A Conditional Expectation Approach to Model Selection and Active Learning under Covariate Shift

Masashi Sugiyama (sugi@cs.titech.ac.jp)
Department of Computer Science, Tokyo Institute of Technology
2-12-1 O-okayama, Meguro-ku, Tokyo 152-8552, Japan

Neil Rubens (neil@sg.cs.titech.ac.jp)
Department of Computer Science, Tokyo Institute of Technology
2-12-1 O-okayama, Meguro-ku, Tokyo 152-8552, Japan

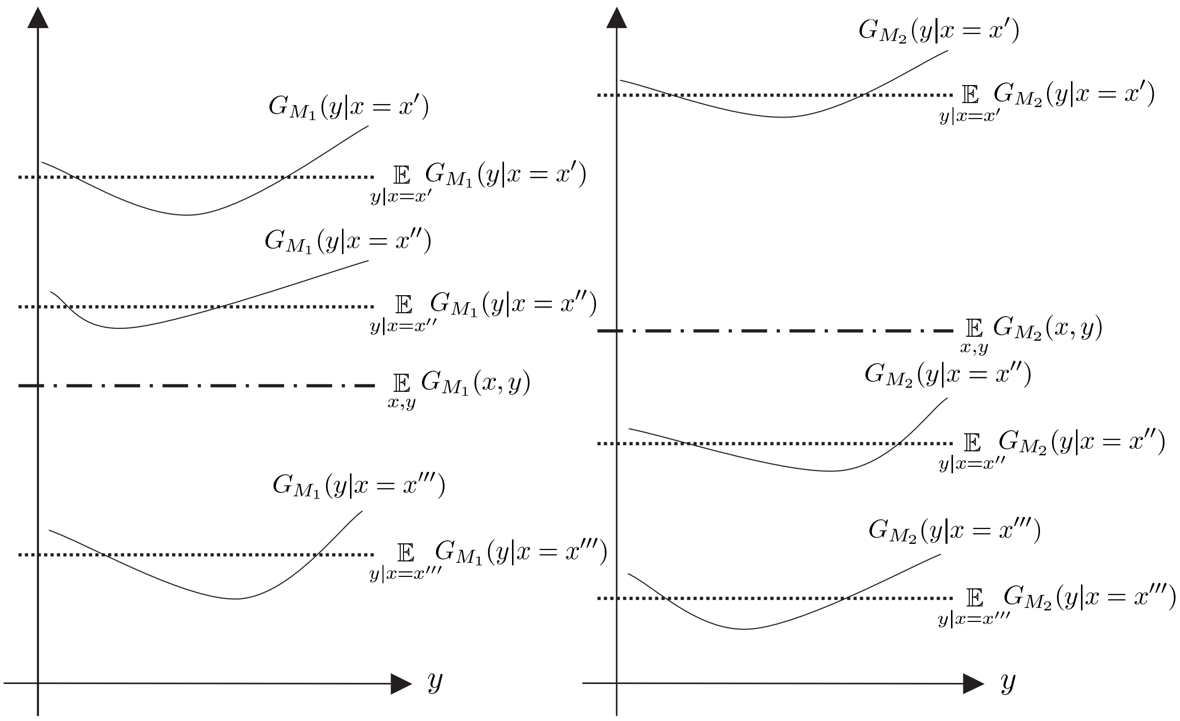
Klaus-Robert Müller (klaus@first.fraunhofer.de)
Department of Computer Science, Technical University of Berlin
Franklinstrasse 28/29, 10587 Berlin, Germany

Abstract

In the previous chapter, Kanamori and Shimodaira provided generalization error estimators which can be used for model selection and active learning. The accuracy of these estimators is theoretically guaranteed in terms of the expectation over realizations of training *input-output* samples. In practice, we are only given a single realization of training samples. Therefore, ideally, we want to have an estimator of the generalization error that is accurate in each *single trial*. However, we may not be able to avoid taking the expectation over the training output noise since it is not generally possible to know the realized value of noise. On the other hand, the location of the training input points is accessible by nature. Motivated by this fact, we propose to estimate the generalization error *without* taking the expectation over training input points. That is, we evaluate the unbiasedness of the generalization error in terms of the *conditional* expectation of training output noise given training input points.

1 Conditional Expectation Analysis of Generalization Error

In order to illustrate a possible advantage of the conditional expectation approach, let us consider a simple model selection scenario where we have only one training sample (x, y) (see Figure 1). The solid curves in Figure 1(a) depict $G_{M_1}(y|x)$, the generalization


$G_{M_1}(y|x = x')$
$\mathop{\mathbb{E}}_{y|x=x'} G_{M_1}(y|x = x')$
$G_{M_1}(y|x = x'')$
$\mathop{\mathbb{E}}_{y|x=x''} G_{M_1}(y|x = x'')$
$\mathop{\mathbb{E}}_{x,y} G_{M_1}(x, y)$
$G_{M_1}(y|x = x''')$
$\mathop{\mathbb{E}}_{y|x=x'''} G_{M_1}(y|x = x''')$
y
$G_{M_2}(y|x = x')$
$\mathop{\mathbb{E}}_{y|x=x'} G_{M_2}(y|x = x')$
$\mathop{\mathbb{E}}_{x,y} G_{M_2}(x, y)$
$G_{M_2}(y|x = x'')$
$\mathop{\mathbb{E}}_{y|x=x''} G_{M_2}(y|x = x'')$
$G_{M_2}(y|x = x''')$
$\mathop{\mathbb{E}}_{y|x=x'''} G_{M_2}(y|x = x''')$
y

(a) Generalization error for model M_1 (b) Generalization error for model M_2

Figure 1: Schematic illustrations of the conditional expectation and full expectation of the generalization error.

error for a model M_1 as a function of the (noisy) training output value y given a training input point x. The three solid curves correspond to the cases where the realization of the training input point x is x', x'', and x''', respectively. The value of the generalization error for the model M_1 in the full expectation approach is depicted by the dash-dotted line, where the expectation is taken over both the training input point x and the training output value y (this corresponds to the mean of the three solid curves). The values of the generalization error in the conditional expectation approach are depicted by the dotted lines, where the expectation is taken only over the training output value y, conditioned on $x = x', x'', x'''$, respectively (this corresponds to the mean value of each solid curve). The graph in Figure 1(b) depicts the generalization errors for a model M_2 in the same manner.

In the full expectation framework, the model M_1 is judged to be better than M_2 regardless of the realization of the training input point since the dash-dotted line in Figure 1(a) is lower than that in Figure 1(b). However, M_2 is actually better than M_1 if x'' or x''' is realized as x. In the conditional expectation framework, the goodness of the model is adaptively evaluated depending on the realization of the training input point x. This illustrates that the conditional expectation framework *can* indeed provide a better model choice than the full expectation framework.

In this chapter, we address the problems of model selection and active learning in the

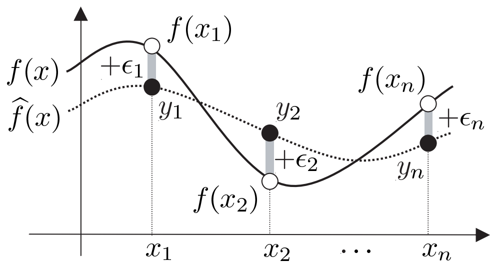


Figure 2: Regression problem of learning $f(x)$ from $\{(x_i, y_i)\}_{i=1}^n$. $\{\epsilon_i\}_{i=1}^n$ are i.i.d. noise with mean zero and variance σ^2, and $\widehat{f}(x)$ is a learned function.

conditional expectation framework. The rest of this chapter is organized as follows. After the problem formulation in Section 2, we introduce a model selection criterion (Section 3) and an active learning criterion (Section 4) in the conditional expectation framework and show that they are more advantageous than the full expectation methods in the context of approximate linear regression. Then we discuss how model selection and active learning can be combined in Section 5. Finally we give concluding remarks and future prospects in Section 6.

2 Linear Regression under Covariate Shift

In this section, we formulate a linear regression problem with covariate shift.

2.1 Statistical Formulation of Linear Regression

Let us consider a regression problem of estimating an unknown input-output dependency from training samples. Let $\{(x_i, y_i)\}_{i=1}^n$ be the training samples, where $x_i \in \mathcal{X} \subset \mathbb{R}^d$ is an i.i.d. training input point following a probability distribution $P_{\mathrm{tr}}(x)$ and $y_i \in \mathcal{Y} \subset \mathbb{R}$ is a corresponding training output value following a conditional probability distribution $P(y|x = x_i)$. We denote the conditional mean of $P(y|x)$ by $f(x)$ and assume that the conditional variance is σ^2, which is independent of x. Then $P(y|x)$ may be regarded as consisting of the true output $f(x)$ and the noise ϵ with mean 0 and variance σ^2 (see Figure 2).

Let us employ a linear regression model for learning $f(x)$.

$$\widehat{f}(x; \boldsymbol{\alpha}) = \sum_{\ell=1}^{t} \alpha_\ell \varphi_\ell(x), \tag{1}$$

where $\{\alpha_\ell\}_{\ell=1}^t$ are parameters to be learned and $\{\varphi_\ell(x)\}_{\ell=1}^t$ are fixed basis functions. A model $\widehat{f}(x; \boldsymbol{\alpha})$ is said to be *correctly specified* if there exists a parameter $\boldsymbol{\alpha}^*$ such that

$$\widehat{f}(x; \boldsymbol{\alpha}^*) = f(x). \tag{2}$$

Otherwise the model is said to be *misspecified.* In the following, we do not assume that the model is correct.

