In J. E. Gentle, W. Härdle, and Y. Mori (Eds.),
Handbook of Computational Statistics: Concepts and Methods,
2nd edition. Chapter 31, pp.927–952, Springer, Berlin, Germany, 2012.

Learning under Non-stationarity: Covariate Shift Adaptation by Importance Weighting

Masashi Sugiyama
Tokyo Institute of Technology
2-12-1 O-okayama, Meguro-ku, Tokyo 152-8552, Japan.
sugi@cs.titech.ac.jp http://sugiyama-www.cs.titech.ac.jp/˜sugi

Abstract

The goal of supervised learning is to estimate an underlying input-output function from its input-output training samples so that output values for unseen test input points can be predicted. A common assumption in supervised learning is that the training input points follow the *same* probability distribution as the test input points. However, this assumption is not satisfied, for example, when outside of the training region is extrapolated. The situation where the training and test input points follow *different* distributions while the conditional distribution of output values given input points is unchanged is called *covariate shift*. Since almost all existing learning methods assume that the training and test samples are drawn from the same distribution, their fundamental theoretical properties such as consistency or efficiency no longer hold under covariate shift. In this chapter, we review recently proposed techniques for covariate shift adaptation.

1 Introduction

The goal of supervised learning is to infer an unknown input-output dependency from training samples, by which output values for unseen test input points can be predicted. When developing a method of supervised learning, it is commonly assumed that the input points in the training set and the input points used for testing follow the *same* probability distribution (Wahba, 1990; Bishop, 1995; Vapnik, 1998; Duda et al., 2001; Hastie et al., 2001; Schölkopf & Smola, 2002). However, this common assumption is not fulfilled, for example, when outside of the training region is extrapolated or when training input points are designed by an active learning (a.k.a. experimental design) algorithm (Wiens, 2000; Kanamori & Shimodaira, 2003; Sugiyama, 2006; Kanamori, 2007; Sugiyama & Nakajima, 2009). Situations where training and test input points follow different probability distributions but the conditional distributions of output values given input points are unchanged are called *covariate shift* (Shimodaira, 2000). In this chapter, we review recently proposed techniques for alleviating for the influence of covariate shift.

Under covariate shift, standard learning techniques such as maximum likelihood estimation are biased. It was shown that the bias caused by covariate shift can be asymptotically canceled by weighting the loss function according to the *importance*—the ratio of test and training input densities (Shimodaira, 2000; Zadrozny, 2004; Sugiyama & Müller, 2005; Sugiyama et al., 2007; Quiñonero-Candela et al., 2009; Sugiyama & Kawanabe, 2011). Similarly, standard model selection criteria such as cross-validation (Stone, 1974; Wahba, 1990) or Akaike's information criterion (Akaike, 1974) lose their unbiasedness under covariate shift. It was shown that proper unbiasedness can also be recovered by modifying the methods based on importance weighting (Shimodaira, 2000; Zadrozny, 2004; Sugiyama & Müller, 2005; Sugiyama et al., 2007).

As explained above, the importance weight plays a central role in covariate shift adaptation. However, since the importance weight is unknown in practice, it should be estimated from data. A naive approach to this task is to first use kernel density estimation (Härdle et al., 2004) for obtaining estimators of the training and test input densities, and then taking the ratio of the estimated densities. However, division by estimated quantities can magnify the estimation error, so directly estimating the importance weight in a single-shot process would be more preferable. Following this idea, various methods for directly estimating the importance have been explored (Silverman, 1978; Ćwik & Mielniczuk, 1989; Qin, 1998; Cheng & Chu, 2004; Huang et al., 2007; Bickel et al., 2007; Sugiyama et al., 2008; Kanamori et al., 2009a). These direct estimation approaches have been demonstrated to be more accurate than the two-step density estimation approach.

Examples of successful real-world applications of covariate shift adaptation include brain-computer interface (Sugiyama et al., 2007), robot control (Hachiya et al., 2009; Akiyama et al., 2010; Hachiya et al., 2011), speaker identification (Yamada et al., 2010a), age prediction from face images (Ueki et al., 2011), wafer alignment in semiconductor exposure apparatus (Sugiyama & Nakajima, 2009), and natural language processing (Tsuboi et al., 2009).

The rest of this chapter is organized as follows. In Section 2, the problem of supervised learning under covariate shift is mathematically formulated. In Section 3, various learning methods under covariate shift are introduced. In Section 4, the issue of model selection under covariate shift is addressed. In Section 5, methods of importance estimation are reviewed. Finally, we conclude in Section 6.

A more extensive description of covariate shift adaptation techniques is available in Sugiyama and Kawanabe (2011).

2 Formulation of Supervised Learning under Covariate Shift

In this section, we formulate the supervised learning problem under covariate shift.

Let us consider the supervised learning problem of estimating an unknown input-

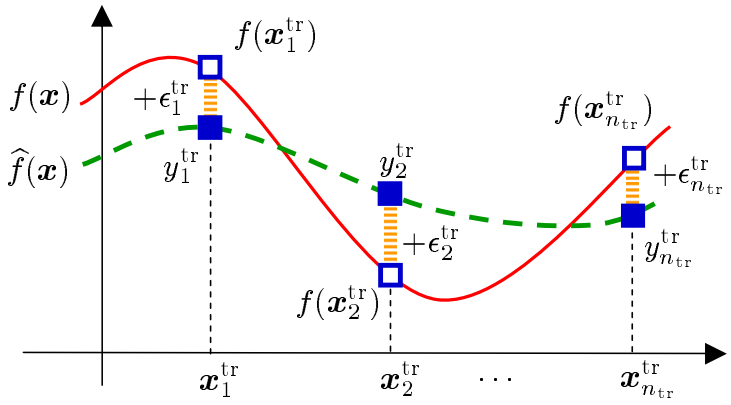


Figure 1: Framework of supervised learning.

output dependency from training samples. Let

$$\{(\boldsymbol{x}_i^{\mathrm{tr}}, y_i^{\mathrm{tr}}) \mid \boldsymbol{x}_i^{\mathrm{tr}} \in \mathcal{X} \subset \mathbb{R}^d, y_i^{\mathrm{tr}} \in \mathcal{Y} \subset \mathbb{R}\}_{i=1}^{n_{\mathrm{tr}}},$$

be the training samples. $\boldsymbol{x}_i^{\mathrm{tr}}$ is a training input point drawn from a probability distribution with density $p_{\mathrm{tr}}(\boldsymbol{x})$. y_i^{tr} is a training output value following a conditional probability distribution with conditional density $p(y|\boldsymbol{x} = \boldsymbol{x}_i^{\mathrm{tr}})$. $p(y|\boldsymbol{x})$ may be regarded as the sum of the true output $f(\boldsymbol{x})$ and noise ϵ:

$$y = f(\boldsymbol{x}) + \epsilon.$$

We assume that the noise ϵ has mean 0 and variance σ^2. This formulation is summarized in Figure 1.

Let $(\boldsymbol{x}^{\mathrm{te}}, y^{\mathrm{te}})$ be a test sample, which is not given to the user in the training phase, but will be provided in the test phase in the future. $\boldsymbol{x}^{\mathrm{te}} \in \mathcal{X}$ is a test input point following a probability distribution with density $p_{\mathrm{te}}(\boldsymbol{x})$, which is different from $p_{\mathrm{tr}}(\boldsymbol{x})$. $y^{\mathrm{te}} \in \mathcal{Y}$ is a test output value following $p(y|\boldsymbol{x} = \boldsymbol{x}^{\mathrm{te}})$, which is the same conditional density as the training phase. The goal of supervised learning is to obtain an approximation $\widehat{f}(\boldsymbol{x})$ to the true function $f(\boldsymbol{x})$ for predicting the test output value y^{te}. More formally, we would like to obtain the approximation $\widehat{f}(\boldsymbol{x})$ that minimizes the test error expected over all test samples (which is called the *generalization error*):

$$G := \mathop{\mathbb{E}}_{\boldsymbol{x}^{\mathrm{te}}} \mathop{\mathbb{E}}_{y^{\mathrm{te}}} \left[\mathrm{loss}(\widehat{f}(\boldsymbol{x}^{\mathrm{te}}), y^{\mathrm{te}}) \right],$$

where $\mathbb{E}_{\boldsymbol{x}^{\mathrm{te}}}$ denotes the expectation over $\boldsymbol{x}^{\mathrm{te}}$ drawn from $p_{\mathrm{te}}(\boldsymbol{x})$ and $\mathbb{E}_{y^{\mathrm{te}}}$ denotes the expectation over y^{te} drawn from $p(y|\boldsymbol{x} = \boldsymbol{x}^{\mathrm{te}})$. $\mathrm{loss}(\widehat{y}, y)$ is the loss function which measures the discrepancy between the true output value y and its estimate $\widehat{y}$. When the output domain $\mathcal{Y}$ is continuous, the problem is called *regression* and the *squared-loss* is often used.

$$\mathrm{loss}(\widehat{y}, y) = (\widehat{y} - y)^2.$$

On the other hand, when $\mathcal{Y} = \{+1, -1\}$, the problem is called (binary) *classification* and the *0/1-loss* is a typical choice.

$$\mathrm{loss}(\widehat{y}, y) = \begin{cases} 0 & \text{if } \mathrm{sgn}(\widehat{y}) = y, \\ 1 & \text{otherwise,} \end{cases}$$

where $\mathrm{sgn}(y) = +1$ if $y \geq 0$ and $\mathrm{sgn}(y) = -1$ if $y < 0$. Note that the 0/1-loss corresponds to the misclassification rate.

We use a parametric function $\widehat{f}(\boldsymbol{x}; \boldsymbol{\theta})$ for learning, where $\boldsymbol{\theta}$ is a parameter. A model $\widehat{f}(\boldsymbol{x}; \boldsymbol{\theta})$ is said to be *correctly specified* if there exists a parameter $\boldsymbol{\theta}^*$ such that $\widehat{f}(\boldsymbol{x}; \boldsymbol{\theta}^*) = f(\boldsymbol{x})$; otherwise the model is said to be *misspecified*. In practice, the model used for learning would be misspecified to a greater or less extent since we do not generally have enough prior knowledge for correctly specifying the model. Thus it is important to consider misspecified models when developing machine learning algorithms.

In standard supervised learning theories (Wahba, 1990; Bishop, 1995; Vapnik, 1998; Duda et al., 2001; Hastie et al., 2001; Schölkopf & Smola, 2002), the test input point $\boldsymbol{x}^{\mathrm{te}}$ is assumed to follow the same distribution as the training input point $\boldsymbol{x}^{\mathrm{tr}}$. On the other hand, in this chapter, we consider the situation called *covariate shift* (Shimodaira, 2000), i.e., the training input point $\boldsymbol{x}^{\mathrm{tr}}$ and the test input point $\boldsymbol{x}^{\mathrm{te}}$ have different distributions. Under covariate shift, most of the standard learning techniques do not work well due to the differing distributions. Below, we review recently developed techniques for mitigating the influence of covariate shift.

