society of America*, 1(5):263–278, 1953.

Takafumi Kanamori, Shohei Hido, and Masashi Sugiyama. A least-squares approach to direct importance estimation. *Journal of Machine Learning Research*, 10(7):1391–1445, 2009.

Jingchen Ke, Chen Gong, Tongliang Liu, Lin Zhao, and Dacheng Tao. Laplacian welsch regularization for robust semisupervised learning. *IEEE Transactions on Cybernetics*, PP (99):1–14, 2020.

Vladimir Koltchinskii. Rademacher penalties and structural risk minimization. *IEEE Transactions on Information Theory*, 47(5):1902–1914, July 2001.

Michel Ledoux and Michel Talagrand. *Probability in Banach Spaces: Isoperimetry and Processes*. Springer Science & Business Media, 2013.

Yan Li, Hiroyuki Kambara, Yasuharu Koike, and Masashi Sugiyama. Application of covariate shift adaptation techniques in brain–computer interfaces. *IEEE Transactions on Biomedical Engineering*, 57(6):1318–1324, 2010.

Joaquin Quionero-Candela, Masashi Sugiyama, Anton Schwaighofer, and Neil D. Lawrence. *Dataset Shift in Machine Learning*. The MIT Press, 2009.

Herbert Robbins and Sutton Monro. A stochastic approximation method. *The Annals of Mathematical Statistics*, pages 400–407, 1951.

Bernhard Schölkopf and Alexander J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT press, 2001.

Hidetoshi Shimodaira. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference*, 90(2):227–244, 2000.

Masashi Sugiyama and Motoaki Kawanabe. *Machine Learning in Non-stationary Environments: Introduction to Covariate Shift Adaptation*. MIT press, 2012.

Masashi Sugiyama and Klaus-Robert Müller. Input-dependent estimation of generalization error under covariate shift. *Statistics & Decisions*, 23(4):249–279, 2005.

Masashi Sugiyama, Matthias Krauledat, and Klaus-Robert Müller. Covariate shift adaptation by importance weighted cross validation. *Journal of Machine Learning Research*, 8 (5):985–1005, 2007.

Masashi Sugiyama, Shinichi Nakajima, Hisashi Kashima, Paul von Buenau, and Motoaki Kawanabe. Direct importance estimation with model selection and its application to covariate shift adaptation. In *Advances in Neural Information Processing Systems*, pages 1433–1440, 2008.

Vladimir N. Vapnik. *Statistical Learning Theory*. Wiley-Interscience, 1998.

Grace Wahba. *Spline Models for Observational Data*, volume 59. SIAM, 1990.

Frank Wilcoxon. Individual comparisons by ranking methods. *Biometrics Bulletin*, 1(6): 80–83, 1945.

Makoto Yamada, Masashi Sugiyama, and Tomoko Matsui. Semi-supervised speaker identification under covariate shift. *Signal Processing*, 90(8):2353–2361, 2010.

Makoto Yamada, Taiji Suzuki, Takafumi Kanamori, Hirotaka Hachiya, and Masashi Sugiyama. Relative density-ratio estimation for robust distribution comparison. In *Advances in Neural Information Processing Systems*, pages 594–602, 2011.

Xulei Yang, Qing Song, and Yue Wang. A weighted support vector machine for data classification. *International Journal of Pattern Recognition and Artificial Intelligence*, 21 (05):961–976, 2007.

Bianca Zadrozny. Learning and evaluating classifiers under sample selection bias. In *Proceedings of the Twenty-first International Conference on Machine Learning*, page 114, 2004.

Appendix A. Proof of Lemma 3

Proof Let $\Phi(S) = \sup_{f\in\mathcal{F},g\in\mathcal{G}_+}\left(J_{\mathrm{UB}}(f,g) - \widehat{J}_{\mathrm{UB}}(f,g;S)\right)$ and S' be a set differing from S on exactly one sample point. Then, since the difference of suprema does not exceed the supremum of the difference, we have

$$\Phi(S') - \Phi(S) \le \sup_{f\in\mathcal{F},g\in\mathcal{G}_+}\left(\widehat{J}_{\mathrm{UB}}(f,g;S) - \widehat{J}_{\mathrm{UB}}(f,g;S')\right).$$