Let us consider a test sample, which is not given to the user in the training phase, but will be given in a future test phase. We denote the test sample by $(x^{\rm te}, y^{\rm te})$, where $x^{\rm te} \in \mathcal{X}$ is a test input point and $y^{\rm te} \in \mathcal{Y}$ is a corresponding test output value. The goal of regression is to determine the value of the parameter $\boldsymbol{\alpha}$ so that the generalization error G (the test error expected over test samples) is minimized:

$$G \equiv \mathbf{E}_{x^{\rm te}, y^{\rm te}} \left[(\widehat{f}(x^{\rm te}; \boldsymbol{\alpha}) - y^{\rm te})^2 \right], \tag{3}$$

where $\mathbf{E}_{x^{\rm te}, y^{\rm te}} [\cdot]$ denotes the expectation over $(x^{\rm te}, y^{\rm te})$.

2.2 Covariate Shift

In standard supervised learning theories, the test sample $(x^{\rm te}, y^{\rm te})$ is assumed to follow the joint distribution $P(y|x)P_{\rm tr}(x)$, which is the same as the training samples (e.g., Wahba, 1990; Bishop, 1995; Vapnik, 1998; Duda et al., 2001; Hastie et al., 2001; Schölkopf and Smola, 2002). On the other hand, here, we consider the *covariate shift* situation, i.e., the conditional distribution $P(y|x)$ remains unchanged, but the test input point $x^{\rm te}$ follows a different probability distribution $P_{\rm te}(x)$.

Let $p_{\rm tr}(x)$ and $p_{\rm te}(x)$ be the probability density functions corresponding to the input distributions $P_{\rm tr}(x)$ and $P_{\rm te}(x)$, respectively. We assume that $p_{\rm tr}(x)$ and $p_{\rm te}(x)$ are strictly positive over the entire domain $\mathcal{X}$.

2.3 Functional Analytic View of Linear Regression

Technically, we assume that the target function $f(x)$ and the basis functions $\{\varphi_\ell(x)\}_{\ell=1}^{t}$ are included in a functional Hilbert space $\mathcal{F}$, where the inner product and the norm in $\mathcal{F}$ are defined by

$$\langle f, g \rangle_{\mathcal{F}} = \int_{\mathcal{X}} (f(x) - g(x))^2 \, p_{\rm te}(x) dx, \tag{4}$$

$$\|f\|_{\mathcal{F}} = \sqrt{\langle f, f \rangle_{\mathcal{F}}}. \tag{5}$$

Then the generalization error G (3) is expressed in terms of $\mathcal{F}$ as

$$G = \left\| \widehat{f} - f \right\|_{\mathcal{F}}^2 + \sigma^2 \tag{6}$$

Given our linear regression model (1), the learning target function $f(x)$ can be decomposed as

$$f(x) = g(x) + \delta r(x), \tag{7}$$

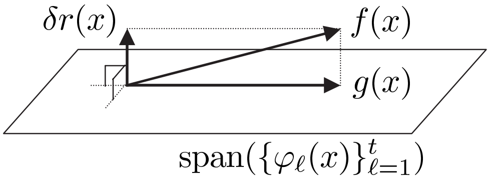


Figure 3: Decomposition of $f(x)$ in a functional Hilbert space $\mathcal{F}$.

where $g(x)$ is the optimal approximation in the model (1):

$$g(x;\boldsymbol{\alpha}^*) = \sum_{\ell=1}^{t} \alpha_\ell^* \varphi_\ell(x). \tag{8}$$

$\boldsymbol{\alpha}^*$ is the unknown optimal parameter under G:

$$\boldsymbol{\alpha}^* \equiv \underset{\boldsymbol{\alpha}}{\mathrm{argmin}}\, G. \tag{9}$$

$r(x)$ is the residual function orthogonal to $\{\varphi_\ell(x)\}_{\ell=1}^t$ in $\mathcal{F}$, i.e.,

$$\langle r, \varphi_\ell \rangle_{\mathcal{F}} = 0 \quad \text{for } \ell = 1, 2, \ldots, t. \tag{10}$$

Without loss of generality, we normalize $r(x)$ as

$$\|r\|_{\mathcal{F}} = 1. \tag{11}$$

Thus the function $r(x)$ governs the nature of the model error and δ (≥ 0) is the magnitude of the error.

Geometrically, in the functional Hilbert space $\mathcal{F}$, $g(x)$ is the orthogonal projection of $f(x)$ onto the subspace spanned by $\{\varphi_\ell(x)\}_{\ell=1}^t$ and $\delta r(x)$ is the residual (see Figure 3).

Let U be a $t \times t$ matrix with the (ℓ, ℓ')-th element

$$U_{\ell,\ell'} = \langle \varphi_\ell, \varphi_{\ell'} \rangle_{\mathcal{F}}. \tag{12}$$

In the following theoretical analysis, we assume that U is accessible.

2.4 Parameter Learning

We learn the parameter $\boldsymbol{\alpha}$ in our linear regression model (1) by a linear learning method, i.e., a learned parameter $\widehat{\boldsymbol{\alpha}}$ is given by the following form:

$$\widehat{\boldsymbol{\alpha}} = L\boldsymbol{y}, \tag{13}$$

where L is a $t \times n$ matrix called the *learning matrix* and

$$\boldsymbol{y} = (y_1, y_2, \ldots, y_n)^\top. \tag{14}$$

We assume that L does not depend on the noise in $\boldsymbol{y}$.

Adaptive importance weighted least squares (AIWLS) introduced in Chapter 6 is an example of linear learning methods:

$$\widehat{\boldsymbol{\alpha}}_{\mathrm{AIWLS}} \equiv \mathop{\mathrm{argmin}}_{\boldsymbol{\alpha}} \left[\sum_{i=1}^{n} \left(\frac{p_{\mathrm{te}}(x_i)}{p_{\mathrm{tr}}(x_i)} \right)^{\lambda} (\widehat{f}(x_i; \boldsymbol{\alpha}) - y_i)^2 \right], \tag{15}$$

where $0 \leq \lambda \leq 1$. We call λ a *flattening parameter* since it flattens the importance weights. The corresponding learning matrix L_{AIWLS} is given by

$$L_{\mathrm{AIWLS}} = (X^{\top} W^{\lambda} X)^{-1} X^{\top} W^{\lambda}, \tag{16}$$

where W is the diagonal matrix with diagonal element being the *importance*:

$$W_{i,i} = \frac{p_{\mathrm{te}}(x_i)}{p_{\mathrm{tr}}(x_i)}. \tag{17}$$

In the following, we assume that the importance is known. If it is unknown, we may estimate it by proper methods such as *kernel mean matching* (KMM, see Chapter 8), *kernel logistic regression* (see Chapter 9), the *Kullback-Leibler importance estimation procedure* (KLIEP, see Sugiyama et al., 2008).

3 Model Selection

In this section, we address the problem of model selection in the conditional expectation framework. Here, the term 'model' refers to the number t and the type $\{\varphi_{\ell}(x)\}_{\ell=1}^{t}$ of basis functions. Some tuning parameters contained in the learning matrix L, e.g., the flatteining parameter λ in AIWLS (15), are also included in the model.

The goal of model selection is to choose the best model M^* from a model set $\mathcal{M}$ such that the generalization error G is minimized.

$$M^* \equiv \mathop{\mathrm{argmin}}_{M \in \mathcal{M}} G(M). \tag{18}$$

The true generalization error G is inaccessible since it contains the unknown target function $f(x)$ (see (6))—in practice, we replace G by its estimator $\widehat{G}$. Therefore, the main goal of model selection research is to obtain an accurate estimator of the generalization error.

In this section, we introduce a generalization error estimator called the *importance-weighted subspace information criterion* (IWSIC) (Sugiyama and Müller, 2005). IWSIC is an extension of SIC, which is a generalization error estimator derived within the conditional expectation framework (Sugiyama and Ogawa, 2001, 2002; Sugiyama and Müller, 2002). IWSIC is shown to possess proper unbiasedness even under covariate shift. For simplicity, we consider fixed basis functions $\{\varphi_{\ell}(x)\}_{\ell=1}^{t}$ and focus on choosing the flattening parameter λ in AIWLS (15). However, IWSIC can be generally used for choosing basis functions and moreover the learning matrix L.

3.1 IWSIC

The generalization error G (3) is expressed as

$$\begin{aligned} G &= \left\|\widehat{f}\right\|_{\mathcal{F}}^2 - 2\left\langle \widehat{f}, g + \delta r\right\rangle_{\mathcal{F}} + \|f\|_{\mathcal{F}}^2 + \sigma^2 \\ &= \langle U\boldsymbol{\alpha}, \boldsymbol{\alpha}\rangle - 2\langle U\boldsymbol{\alpha}, \boldsymbol{\alpha}^*\rangle + C + \sigma^2, \end{aligned} \tag{19}$$

where C is constant:

$$C \equiv \|f\|_{\mathcal{F}}^2. \tag{20}$$

In (19), the first term $\langle U\boldsymbol{\alpha}, \boldsymbol{\alpha}\rangle$ is accessible and the third term C and the fourth term σ^2 are constants independent of the model. For this reason, we focus on estimating the second term '$\langle U\boldsymbol{\alpha}, \boldsymbol{\alpha}^*\rangle$'. Let

$$G' \equiv \langle U\boldsymbol{\alpha}, \boldsymbol{\alpha}\rangle - 2\langle U\boldsymbol{\alpha}, \boldsymbol{\alpha}^*\rangle = G - C - \sigma^2, \tag{21}$$

which is an essential part of G.

A basic idea of IWSIC is to replace the unknown $\boldsymbol{\alpha}^*$ by its linear estimator $\widetilde{\boldsymbol{\alpha}}$:

$$\widetilde{\boldsymbol{\alpha}} \equiv \widetilde{L}\boldsymbol{y}, \tag{22}$$

where

$$\widetilde{L} \equiv (X^\top W X)^{-1} X^\top W. \tag{23}$$

Note that $\widetilde{\boldsymbol{\alpha}}$ is an unbiased estimator of $\boldsymbol{\alpha}^*$ if the model is correct (i.e., $\delta = 0$); otherwise it is asymptotically unbiased in general.