3 Function Learning under Covariate Shift

A standard method to learn the parameter $\boldsymbol{\theta}$ in the model $\widehat{f}(\boldsymbol{x}; \boldsymbol{\theta})$ would be *empirical risk minimization* (ERM) (Vapnik, 1998; Schölkopf & Smola, 2002):

$$\widehat{\boldsymbol{\theta}}_{\mathrm{ERM}} := \operatorname*{argmin}_{\boldsymbol{\theta}} \left[\frac{1}{n_{\mathrm{tr}}} \sum_{i=1}^{n_{\mathrm{tr}}} \mathrm{loss}(\widehat{f}(\boldsymbol{x}_i^{\mathrm{tr}}; \boldsymbol{\theta}), y_i^{\mathrm{tr}}) \right].$$

If $p_{\mathrm{tr}}(\boldsymbol{x}) = p_{\mathrm{te}}(\boldsymbol{x})$, $\widehat{\boldsymbol{\theta}}_{\mathrm{ERM}}$ converges to the optimal parameter $\boldsymbol{\theta}^*$ (Shimodaira, 2000):

$$\boldsymbol{\theta}^* := \operatorname*{argmin}_{\boldsymbol{\theta}} [G].$$

However, under covariate shift where $p_{\mathrm{tr}}(\boldsymbol{x}) \neq p_{\mathrm{te}}(\boldsymbol{x})$, $\widehat{\boldsymbol{\theta}}_{\mathrm{ERM}}$ does not converge to $\boldsymbol{\theta}^*$ if the model is misspecified[1].

In this section, we review various learning methods for covariate shift adaptation and show their numerical examples.

[1] $\widehat{\boldsymbol{\theta}}_{\mathrm{ERM}}$ still converges to $\boldsymbol{\theta}^*$ under covariate shift if the model is correctly specified.

3.1 Importance Weighting Techniques for Covariate Shift Adaptation

Here, we introduce various regression and classification techniques for covariate shift adaptation.

3.1.1 Importance Weighted ERM

The inconsistency of ERM is due to the difference between training and test input distributions. *Importance sampling* (Fishman, 1996) is a standard technique to compensate for the difference of distributions. The following identity shows the essence of importance sampling:

$$\mathop{\mathbb{E}}_{\boldsymbol{x}^{\mathrm{te}}}[g(\boldsymbol{x}^{\mathrm{te}})] = \int g(\boldsymbol{x}) p_{\mathrm{te}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} = \int g(\boldsymbol{x}) \frac{p_{\mathrm{te}}(\boldsymbol{x})}{p_{\mathrm{tr}}(\boldsymbol{x})} p_{\mathrm{tr}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} = \mathop{\mathbb{E}}_{\boldsymbol{x}^{\mathrm{tr}}}\left[g(\boldsymbol{x}^{\mathrm{tr}}) \frac{p_{\mathrm{te}}(\boldsymbol{x}^{\mathrm{tr}})}{p_{\mathrm{tr}}(\boldsymbol{x}^{\mathrm{tr}})} \right],$$

where $\mathbb{E}_{\boldsymbol{x}^{\mathrm{tr}}}$ and $\mathbb{E}_{\boldsymbol{x}^{\mathrm{te}}}$ denote the expectations over $\boldsymbol{x}^{\mathrm{tr}}$ and $\boldsymbol{x}^{\mathrm{te}}$ drawn from $p_{\mathrm{tr}}(\boldsymbol{x})$ and $p_{\mathrm{te}}(\boldsymbol{x})$, respectively. The quantity

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

is called the *importance*. The above identity shows that the expectation of a function $g(\boldsymbol{x})$ over $p_{\mathrm{te}}(\boldsymbol{x})$ can be computed by the importance-weighted expectation of $g(\boldsymbol{x})$ over $p_{\mathrm{tr}}(\boldsymbol{x})$. Thus the difference of distributions can be systematically adjusted by importance weighting.

Applying the above importance weighting technique to ERM, we obtain *importance-weighted ERM* (IWERM):

$$\widehat{\boldsymbol{\theta}}_{\mathrm{IWERM}} := \mathop{\mathrm{argmin}}_{\boldsymbol{\theta}} \left[\frac{1}{n_{\mathrm{tr}}} \sum_{i=1}^{n_{\mathrm{tr}}} \frac{p_{\mathrm{te}}(\boldsymbol{x}_i^{\mathrm{tr}})}{p_{\mathrm{tr}}(\boldsymbol{x}_i^{\mathrm{tr}})} \mathrm{loss}(\widehat{f}(\boldsymbol{x}_i^{\mathrm{tr}}; \boldsymbol{\theta}), y_i^{\mathrm{tr}}) \right].$$

$\widehat{\boldsymbol{\theta}}_{\mathrm{IWERM}}$ converges to $\boldsymbol{\theta}^*$ under covariate shift, even if the model is misspecified (Shimodaira, 2000). In practice, IWERM may be *regularized*, e.g., by slightly flattening the importance weight and/or adding a penalty term as

$$\mathop{\mathrm{argmin}}_{\boldsymbol{\theta}} \left[\frac{1}{n_{\mathrm{tr}}} \sum_{i=1}^{n_{\mathrm{tr}}} \left(\frac{p_{\mathrm{te}}(\boldsymbol{x}_i^{\mathrm{tr}})}{p_{\mathrm{tr}}(\boldsymbol{x}_i^{\mathrm{tr}})} \right)^{\gamma} \mathrm{loss}(\widehat{f}(\boldsymbol{x}_i^{\mathrm{tr}}; \boldsymbol{\theta}), y_i^{\mathrm{tr}}) + \lambda \boldsymbol{\theta}^{\top} \boldsymbol{\theta} \right],$$

where $0 \leq \gamma \leq 1$ is the flattening parameter, $\lambda \geq 0$ is the regularization parameter, and $^{\top}$ denotes the transpose of a matrix or a vector.

3.1.2 Importance-Weighted Regression Methods

Least-squares (LS) would be one of the most fundamental regression techniques. The importance-weighted regression method for the squared-loss (see Figure 2), called

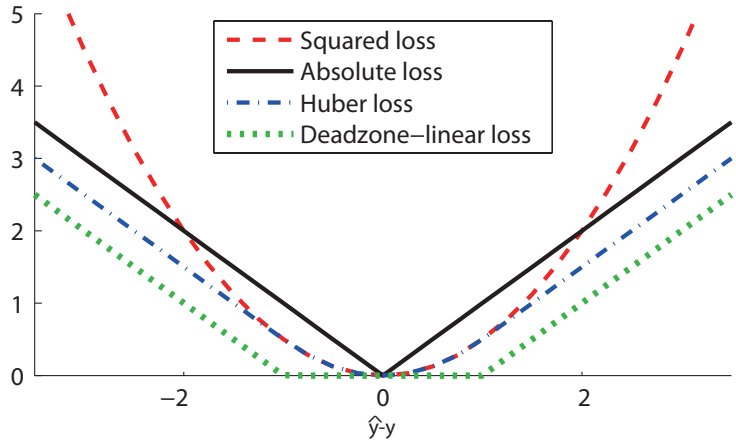


Figure 2: Loss functions for regression. y is the true output value at an input point and $\widehat{y}$ is its estimate.

importance-weighted LS (IWLS), is given as follows:

$$\widehat{\boldsymbol{\theta}}_{\mathrm{IWLS}} := \operatorname*{argmin}_{\boldsymbol{\theta}} \left[\frac{1}{n_{\mathrm{tr}}} \sum_{i=1}^{n_{\mathrm{tr}}} \left(\frac{p_{\mathrm{te}}(\boldsymbol{x}_i^{\mathrm{tr}})}{p_{\mathrm{tr}}(\boldsymbol{x}_i^{\mathrm{tr}})} \right)^{\gamma} \left(\widehat{f}(\boldsymbol{x}_i^{\mathrm{tr}}; \boldsymbol{\theta}) - y_i^{\mathrm{tr}} \right)^2 + \lambda \boldsymbol{\theta}^\top \boldsymbol{\theta} \right]. \tag{1}$$

Let us employ the following linear model:

$$\widehat{f}(\boldsymbol{x}; \boldsymbol{\theta}) = \sum_{\ell=1}^{b} \theta_\ell \phi_\ell(\boldsymbol{x}), \tag{2}$$

where $\{\phi_\ell(\boldsymbol{x})\}_{\ell=1}^b$ are fixed linearly-independent basis functions. Then the solution $\widehat{\boldsymbol{\theta}}_{\mathrm{IWLS}}$ is given *analytically* as

$$\widehat{\boldsymbol{\theta}}_{\mathrm{IWLS}} = (\boldsymbol{X}^{\mathrm{tr}\top} \boldsymbol{W}^\gamma \boldsymbol{X}^{\mathrm{tr}} + n_{\mathrm{tr}} \lambda \boldsymbol{I}_b)^{-1} \boldsymbol{X}^{\mathrm{tr}\top} \boldsymbol{W}^\gamma \boldsymbol{y}^{\mathrm{tr}}, \tag{3}$$

where $\boldsymbol{X}^{\mathrm{tr}}$ is the *design matrix*, i.e., $\boldsymbol{X}^{\mathrm{tr}}$ is the $n_{\mathrm{tr}} \times b$ matrix with the (i, ℓ)-th element $X_{i,\ell}^{\mathrm{tr}} = \phi_\ell(\boldsymbol{x}_i^{\mathrm{tr}})$, $\boldsymbol{W}$ is the diagonal matrix with the i-th diagonal element $\frac{p_{\mathrm{te}}(\boldsymbol{x}_i^{\mathrm{tr}})}{p_{\mathrm{tr}}(\boldsymbol{x}_i^{\mathrm{tr}})}$, $\boldsymbol{I}_b$ is the b-dimensional identity matrix, and $\boldsymbol{y}^{\mathrm{tr}}$ is the n_{tr}-dimensional vector with the i-th element y_i^{tr}.

The LS method often suffers from excessive sensitivity to *outliers* (i.e., irregular values) and less reliability. A popular alternative is *importance-weighted least absolute* (IWLA) regression—instead of the squared loss, the absolute loss is used (see Figure 2):

$$\widehat{\boldsymbol{\theta}}_{\mathrm{IWLA}} = \operatorname*{argmin}_{\boldsymbol{\theta}} \left[\frac{1}{n_{\mathrm{tr}}} \sum_{i=1}^{n_{\mathrm{tr}}} \left(\frac{p_{\mathrm{te}}(\boldsymbol{x}_i^{\mathrm{tr}})}{p_{\mathrm{tr}}(\boldsymbol{x}_i^{\mathrm{tr}})} \right)^{\gamma} \left| \widehat{f}(\boldsymbol{x}_i^{\mathrm{tr}}; \boldsymbol{\theta}) - y_i^{\mathrm{tr}} \right| + \lambda \boldsymbol{\theta}^\top \boldsymbol{\theta} \right].$$

For the linear model (2), the above optimization problem is reduced to a quadratic program, which can be solved by a standard optimization software. If the regularization term

$\boldsymbol{\theta}^\top\boldsymbol{\theta}$ is replaced by the ℓ_1-regularizer $\sum_{\ell=1}^{b}|\theta_\ell|$, the optimization problem is reduced to a linear program, which may be solved more efficiently. Furthermore, the ℓ_1-regularizer is known to induce a *sparse* solution (Williams, 1995; Tibshirani, 1996; Chen et al., 1998).

Although the LA method is robust against outliers, it tends to have a large variance when the noise is Gaussian. The use of the *Huber loss* can mitigate this problem:

$$\widehat{\boldsymbol{\theta}}_{\mathrm{Huber}} = \operatorname*{argmin}_{\boldsymbol{\theta}} \left[\frac{1}{n_{\mathrm{tr}}} \sum_{i=1}^{n_{\mathrm{tr}}} \left(\frac{p_{\mathrm{te}}(\boldsymbol{x}_i^{\mathrm{tr}})}{p_{\mathrm{tr}}(\boldsymbol{x}_i^{\mathrm{tr}})} \right)^{\gamma} \rho_\tau \left(\widehat{f}(\boldsymbol{x}_i^{\mathrm{tr}};\boldsymbol{\theta}) - y_i^{\mathrm{tr}} \right) + \lambda \boldsymbol{\theta}^\top\boldsymbol{\theta} \right],$$

where τ (≥ 0) is the robustness parameter and ρ_τ is the Huber loss defined as follows (see Figure 2):

$$\rho_\tau(y) := \begin{cases} \frac{1}{2}y^2 & \text{if } |y| \leq \tau, \\ \tau|y| - \frac{1}{2}\tau^2 & \text{if } |y| > \tau. \end{cases}$$

Thus, the squared loss is applied to 'good' samples with small fitting error, and the absolute loss is applied to 'bad' samples with large fitting error. The above optimization problem can be reduced to a quadratic program (Mangasarian & Musicant, 2000), which can be solved by a standard optimization software.