If the differing sample point is a training sample, then

$$\Phi(S') - \Phi(S) \le 2MG\cdot\frac{2}{n_{\mathrm{tr}}}MG + M^2\cdot\frac{1}{n_{\mathrm{tr}}}G^2 = \frac{5}{n_{\mathrm{tr}}}M^2G^2.$$

On the other hand, if the differing sample point is a test sample, then

$$\Phi(S') - \Phi(S) \le M^2\cdot\frac{2}{n_{\mathrm{te}}}\cdot 2G \le \frac{5}{n_{\mathrm{te}}}M^2G^2.$$

Similarly, we can obtain the same result for bounding $\Phi(S)-\Phi(S')$. Then, by McDiarmid's inequality, for any $\delta > 0$, with probability at least $1-\delta$, the following holds:

$$\Phi(S) \le \mathbb{E}_S[\Phi(S)] + 5M^2G^2\sqrt{\frac{\log\frac{1}{\delta}}{2}}\left(\frac{1}{\sqrt{n_{\mathrm{tr}}}} + \frac{1}{\sqrt{n_{\mathrm{te}}}}\right).$$

Let $S_{\mathrm{tr}} = \left\{\left(\boldsymbol{x}_i^{\mathrm{tr}}, y_i^{\mathrm{tr}}\right)\right\}_{i=1}^{n_{\mathrm{tr}}}$ and $S_{\mathrm{te}} = \left\{\boldsymbol{x}_i^{\mathrm{te}}\right\}_{i=1}^{n_{\mathrm{te}}}$. We next bound the expectation in the right-hand side:

$$\mathbb{E}_S[\Phi(S)] = \mathbb{E}_S\left[\sup_{f\in\mathcal{F},g\in\mathcal{G}_+}\left(J_{\mathrm{UB}}(f,g) - \widehat{J}_{\mathrm{UB}}(f,g;S)\right)\right] \le \text{(I)} + M^2\text{(II)} + 2M^2\text{(III)},$$

where

$$\begin{aligned}
\text{(I)} &= \mathbb{E}_{S_{\mathrm{tr}}}\Bigg[\sup_{f\in\mathcal{F},g\in\mathcal{G}_+}\Bigg(\left(\mathbb{E}_{(\boldsymbol{x}^{\mathrm{tr}},y^{\mathrm{tr}})}\left[\ell_{\mathrm{UB}}(f(\boldsymbol{x}^{\mathrm{tr}}),y^{\mathrm{tr}})g(\boldsymbol{x}^{\mathrm{tr}})\right]\right)^2 \\
&\qquad - \left(\frac{1}{n_{\mathrm{tr}}}\sum_{i=1}^{n_{\mathrm{tr}}}\ell_{\mathrm{UB}}(f(\boldsymbol{x}_i^{\mathrm{tr}}),y_i^{\mathrm{tr}})g(\boldsymbol{x}_i^{\mathrm{tr}})\right)^2\Bigg)\Bigg], \\
\text{(II)} &= \mathbb{E}_{S_{\mathrm{tr}}}\left[\sup_{g\in\mathcal{G}_+}\left(\mathbb{E}_{\boldsymbol{x}^{\mathrm{tr}}}\left[g^2(\boldsymbol{x}^{\mathrm{tr}})\right] - \frac{1}{n_{\mathrm{tr}}}\sum_{i=1}^{n_{\mathrm{tr}}}g^2(\boldsymbol{x}_i^{\mathrm{tr}})\right)\right], \\
\text{(III)} &= \mathbb{E}_{S_{\mathrm{te}}}\left[\sup_{g\in\mathcal{G}_+}\left(\frac{1}{n_{\mathrm{te}}}\sum_{i=1}^{n_{\mathrm{te}}}g(\boldsymbol{x}_i^{\mathrm{te}}) - \mathbb{E}_{\boldsymbol{x}^{\mathrm{te}}\sim p_{\mathrm{te}}(\boldsymbol{x})}\left[g(\boldsymbol{x}^{\mathrm{te}})\right]\right)\right].
\end{aligned}$$