However, simply replacing $\boldsymbol{\alpha}^*$ by $\widetilde{\boldsymbol{\alpha}}$ induces a bias in generalization error estimation since the same sample $\boldsymbol{y}$ is used for obtaining $\widehat{\boldsymbol{\alpha}}$ and $\widetilde{\boldsymbol{\alpha}}$—here, we are addressing the bias in terms of the conditional expectation over training output values $\{y_i\}_{i=1}^n$ given training input points $\{x_i\}_{i=1}^n$. The bias can be expressed as

$$\mathbf{E}_{\boldsymbol{y}}\left[\langle U\boldsymbol{\alpha}, \widetilde{\boldsymbol{\alpha}}\rangle - \langle U\boldsymbol{\alpha}, \boldsymbol{\alpha}^*\rangle\right] = \mathbf{E}_{\boldsymbol{y}}\left[\left\langle U\boldsymbol{\alpha}, \widetilde{L}(\boldsymbol{y} - \boldsymbol{z})\right\rangle\right], \tag{24}$$

where $\mathbf{E}_{\boldsymbol{y}}[\cdot]$ denotes the expectation over $\boldsymbol{y}$ (or equivalently $\{\epsilon_i\}_{i=1}^n$) and

$$\boldsymbol{z} \equiv (f(x_1), f(x_2), \ldots, f(x_n))^\top. \tag{25}$$

Based on (24), we define

$$\text{preIWSIC} \equiv \langle U\boldsymbol{\alpha}, \boldsymbol{\alpha}\rangle - 2\langle U\boldsymbol{\alpha}, \widetilde{\boldsymbol{\alpha}}\rangle + 2\mathbf{E}_{\boldsymbol{y}}\left[\left\langle U\boldsymbol{\alpha}, \widetilde{L}(\boldsymbol{y} - \boldsymbol{z})\right\rangle\right]. \tag{26}$$

If we can compute (or approximate) the third term in preIWSIC (26), the entire criterion becomes accessible and therefore it can be used for model selection.

If the learning matrix L is determined based on AIWLS (15), we have

$$\mathbf{E}_{\boldsymbol{y}}\left[\left\langle U\widehat{\boldsymbol{\alpha}}, \widetilde{L}(\boldsymbol{y} - \boldsymbol{z})\right\rangle\right] = \sigma^2 \mathbf{tr}\left(UL\widetilde{L}^\top\right). \tag{27}$$

Let us replace the unknown noise variance σ^2 by an ordinary estimator $\widehat{\sigma}^2$:

$$\widehat{\sigma}^2 \equiv \frac{\left\|X(X^\top X)^{-1}X^\top \boldsymbol{y} - \boldsymbol{y}\right\|^2}{n-t}, \tag{28}$$

which is known to be unbiased if $\delta = 0$. Summarizing the above approximations, we have IWSIC:

$$\text{IWSIC} \equiv \langle U\widehat{\boldsymbol{\alpha}}, \widehat{\boldsymbol{\alpha}}\rangle - 2\,\langle U\widehat{\boldsymbol{\alpha}}, \widetilde{\boldsymbol{\alpha}}\rangle + 2\widehat{\sigma}^2\mathbf{tr}\left(UL\widetilde{L}^\top\right). \tag{29}$$

IWSIC satisfies

$$\mathbf{E}_{\boldsymbol{y}}\left[\text{IWSIC} - G'\right] = O_p(\delta n^{-\frac{1}{2}}), \tag{30}$$

where O_p denotes the asymptotic order in probability. This means that IWSIC is an exact unbiased estimator of the essential generalization error G' if the model is correct (i.e., $\delta = 0$); generally, IWSIC is asymptotically unbiased with asymptotic order $n^{-\frac{1}{2}}$. In addition to the unbiasedness, IWSIC is shown to be useful for comparing the generalization error of two different models (Sugiyama and Müller, 2005).

Eq. (30) further shows that the bias of IWSIC is proportional to the model error δ. Thus IWSIC is more accurate if the target model has a smaller model error. This is practically a useful property in model selection because of the following reason. The goal of model selection is to choose the best model from a model set $\mathcal{M}$. The set $\mathcal{M}$ may contain various models including good ones and poor ones. In practice, it may not be difficult to distinguish very poor models from good ones; just using a rough estimator of the generalization error would be enough for this purpose. Therefore, what is really important in model selection is how to choose a very good model from a set of good models. Usually good models have small model errors and IWSIC is accurate for such models. For this reason, IWSIC is most useful when choosing a very good model from a set of good models.

A variance reduction method of SIC is discussed in Sugiyama et al. (2004), which could be used for further improving the model selection performance of IWSIC. IWSIC can also be extended to the situation where the learning transformation L is non-linear (Sugiyama, 2007).

In the above discussion, the matrix U (see (12)) and the importance $\{p_{\mathrm{te}}(x_i)/p_{\mathrm{tr}}(x_i)\}_{i=1}^n$ (see (17)) are assumed known. Even when they are estimated from data, the unbiasedness of IWSIC is still approximately maintained (Sugiyama and Müller, 2005).

3.2 Relation to Other Model Selection Methods

IWSIC is shown to possess proper unbiasedness within the conditional expectation framework. Here, we qualitatively compare IWSIC with other model selection methods.

3.2.1 Importance-Weighted AIC

The modified AIC given in Chapter 6, which we refer to as *Important-Weighted AIC* (IWAIC) here, is unbiased in terms of the full expectation over the training set.

For the linear regression model (1) with the linear learning method (13), IWAIC is expressed as follows (we properly shifted and rescaled it for better comparison):

$$\mathrm{IWAIC} = \left\langle \widehat{U}\widehat{\boldsymbol{\alpha}}, \widehat{\boldsymbol{\alpha}} \right\rangle - 2\left\langle \widehat{U}\widehat{\boldsymbol{\alpha}}, \widetilde{\boldsymbol{\alpha}} \right\rangle + 2\mathbf{tr}\left(\widehat{U}L\widehat{\Sigma}\widetilde{L}^\top \right), \tag{31}$$

where

$$\widehat{U} \equiv \frac{1}{n}X^\top WX, \tag{32}$$

and $\widehat{\Sigma}$ is the diagonal matrix with the i-th diagonal element

$$\widehat{\Sigma}_{i,i} \equiv (y_i - \widehat{f}(x_i; \widehat{\boldsymbol{\alpha}}))^2. \tag{33}$$

The appearances of IWAIC and IWSIC are similar but different in two aspects.

(i) The matrix U in IWSIC is replaced by its empirical estimate $\widehat{U}$ in IWAIC.

(ii) Instead of $\widehat{\Sigma}$ in IWAIC, $\widehat{\sigma}^2\mathbf{I}$ is used in IWSIC, where $\mathbf{I}$ denotes the identity matrix.

IWAIC satisfies

$$\mathbf{E}_{X,\boldsymbol{y}}\left[\mathrm{IWAIC} - G'\right] = o(n^{-1}), \tag{34}$$

where $\mathbf{E}_{X,\boldsymbol{y}}[\cdot]$ denotes the expectation over $\{(x_i, y_i)\}_{i=1}^n$. This shows that IWAIC has a smaller asymptotic bias in the full expectation analysis. On the other hand, if only the conditional expectation of training output values $\boldsymbol{y}$ given training input points X is taken, IWAIC satisfies

$$\mathbf{E}_{\boldsymbol{y}}\left[\mathrm{IWAIC} - G'\right] = O_p(n^{-\frac{1}{2}}), \tag{35}$$

which is the same asymptotic order as IWSIC (see (30)). However, a crucial difference is that the bias of IWAIC is not proportional to the model error δ. In approximately linear regression where the model error is $\delta = o(1)$ with respect to n, the bias of IWSIC is

$$\mathbf{E}_{\boldsymbol{y}}\left[\mathrm{IWSIC} - G'\right] = o_p(n^{-\frac{1}{2}}), \tag{36}$$

which is smaller than IWAIC. Thus IWSIC is more accurate than IWAIC in approximate linear regression.

Note that the range of IWAIC is not limited to linear regression; it can be applied to any statistically regular models (Watanabe, 2001) and any smooth loss functions.

3.2.2 Importance-Weighted CV

Cross validation (CV) is another popular method for model selection (Stone, 1974; Wahba, 1990), which gives an estimate of the generalization error G. Under covariate shift, a variant of CV called *importance-weighted CV* (IWCV) has proper unbiasedness (Sugiyama et al., 2007). In IWCV, the training set $\mathcal{T} = \{(x_i, y_i)\}_{i=1}^n$ is randomly divided into k

disjoint subsets $\{\mathcal{T}_i\}_{i=1}^k$ with (approximately) same size. The *k-fold IWCV* estimate of the generalization error G is given by

$$k\mathrm{IWCV} \equiv \frac{1}{k}\sum_{r=1}^{k}\frac{1}{|\mathcal{T}_r|}\sum_{(x,y)\in\mathcal{T}_r}(\widehat{f}(x;\widehat{\boldsymbol{\alpha}}_{\mathcal{T}_r})-y)^2, \tag{37}$$

where $\widehat{f}(x;\widehat{\boldsymbol{\alpha}}_{\mathcal{T}_r})$ is a function learned from $\{\mathcal{T}_i\}_{i\neq r}$. That is, $\mathcal{T}_r$ is not used for learning, but is used for computing the validation error. When $k=n$, kIWCV is particularly called *leave-one-out IWCV* (LOOIWCV):

$$\mathrm{LOOIWCV} \equiv \frac{1}{n}\sum_{r=1}^{n}(\widehat{f}(x_r;\widehat{\boldsymbol{\alpha}}_r)-y_r)^2, \tag{38}$$

where $\widehat{f}(x;\widehat{\boldsymbol{\alpha}}_r)$ is a function learned from $\{(x_i,y_i)\}_{i\neq r}$.