Another variant of the IWLA is *importance-weighted support vector regression* (IWSVR):

$$\widehat{\boldsymbol{\theta}}_{\mathrm{SVR}} = \operatorname*{argmin}_{\boldsymbol{\theta}} \left[\frac{1}{n_{\mathrm{tr}}} \sum_{i=1}^{n_{\mathrm{tr}}} \left(\frac{p_{\mathrm{te}}(\boldsymbol{x}_i^{\mathrm{tr}})}{p_{\mathrm{tr}}(\boldsymbol{x}_i^{\mathrm{tr}})} \right)^{\gamma} \left| \widehat{f}(\boldsymbol{x}_i^{\mathrm{tr}};\boldsymbol{\theta}) - y_i^{\mathrm{tr}} \right|_\epsilon + \lambda \boldsymbol{\theta}^\top\boldsymbol{\theta} \right],$$

where $|\cdot|_\epsilon$ is the *deadzone-linear loss* (or *Vapnik's ϵ-insensitive loss*) defined as follows (see Figure 2):

$$|x|_\epsilon := \begin{cases} 0 & \text{if } |x| \leq \epsilon, \\ |x| - \epsilon & \text{if } |x| > \epsilon. \end{cases}$$

For the linear model (2), the above optimization problem is reduced to a quadratic program (Vapnik, 1998), which can be solved by a standard optimization software. If the regularization term $\boldsymbol{\theta}^\top\boldsymbol{\theta}$ is replaced by the ℓ_1-regularizer $\sum_{\ell=1}^{b}|\theta_\ell|$, the optimization problem is reduced to a linear program.

3.1.3 Importance-Weighted Classification Methods

In the binary classification scenario where $\mathcal{Y} = \{+1, -1\}$, *Fisher discriminant analysis* (FDA) (Fisher, 1936), *logistic regression* (LR) (Hastie et al., 2001), *support vector machine* (SVM) (Vapnik, 1998; Schölkopf & Smola, 2002), and *boosting* (Freund & Schapire, 1996; Breiman, 1998; Friedman et al., 2000) would be popular learning algorithms. They can be regarded as ERM methods with different loss functions (see Figure 3).

An importance-weighted version of FDA, IWFDA, is given by

$$\widehat{\boldsymbol{\theta}}_{\mathrm{IWFDA}} := \operatorname*{argmin}_{\boldsymbol{\theta}} \left[\frac{1}{n_{\mathrm{tr}}} \sum_{i=1}^{n_{\mathrm{tr}}} \left(\frac{p_{\mathrm{te}}(\boldsymbol{x}_i^{\mathrm{tr}})}{p_{\mathrm{tr}}(\boldsymbol{x}_i^{\mathrm{tr}})} \right)^{\gamma} \left(1 - y_i^{\mathrm{tr}} \widehat{f}(\boldsymbol{x}_i^{\mathrm{tr}};\boldsymbol{\theta}) \right)^2 + \lambda \boldsymbol{\theta}^\top\boldsymbol{\theta} \right],$$

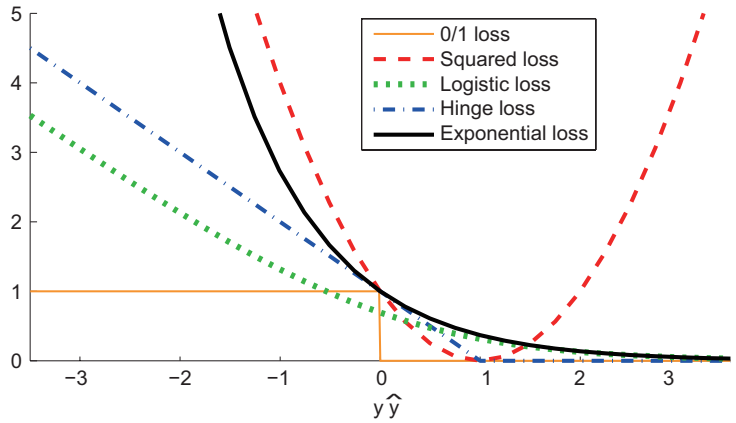


Figure 3: Loss functions for classification. y is the true output value at an input point and $\widehat{y}$ is its estimate.

which is essentially equivalent to Eq.(1) since $(y_i^{\mathrm{tr}})^2 = 1$.

An importance-weighted version of LR, IWLR, is given by

$$\widehat{\boldsymbol{\theta}}_{\mathrm{IWLR}} := \operatorname*{argmin}_{\boldsymbol{\theta}} \left[\sum_{i=1}^{n_{\mathrm{tr}}} \left(\frac{p_{\mathrm{te}}(\boldsymbol{x}_i^{\mathrm{tr}})}{p_{\mathrm{tr}}(\boldsymbol{x}_i^{\mathrm{tr}})} \right)^{\gamma} \log\left(1 + \exp\left(-y_i^{\mathrm{tr}} \widehat{f}(\boldsymbol{x}_i^{\mathrm{tr}}; \boldsymbol{\theta})\right)\right) + \lambda \boldsymbol{\theta}^\top \boldsymbol{\theta} \right],$$

which is usually solved by (quasi-)Newton methods.

An importance-weighted version of SVM, IWSVM, is given by

$$\widehat{\boldsymbol{\theta}}_{\mathrm{IWSVM}} := \operatorname*{argmin}_{\boldsymbol{\theta}} \left[\sum_{i=1}^{n_{\mathrm{tr}}} \left(\frac{p_{\mathrm{te}}(\boldsymbol{x}_i^{\mathrm{tr}})}{p_{\mathrm{tr}}(\boldsymbol{x}_i^{\mathrm{tr}})} \right)^{\gamma} \max\left(0, 1 - y_i^{\mathrm{tr}} \widehat{f}(\boldsymbol{x}_i^{\mathrm{tr}}; \boldsymbol{\theta})\right) + \lambda \boldsymbol{\theta}^\top \boldsymbol{\theta} \right],$$

whose solution can be obtained by a standard quadratic programming solver.

An importance-weighted version of Boosting, IWB, is given by

$$\widehat{\boldsymbol{\theta}}_{\mathrm{IWB}} := \operatorname*{argmin}_{\boldsymbol{\theta}} \left[\sum_{i=1}^{n_{\mathrm{tr}}} \left(\frac{p_{\mathrm{te}}(\boldsymbol{x}_i^{\mathrm{tr}})}{p_{\mathrm{tr}}(\boldsymbol{x}_i^{\mathrm{tr}})} \right)^{\gamma} \exp\left(-y_i^{\mathrm{tr}} \widehat{f}(\boldsymbol{x}_i^{\mathrm{tr}}; \boldsymbol{\theta})\right) + \lambda \boldsymbol{\theta}^\top \boldsymbol{\theta} \right],$$

which is usually solved by stage-wise optimization.

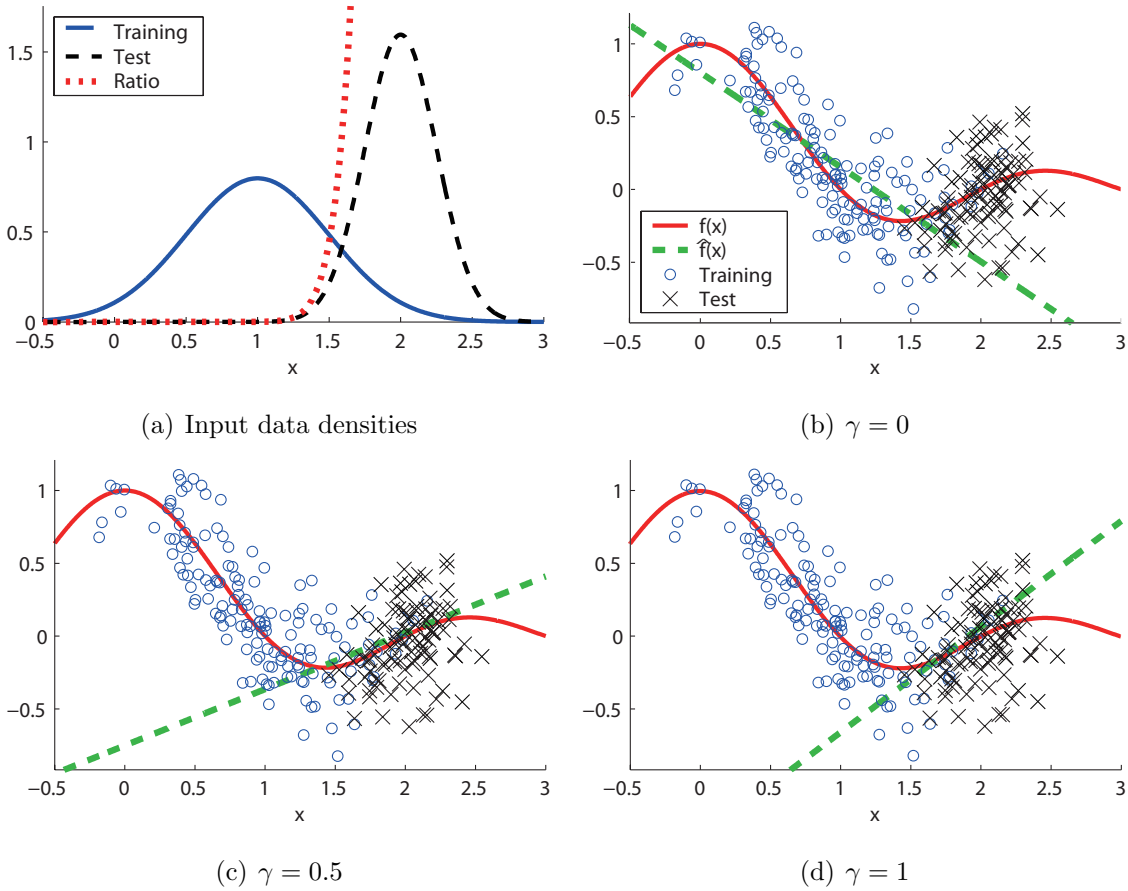
3.2 Numerical Examples

Here we illustrate the behavior of IWERM using toy regression and classification data sets.

3.2.1 Regression

Let us consider one-dimensional regression problem. Let the learning target function be $f(x) = \mathrm{sinc}(x)$, and let the training and test input densities be

$$p_{\mathrm{tr}}(x) = N(x; 1, (1/2)^2) \quad \text{and} \quad p_{\mathrm{te}}(x) = N(x; 2, (1/4)^2),$$



(a) Input data densities

(b) $\gamma = 0$

(c) $\gamma = 0.5$

(d) $\gamma = 1$

Figure 4: An illustrative regression example with covariate shift. (a) The probability density functions of the training and test input points and their ratio (i.e., the importance). (b)–(d) The learning target function $f(x)$ (the solid line), training samples ('$\circ$'), a learned function $\widehat{f}(x)$ (the dashed line), and test samples ('$\times$').