Then we bound the above three terms as follows:

$$
\begin{aligned}
\text{(I)} \leq\ & \mathbb{E}_{S_{\mathrm{tr}}}\left[\sup_{f\in\mathcal{F},g\in\mathcal{G}_+}\left(\mathbb{E}_{\tilde{S}_{\mathrm{tr}}}\left(\frac{1}{n_{\mathrm{tr}}}\sum_{i=1}^{n_{\mathrm{tr}}}\ell_{\mathrm{UB}}(f(\tilde{\boldsymbol{x}}_i^{\mathrm{tr}}),\tilde{y}_i^{\mathrm{tr}})g(\tilde{\boldsymbol{x}}_i^{\mathrm{tr}})\right)^2-\left(\frac{1}{n_{\mathrm{tr}}}\sum_{i=1}^{n_{\mathrm{tr}}}\ell_{\mathrm{UB}}(f(\boldsymbol{x}_i^{\mathrm{tr}}),y_i^{\mathrm{tr}})g(\boldsymbol{x}_i^{\mathrm{tr}})\right)^2\right)\right]\\
& \text{(points in } \tilde{S}_{\mathrm{tr}} \text{ are sampled in an i.i.d. fashion from } p_{\mathrm{tr}}(\boldsymbol{x},y))\\
\leq\ & \mathbb{E}_{S_{\mathrm{tr}},\tilde{S}_{\mathrm{tr}}}\left[\sup_{f\in\mathcal{F},g\in\mathcal{G}_+}\left(\left(\frac{1}{n_{\mathrm{tr}}}\sum_{i=1}^{n_{\mathrm{tr}}}\ell_{\mathrm{UB}}(f(\tilde{\boldsymbol{x}}_i^{\mathrm{tr}}),\tilde{y}_i^{\mathrm{tr}})g(\tilde{\boldsymbol{x}}_i^{\mathrm{tr}})\right)^2-\left(\frac{1}{n_{\mathrm{tr}}}\sum_{i=1}^{n_{\mathrm{tr}}}\ell_{\mathrm{UB}}(f(\boldsymbol{x}_i^{\mathrm{tr}}),y_i^{\mathrm{tr}})g(\boldsymbol{x}_i^{\mathrm{tr}})\right)^2\right)\right]\\
\leq\ & \mathbb{E}_{S_{\mathrm{tr}},\tilde{S}_{\mathrm{tr}}}\left[\sup_{f\in\mathcal{F},g\in\mathcal{G}_+}\frac{2MG}{n_{\mathrm{tr}}}\left|\sum_{i=1}^{n_{\mathrm{tr}}}\left(\ell_{\mathrm{UB}}(f(\tilde{\boldsymbol{x}}_i^{\mathrm{tr}}),\tilde{y}_i^{\mathrm{tr}})g(\tilde{\boldsymbol{x}}_i^{\mathrm{tr}})-\ell_{\mathrm{UB}}(f(\boldsymbol{x}_i^{\mathrm{tr}}),y_i^{\mathrm{tr}})g(\boldsymbol{x}_i^{\mathrm{tr}})\right)\right|\right]\\
=\ & \mathbb{E}_{\boldsymbol{\sigma},S_{\mathrm{tr}},\tilde{S}_{\mathrm{tr}}}\left[\sup_{f\in\mathcal{F},g\in\mathcal{G}_+}\frac{2MG}{n_{\mathrm{tr}}}\left|\sum_{i=1}^{n_{\mathrm{tr}}}\sigma_i\left(\ell_{\mathrm{UB}}(f(\tilde{\boldsymbol{x}}_i^{\mathrm{tr}}),\tilde{y}_i^{\mathrm{tr}})g(\tilde{\boldsymbol{x}}_i^{\mathrm{tr}})-\ell_{\mathrm{UB}}(f(\boldsymbol{x}_i^{\mathrm{tr}}),y_i^{\mathrm{tr}})g(\boldsymbol{x}_i^{\mathrm{tr}})\right)\right|\right]\\
& (\{\sigma_i\}_{i=1}^{n_{\mathrm{tr}}} \text{ is a Rademacher sequence})\\
\leq\ & 4MG\mathbb{E}_{\boldsymbol{\sigma},S_{\mathrm{tr}}}\left[\sup_{f\in\mathcal{F},g\in\mathcal{G}_+}\left|\frac{1}{n_{\mathrm{tr}}}\sum_{i=1}^{n_{\mathrm{tr}}}\sigma_i\ell_{\mathrm{UB}}(f(\boldsymbol{x}_i^{\mathrm{tr}}),y_i^{\mathrm{tr}})g(\boldsymbol{x}_i^{\mathrm{tr}})\right|\right]\\
\leq\ & 4MG\mathbb{E}_{\boldsymbol{\sigma},S_{\mathrm{tr}}}\left[\sup_{f\in\mathcal{F},g\in\mathcal{G}}\frac{1}{n_{\mathrm{tr}}}\sum_{i=1}^{n_{\mathrm{tr}}}\sigma_i\ell_{\mathrm{UB}}(f(\boldsymbol{x}_i^{\mathrm{tr}}),y_i^{\mathrm{tr}})g(\boldsymbol{x}_i^{\mathrm{tr}})\right]\\
\leq\ & 2MG\Bigg(\mathbb{E}_{\boldsymbol{\sigma},S_{\mathrm{tr}}}\left[\sup_{f\in\mathcal{F},g\in\mathcal{G}}\frac{1}{n_{\mathrm{tr}}}\sum_{i=1}^{n_{\mathrm{tr}}}\sigma_i\left(\ell_{\mathrm{UB}}(f(\boldsymbol{x}_i^{\mathrm{tr}}),y_i^{\mathrm{tr}})+g(\boldsymbol{x}_i^{\mathrm{tr}})\right)^2\right]\\
& \qquad +\mathbb{E}_{\boldsymbol{\sigma},S_{\mathrm{tr}}}\left[\sup_{f\in\mathcal{F}}\frac{1}{n_{\mathrm{tr}}}\sum_{i=1}^{n_{\mathrm{tr}}}\sigma_i\ell_{\mathrm{UB}}^2(f(\boldsymbol{x}_i^{\mathrm{tr}}),y_i^{\mathrm{tr}})\right]+\mathbb{E}_{\boldsymbol{\sigma},S_{\mathrm{tr}}}\left[\sup_{g\in\mathcal{G}}\frac{1}{n_{\mathrm{tr}}}\sum_{i=1}^{n_{\mathrm{tr}}}\sigma_i g^2(\boldsymbol{x}_i^{\mathrm{tr}})\right]\Bigg)\\
\leq\ & 2MG\Bigg(2\left(M+G\right)\mathbb{E}_{\boldsymbol{\sigma},S_{\mathrm{tr}}}\left[\sup_{f\in\mathcal{F},g\in\mathcal{G}}\frac{1}{n_{\mathrm{tr}}}\sum_{i=1}^{n_{\mathrm{tr}}}\sigma_i\left(\ell_{\mathrm{UB}}(f(\boldsymbol{x}_i^{\mathrm{tr}}),y_i^{\mathrm{tr}})+g(\boldsymbol{x}_i^{\mathrm{tr}})\right)\right]\\
& \qquad +2M\mathbb{E}_{\boldsymbol{\sigma},S_{\mathrm{tr}}}\left[\sup_{f\in\mathcal{F}}\frac{1}{n_{\mathrm{tr}}}\sum_{i=1}^{n_{\mathrm{tr}}}\sigma_i\ell_{\mathrm{UB}}(f(\boldsymbol{x}_i^{\mathrm{tr}}),y_i^{\mathrm{tr}})\right]+2G\mathbb{E}_{\boldsymbol{\sigma},S_{\mathrm{tr}}}\left[\sup_{g\in\mathcal{G}}\frac{1}{n_{\mathrm{tr}}}\sum_{i=1}^{n_{\mathrm{tr}}}\sigma_i g(\boldsymbol{x}_i^{\mathrm{tr}})\right]\Bigg)\\
& \text{(Ledoux-Talagrand contraction lemma (Ledoux and Talagrand, 2013))}\\
\leq\ & 4MG\left(2M+G\right)L\mathfrak{R}_{n_{\mathrm{tr}}}(\mathcal{F})+4MG\left(M+2G\right)\mathfrak{R}_{n_{\mathrm{tr}}}(\mathcal{G}),\ \text{(Ledoux-Talagrand contraction lemma)}
\end{aligned}
$$