LOOIWCV is almost unbiased in the full expectation framework.

$$\mathbf{E}_{X,\boldsymbol{y}}\left[\mathrm{LOOIWCV}\right] = G^{(n-1)} \approx G^{(n)}, \tag{39}$$

where $G^{(n)}$ is the expected generalization error over all the training set with size n:

$$G^{(n)} = \mathbf{E}_{X,\boldsymbol{y}}\left[G\right]. \tag{40}$$

Thus LOOIWCV with n training samples is an exact unbiased estimator of the expected generalization error with $n-1$ training samples. However, in the conditional expectation framework, its unbiasedness is only asymptotic:

$$\mathbf{E}_{\boldsymbol{y}}\left[\mathrm{LOOIWCV}\right] = \mathbf{E}_{\boldsymbol{y}}\left[G\right] + O_p(n^{-\frac{1}{2}}). \tag{41}$$

This means that LOOIWCV has the same asymptotic order as IWSIC (see (30)). However, the bias of IWSIC is proportional to the model error δ, so IWSIC has a smaller bias than LOOIWCV in approximately linear regression.

Note that the unbiasedness of IWCV is valid for any loss function, any model, and any parameter learning method; even non-parametric learning methods are allowed.

3.3 Numerical Examples

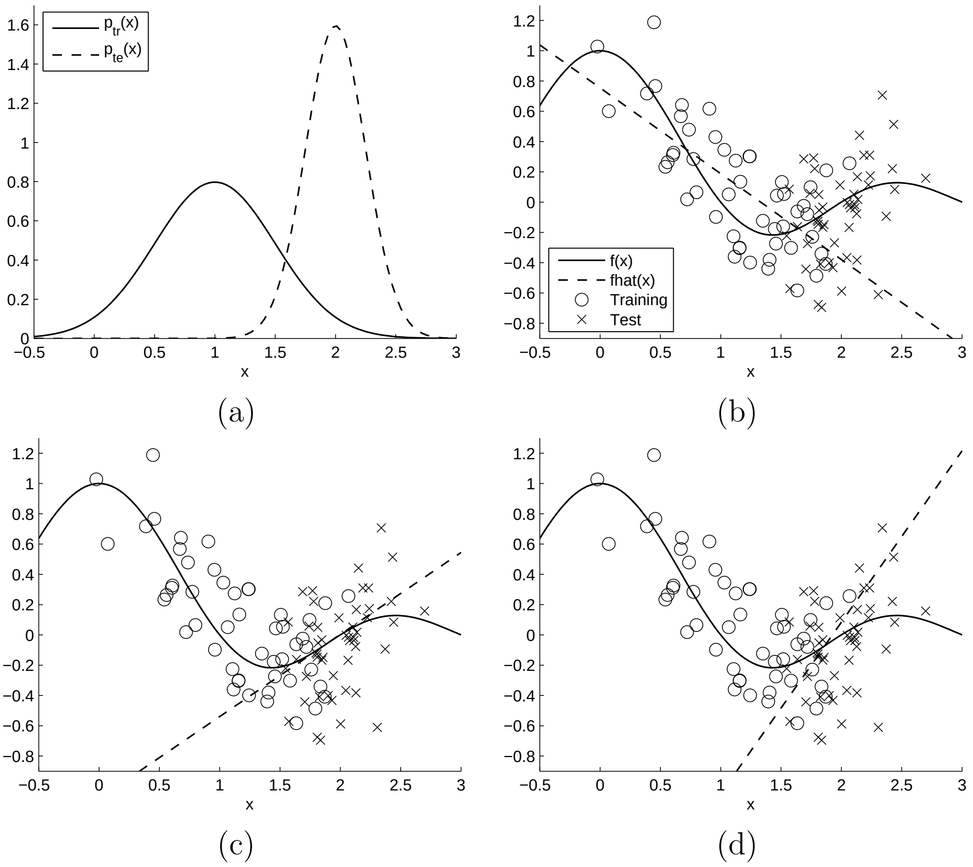
Here, we illustrate how IWSIC works through numerical experiments.

Let the input dimension be $d=1$ and the target function $f(x)$ be

$$f(x) = \mathrm{sinc}(x). \tag{42}$$

We use the following linear regression model for learning:

$$\widehat{f}(x) = \alpha_0 + \alpha_1 x. \tag{43}$$



(a) (b) (c) (d)

Figure 4: (a) Training and test input densities. (b), (c), and (d) Learning target function and functions learned by AIWLS with $\lambda = 0, 0.5, 1$.

We determine the parameters α_0 and α_1 by AIWLS (15). Let the training input distribution be Gaussian with mean 1 and standard deviation 1/2, and let the test input distribution be Gaussian with mean 2 and standard deviation 1/4. Let the conditional distribution $P(y|x)$ be Gaussian with mean $\mathrm{sinc}(x)$ and standard deviation 1/2, and let the number of training samples be

$$n = 50, 100, 200. \tag{44}$$

The above setting is summarized in Figure 4(a).

In addition to the training samples, we draw 1000 test unlabeled samples and estimate the importance by KLIEP (Sugiyama et al., 2008) using these data samples. Figure 4(b)–(d) depicts examples of functions learned by AIWLS with flattening parameter $\lambda = 0, 0.5, 1$. Our model selection task here is to choose the flattening parameter λ in AIWLS from

$$\lambda = 0, 0.1, 0.2, \ldots, 1. \tag{45}$$

We use IWSIC, IWAIC and IWCV for the selection of λ. The simulation is repeated 30000 times for each n. The obtained generalization error by each model selection method is

Table 1: Means and standard deviations of generalization error. All values in the table are multiplied by 10^2. The best method and comparable ones by the t-test at the significance level 5% are marked by '∘'.

n	IWSIC	IWAIC	IWCV
50	°12.01±10.86	13.50±13.54	12.22±11.85
100	°8.57±3.92	9.01±4.56	°8.63±4.01
200	°7.34±1.85	7.54±2.16	°7.37±1.95

summarized in Table 1, showing that IWSIC is significantly better than other approaches particularly when n is small.

4 Active Learning

In this section, we address the problem of active learning in the conditional expectation framework. The goal of (batch) active learning is to choose training input points $\{x_i\}_{i=1}^n$ such that the generalization error G is minimized. However, directly optimizing $\{x_i\}_{i=1}^n$ may be computationally hard since n input points of d dimensions needs to be simultaneously optimized. Here, we avoid this difficulty by optimizing the training input density $p_{\mathrm{tr}}(x)$ from which we draw training input points:

$$p^*_{\mathrm{tr}} \equiv \operatorname*{argmin}_{p_{\mathrm{tr}}} G(p_{\mathrm{tr}}). \tag{46}$$

The true generalization error G is inaccessible since it contains the unknown target function $f(x)$. Therefore, the main goal of active learning research is to obtain an accurate estimator of the generalization error, which is actually the same as model selection. However, generalization error estimation in active learning is generally harder than model selection since the generalization error has to be estimated *before* observing training output values $\{y_i\}_{i=1}^n$.

We assume that the test input density $p_{\mathrm{te}}(x)$ is known and the parameter $\boldsymbol{\alpha}$ is learned by IWLS, i.e.,

$$\widehat{\boldsymbol{\alpha}}_{\mathrm{IWLS}} \equiv \operatorname*{argmin}_{\boldsymbol{\alpha}} \left[\sum_{i=1}^n \frac{p_{\mathrm{te}}(x_i)}{p_{\mathrm{tr}}(x_i)} (\widehat{f}(x_i;\boldsymbol{\alpha}) - y_i)^2 \right]. \tag{47}$$

The corresponding learning matrix L_{IWLS} is given by

$$L_{\mathrm{IWLS}} = (X^\top W X)^{-1} X^\top W. \tag{48}$$

In this section, we introduce an active learning method called *ALICE* (Active Learning using Importance-weighted least-squares learning based on Conditional Expectation of the generalization error) (Sugiyama, 2006). ALICE is an extension of the traditional *variance-only* method (Fedorov, 1972; Cohn et al., 1996; Fukumizu, 2000) to approximately correct models (see Section 4.2 for detail).

4.1 ALICE

The conditional expectation of the generalization error G over training output values $\{y_i\}_{i=1}^n$ given training input points $\{x_i\}_{i=1}^n$ can be decomposed as

$$\mathbf{E}_{\boldsymbol{y}}\left[G\right] = B + V + \delta^2 + \sigma^2, \tag{49}$$

where

$$B \equiv \left\| g - \mathbf{E}_{\boldsymbol{y}}\left[\widehat{f}\right] \right\|_{\mathcal{F}}^2, \tag{50}$$

$$V \equiv \mathbf{E}_{\boldsymbol{y}}\left[\left\| \widehat{f} - \mathbf{E}_{\boldsymbol{y}}\left[\widehat{f}\right] \right\|_{\mathcal{F}}^2\right] = \sigma^2 \mathbf{tr}\left(U L_{\mathrm{IWLS}} L_{\mathrm{IWLS}}^\top\right). \tag{51}$$

B is the squared conditional bias and V is the conditional variance of the learned function. δ^2 and σ^2 are constants. Let

$$G'' \equiv G - \delta^2 - \sigma^2, \tag{52}$$

which is an essential part of the generalization error G. Note that it is different from G' (cf. (21)).