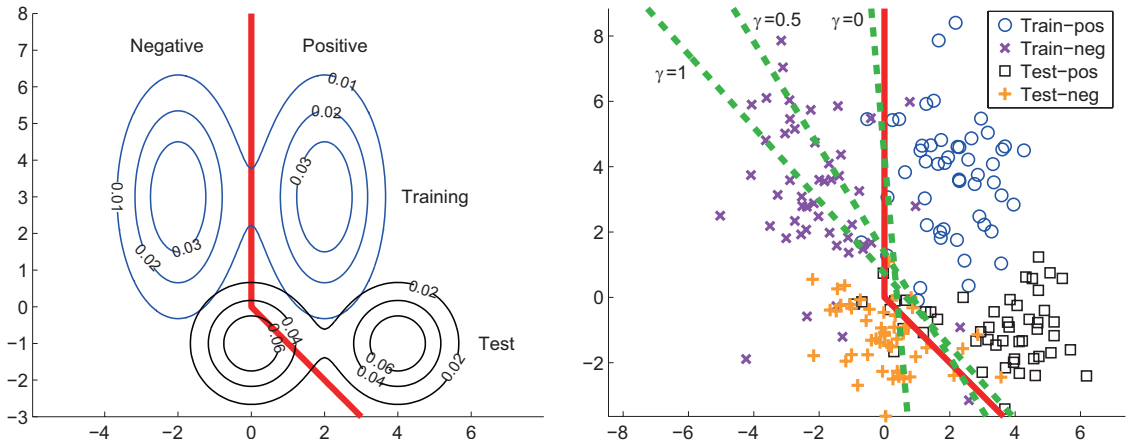
where $N(x; \mu, \sigma^2)$ denotes the Gaussian density with mean μ and variance σ^2. As illustrated in Figure 4(a), we are considering a (weak) extrapolation problem since the training input points are distributed in the left-hand side of the input domain and the test input points are distributed in the right-hand side.

We create the training output value $\{y_i^{\mathrm{tr}}\}_{i=1}^{n_{\mathrm{tr}}}$ as $y_i^{\mathrm{tr}} = f(x_i^{\mathrm{tr}}) + \epsilon_i^{\mathrm{tr}}$, where $\{\epsilon_i^{\mathrm{tr}}\}_{i=1}^{n_{\mathrm{tr}}}$ are i.i.d. noise drawn from $N(\epsilon; 0, (1/4)^2)$. Let the number of training samples be $n_{\mathrm{tr}} = 150$, and we use the following linear model:

$$\widehat{f}(x; \boldsymbol{\theta}) = \theta_1 x + \theta_2.$$

The parameter $\boldsymbol{\theta}$ is learned by IWLS.

Here we fix the regularization parameter to $\lambda = 0$, and compare the performance of IWLS for $\gamma = 0, 0.5, 1$. When $\gamma = 0$, a good approximation of the left-hand side of the sinc function can be obtained (see Figure 4(b)). However, this is not appropriate



(a) Optimal decision boundary (the thick solid line) and contours of training and test input densities (thin solid lines).

(b) Optimal decision boundary (solid line) and learned boundaries (dashed lines). '○' and '×' denote the positive and negative training samples, while '□' and '+' denote the positive and negative test samples.

Figure 5: An illustrative classification example with covariate shift.

for estimating the test output values ('×' in the figure). Thus, IWLS with $\gamma = 0$ (i.e., ordinary LS) results in a large test error. Figure 4(d) depicts the learned function when $\gamma = 1$, which tends to approximate the test output values well, but having a large variance. Figure 4(c) depicts a learned function when $\gamma = 0.5$, which yields even better estimation of the test output values for this particular data realization.

3.2.2 Classification

Let us consider a binary classification problem on the two-dimensional input space. Let the class posterior probabilities given input $\boldsymbol{x}$ be

$$p(y = +1 \,|\, \boldsymbol{x}) = \frac{1}{2}\left(1 + \tanh\left(x^{(1)} + \min(0, x^{(2)})\right)\right), \tag{4}$$

where $\boldsymbol{x} = (x^{(1)}, x^{(2)})^\top$ and $p(y = -1 \,|\, \boldsymbol{x}) = 1 - p(y = +1 \,|\, \boldsymbol{x})$. The optimal decision boundary, i.e., a set of all $\boldsymbol{x}$ such that $p(y = +1 \,|\, \boldsymbol{x}) = p(y = -1 \,|\, \boldsymbol{x}) = 1/2$ is illustrated in Figure 5(a).

Let the training and test input densities be

$$p_{\mathrm{tr}}(\boldsymbol{x}) = \frac{1}{2}N\left(\boldsymbol{x}; \begin{bmatrix}-2\\3\end{bmatrix}, \begin{bmatrix}1 & 0\\0 & 4\end{bmatrix}\right) + \frac{1}{2}N\left(\boldsymbol{x}; \begin{bmatrix}2\\3\end{bmatrix}, \begin{bmatrix}1 & 0\\0 & 4\end{bmatrix}\right),$$
$$p_{\mathrm{te}}(\boldsymbol{x}) = \frac{1}{2}N\left(\boldsymbol{x}; \begin{bmatrix}0\\-1\end{bmatrix}, \begin{bmatrix}1 & 0\\0 & 1\end{bmatrix}\right) + \frac{1}{2}N\left(\boldsymbol{x}; \begin{bmatrix}4\\-1\end{bmatrix}, \begin{bmatrix}1 & 0\\0 & 1\end{bmatrix}\right),$$

where $N(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ is the multivariate Gaussian density with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$. This setup implies that we are considering a (weak) extrapolation problem. Contours of the training and test input densities are illustrated in Figure 5(a).

Let the number of training samples be $n_{\mathrm{tr}} = 500$, and we create training input points $\{\boldsymbol{x}_i^{\mathrm{tr}}\}_{i=1}^{n_{\mathrm{tr}}}$ following $p_{\mathrm{tr}}(\boldsymbol{x})$ and training output labels $\{y_i^{\mathrm{tr}}\}_{i=1}^{n_{\mathrm{tr}}}$ following $p(y|\boldsymbol{x} = \boldsymbol{x}_i^{\mathrm{tr}})$. Similarly, let the number of test samples be $n_{\mathrm{te}} = 500$, and we create n_{te} test input points $\{\boldsymbol{x}_j^{\mathrm{te}}\}_{j=1}^{n_{\mathrm{te}}}$ following $p_{\mathrm{te}}(\boldsymbol{x})$ and test output labels $\{y_j^{\mathrm{te}}\}_{j=1}^{n_{\mathrm{te}}}$ following $p(y|\boldsymbol{x} = \boldsymbol{x}_j^{\mathrm{te}})$. We use the following linear model:

$$\widehat{f}(\boldsymbol{x}; \boldsymbol{\theta}) = \theta_1 x^{(1)} + \theta_2 x^{(2)} + \theta_3.$$

The parameter $\boldsymbol{\theta}$ is learned by IWFDA.

Here we fix the regularization parameter to $\lambda = 0$, and compare the performance of IWFDA for $\gamma = 0, 0.5, 1$. Figure 5(b) depicts an example of realizations of training and test samples, and decision boundaries obtained by IWFDA. For this particular realization of data samples, $\gamma = 0.5$ or 1 works better than $\gamma = 0$.

4 Model Selection under Covariate Shift

As illustrated in the previous section, importance-weighting is a promising approach to covariate shift adaptation, given that the flattening parameter γ is chosen appropriately. Although $\gamma = 0.5$ worked well both for the toy regression and classification experiments in the previous section, $\gamma = 0.5$ may not always be the best choice. Indeed, an appropriate value of γ depends on the learning target function, models, the noise level in the training samples, etc. Therefore, *model selection* needs to be appropriately carried out for enhancing the generalization capability under covariate shift.

The goal of model selection is to determine the model (e.g, basis functions, the flattening parameter γ, and the regularization parameter λ) so that the generalization error is minimized (Akaike, 1970; Mallows, 1973; Akaike, 1974; Takeuchi, 1976; Schwarz, 1978; Rissanen, 1978; Craven & Wahba, 1979; Akaike, 1980; Rissanen, 1987; Shibata, 1989; Wahba, 1990; Efron & Tibshirani, 1993; Murata et al., 1994; Konishi & Kitagawa, 1996; Ishiguro et al., 1997; Vapnik, 1998; Sugiyama & Ogawa, 2001; Sugiyama & Müller, 2002; Sugiyama et al., 2004). The true generalization error is not accessible since it contains the unknown learning target function. Thus, some generalization error estimator needs to be used instead. However, standard generalization error estimators such as *cross-validation* (CV) are heavily biased under covariate shift, and therefore they are no longer reliable. In this section, we review a modified CV method that possesses proper unbiasedness even under covariate shift.

4.1 Importance-Weighted Cross-Validation

One of the popular techniques for estimating the generalization error is CV (Stone, 1974; Wahba, 1990). CV has been shown to give an *almost* unbiased estimate of the general-

ization error with finite samples (Luntz & Brailovsky, 1969; Schölkopf & Smola, 2002). However, such almost unbiasedness is no longer fulfilled under covariate shift.

To cope with this problem, a variant of CV called *importance-weighted CV* (IWCV) has been proposed (Sugiyama et al., 2007). Let us randomly divide the training set $\mathcal{Z} = \{(\boldsymbol{x}_i^{\mathrm{tr}}, y_i^{\mathrm{tr}})\}_{i=1}^{n_{\mathrm{tr}}}$ into k disjoint non-empty subsets $\{\mathcal{Z}_i\}_{i=1}^{k}$ of (approximately) the same size. Let $\widehat{f}_{\mathcal{Z}_i}(\boldsymbol{x})$ be a function learned from $\{\mathcal{Z}_{i'}\}_{i' \neq i}$ (i.e., without $\mathcal{Z}_i$). Then the *k-fold IWCV* (kIWCV) estimate of the generalization error G is given by

$$\widehat{G}_{k\mathrm{IWCV}} = \frac{1}{k} \sum_{i=1}^{k} \frac{1}{|\mathcal{Z}_i|} \sum_{(\boldsymbol{x},y) \in \mathcal{Z}_i} \frac{p_{\mathrm{te}}(\boldsymbol{x})}{p_{\mathrm{tr}}(\boldsymbol{x})} \mathrm{loss}(\widehat{f}_{\mathcal{Z}_i}(\boldsymbol{x}), y),$$

where $|\mathcal{Z}_i|$ is the number of samples in the subset $\mathcal{Z}_i$.