$$
\begin{aligned}
\text{(II)} =\ & \mathbb{E}_{S_{\mathrm{tr}}}\left[\sup_{g\in\mathcal{G}_+}\left(\mathbb{E}_{\tilde{S}_{\mathrm{tr}}}\left[\frac{1}{n_{\mathrm{tr}}}\sum_{i=1}^{n_{\mathrm{tr}}}g^2(\tilde{\boldsymbol{x}}_i^{\mathrm{tr}})\right]-\frac{1}{n_{\mathrm{tr}}}\sum_{i=1}^{n_{\mathrm{tr}}}g^2(\boldsymbol{x}_i^{\mathrm{tr}})\right)\right]\\
\leq\ & \mathbb{E}_{S_{\mathrm{tr}},\tilde{S}_{\mathrm{tr}}}\left[\sup_{g\in\mathcal{G}_+}\frac{1}{n_{\mathrm{tr}}}\sum_{i=1}^{n_{\mathrm{tr}}}\left(g^2(\tilde{\boldsymbol{x}}_i^{\mathrm{tr}})-g^2(\boldsymbol{x}_i^{\mathrm{tr}})\right)\right]=\mathbb{E}_{\boldsymbol{\sigma},S_{\mathrm{tr}},\tilde{S}_{\mathrm{tr}}}\left[\sup_{g\in\mathcal{G}_+}\frac{1}{n_{\mathrm{tr}}}\sum_{i=1}^{n_{\mathrm{tr}}}\sigma_i\left(g^2(\tilde{\boldsymbol{x}}_i^{\mathrm{tr}})-g^2(\boldsymbol{x}_i^{\mathrm{tr}})\right)\right]\\
\leq\ & 2\mathbb{E}_{\boldsymbol{\sigma},S_{\mathrm{tr}}}\left[\sup_{g\in\mathcal{G}_+}\frac{1}{n_{\mathrm{tr}}}\sum_{i=1}^{n_{\mathrm{tr}}}\sigma_i g^2(\boldsymbol{x}_i^{\mathrm{tr}})\right]\\
\leq\ & 4G\mathfrak{R}_{n_{\mathrm{tr}}}(\mathcal{G}),\quad \text{(Ledoux-Talagrand contraction lemma)}
\end{aligned}
$$