The bias term B depends on the unknown target function $f(x)$. Therefore, it is generally not possible to estimate the bias term B before observing $\{y_i\}_{i=1}^n$ since we have no information on the target function $f(x)$. On the other hand, the variance term V only depends on the learned function, and (51) implies that V can be computed without $\{y_i\}_{i=1}^n$ up to the scaling factor σ^2, which is an unknown noise variance. The basic idea of variance-only active learning methods is to guarantee that B can be safely ignored and focus on evaluating V/σ^2; when IWLS (47) is used for parameter learning, we can show that

$$B = O_p(\delta^2 n^{-1}), \tag{53}$$

$$V = O_p(n^{-1}). \tag{54}$$

Based on these, ALICE is defined as

$$\mathrm{ALICE} \equiv \mathbf{tr}\left(U L_{\mathrm{IWLS}} L_{\mathrm{IWLS}}^\top\right). \tag{55}$$

The use of ALICE can be justified in approximate linear regression, i.e., if the model error is $\delta = o(1)$ with respect to n, ALICE satisfies

$$\sigma^2 \mathrm{ALICE} - G'' = o_p(n^{-1}). \tag{56}$$

4.2 Relation to Other Active Learning Methods

ALICE is shown to be a sound active learning criterion in approximately linear regression. Here, we qualitatively compare ALICE with other active learning methods.

4.2.1 Traditional Variance-Only Method with Ordinary Least Squares

A traditional approach to variance-only active learning employs *ordinary least squares* (OLS) for parameter learning, i.e.,

$$\widehat{\boldsymbol{\alpha}}_{\mathrm{OLS}} \equiv \mathop{\mathrm{argmin}}_{\boldsymbol{\alpha}} \left[\sum_{i=1}^{n} (\widehat{f}(x_i;\boldsymbol{\alpha}) - y_i)^2 \right]. \tag{57}$$

The corresponding learning matrix L_{OLS} is given by

$$L_{\mathrm{OLS}} = (X^\top X)^{-1} X^\top. \tag{58}$$

Based on OLS, an active learning criterion, which we refer to as the *Variance-Only criterion with Least-Squares* (VOLS) here, is given as follows (Fedorov, 1972; Cohn et al., 1996; Fukumizu, 2000):

$$\mathrm{VOLS} = \mathbf{tr}\left(U L_{\mathrm{OLS}} L_{\mathrm{OLS}}^\top\right). \tag{59}$$

The use of VOLS is justified also in approximate linear regression, i.e., if the model error is $\delta = o(n^{-\frac{1}{2}})$, VOLS satisfies the following property (Sugiyama, 2006):

$$\sigma^2 \mathrm{VOLS} - G'' = o_p(n^{-1}). \tag{60}$$

However, the condition on the model error δ is stronger than for ALICE.

4.2.2 Full Expectation Variance-Only Method

Within the full expectation framework, Kanamori and Shimodaira (2003) proved that the expected generalization error is asymptotically expressed as follows (see also Chapter 6):

$$\mathbf{E}_{X,\boldsymbol{y}}\left[G''\right] = \frac{1}{n}\mathbf{tr}\left(U^{-1}H\right) + O(n^{-\frac{3}{2}}), \tag{61}$$

where H is the n-dimensional square matrix defined by

$$H = S + \sigma^2 T. \tag{62}$$

S and T are the t-dimensional square matrices with the (ℓ, ℓ')-th elements

$$S_{\ell,\ell'} = \int_{\mathcal{X}} \varphi_\ell(x)\varphi_{\ell'}(x)(\delta r(x))^2 \frac{(p_{\mathrm{te}}(x))^2}{p_{\mathrm{tr}}(x)} dx, \tag{63}$$

$$T_{\ell,\ell'} = \int_{\mathcal{X}} \varphi_\ell(x)\varphi_{\ell'}(x) \frac{(p_{\mathrm{te}}(x))^2}{p_{\mathrm{tr}}(x)} dx. \tag{64}$$

Note that $\frac{1}{n}\mathbf{tr}\left(U^{-1}S\right)$ corresponds to the squared bias while $\frac{\sigma^2}{n}\mathbf{tr}\left(U^{-1}T\right)$ corresponds to the variance. T is accessible by assumption, but S is not (due to $\delta r(x)$).

Based on this decomposition, a variance-only active learning criterion, which we refer to as the *Full Expectation Variance-Only* (FEVO) method, is given as follows (Wiens, 2000):

$$\mathrm{FEVO} = \frac{1}{n}\mathbf{tr}\left(U^{-1}T\right). \tag{65}$$

Sugiyama (2006) proved that the use of FEVO is also justified in approximate linear regression, i.e., if the model error is $\delta = o(1)$ with respect to n, FEVO satisfies

$$\sigma^2\mathrm{FEVO} - G'' = o(n^{-1}). \tag{66}$$

This implies that the asymptotic order of FEVO is the same as ALICE. Furthermore, ALICE and FEVO are actually equivalent asymptotically, i.e.,

$$\mathrm{ALICE} - \mathrm{FEVO} = O_p(n^{-\frac{3}{2}}). \tag{67}$$

However, they are different in the order of $O_p(n^{-1})$. To investigate this difference more precisely, let us measure the goodness of a generalization error estimator $\widehat{G}$ by

$$\mathbf{E}_y\left[(\widehat{G} - G'')^2\right]. \tag{68}$$

If $\delta = o(n^{-\frac{1}{4}})$ and terms of $o_p(n^{-3})$ are ignored, we have

$$\mathbf{E}_y\left[(\sigma^2\mathrm{ALICE} - G'')^2\right] \leq \mathbf{E}_y\left[(\sigma^2\mathrm{FEVO} - G'')^2\right]. \tag{69}$$

Thus, for approximate linear regression with $\delta = o(n^{-\frac{1}{4}})$, ALICE is a more accurate estimator of the generalization error than FEVO in the above sense.

FEVO does not depend on the realization of training input points $\{x_i\}_{i=1}^n$ (though it does depend on the training input density $p_{\mathrm{tr}}(x)$). Thanks to this property, the optimal training input density $\widehat{p}_{\mathrm{tr}}(x)$ can be obtained in a close-form as follows (Wiens, 2000):

$$\widehat{p}_{\mathrm{tr}}(x) = \frac{\widehat{h}(x)}{\int_{\mathcal{X}}\widehat{h}(x)dx}, \tag{70}$$

where

$$\widehat{h}(x) = p_{\mathrm{te}}(x)\left(\sum_{\ell,\ell'=1}^{t}[U^{-1}]_{\ell,\ell'}\varphi_\ell(x)\varphi_{\ell'}(x)\right)^{\frac{1}{2}}. \tag{71}$$

4.2.3 Full Expectation Bias-Variance Method

Another idea of approximating H in (61) is a two-stage sampling scheme introduced in Chapter 6: in the first stage, $\widetilde{n}$ $(\leq n)$ training input points $\{\widetilde{x}_i\}_{i=1}^{\widetilde{n}}$ are created independently following the test input distribution with density $p_{\mathrm{te}}(x)$, and the corresponding training output values $\{\widetilde{y}_i\}_{i=1}^{\widetilde{n}}$ are gathered. Then a consistent estimator $\widetilde{H}$ of the unknown matrix H in (61) can be obtained based on $\{(\widetilde{x}_i, \widetilde{y}_i)\}_{i=1}^{\widetilde{n}}$ as

$$\widetilde{H}_{\ell,\ell'} = \frac{1}{\widetilde{n}} \sum_{i=1}^{\widetilde{n}} \frac{p_{\mathrm{te}}(\widetilde{x}_i)}{p_{\mathrm{tr}}(\widetilde{x}_i)} (\widetilde{y}_i - \widehat{f}(\widetilde{x}_i; \widetilde{\boldsymbol{\alpha}}_{\mathrm{OLS}}))^2 \varphi_\ell(\widetilde{x}_i)\varphi_{\ell'}(\widetilde{x}_i), \tag{72}$$

where $\widetilde{\boldsymbol{\alpha}}_{\mathrm{OLS}}$ is obtained from $\{(\widetilde{x}_i, \widetilde{y}_i)\}_{i=1}^{\widetilde{n}}$ by OLS (57). This corresponds to estimating the bias term S and the noise variance σ^2 from $\{(\widetilde{x}_i, \widetilde{y}_i)\}_{i=1}^{\widetilde{n}}$. U^{-1} is also replaced by a consistent estimator $\widetilde{U}^{-1}$:

$$\widetilde{U}_{\ell,\ell'} = \frac{1}{\widetilde{n}} \sum_{i=1}^{\widetilde{n}} \varphi_\ell(\widetilde{x}_i)\varphi_{\ell'}(\widetilde{x}_i). \tag{73}$$

Based on these approximations, an active learning criterion, which we refer to as the *Full Expectation Bias-Variance* (FEBV) method here, is given as

$$\mathrm{FEBV} = \frac{1}{n}\mathbf{tr}\left(\widetilde{U}^{-1}\widetilde{H}\right). \tag{74}$$

In the second stage, this criterion is used for optimizing the location of the remaining $n - \widetilde{n}$ training input points. Kanamori and Shimodaira (2003) proved that the use of FEBV can be justified for misspecified models, i.e., for $\delta = O(1)$ with respect to n, FEBV satisfies

$$\sigma^2\mathrm{FEBV} - G'' = o(n^{-1}). \tag{75}$$

The order of δ required above is weaker than that required in ALICE or FEVO. Therefore, FEBV theoretically has a wider range of applications. However, this strong theoretical property is not necessarily useful in practice since learning with totally misspecified models (i.e., $\delta = O(1)$) may not work well due to large model errors. Furthermore, due to the two-stage sampling scheme, FEBV allows us to choose only $n - \widetilde{n}$ training input points. This can be very restrictive when the total number n is not so large.