When $k = n_{\mathrm{tr}}$, kIWCV is particularly called *IW leave-one-out CV* (IWLOOCV):

$$\widehat{G}_{\mathrm{IWLOOCV}} = \frac{1}{n_{\mathrm{tr}}} \sum_{i=1}^{n_{\mathrm{tr}}} \frac{p_{\mathrm{te}}(\boldsymbol{x}_i^{\mathrm{tr}})}{p_{\mathrm{tr}}(\boldsymbol{x}_i^{\mathrm{tr}})} \mathrm{loss}(\widehat{f}_i(\boldsymbol{x}_i^{\mathrm{tr}}), y_i^{\mathrm{tr}}),$$

where $\widehat{f}_i(\boldsymbol{x})$ is a function learned from $\{(\boldsymbol{x}_{i'}^{\mathrm{tr}}, y_{i'}^{\mathrm{tr}})\}_{i' \neq i}$ (i.e., without $(\boldsymbol{x}_i^{\mathrm{tr}}, y_i^{\mathrm{tr}})$). It was proved that IWLOOCV gives an *almost* unbiased estimate of the generalization error even under covariate shift (Sugiyama et al., 2007). More precisely, IWLOOCV for n_{tr} training samples gives an unbiased estimate of the generalization error for $n_{\mathrm{tr}} - 1$ training samples:

$$\mathop{\mathbb{E}}_{\{\boldsymbol{x}_i^{\mathrm{tr}}\}_{i=1}^{n_{\mathrm{tr}}}} \mathop{\mathbb{E}}_{\{y_i^{\mathrm{tr}}\}_{i=1}^{n_{\mathrm{tr}}}} \left[\widehat{G}_{\mathrm{IWLOOCV}}\right] = \mathop{\mathbb{E}}_{\{\boldsymbol{x}_i^{\mathrm{tr}}\}_{i=1}^{n_{\mathrm{tr}}}} \mathop{\mathbb{E}}_{\{y_i^{\mathrm{tr}}\}_{i=1}^{n_{\mathrm{tr}}}} [G'] \approx \mathop{\mathbb{E}}_{\{\boldsymbol{x}_i^{\mathrm{tr}}\}_{i=1}^{n_{\mathrm{tr}}}} \mathop{\mathbb{E}}_{\{y_i^{\mathrm{tr}}\}_{i=1}^{n_{\mathrm{tr}}}} [G],$$

where $\mathbb{E}_{\{\boldsymbol{x}_i^{\mathrm{tr}}\}_{i=1}^{n_{\mathrm{tr}}}}$ denotes the expectation over $\{\boldsymbol{x}_i^{\mathrm{tr}}\}_{i=1}^{n_{\mathrm{tr}}}$ drawn i.i.d. from $p_{\mathrm{tr}}(\boldsymbol{x})$, $\mathbb{E}_{\{y_i^{\mathrm{tr}}\}_{i=1}^{n_{\mathrm{tr}}}}$ denotes the expectation over $\{y_i^{\mathrm{tr}}\}_{i=1}^{n_{\mathrm{tr}}}$ each drawn from $p(y|\boldsymbol{x} = \boldsymbol{x}_i^{\mathrm{tr}})$, and G' denotes the generalization error for $n_{\mathrm{tr}} - 1$ training samples. A similar proof is also possible for kIWCV, but the bias is slightly larger (Hastie et al., 2001).

The almost unbiasedness of IWCV holds for any loss function, any model, and any parameter learning method; even non-identifiable models (Watanabe, 2009) or non-parametric learning methods (Schölkopf & Smola, 2002) are allowed. Thus IWCV is a highly flexible model selection technique under covariate shift. For other model selection criteria under covariate shift, see Shimodaira (2000) for regular models with smooth losses and Sugiyama and Müller (2005) for linear models with the squared loss.

4.2 Numerical Examples

Here we illustrate the behavior of IWCV using the same toy data sets as Section 3.2.

4.2.1 Regression

Let us continue the one-dimensional regression simulation in Section 3.2.1.

As illustrated in Figure 4 in Section 3.2.1, IWLS with flattening parameter $\gamma = 0.5$ appears to work well for that particular realization of data samples. However, the best value of γ would depend on the realization of samples. In order to investigate this systematically, let us repeat the simulation 1000 times with different random seeds, i.e., in each run $\{(x_i^{\mathrm{tr}}, \epsilon_i^{\mathrm{tr}})\}_{i=1}^{n_{\mathrm{tr}}}$ are randomly drawn and the scores of 10-fold IWCV and 10-fold ordinary CV are calculated for $\gamma = 0, 0.1, 0.2, \ldots, 1$. The means and standard deviations of the generalization error G and its estimate by each method are depicted as functions of γ in Figure 6. The graphs show that IWCV gives very accurate unbiased estimates of the generalization error, while ordinary CV is heavily biased.

Next we investigate the model selection performance. The flattening parameter γ is chosen from $\{0, 0.1, 0.2, \ldots, 1\}$ so that the score of each model selection criterion is minimized. The mean and standard deviation of the generalization error G of the learned function obtained by each method over 1000 runs are described in Table 1. This shows that IWCV gives significantly smaller generalization errors than ordinary CV, under the *t-test* (Henkel, 1976) at the significance level 5%. For reference, the generalization error when the flattening parameter γ is chosen optimally (i.e., in each trial, γ is chosen so that the true generalization error is minimized) is described as 'Optimal' in the table. The result shows that the performance of IWCV is rather close to that of the optimal choice.

4.2.2 Classification

Let us continue the toy classification simulation in Section 3.2.2.

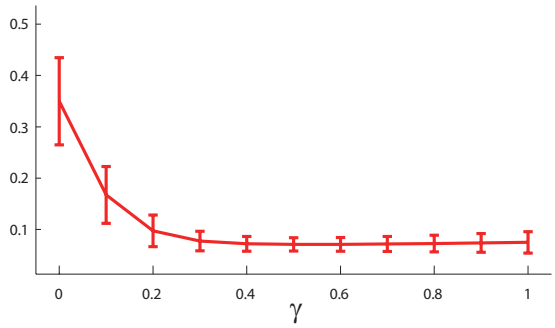
In Figure 5(b) in Section 3.2.2, IWFDA with a middle/large flattening parameter γ appears to work well for that particular realization of samples. Here, we investigate the choice of the flattening parameter value by IWCV and ordinary CV. Figure 7 depicts the means and standard deviations of the generalization error G (i.e., the misclassification rate) and its estimate by each method over 1000 runs, as functions of the flattening parameter γ in IWFDA. The graphs clearly show that IWCV gives much better estimates of the generalization error than ordinary CV.

Next we investigate the model selection performance. The flattening parameter γ is chosen from $\{0, 0.1, 0.2, \ldots, 1\}$ so that the score of each model selection criterion is minimized. The mean and standard deviation of the generalization error G of the learned function obtained by each method over 1000 runs are described in Table 2. The table shows that IWCV gives significantly smaller test errors than ordinary CV, and the performance of IWCV is rather close to that of the optimal choice.

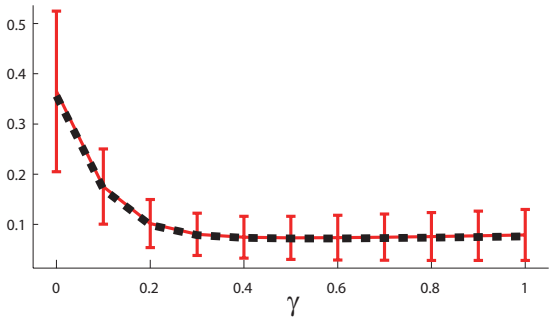
5 Importance Estimation

In the previous sections, we have seen that the importance weight

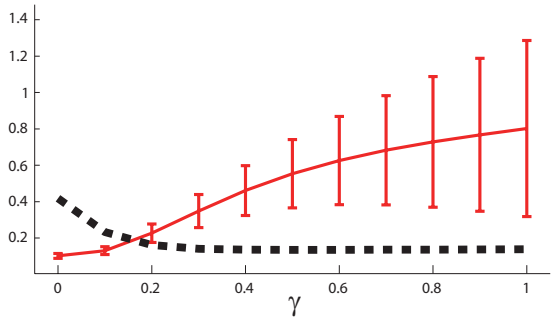
$$w(\boldsymbol{x}) = \frac{p_{\mathrm{te}}(\boldsymbol{x})}{p_{\mathrm{tr}}(\boldsymbol{x})}$$



(a) True generalization error



(b) IWCV score

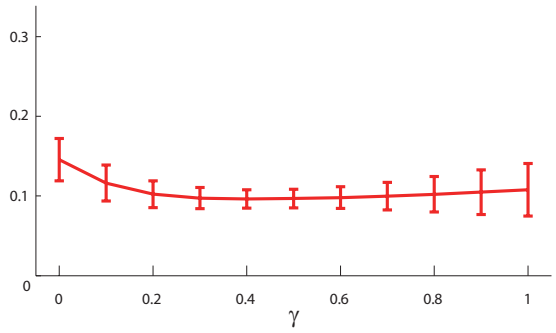


(c) Ordinary CV score

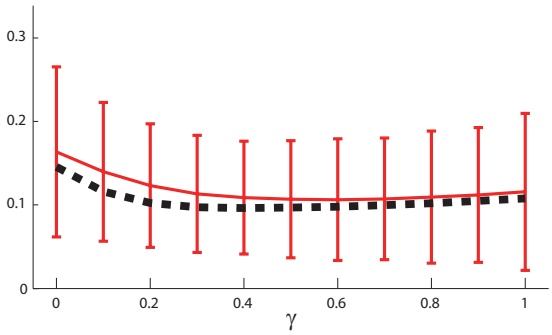
Figure 6: Generalization error and its estimates obtained by IWCV and ordinary CV as functions of the flattening parameter γ in IWLS for the regression examples in Figure 4. Thick dashed curves in the bottom graphs depict the true generalization error for clear comparison.

Table 1: The mean and standard deviation of the generalization error G obtained by each method for the toy regression data set. The best method and comparable ones by the t-test at the significance level 5% are indicated by '$\circ$'. For reference, the generalization error obtained with the optimal γ (i.e., the minimum generalization error) is described as 'Optimal'.

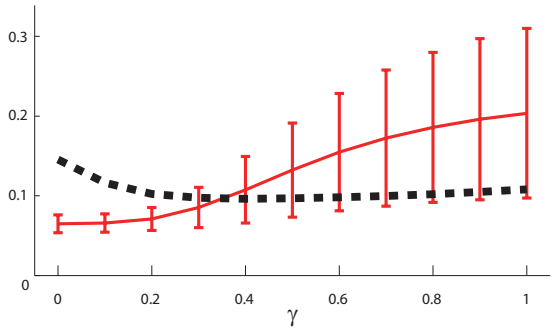
IWCV	Ordinary CV	Optimal
$^{\circ}0.077 \pm 0.020$	0.356 ± 0.086	0.069 ± 0.011



(a) True generalization error



(b) IWCV score



(c) Ordinary CV score

Figure 7: The generalization error G (i.e., the misclassification rate) and its estimates obtained by IWCV and ordinary CV as functions of the flattening parameter γ in IWFDA for the toy classification examples in Figure 5. Thick dashed curves in the bottom graphs depict the true generalization error for clear comparison.

Table 2: The mean and standard deviation of the generalization error G (i.e., the misclassification rate) obtained by each method for the toy classification data set. The best method and comparable ones by the t-test at the significance level 5% are indicated by '$\circ$'. For reference, the generalization error obtained with the optimal γ (i.e., the minimum generalization error) is described as 'Optimal'.

IWCV	Ordinary CV	Optimal
$^{\circ}0.108 \pm 0.027$	0.131 ± 0.029	0.091 ± 0.009

plays a central role in covariate shift adaptation. However, the importance weight is unknown in practice and needs to be estimated from data. In this section, we review importance estimation methods.

Here we assume that in addition to the training input samples $\{\boldsymbol{x}_i^{\mathrm{tr}}\}_{i=1}^{n_{\mathrm{tr}}}$ drawn independently from $p_{\mathrm{tr}}(\boldsymbol{x})$, we are given test input samples $\{\boldsymbol{x}_j^{\mathrm{te}}\}_{j=1}^{n_{\mathrm{te}}}$ drawn independently from $p_{\mathrm{te}}(\boldsymbol{x})$. Thus the goal of the importance estimation problem addressed here is to estimate the importance function $w(\boldsymbol{x})$ from $\{\boldsymbol{x}_i^{\mathrm{tr}}\}_{i=1}^{n_{\mathrm{tr}}}$ and $\{\boldsymbol{x}_j^{\mathrm{te}}\}_{j=1}^{n_{\mathrm{te}}}$.