$$
\begin{aligned}
\text{(III)} &= \mathbb{E}_{S_{\mathrm{te}}}\left[\sup_{g\in\mathcal{G}_+}\left(\frac{1}{n_{\mathrm{te}}}\sum_{i=1}^{n_{\mathrm{te}}} g(\boldsymbol{x}_i^{\mathrm{te}}) - \mathbb{E}_{\tilde{S}_{\mathrm{te}}}\left[\frac{1}{n_{\mathrm{te}}}\sum_{i=1}^{n_{\mathrm{te}}} g(\tilde{\boldsymbol{x}}_i^{\mathrm{te}})\right]\right)\right] \\
&\leq \mathbb{E}_{S_{\mathrm{te}},\tilde{S}_{\mathrm{te}}}\left[\sup_{g\in\mathcal{G}_+}\frac{1}{n_{\mathrm{te}}}\sum_{i=1}^{n_{\mathrm{te}}}\left(g(\boldsymbol{x}_i^{\mathrm{te}}) - g(\tilde{\boldsymbol{x}}_i^{\mathrm{te}})\right)\right] = \mathbb{E}_{\boldsymbol{\sigma},S_{\mathrm{te}},\tilde{S}_{\mathrm{te}}}\left[\sup_{g\in\mathcal{G}_+}\frac{1}{n_{\mathrm{te}}}\sum_{i=1}^{n_{\mathrm{te}}}\sigma_i\left(g(\boldsymbol{x}_i^{\mathrm{te}}) - g(\tilde{\boldsymbol{x}}_i^{\mathrm{te}})\right)\right] \\
&\leq 2\mathbb{E}_{\boldsymbol{\sigma},S_{\mathrm{te}}}\left[\sup_{g\in\mathcal{G}_+}\frac{1}{n_{\mathrm{te}}}\sum_{i=1}^{n_{\mathrm{te}}}\sigma_i g(\boldsymbol{x}_i^{\mathrm{te}})\right] \leq 2\mathfrak{R}_{n_{\mathrm{te}}}(\mathcal{G}).
\end{aligned}
$$