Note that the range of FEBV is not restricted to linear regression; it can be applied to any statistically regular models (Watanabe, 2001) and any smooth loss functions.

4.3 Numerical Examples

Here, we illustrate how ALICE works through numerical experiments.

Let the input dimension be $d = 1$ and the target function $f(x)$ be

$$f(x) = 1 - x + x^2 + \delta r(x), \tag{76}$$

where

$$r(x) = \delta\frac{z^3 - 3z}{\sqrt{6}} \quad \text{with} \quad z = \frac{x - 0.2}{0.4}. \tag{77}$$

We use the following linear regression model for learning:

$$\widehat{f}(x) = \alpha_0 + \alpha_1 x + \alpha_2 x^2. \tag{78}$$

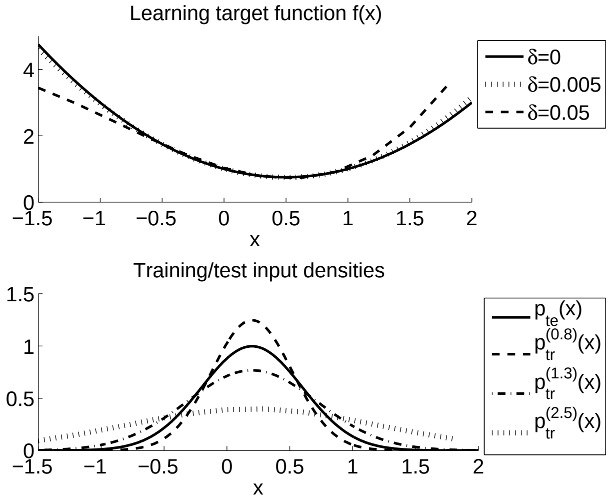


Figure 5: Target function and training/test input densities. $p_{\mathrm{tr}}^{(c)}(x)$ denotes the training input density with width parameter c.

Note that for this regression model, the residual function $r(x)$ fulfills (10) and (11). Let us consider the following three cases.

$$\delta = 0, 0.005, 0.05, \tag{79}$$

which correspond to "*correctly specified*", "*approximately correct*", and "*misspecified*" cases, respectively.

Let the test input distribution be Gaussian with mean 0.2 and standard deviation 0.4, which is assumed to be known in this illustrative simulation. Let us gather 100 training samples by active learning. Our task here is to choose the training input distribution from a set of Gaussians with mean 0.2 and standard deviation 0.4c, where

$$c = 0.8, 0.9, 1.0, \ldots, 2.5. \tag{80}$$

We add i.i.d. Gaussian noise with mean zero and standard deviation 0.3 to the training output values. The above setting is summarized in Figure 5. We repeat this simulation 1000 times for each δ.

In Table 2, the mean and standard deviation of the generalization error obtained by each method are described. FEVO* denotes the case where the closed-form solution of FEVO (see (70)) is used.

When $\delta = 0$, VOLS works significantly better than other methods. Actually, in this case, training input densities that approximately minimize the generalization error were successfully found by ALICE, FEVO, FEBV, and VOLS. This implies that the difference in the obtained error is caused not by the quality of the active learning criteria, but by the difference between IWLS and OLS since IWLS generally has larger variance than OLS (Shimodaira, 2000). Therefore, when $\delta = 0$, OLS would be more accurate than IWLS since both IWLS and OLS are unbiased. Although ALICE, FEVO, and FEBV

Table 2: The mean and standard deviation of the generalization error $G-\sigma^2$ obtained by each method for the toy data set. The best method and comparable ones by the t-test at the significance level 5% are marked by '$\circ$'. The value of VOLS for $\delta = 0.05$ is extremely large but it is not a typo. All values in the table are multiplied by 10^3.

δ	ALICE	FEVO	FEVO*	FEBV	VOLS	Passive
0	2.08±1.95	2.40±2.15	2.32±2.02	3.09±3.03	°1.31±1.70	3.11±2.78
0.005	°2.10±1.96	2.43±2.15	2.35±2.02	3.13±3.00	2.53±2.23	3.14±2.78
0.05	°4.61±2.12	4.89±2.26	4.84±2.14	5.95±3.58	124±7.4	6.01±3.43

are outperformed by VOLS, they still work better than Passive (training input density is equal to the test input density). Note that ALICE is significantly better than FEVO, FEBV, and Passive by the t-test.

When $\delta = 0.005$, ALICE gives significantly smaller errors than other methods. All the methods except VOLS work similarly to the case with $\delta = 0$, while VOLS tends to perform poorly. This result is surprising since the learning target functions with $\delta = 0$ and $\delta = 0.005$ are visually almost the same, as illustrated in the top graph of Figure 5. Therefore, intuitively the result when $\delta = 0.005$ should not be much different from the result when $\delta = 0$. However, this slight difference appears to make VOLS unreliable. Other methods are shown to be robust against model misspecification.

When $\delta = 0.05$, ALICE again works significantly better than others. FEVO still works reasonably well. The performance of FEBV is slightly degraded, although it is still better than Passive. VOLS gives extremely large errors.

The above results are summarized as follows. For all three cases ($\delta = 0, 0.005, 0.05$), ALICE, FEVO, and FEBV work reasonably well and consistently outperform Passive. Among them, ALICE appears to be better than FEVO and FEBV for all three cases. VOLS works excellently for correctly specified models, although it tends to perform poorly once the correctness of the model is violated. Therefore, ALICE is shown to be robust against model misspecification and therefore work well.

5 Active Learning with Model Selection

The problems of model selection and active learning share a common goal—minimizing the generalization error (see (18) and (46)). However, they have been studied separately as two independent problems so far. If models and training input points are optimized at the same time, the generalization performance could be further improved. We call the problem of simultaneously optimizing training input points and models *active learning with model selection*:

$$\min_{M, p_{\mathrm{tr}}} G(M, p_{\mathrm{tr}}). \tag{81}$$

This is the problem we address in this section.

5.1 Direct Approach and Active Learning/Model Selection Dilemma

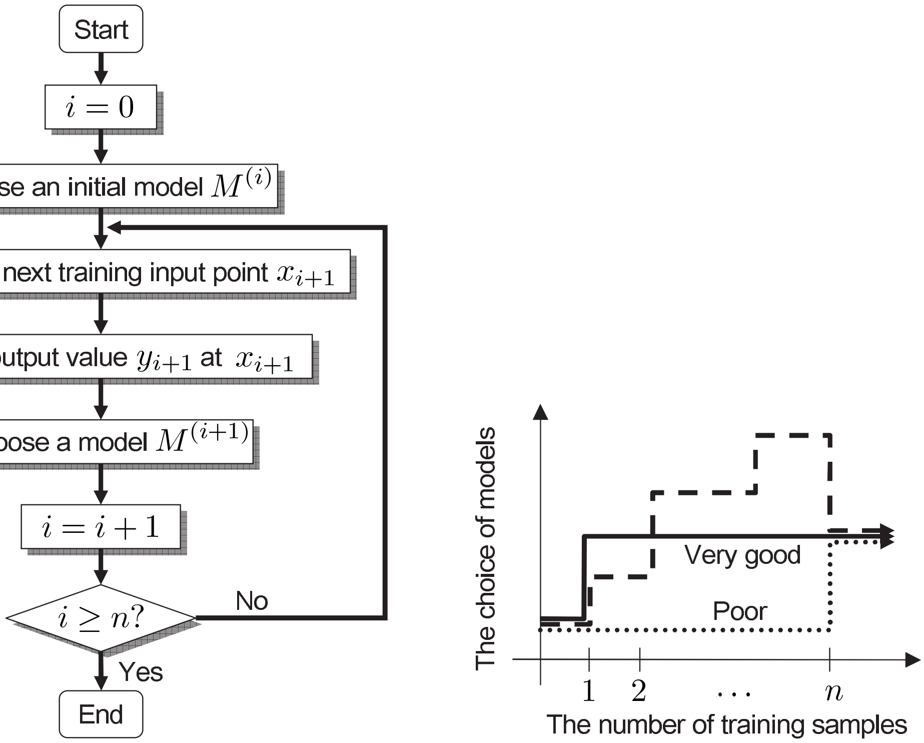
A naive and direct solution to (81) would be to simultaneously optimize M and p_{tr}. However, this direct approach may not be possible by simply combining an existing active learning method and an existing model selection method in a batch manner due to the *active learning/model selection dilemma*: when choosing the model M with existing model selection methods, the training input points (or the training input density) must have been fixed and the corresponding training output values must have been gathered (Akaike, 1974; Rissanen, 1978; Schwarz, 1978; Craven and Wahba, 1979; Shimodaira, 2000; Sugiyama and Müller, 2005). On the other hand, when selecting the training input density with existing active learning methods, the model must have been fixed (Fedorov, 1972; MacKay, 1992b; Cohn et al., 1996; Fukumizu, 2000; Wiens, 2000; Kanamori and Shimodaira, 2003; Sugiyama, 2006). For example, IWSIC (29) can not be computed without fixing the training input density (and without training output values) and ALICE (55) can not be computed without fixing the model.

If there exist training input points which are optimal for all model candidates, it is possible to solve both active learning and model selection without regard to the dilemma: choose the training input points for some model by some active learning method (e.g., ALICE), gather corresponding training output values, and perform model selection using some method (e.g., IWSIC). It is shown that such common optimal training input points exist for correctly specified trigonometric polynomial models (Sugiyama and Ogawa, 2003). However, such common optimal training input points may not exist in general and thus the range of application of this approach is limited.

5.2 Sequential Approach

A standard approach to coping with the active learning/model selection dilemma for arbitrary models would be the *sequential approach* (MacKay, 1992a), i.e., in an iterative manner, a model is chosen by a model selection method and the next input point (or a small portion) is optimized for the chosen model by an active learning method (see Figure 6(a)).