5.1 Kernel Density Estimation

Kernel density estimation (KDE) is a non-parametric technique to estimate a probability density function $p(\boldsymbol{x})$ from its i.i.d. samples $\{\boldsymbol{x}_i\}_{i=1}^{n}$. For the Gaussian kernel

$$K_\sigma(\boldsymbol{x}, \boldsymbol{x}') = \exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{x}'\|^2}{2\sigma^2}\right), \tag{5}$$

KDE is expressed as

$$\widehat{p}(\boldsymbol{x}) = \frac{1}{n_{\mathrm{tr}}(2\pi\sigma^2)^{d/2}} \sum_{\ell=1}^{n} K_\sigma(\boldsymbol{x}, \boldsymbol{x}_\ell).$$

The performance of KDE depends on the choice of the kernel width σ. It can be optimized by cross-validation (CV) as follows (Härdle et al., 2004): First, divide the samples $\{\boldsymbol{x}_i\}_{i=1}^{n}$ into k disjoint non-empty subsets $\{\mathcal{X}_r\}_{r=1}^{k}$ of (approximately) the same size. Then obtain a density estimator $\widehat{p}_{\mathcal{X}_r}(\boldsymbol{x})$ from $\{\mathcal{X}_i\}_{i\neq r}$ (i.e., without $\mathcal{X}_r$), and compute its log-likelihood for the hold-out subset $\mathcal{X}_r$:

$$\frac{1}{|\mathcal{X}_r|} \sum_{\boldsymbol{x}\in\mathcal{X}_r} \log \widehat{p}_{\mathcal{X}_r}(\boldsymbol{x}),$$

where $|\mathcal{X}|$ denotes the number of elements in the set $\mathcal{X}$. Repeat this procedure for $r = 1, 2, \ldots, k$ and choose the value of σ such that the average of the above hold-out log-likelihood over all r is maximized. Note that the average hold-out log-likelihood is an almost unbiased estimate of the Kullback-Leibler divergence from $p(\boldsymbol{x})$ to $\widehat{p}(\boldsymbol{x})$, up to an irrelevant constant.

KDE can be used for importance estimation by first obtaining density estimators $\widehat{p}_{\mathrm{tr}}(\boldsymbol{x})$ and $\widehat{p}_{\mathrm{te}}(\boldsymbol{x})$ separately from $\{\boldsymbol{x}_i^{\mathrm{tr}}\}_{i=1}^{n_{\mathrm{tr}}}$ and $\{\boldsymbol{x}_j^{\mathrm{te}}\}_{j=1}^{n_{\mathrm{te}}}$, and then estimating the importance by $\widehat{w}(\boldsymbol{x}) = \widehat{p}_{\mathrm{te}}(\boldsymbol{x})/\widehat{p}_{\mathrm{tr}}(\boldsymbol{x})$. However, division by an estimated density can magnify the estimation error, so directly estimating the importance weight in a single-shot process would be more preferable.

5.2 Kullback-Leibler Importance Estimation Procedure

The *Kullback-Leibler importance estimation procedure* (KLIEP) (Sugiyama et al., 2008) directly gives an estimate of the importance function without going through density estimation by matching the two densities $p_{\mathrm{tr}}(\boldsymbol{x})$ and $p_{\mathrm{te}}(\boldsymbol{x})$ in terms of the *Kullback-Leibler divergence* (Kullback & Leibler, 1951).

Let us model the importance weight $w(\boldsymbol{x})$ by the following kernel model:

$$\widehat{w}(\boldsymbol{x}) = \sum_{\ell=1}^{n_{\mathrm{te}}} \alpha_\ell K_\sigma(\boldsymbol{x}, \boldsymbol{x}_\ell^{\mathrm{te}}),$$

where $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_{n_{\mathrm{te}}})^\top$ are parameters to be learned from data samples and $K_\sigma(\boldsymbol{x}, \boldsymbol{x}')$ is the Gaussian kernel (see Eq.(5)). An estimate of the density $p_{\mathrm{te}}(\boldsymbol{x})$ is given by using the model $\widehat{w}(\boldsymbol{x})$ as $\widehat{p}_{\mathrm{te}}(\boldsymbol{x}) = \widehat{w}(\boldsymbol{x}) p_{\mathrm{tr}}(\boldsymbol{x})$. In KLIEP, the parameters $\boldsymbol{\alpha}$ are determined so that the Kullback-Leibler divergence from $p_{\mathrm{te}}(\boldsymbol{x})$ to $\widehat{p}_{\mathrm{te}}(\boldsymbol{x})$ is minimized:

$$\mathrm{KL}(\boldsymbol{\alpha}) := \mathop{\mathbb{E}}_{\boldsymbol{x}^{\mathrm{te}}} \left[\log \frac{p_{\mathrm{te}}(\boldsymbol{x}^{\mathrm{te}})}{\widehat{w}(\boldsymbol{x}^{\mathrm{te}}) p_{\mathrm{tr}}(\boldsymbol{x}^{\mathrm{te}})} \right] = \mathop{\mathbb{E}}_{\boldsymbol{x}^{\mathrm{te}}} \left[\log \frac{p_{\mathrm{te}}(\boldsymbol{x}^{\mathrm{te}})}{p_{\mathrm{tr}}(\boldsymbol{x}^{\mathrm{te}})} \right] - \mathop{\mathbb{E}}_{\boldsymbol{x}^{\mathrm{te}}} \left[\log \widehat{w}(\boldsymbol{x}^{\mathrm{te}}) \right],$$

where $\mathbb{E}_{\boldsymbol{x}^{\mathrm{te}}}$ denotes the expectation over $\boldsymbol{x}^{\mathrm{te}}$ drawn from $p_{\mathrm{te}}(\boldsymbol{x})$. The first term is a constant, so it can be safely ignored. We define the negative of the second term by KL':

$$\mathrm{KL}'(\boldsymbol{\alpha}) := \mathop{\mathbb{E}}_{\boldsymbol{x}^{\mathrm{te}}} \left[\log \widehat{w}(\boldsymbol{x}^{\mathrm{te}}) \right]. \tag{6}$$

Since $\widehat{p}_{\mathrm{te}}(\boldsymbol{x})$ $(= \widehat{w}(\boldsymbol{x}) p_{\mathrm{tr}}(\boldsymbol{x}))$ is a probability density function, it should satisfy

$$1 = \int_{\mathcal{D}} \widehat{p}_{\mathrm{te}}(\boldsymbol{x}) d\boldsymbol{x} = \int_{\mathcal{D}} \widehat{w}(\boldsymbol{x}) p_{\mathrm{tr}}(\boldsymbol{x}) d\boldsymbol{x} = \mathop{\mathbb{E}}_{\boldsymbol{x}^{\mathrm{tr}}} \left[\widehat{w}(\boldsymbol{x}^{\mathrm{tr}}) \right]. \tag{7}$$

The KLIEP optimization problem is given by replacing the expectations in Eqs.(6) and (7) with empirical averages:

$$\max_{\{\alpha_\ell\}_{\ell=1}^{n_{\mathrm{te}}}} \left[\sum_{j=1}^{n_{\mathrm{te}}} \log \left(\sum_{\ell=1}^{n_{\mathrm{te}}} \alpha_\ell K(\boldsymbol{x}_j^{\mathrm{te}}, \boldsymbol{x}_\ell^{\mathrm{te}}) \right) \right]$$
$$\text{subject to } \frac{1}{n_{\mathrm{tr}}} \sum_{\ell=1}^{n_{\mathrm{te}}} \alpha_\ell \left(\sum_{i=1}^{n_{\mathrm{tr}}} K(\boldsymbol{x}_i^{\mathrm{tr}}, \boldsymbol{x}_\ell^{\mathrm{te}}) \right) = 1 \quad \text{and} \quad \alpha_1, \alpha_2, \ldots, \alpha_{n_{\mathrm{te}}} \geq 0.$$

This is a *convex* optimization problem and the global solution—which tends to be *sparse* (Boyd & Vandenberghe, 2004)—can be obtained, e.g., by simply performing gradient ascent and feasibility satisfaction iteratively. A pseudo code is summarized in Figure 8. The Gaussian width σ can be optimized by CV over KL', where only the test samples $\{\boldsymbol{x}_j^{\mathrm{te}}\}_{j=1}^{n_{\mathrm{te}}}$ are divided into k disjoint subsets (Sugiyama et al., 2008).

A MATLAB® implementation of the entire KLIEP algorithm is available from the following web page.

http://sugiyama-www.cs.titech.ac.jp/~sugi/software/KLIEP/

Input: $\{\boldsymbol{x}_i^{\mathrm{tr}}\}_{i=1}^{n_{\mathrm{tr}}}$, $\{\boldsymbol{x}_j^{\mathrm{te}}\}_{j=1}^{n_{\mathrm{te}}}$, and σ
Output: $\widehat{w}(\boldsymbol{x})$

$A_{j,\ell} \longleftarrow K_\sigma(\boldsymbol{x}_j^{\mathrm{te}}, \boldsymbol{x}_\ell^{\mathrm{te}})$ for $j, \ell = 1, 2, \ldots, n_{\mathrm{te}}$;
$b_\ell \longleftarrow \frac{1}{n_{\mathrm{tr}}} \sum_{i=1}^{n_{\mathrm{tr}}} K_\sigma(\boldsymbol{x}_i^{\mathrm{tr}}, \boldsymbol{x}_\ell^{\mathrm{te}})$ for $\ell = 1, 2, \ldots, n_{\mathrm{te}}$;
Initialize $\boldsymbol{\alpha}$ $(> \mathbf{0}_{n_{\mathrm{te}}})$ and ε $(0 < \varepsilon \ll 1)$;
Repeat until convergence
$\boldsymbol{\alpha} \longleftarrow \boldsymbol{\alpha} + \varepsilon \boldsymbol{A}^\top (\mathbf{1}_{n_{\mathrm{te}}} ./ \boldsymbol{A}\boldsymbol{\alpha})$; % Gradient ascent
$\boldsymbol{\alpha} \longleftarrow \boldsymbol{\alpha} + (1 - \boldsymbol{b}^\top \boldsymbol{\alpha}) \boldsymbol{b} / (\boldsymbol{b}^\top \boldsymbol{b})$; % Constraint satisfaction
$\boldsymbol{\alpha} \longleftarrow \max(\mathbf{0}_{n_{\mathrm{te}}}, \boldsymbol{\alpha})$; % Constraint satisfaction
$\boldsymbol{\alpha} \longleftarrow \boldsymbol{\alpha} / (\boldsymbol{b}^\top \boldsymbol{\alpha})$; % Constraint satisfaction
end
$\widehat{w}(\boldsymbol{x}) \longleftarrow \sum_{\ell=1}^{n_{\mathrm{te}}} \alpha_\ell K_\sigma(\boldsymbol{x}, \boldsymbol{x}_\ell^{\mathrm{te}})$;

Figure 8: Pseudo code of KLIEP. $\mathbf{0}_{n_{\mathrm{te}}}$ denotes the n_{te}-dimensional vector with all zeros, and $\mathbf{1}_{n_{\mathrm{te}}}$ denotes the n_{te}-dimensional vector with all ones. './' indicates the element-wise division, and inequalities and the 'max' operation for vectors are applied in the element-wise manner.

5.3 Numerical Examples

Here, we illustrate the behavior of the KLIEP method.