By summarizing all the results above, we complete the proof.[4] ■

Appendix B. Proof of Theorem 5

Proof Let $(f_J^*, g_J^*) = \arg\min_{(f,g)\in\mathcal{F}\times\mathcal{G}} J_{\mathrm{UB}}(f,g)$. Then for any $\delta > 0$, by McDiarmid's inequality, with probability at least $1-\delta$, we have

$$
\widehat{J}_{\mathrm{UB}}(f_J^*, g_J^*; S) \leq \mathbb{E}_S[\widehat{J}_{\mathrm{UB}}(f_J^*, g_J^*; S)] + 5M^2G^2\sqrt{\frac{\log\frac{1}{\delta}}{2}}\left(\frac{1}{\sqrt{n_{\mathrm{tr}}}} + \frac{1}{\sqrt{n_{\mathrm{te}}}}\right).
$$

Since $\mathbb{E}\left[X^2\right] = (\mathbb{E}[X])^2 + \mathrm{Var}[X]$, we have

$$
\mathbb{E}_S[\widehat{J}_{\mathrm{UB}}(f_J^*, g_J^*; S)] = J_{\mathrm{UB}}(f_J^*, g_J^*) + \frac{1}{n_{\mathrm{tr}}}\mathrm{Var}\left[\ell_{\mathrm{UB}}(f(\tilde{\boldsymbol{x}}_1^{\mathrm{tr}}), \tilde{y}_1^{\mathrm{tr}})g(\tilde{\boldsymbol{x}}_1^{\mathrm{tr}})\right] \leq J_{\mathrm{UB}}(f_J^*, g_J^*) + \frac{1}{n_{\mathrm{tr}}}M^2G^2,
$$

and thus,

$$
\widehat{J}_{\mathrm{UB}}(f_J^*, g_J^*; S) \leq J_{\mathrm{UB}}(f_J^*, g_J^*) + 5M^2G^2\sqrt{\frac{\log\frac{1}{\delta}}{2}}\left(\frac{1}{\sqrt{n_{\mathrm{tr}}}} + \frac{1}{\sqrt{n_{\mathrm{te}}}}\right) + M^2G^2\frac{1}{n_{\mathrm{tr}}}.
$$

Therefore, according to (2), (3) and (6), we have

$$
\begin{aligned}
&\frac{1}{2}R^2(\widehat{f}) - J_{\mathrm{UB}}(f_J^*, g_J^*) \\
&\leq \left(J_{\mathrm{UB}}(\widehat{f},\widehat{g}) - \widehat{J}_{\mathrm{UB}}(\widehat{f},\widehat{g};S)\right) + \left(\widehat{J}_{\mathrm{UB}}(\widehat{f},\widehat{g};S) - \widehat{J}_{\mathrm{UB}}(f_J^*, g_J^*;S)\right) + \left(\widehat{J}_{\mathrm{UB}}(f_J^*, g_J^*;S) - J_{\mathrm{UB}}(f_J^*, g_J^*)\right) \\
&\leq \left(8MG\left(M+G\right)\left(L\mathfrak{R}_{n_{\mathrm{tr}}}^{\mathrm{tr}}(\mathcal{F}) + \mathfrak{R}_{n_{\mathrm{tr}}}^{\mathrm{tr}}(\mathcal{G})\right) + 4M^2\mathfrak{R}_{n_{\mathrm{te}}}^{\mathrm{te}}(\mathcal{G}) + 5M^2G^2\sqrt{\frac{\log\frac{1}{\delta}}{2}}\left(\frac{1}{\sqrt{n_{\mathrm{tr}}}} + \frac{1}{\sqrt{n_{\mathrm{te}}}}\right)\right) \\
&\quad + 0 + \left(5M^2G^2\sqrt{\frac{\log\frac{1}{\delta}}{2}}\left(\frac{1}{\sqrt{n_{\mathrm{tr}}}} + \frac{1}{\sqrt{n_{\mathrm{te}}}}\right) + M^2G^2\frac{1}{n_{\mathrm{tr}}}\right).
\end{aligned}
$$

Rearranging the equation above, we obtain Eq. (8). ■

[4] In fact, the bound presented in Lemma 3 is looser than the result that we obtained here. We did this for saving the space and making the bound more readable.