In the sequential approach, the chosen model $M^{(i)}$ varies through the online learning process (see the dashed line in Figure 6(b)), where $M^{(i)}$ denotes the model chosen at the i-th step. We refer to this phenomenon as the *model drift*. The model drift phenomenon could be a weakness of the sequential approach since the location of optimal training input points depends *strongly* on the target model in active learning; a good training input point for one model could be poor for another model. Depending on the transition of the chosen models, the sequential approach can work very well. For example, when the transition of the model is the solid line in Figure 6(b), most of the training input points are chosen for the finally selected model $M^{(n)}$ and the sequential approach has an excellent performance. However, when the transition of the model is the dotted line in Figure 6(b), the performance becomes poor since most of the training input points


Start
$i = 0$
Choose an initial model $M^{(i)}$
Choose the next training input point x_{i+1}
Gather output value y_{i+1} at x_{i+1}
Choose a model $M^{(i+1)}$
$i = i + 1$
$i \geq n?$
No
Yes
End
The choice of models
Very good
Poor
1
2
$\cdots$
n
The number of training samples

(a) Diagram (b) Transition of chosen models

Figure 6: Sequential approach to active learning with model selection.

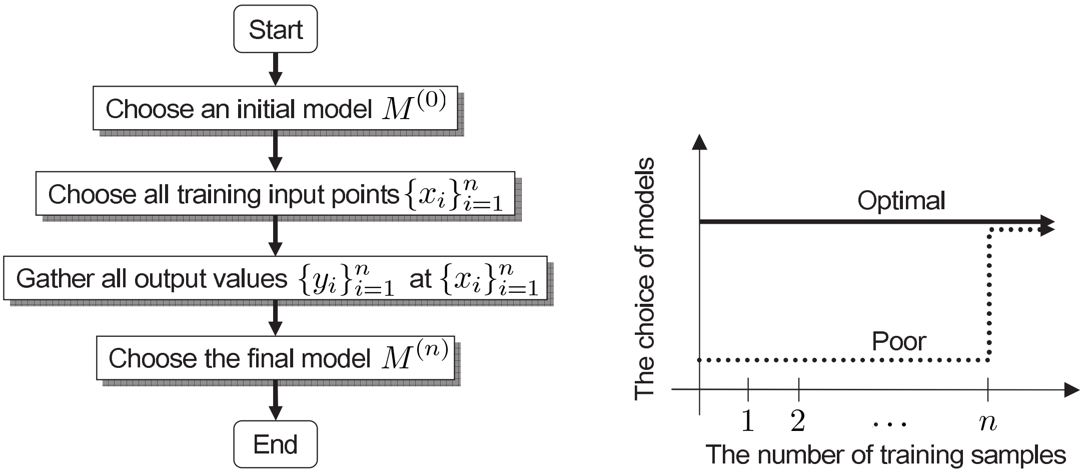
are chosen for other models. Note that we can *not* control the transition of the model properly since we do not know a priori which model will be chosen in the end. Therefore, the performance of the sequential approach is unstable.

Another issue that needs to be taken into account in the sequential approach is that the training input points are not i.i.d. in general—the choice of the $(i+1)$-th training input point x_{i+1} depends on the previously gathered samples $\{(x_j, y_j)\}_{j=1}^{i}$. Since standard model selection methods and active learning methods require the i.i.d. assumption for establishing their statistical properties such as unbiasedness and consistency, they may not be directly employed in the sequential approach (Bach, 2007).

IWSIC (29) and ALICE (55) also suffer from the violation of the i.i.d. condition and loose their unbiasedness and consistency. However, this problem can be easily settled by slightly modifying the criteria. Suppose we draw u input points from $p_{\mathrm{tr}}^{(i)}(x)$ in each iteration (let $n = uv$, where v is the number of iterations). If u tends to infinity, simply redefining the diagonal matrix W as follows makes IWSIC and ALICE still asymptotically unbiased and consistent:

$$W_{k,k} = \frac{p_{\mathrm{te}}(x_k)}{p_{\mathrm{tr}}^{(i)}(x_k)}, \tag{82}$$

where $k = (i-1)u + j$, $i = 1, 2, \ldots, v$, and $j = 1, 2, \ldots, u$. This would be another advantage of the conditional expectation approach.


Start
Choose an initial model $M^{(0)}$
Choose all training input points $\{x_i\}_{i=1}^{n}$
Gather all output values $\{y_i\}_{i=1}^{n}$ at $\{x_i\}_{i=1}^{n}$
Choose the final model $M^{(n)}$
End
The choice of models
Optimal
Poor
1
2
$\cdots$
n
The number of training samples

(a) Diagram (b) Transition of chosen models

Figure 7: Batch approach to active learning with model selection.

5.3 Batch Approach

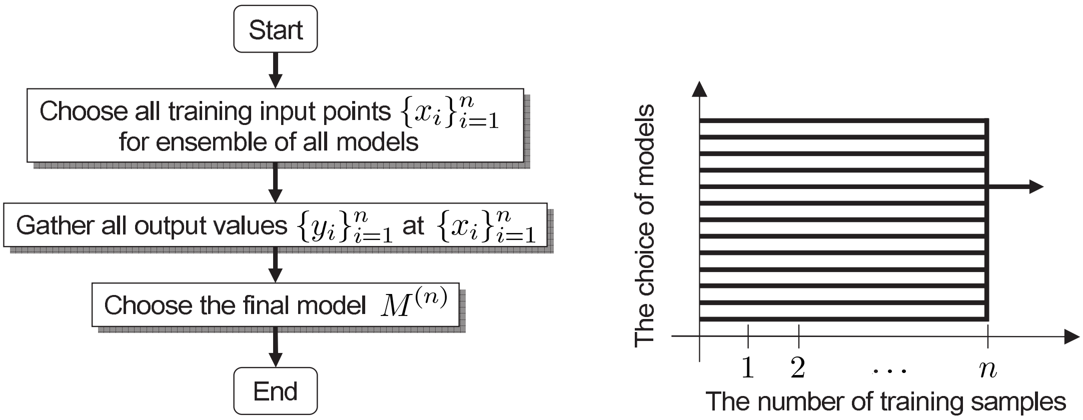
An alternative approach to active learning with model selection is to choose all the training input points for an initially chosen model $M^{(0)}$. We refer to this approach as the *batch approach* (see Figure 7(a)). Due to the batch nature, this approach does not suffer from the model drift (cf. Figure 6(b)); the batch approach can be optimal in terms of active learning if an initially chosen model $M^{(0)}$ agrees with the finally chosen model $M^{(n)}$ (see the solid line in Figure 7(b)).

The performance of this batch approach heavily depends on the initial model $M^{(0)}$. In order to choose the initial model appropriately, we may need a generalization error estimator that can be computed before observing training output values—for example, ALICE (55). However, this does not work well since ALICE only evaluates the variance of the estimator; thus using ALICE for choosing the initial model $M^{(0)}$ merely results in always selecting the simplest model in the candidates. Note that this problem is not specific to ALICE, but is common to most generalization error estimators since it is generally not possible to estimate the bias before observing training output values. For this reason, in practice, we may have to choose the initial model $M^{(0)}$ *randomly*. If we have some prior preference of models, $P(M)$, we may draw the initial model according to it.

Due to the randomness of the initial model choice, the performance of the batch approach may be unstable (see the dashed line in Figure 7(b)).

5.4 Ensemble Active Learning

The weakness of the batch approach lies in the fact that the training input points chosen by an active learning method are *overfitted* to the initially chosen model—the training input points optimized for the initial model could be poor if a different model is chosen


Start
Choose all training input points $\{x_i\}_{i=1}^n$ for ensemble of all models
Gather all output values $\{y_i\}_{i=1}^n$ at $\{x_i\}_{i=1}^n$
Choose the final model $M^{(n)}$
End
The choice of models
1
2
···
n
The number of training samples

(a) Diagram (b) Transition of chosen models

Figure 8: Ensemble approach to active learning with model selection.

later.

We may reduce the risk of overfitting by not optimizing the training input density *specifically* for a single model, but by optimizing it for *all* model candidates (see Figure 8). This allows all the models to contribute to the optimization of the training input density and thus we can hedge the risk of overfitting to a single (possibly inferior) model. Since this approach could be viewed as applying a popular idea of *ensemble learning* to the problem of active learning, this method is called *ensemble active learning* (EAL).

This idea could be realized by determining the training input density so that the *expected* generalization error over *all* model candidates is minimized:

$$\min_{p_{\mathrm{tr}}} \sum_{M} \mathrm{ALICE}_M(p_{\mathrm{tr}})P(M), \tag{83}$$

where ALICE_M denotes ALICE for a model M and $P(M)$ is the prior preference of the model M. If no prior information on goodness of the models is available, the uniform prior may be simply used.

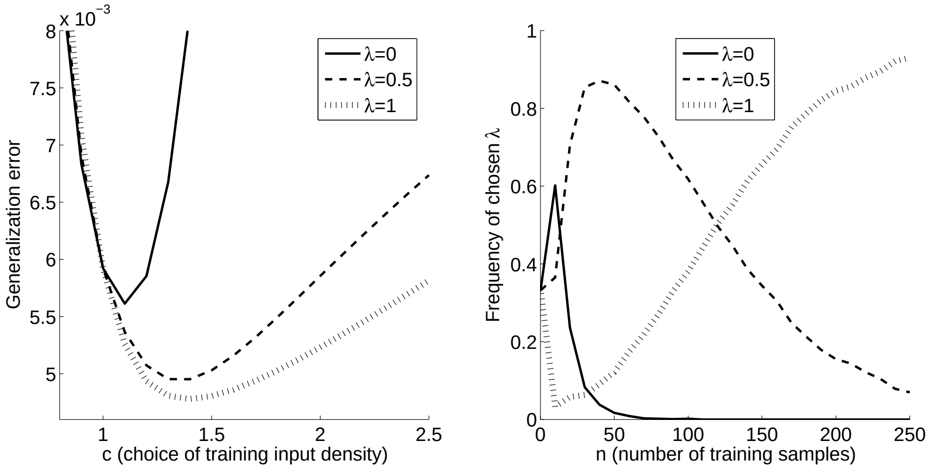
5.5 Numerical Examples

Here, we illustrate how the ensemble active learning method behaves through numerical experiments.