Let us consider the following one-dimensional importance estimation problem:

$$p_{\mathrm{tr}}(x) = N(x; 1, (1/2)^2) \ \text{ and } \ p_{\mathrm{te}}(x) = N(x; 2, (1/4)^2).$$

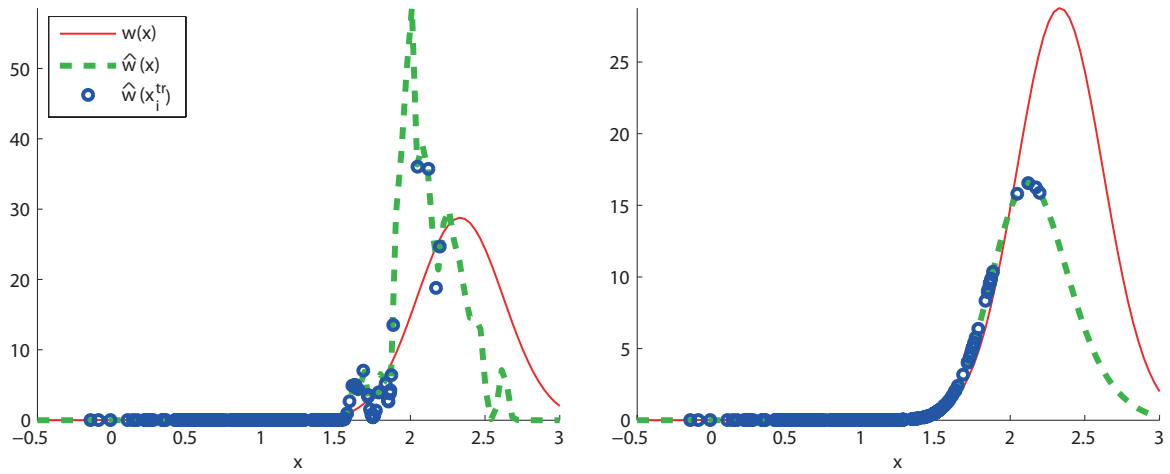
Let the number of training samples be $n_{\mathrm{tr}} = 200$ and the number of test samples be $n_{\mathrm{te}} = 1000$.

Figure 9 depicts the true importance and its estimates by KLIEP, where three different Gaussian widths $\sigma = 0.02, 0.2, 0.8$ are tested. The graphs show that the performance of KLIEP is highly dependent on the Gaussian width. More specifically, the estimated importance function $\widehat{w}(x)$ is highly fluctuated when σ is small, while it is overly smoothed when σ is large. When σ is chosen appropriately, KLIEP seems to work reasonably well for this example.

Figure 10 depicts the values of the true J (see Eq.(6)) and its estimate by 5-fold CV; the means, the 25 percentiles, and the 75 percentiles over 100 trials are plotted as functions of the Gaussian width σ. This shows that CV gives a very good estimate of J, which results in an appropriate choice of σ.

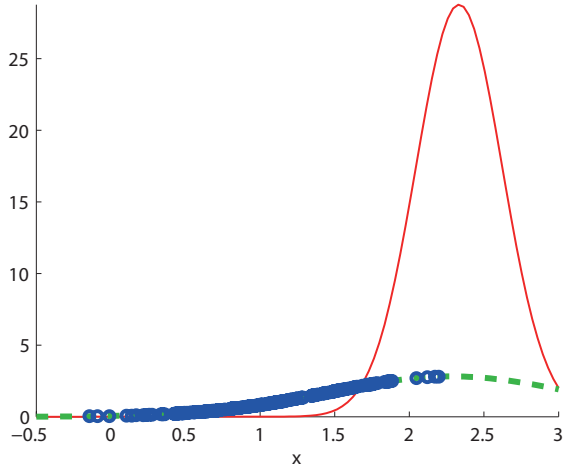
6 Conclusions and Outlook

In standard supervised learning theories, test input points are assumed to follow the same probability distribution as training input points. However, this assumption is often violated in real-world learning problems. In this chapter, we reviewed recently proposed



(a) Gaussian width $\sigma = 0.02$

(b) Gaussian width $\sigma = 0.2$



(c) Gaussian width $\sigma = 0.8$

Figure 9: Results of importance estimation by KLIEP. $w(x)$ is the true importance function and $\widehat{w}(x)$ is its estimation obtained by KLIEP.

techniques for covariate shift adaptation, including importance-weighted empirical risk minimization, importance-weighted cross-validation, and direct importance estimation.

In Section 5, we introduced the KLIEP algorithm for importance estimation, where linearly-parameterized models were used. It was shown that the KLIEP idea can also be naturally applied to log-linear models (Tsuboi et al., 2009), Gaussian mixture models (Yamada & Sugiyama, 2009), and probabilistic principal component analysis mixture models (Yamada et al., 2010b). Other than KLIEP, various methods of direct importance estimation have also been proposed (Silverman, 1978; Ćwik & Mielniczuk, 1989; Qin, 1998; Cheng & Chu, 2004; Huang et al., 2007; Bickel et al., 2007; Kanamori et al., 2009a). Among them, the method proposed in Kanamori et al. (2009a) called *uncon-*

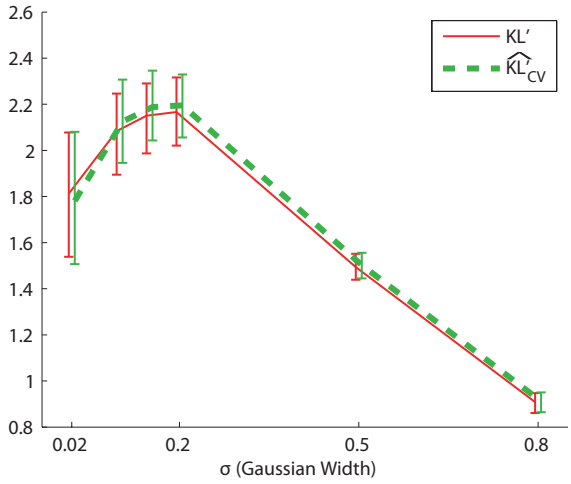


Figure 10: Model selection curve for KLIEP. KL′ is the true score of an estimated importance (see Eq.(6)) and $\widehat{\mathrm{KL}'}_{\mathrm{CV}}$ is its estimate by 5-fold CV.

strained least-squares importance fitting (uLSIF) gives an analytic-form solution and the solution can be computed very efficiently in a stable manner. Thus it can be applied to large-scale data sets.

Recently, importance estimation methods which incorporate dimensionality reduction have been developed. A method proposed by Sugiyama et al. (2010a) uses a supervised dimensionality reduction technique called *local Fisher discriminant analysis* (Sugiyama, 2007) for identifying a subspace in which two densities are significantly different (which is called the *hetero-distributional subspace*). Another method proposed by Sugiyama et al. (2011) tries to find the hetero-distributional subspace by directly minimizing the discrepancy between the two distributions. Theoretical analysis of importance estimation has also been conducted thoroughly (Silverman, 1978; Ćwik & Mielniczuk, 1989; Gijbels & Mielniczuk, 1995; Jacob & Oliveira, 1997; Qin, 1998; Cheng & Chu, 2004; Bensaid & Fabre, 2007; Nguyen et al., 2010; Sugiyama et al., 2008; Chen et al., 2009; Kanamori et al., 2009b; Kanamori et al., 2010).

It has been shown that various statistical data processing tasks can be solved through importance estimation (Sugiyama et al., 2009; Sugiyama et al., 2012), including multi-task learning (Bickel et al., 2007), inlier-based outlier detection (Silverman, 1978; Hido et al., 2008; Smola et al., 2009; Hido et al., 2011), change detection in time series (Kawahara & Sugiyama, 2011), mutual information estimation (Suzuki et al., 2008; Suzuki et al., 2009b), independent component analysis (Suzuki & Sugiyama, 2011), feature selection (Suzuki et al., 2009a), sufficient dimension reduction (Suzuki & Sugiyama, 2010), causal inference (Yamada & Sugiyama, 2010), conditional density estimation (Sugiyama et al., 2010b), and probabilistic classification (Sugiyama, 2010). Thus, following this line of research, further improving the accuracy and computational efficiency of importance estimation as well as further exploring possible application of importance estimation would be a promising direction to be pursued.

Acknowledgments

The author was supported by MEXT Grant-in-Aid for Young Scientists (A) 20680007, SCAT, and AOARD.

References

Akaike, H. (1970). Statistical predictor identification. *Annals of the Institute of Statistical Mathematics*, *22*, 203–217.

Akaike, H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, *AC-19*, 716–723.

Akaike, H. (1980). Likelihood and the Bayes procedure. *Bayesian Statistics* (pp. 141–166). Valencia, Spain: Valencia University Press.

Akiyama, T., Hachiya, H., & Sugiyama, M. (2010). Efficient exploration through active learning for value function approximation in reinforcement learning. *Neural Networks*, *23*, 639–648.

Bensaid, N., & Fabre, J. P. (2007). Optimal asymptotic quadratic error of kernel estimators of Radon-Nikodym derivatives for strong mixing data. *Journal of Nonparametric Statistics*, *19*, 77–88.

Bickel, S., Brückner, M., & Scheffer, T. (2007). Discriminative learning for differing training and test distributions. *Proceedings of the 24th International Conference on Machine Learning (ICML2007)* (pp. 81–88).

Bishop, C. M. (1995). *Neural networks for pattern recognition*. Oxford, UK: Clarendon Press.

Boyd, S., & Vandenberghe, L. (2004). *Convex optimization*. Cambridge, UK: Cambridge University Press.

Breiman, L. (1998). Arcing classifiers. *The Annals of Statistics*, *26*, 801–849.

Chen, S.-M., Hsu, Y.-S., & Liaw, J.-T. (2009). On kernel estimators of density ratio. *Statistics*, *43*, 463–479.

Chen, S. S., Donoho, D. L., & Saunders, M. A. (1998). Atomic decomposition by basis pursuit. *SIAM Journal on Scientific Computing*, *20*, 33–61.

Cheng, K. F., & Chu, C. K. (2004). Semiparametric density estimation under a two-sample density ratio model. *Bernoulli*, *10*, 583–604.

Craven, P., & Wahba, G. (1979). Smoothing noisy data with spline functions: Estimating the correct degree of smoothing by the method of generalized cross-validation. *Numerische Mathematik*, *31*, 377–403.

Ćwik, J., & Mielniczuk, J. (1989). Estimating density ratio with application to discriminant analysis. *Communications in Statistics: Theory and Methods*, *18*, 3057–3069.

Duda, R. O., Hart, P. E., & Stork, D. G. (2001). *Pattern classification*. New York, NY, USA: Wiley. Second edition.

Efron, B., & Tibshirani, R. J. (1993). *An introduction to the bootstrap*. New York, NY, USA: Chapman & Hall/CRC.

Fisher, R. A. (1936). The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, *7*, 179–188.

Fishman, G. S. (1996). *Monte Carlo: Concepts, algorithms, and applications*. Berlin, Germany: Springer-Verlag.

Freund, Y., & Schapire, R. E. (1996). Experiments with a new boosting algorithm. *Proc. 13th International Conference on Machine Learning* (pp. 148–156). Morgan Kaufmann.

Friedman, J., Hastie, T., & Tibshirani, R. (2000). Additive logistic regression: A statistical view of boosting. *The Annals of Statistics*, *28*, 337–407.

Gijbels, I., & Mielniczuk, J. (1995). Asymptotic properties of kernel estimators of the Radon-Nikodym derivative with applications to discriminant analysis. *Statistica Sinica*, *5*, 261–278.

Hachiya, H., Akiyama, T., Sugiyama, M., & Peters, J. (2009). Adaptive importance sampling for value function approximation in off-policy reinforcement learning. *Neural Networks*, *22*, 1399–1410.

Hachiya, H., Peters, J., & Sugiyama, M. (2011). Reward weighted regression with sample reuse. *Neural Computation*, *11*, 2798–2832.

Härdle, W., Müller, M., Sperlich, S., & Werwatz, A. (2004). *Nonparametric and semiparametric models*. Berlin, Germany: Springer.

Hastie, T., Tibshirani, R., & Friedman, J. (2001). *The elements of statistical learning: Data mining, inference, and prediction*. New York, NY, USA: Springer.

Henkel, R. E. (1976). *Tests of significance*. Beverly Hills, CA, USA.: SAGE Publication.

Hido, S., Tsuboi, Y., Kashima, H., Sugiyama, M., & Kanamori, T. (2008). Inlier-based outlier detection via direct density ratio estimation. *Proceedings of IEEE International Conference on Data Mining (ICDM2008)* (pp. 223–232). Pisa, Italy.

Hido, S., Tsuboi, Y., Kashima, H., Sugiyama, M., & Kanamori, T. (2011). Statistical outlier detection using direct density ratio estimation. *Knowledge and Information Systems*, *26*, 309–336.

Huang, J., Smola, A., Gretton, A., Borgwardt, K. M., & Schölkopf, B. (2007). Correcting sample selection bias by unlabeled data. In B. Schölkopf, J. Platt and T. Hoffman (Eds.), *Advances in neural information processing systems 19*, 601–608. Cambridge, MA, USA: MIT Press.

Ishiguro, M., Sakamoto, Y., & Kitagawa, G. (1997). Bootstrapping log likelihood and EIC, an extension of AIC. *Annals of the Institute of Statistical Mathematics*, *49*, 411–434.

Jacob, P., & Oliveira, P. E. (1997). Kernel estimators of general Radon-Nikodym derivatives. *Statistics*, *30*, 25–46.

Kanamori, T. (2007). Pool-based active learning with optimal sampling distribution and its information geometrical interpretation. *Neurocomputing*, *71*, 353–362.

Kanamori, T., Hido, S., & Sugiyama, M. (2009a). A least-squares approach to direct importance estimation. *Journal of Machine Learning Research*, *10*, 1391–1445.

Kanamori, T., & Shimodaira, H. (2003). Active learning algorithm using the maximum weighted log-likelihood estimator. *Journal of Statistical Planning and Inference*, *116*, 149–162.

Kanamori, T., Suzuki, T., & Sugiyama, M. (2009b). *Condition number analysis of kernel-based density ratio estimation* (Technical Report). arXiv.

Kanamori, T., Suzuki, T., & Sugiyama, M. (2010). Theoretical analysis of density ratio estimation. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, *E93-A*, 787–798.

Kawahara, Y., & Sugiyama, M. (2011). Sequential change-point detection based on direct density-ratio estimation. *Statistical Analysis and Data Mining*. to appear.

Konishi, S., & Kitagawa, G. (1996). Generalized information criteria in model selection. *Biometrika*, *83*, 875–890.

Kullback, S., & Leibler, R. A. (1951). On information and sufficiency. *Annals of Mathematical Statistics*, *22*, 79–86.

Luntz, A., & Brailovsky, V. (1969). On estimation of characters obtained in statistical procedure of recognition. *Technicheskaya Kibernetica*, *3*. in Russian.

Mallows, C. L. (1973). Some comments on C_P. *Technometrics*, *15*, 661–675.

Mangasarian, O. L., & Musicant, D. R. (2000). Robust linear and support vector regression. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *22*, 950–955.

Murata, N., Yoshizawa, S., & Amari, S. (1994). Network information criterion — Determining the number of hidden units for an artificial neural network model. *IEEE Transactions on Neural Networks*, *5*, 865–872.

Nguyen, X., Wainwright, M. J., & Jordan, M. I. (2010). Estimating divergence functionals and the likelihood ratio by convex risk minimization. *IEEE Transactions on Information Theory*, *56*, 5847–5861.

Qin, J. (1998). Inferences for case-control and semiparametric two-sample density ratio models. *Biometrika*, *85*, 619–630.

Quiñonero-Candela, J., Sugiyama, M., Schwaighofer, A., & Lawrence, N. (Eds.). (2009). *Dataset shift in machine learning*. Cambridge, MA, USA: MIT Press.

Rissanen, J. (1978). Modeling by shortest data description. *Automatica*, *14*, 465–471.

Rissanen, J. (1987). Stochastic complexity. *Journal of the Royal Statistical Society, Series B*, *49*, 223–239.

Schölkopf, B., & Smola, A. J. (2002). *Learning with kernels*. Cambridge, MA, USA: MIT Press.

Schwarz, G. (1978). Estimating the dimension of a model. *The Annals of Statistics*, *6*, 461–464.

Shibata, R. (1989). Statistical aspects of model selection. In J. C. Willems (Ed.), *From data to model*, 215–240. New York, NY, USA: Springer-Verlag.

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

Silverman, B. W. (1978). Density ratios, empirical likelihood and cot death. *Journal of the Royal Statistical Society, Series C*, *27*, 26–33.

Smola, A., Song, L., & Teo, C. H. (2009). Relative novelty detection. *Proceedings of Twelfth International Conference on Artificial Intelligence and Statistics (AISTATS2009)* (pp. 536–543). Clearwater Beach, FL, USA.

Stone, M. (1974). Cross-validatory choice and assessment of statistical predictions. *Journal of the Royal Statistical Society, Series B*, *36*, 111–147.

Sugiyama, M. (2006). Active learning in approximately linear regression based on conditional expectation of generalization error. *Journal of Machine Learning Research*, *7*, 141–166.

Sugiyama, M. (2007). Dimensionality reduction of multimodal labeled data by local Fisher discriminant analysis. *Journal of Machine Learning Research*, *8*, 1027–1061.

Sugiyama, M. (2010). Superfast-trainable multi-class probabilistic classifier by least-squares posterior fitting. *IEICE Transactions on Information and Systems*, *E93-D*, 2690–2701.

Sugiyama, M., Kanamori, T., Suzuki, T., Hido, S., Sese, J., Takeuchi, I., & Wang, L. (2009). A density-ratio framework for statistical data processing. *IPSJ Transactions on Computer Vision and Applications*, *1*, 183–208.

Sugiyama, M., & Kawanabe, M. (2011). *Machine learning in non-stationary environments: Introduction to covariate shift adaptation*. Cambridge, MA, USA: MIT Press. to appear.

Sugiyama, M., Kawanabe, M., & Chui, P. L. (2010a). Dimensionality reduction for density ratio estimation in high-dimensional spaces. *Neural Networks*, *23*, 44–59.

Sugiyama, M., Kawanabe, M., & Müller, K.-R. (2004). Trading variance reduction with unbiasedness: The regularized subspace information criterion for robust model selection in kernel regression. *Neural Computation*, *16*, 1077–1104.

Sugiyama, M., Krauledat, M., & Müller, K.-R. (2007). Covariate shift adaptation by importance weighted cross validation. *Journal of Machine Learning Research*, *8*, 985–1005.

Sugiyama, M., & Müller, K.-R. (2002). The subspace information criterion for infinite dimensional hypothesis spaces. *Journal of Machine Learning Research*, *3*, 323–359.

Sugiyama, M., & Müller, K.-R. (2005). Input-dependent estimation of generalization error under covariate shift. *Statistics & Decisions*, *23*, 249–279.

Sugiyama, M., & Nakajima, S. (2009). Pool-based active learning in approximate linear regression. *Machine Learning*, *75*, 249–274.

Sugiyama, M., & Ogawa, H. (2001). Subspace information criterion for model selection. *Neural Computation*, *13*, 1863–1889.

Sugiyama, M., Suzuki, T., & Kanamori, T. (2012). *Density ratio estimation in machine learning*. Cambridge, UK: Cambridge University Press. to appear.

Sugiyama, M., Suzuki, T., Nakajima, S., Kashima, H., von Bünau, P., & Kawanabe, M. (2008). Direct importance estimation for covariate shift adaptation. *Annals of the Institute of Statistical Mathematics*, *60*, 699–746.

Sugiyama, M., Takeuchi, I., Suzuki, T., Kanamori, T., Hachiya, H., & Okanohara, D. (2010b). Least-squares conditional density estimation. *IEICE Transactions on Information and Systems*, *E93-D*, 583–594.

Sugiyama, M., Yamada, M., von Bünau, P., Suzuki, T., Kanamori, T., & Kawanabe, M. (2011). Direct density-ratio estimation with dimensionality reduction via least-squares hetero-distributional subspace search. *Neural Networks*, *24*, 183–198.

Suzuki, T., & Sugiyama, M. (2010). Sufficient dimension reduction via squared-loss mutual information estimation. *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics (AISTATS2010)* (pp. 804–811). Sardinia, Italy.

Suzuki, T., & Sugiyama, M. (2011). Least-squares independent component analysis. *Neural Computation*, *23*, 284–301.

Suzuki, T., Sugiyama, M., Kanamori, T., & Sese, J. (2009a). Mutual information estimation reveals global associations between stimuli and biological processes. *BMC Bioinformatics*, *10*, S52.

Suzuki, T., Sugiyama, M., Sese, J., & Kanamori, T. (2008). Approximating mutual information by maximum likelihood density ratio estimation. *Proceedings of ECML-PKDD2008 Workshop on New Challenges for Feature Selection in Data Mining and Knowledge Discovery 2008 (FSDM2008)* (pp. 5–20). Antwerp, Belgium.

Suzuki, T., Sugiyama, M., & Tanaka, T. (2009b). Mutual information approximation via maximum likelihood estimation of density ratio. *Proceedings of 2009 IEEE International Symposium on Information Theory (ISIT2009)* (pp. 463–467). Seoul, Korea.

Takeuchi, K. (1976). Distribution of information statistics and validity criteria of models. *Mathematical Science*, *153*, 12–18. in Japanese.

Tibshirani, R. (1996). Regression shrinkage and subset selection with the lasso. *Journal of the Royal Statistical Society, Series B*, *58*, 267–288.

Tsuboi, Y., Kashima, H., Hido, S., Bickel, S., & Sugiyama, M. (2009). Direct density ratio estimation for large-scale covariate shift adaptation. *Journal of Information Processing*, *17*, 138–155.

Ueki, K., Sugiyama, M., & Ihara, Y. (2011). Lighting condition adaptation for perceived age estimation. *IEICE Transactions on Information and Systems*, *E94-D*, 392–395.

Vapnik, V. N. (1998). *Statistical learning theory.* New York, NY, USA: Wiley.

Wahba, G. (1990). *Spline models for observational data.* Philadelphia, PA, USA: Society for Industrial and Applied Mathematics.

Watanabe, S. (2009). *Algebraic geometry and statistical learning theory.* Cambridge, UK: Cambridge University Press.

Wiens, D. P. (2000). Robust weights and designs for biased regression models: Least squares and generalized M-estimation. *Journal of Statistical Planning and Inference*, *83*, 395–412.

Williams, P. M. (1995). Bayesian regularization and pruning using a Laplace prior. *Neural Computation*, *7*, 117–143.

Yamada, M., & Sugiyama, M. (2009). Direct importance estimation with Gaussian mixture models. *IEICE Transactions on Information and Systems*, *E92-D*, 2159–2162.

Yamada, M., & Sugiyama, M. (2010). Dependence minimizing regression with model selection for non-linear causal inference under non-Gaussian noise. *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence (AAAI2010)* (pp. 643–648). Atlanta, Georgia, USA: The AAAI Press.

Yamada, M., Sugiyama, M., & Matsui, T. (2010a). Semi-supervised speaker identification under covariate shift. *Signal Processing*, *90*, 2353–2361.

Yamada, M., Sugiyama, M., Wichern, G., & Simm, J. (2010b). Direct importance estimation with a mixture of probabilistic principal component analyzers. *IEICE Transactions on Information and Systems*, *E93-D*, 2846–2849.

Zadrozny, B. (2004). Learning and evaluating classifiers under sample selection bias. *Proceedings of the Twenty-First International Conference on Machine Learning (ICML2004)* (pp. 903–910). New York, NY, USA: ACM Press.