We use the same toy example as Section 4.3; the difference is the learning target function and parameter learning methods. In Section 4.3, the target function is changed through δ (see (76)) and IWLS is used; here we fix the target function at $\delta = 0.05$ and use AIWLS (15) for parameter learning. We choose the flattening parameter λ in AIWLS by IWSIC (29) from

$$\lambda = 0, 0.5, 1. \tag{84}$$

The choice of λ corresponds to model selection in this scenario.



(a) (b)

Figure 9: (a) Mean generalization error $G - \sigma^2$ over 1000 trials as a function of training input density c for each λ (when $n = 100$). (b) Frequency of chosen λ over 1000 trials as a function of the number of training samples.

First, we investigate the dependency between the goodness of the training input density (i.e., c) and the model (i.e., λ). For each λ and each c, we draw training input points $\{x_i\}_{i=1}^{100}$ and gather output values $\{y_i\}_{i=1}^{100}$. Then we learn the parameter by AIWLS and compute the generalization error. The mean generalization error over 1000 trials as a function of c for each λ is depicted in Figure 9(a). This graph underlines that the best training input density c could strongly depend on the model λ, implying that a training input density that is good for one model could be poor for others. For example, when the training input density is optimized for the model $\lambda = 0$, $c = 1.1$ would be an excellent choice. However, $c = 1.1$ is not so suitable for other models $\lambda = 0.5, 1$. This figure illustrates a possible weakness of the batch method: when an initially chosen model is significantly different from the finally chosen model, the training input points optimized for the initial model could be less useful for the final model and the performance is degraded.

Next, we investigate the behavior of the sequential approach. In our implementation, 10 training input points are chosen at each iteration. Figure 9(b) depicts the transition of the frequency of chosen λ in the sequential learning process over 1000 trials. It shows that the choice of models varies over the learning process; a smaller λ (which has smaller variance thus low complexity) is favored in the beginning, but a larger λ (which has larger variance thus higher complexity) tends to be chosen as the number of training samples increases. Figure 9(b) illustrates a possible weakness of the sequential method: the target model drifts during the sequential learning process and the training input points designed in an early stage could be poor for the finally chosen model.

Finally, we investigate the generalization performance of each method when the number of training samples to gather is

$$n = 100, 150, 200, 250. \tag{85}$$

Table 3: Means and standard deviations of generalization error for the toy data set. All values in the table are multiplied by 10^3. The best method and comparable ones by the t-test at the significance level 5% are marked by '$\circ$'.

n	Passive	Sequential	Batch	Ensemble
100	5.92±3.28	5.57±2.75	5.65±2.92	$^\circ$5.12±2.50
150	4.77±2.18	4.43±1.77	4.64±1.91	$^\circ$4.11±1.55
200	4.21±1.75	3.89±1.40	4.19±1.60	$^\circ$3.68±1.19
250	3.78±1.32	3.47±1.02	3.91±1.42	$^\circ$3.35±0.92

Table 3 describes the means and standard deviations of the generalization error obtained by the sequential, batch, and ensemble methods; as a baseline, we also included the result of passive learning (or equivalently $c = 1$). The table shows that all three methods tend to outperform passive learning. However, the improvement of the sequential method is not so significant, which would be caused by the model drift phenomenon (see Figure 9). The batch method also does not provide significant improvement due to the overfitting to the randomly chosen initial model (see Figure 9(a)). On the other hand, the proposed ensemble method does not suffer from these problems and works significantly better than other methods.

6 Conclusions

We introduced a conditional expectation approach to model selection and active learning under covariate shift and proved that it is more accurate than the full expectation approach in approximate linear regression. Furthermore a method to combine active learning and model selection was introduced that was nicely showing its experimental validity.

Future work will consider nonlinear extensions to the proposed methods and study their use for classification. From the practical application viewpoint we will employ covariate shift compensation techniques for BCI following the lines of Sugiyama et al. (2007) and use the novel active learning strategies for improving experimental design in computational chemistry (cf. Warmuth et al., 2003).

Acknowledgments

MS thanks support by MEXT (17700142 and 18300057), the Okawa Foundation, and the Microsoft CORE3 Project. KRM greatfully acknowledges support by the EU (PASCAL), DFG MU 987/4-1 and BMBF. The authors thank for valuable discussions with Motoaki Kawanabe.

References

H. Akaike. A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, AC-19(6):716–723, 1974.

F. Bach. Active learning for misspecified generalized linear models. In B. Schölkopf, J. Platt, and T. Hoffman, editors, *Advances in Neural Information Processing Systems 19*. MIT Press, Cambridge, MA, 2007.

C. M. Bishop. *Neural Networks for Pattern Recognition*. Clarendon Press, Oxford, 1995.

D. A. Cohn, Z. Ghahramani, and M. I. Jordan. Active learning with statistical models. *Journal of Artificial Intelligence Research*, 4:129–145, 1996.

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

R. O. Duda, P. E. Hart, and D. G. Stor. *Pattern Classification*. Wiley, New York, 2001.

V. V. Fedorov. *Theory of Optimal Experiments*. Academic Press, New York, 1972.

K. Fukumizu. Statistical active learning in multilayer perceptrons. *IEEE Transactions on Neural Networks*, 11(1):17–26, 2000.

T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, New York, 2001.

T. Kanamori and H. Shimodaira. Active learning algorithm using the maximum weighted log-likelihood estimator. *Journal of Statistical Planning and Inference*, 116(1):149–162, 2003.

D. J. C. MacKay. Bayesian interpolation. *Neural Computation*, 4(3):415–447, 1992a.

D. J. C. MacKay. Information-based objective functions for active data selection. *Neural Computation*, 4(4):590–604, 1992b.

J. Rissanen. Modeling by shortest data description. *Automatica*, 14(5):465–471, 1978.

B. Schölkopf and A. J. Smola. *Learning with Kernels*. MIT Press, Cambridge, MA, 2002.

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

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

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

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

M. Sugiyama. Generalization error estimation for non-linear learning methods. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, E90-A(7):1496–1499, 2007.

M. Sugiyama, M. Kawanabe, and K.-R. Müller. Trading variance reduction with unbiasedness: The regularized subspace information criterion for robust model selection in kernel regression. *Neural Computation*, 16(5):1077–1104, 2004.

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

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

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

M. Sugiyama, S. Nakajima, H. Kashima, P. von Bünau, and M. Kawanabe. Direct importance estimation with model selection and its application to covariate shift adaptation. In *Advances in Neural Information Processing Systems 20*, Cambridge, MA, 2008. MIT Press.

M. Sugiyama and H. Ogawa. Subspace information criterion for model selection. *Neural Computation*, 13(8):1863–1889, 2001.

M. Sugiyama and H. Ogawa. Optimal design of regularization term and regularization parameter by subspace information criterion. *Neural Networks*, 15(3):349–361, 2002.

M. Sugiyama and H. Ogawa. Active learning with model selection—Simultaneous optimization of sample points and models for trigonometric polynomial models. *IEICE Transactions on Information and Systems*, E86-D(12):2753–2763, 2003.

V. N. Vapnik. *Statistical Learning Theory.* Wiley, New York, 1998.

G. Wahba. *Spline Model for Observational Data.* Society for Industrial and Applied Mathematics, Philadelphia and Pennsylvania, 1990.

M. K. Warmuth, J. Liao, G. Rätsch, M. Mathieson, S. Putta, and C. Lemmen. Active learning with SVMs in the drug discovery process. *Chemical Information and Computer Sciences*, 43(2):667–673, 2003.

S. Watanabe. Algebraic analysis for nonidentifiable learning machines. *Neural Computation*, 13(4):899–933, 2001.

D. P. Wiens. Robust weights and designs for biased regression models: Least squares and generalized M-estimation. *Journal of Statistical Planning and Inference*, 83(2):395–412, 2000.

Is Importance Weighting Needed under Covariate Shift?

Covariate shift matters in parameter learning only when the model used for function learning is misspecified (i.e., the model is so simple that the true learning target function can not be expressed). When the model is correctly (or overly) specified, ordinary maximum likelihood estimation (MLE) is still consistent. Following this fact, there is a criticism that covariate shift adaptation by importance weighting is not needed, but just the use of a complex enough (or a non-parametric) model with ordinary MLE can settle the problem.

However, too complex models result in huge variance, so we need to choose a complex enough but not too complex model for better generalization performance. In order to perform model selection, we often use an unbiased generalization error estimation methods such as Akaike's information criterion or cross-validation. However, these generalization error estimation methods are heavily biased for misspecified models under covariate shift. Instead, importance weighted variants can eliminate the bias and therefore more reliable under covariate shift (see e.g., Chapter 6 and Chapter 7). In the model selection process, we can not avoid computing the generalization error estimates for misspecified models since the purpose is to rule out such models from appropriate ones. For this reason, the use of (and therefore estimation of) the importance weights is indispensable when covariate shift occurs. Thus accurately estimating the importance weights (see e.g., Chapter 8 and Chapter 9) is very important and we need to further investigate this issue.