Journal of Machine Learning Research 3 (2002) 323-359 Submitted 12/01; Revised 8/02; Published 11/02

# The Subspace Information Criterion for Infinite Dimensional Hypothesis Spaces

**Masashi Sugiyama** SUGI@OG.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.FHG.DE
*Fraunhofer FIRST, IDA*
*Kekuléstr. 7, 12489 Berlin and*
*Department of Computer Science, University of Potsdam*
*August-Bebel-Str.89, Haus 4, 14482 Potsdam, Germany*

**Editor:** Nello Cristianini

## Abstract

A central problem in learning is selection of an appropriate model. This is typically done by estimating the *unknown* generalization errors of a set of models to be selected from and then choosing the model with minimal generalization error estimate. In this article, we discuss the problem of model selection and generalization error estimation in the context of kernel regression models, e.g., kernel ridge regression, kernel subset regression or Gaussian process regression.

Previously, a non-asymptotic generalization error estimator called the subspace information criterion (SIC) was proposed, that could be successfully applied to *finite* dimensional subspace models. SIC is an unbiased estimator of the generalization error for the finite sample case under the conditions that the learning target function belongs to a specified reproducing kernel Hilbert space (RKHS) $\mathcal{H}$ and the reproducing kernels centered on training sample points span the whole space $\mathcal{H}$. These conditions hold only if $\dim \mathcal{H} \leq \ell$, where $\ell$ $(< \infty)$ is the number of training examples. Therefore, SIC could be applied only to finite dimensional RKHSs.

In this paper, we extend the range of applicability of SIC, and show that even if the reproducing kernels centered on training sample points do not span the whole space $\mathcal{H}$, SIC is an unbiased estimator of an essential part of the generalization error. Our extension allows the use of any RKHSs including *infinite* dimensional ones, i.e., richer function classes commonly used in Gaussian processes, support vector machines or boosting. We further show that when the kernel matrix is invertible, SIC can be expressed in a much simpler form, making its computation highly efficient. In computer simulations on ridge parameter selection with real and artificial data sets, SIC is compared favorably with other standard model selection techniques for instance leave-one-out cross-validation or an empirical Bayesian method.

**Keywords:** Generalization error, model selection, subspace information criterion, cross-validation, kernel regression, reproducing kernel Hilbert space, finite sample statistics, Gaussian processes, unbiased estimators.

## 1. Introduction

The goal of supervised learning is to obtain an underlying input-output dependency from the given training examples (e.g., Vapnik, 1982, 1995, 1998; Bishop, 1995; Devroye et al., 1996). If the dependency is successfully learned, appropriate output values can be inferred for previously unseen input points, i.e., the learning machine generalizes.

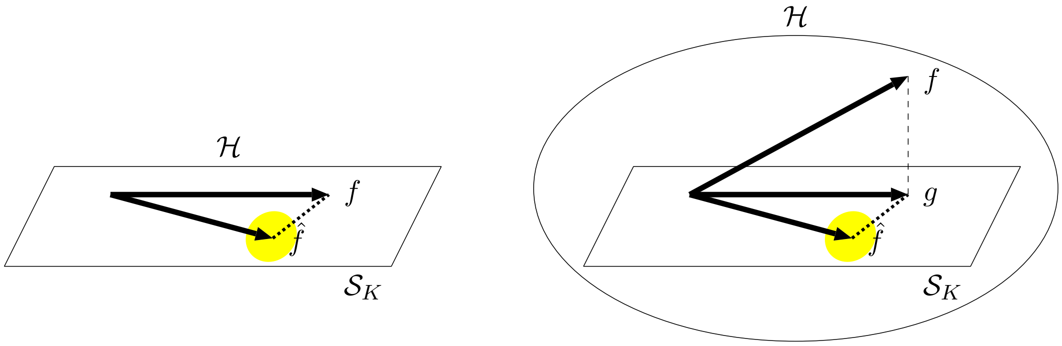
Estimating the generalization capability is one of the central issues in supervised learning. So far, a large number of generalization error estimation methods have been proposed (Mallows, 1964, 1973; Akaike, 1974; Takeuchi, 1976; Sugiura, 1978; Craven and Wahba, 1979; Bunke and Droge, 1984; Wahba, 1990; Murata et al., 1994; Vapnik, 1995; Konishi and Kitagawa, 1996; Cherkassky et al., 1999; Sugiyama and Ogawa, 2001), in particular also from the standpoints of Bayesian statistics (Schwarz, 1978; Akaike, 1980; MacKay, 1992a; Watanabe, 2001) and stochastic complexity (Rissanen, 1978, 1987, 1996; Yamanishi, 1998). An accurate estimator of the generalization error can be used for model selection. Furthermore, it eventually induces a new learning algorithm, e.g., the support vector machine (e.g., Vapnik, 1995; Schölkopf et al., 1998; Burges, 1998; Cristianini and Shawe-Taylor, 2000; Schölkopf et al., 2000; Müller et al., 2001; Schölkopf and Smola, 2002), that is inspired by the VC-theory (Vapnik, 1982, 1995, 1998).

In the development of generalization error estimation methods, asymptotic approximations in the number of training examples are often used (Akaike, 1974; Takeuchi, 1976; Murata et al., 1994; Konishi and Kitagawa, 1996, see also Shibata, 1981; Nishii, 1984 for asymptotic properties and consistency). However, in supervised learning, the small sample case is of high practical importance. One of the remarkable generalization error estimation methods that work with finite samples is the VC-bound (Vapnik, 1995), which gives a probabilistic upper bound of the generalization error. The bound is derived within a general framework so it is applicable to a wide range of models, although in practice the bound can be loose and hard to compute. Note that recent advances like the span bound on the leave-one-out error (Vapnik and Chapelle, 2000) are considerably tighter and work extremely well in practice.

Another generalization error estimation method that works effectively with finite samples is the subspace information criterion (SIC) (Sugiyama and Ogawa, 2001). Among several interesting theoretical properties, SIC is proved to be an unbiased estimator of the generalization error. SIC has been successfully applied to the selection of subspace models in linear regression. Theoretical and experimental comparison of SIC with various model selection methods was given by Sugiyama and Ogawa (2002b), and an analytic form of the optimal ridge parameter was derived based on SIC (Sugiyama and Ogawa, 2002a). SIC can also be applied to image restoration (Sugiyama et al., 2001; Sugiyama and Ogawa, 2002c). Furthermore, SIC was extended to sparse regressors (Tsuda et al., 2002).

SIC is applicable if an unbiased estimate of the learning target function is available. So far, a general method for obtaining such an unbiased estimate was given when the learning target function belongs to a specified reproducing kernel Hilbert space (RKHS) and the reproducing kernels centered on training sample points span the whole RKHS (see Figure 1a). Therefore, SIC could be applied only to finite dimensional RKHSs.

In this paper, we extend the range of application of SIC beyond this rather limiting scenario, and show that even if the reproducing kernels centered on training sample points



(a) $\mathcal{S}_K = \mathcal{H}$ (Original SIC)

(b) $\mathcal{S}_K \subset \mathcal{H}$ (This paper)

Figure 1: Original SIC and extension carried out in this paper. $\mathcal{H}$ is a reproducing kernel Hilbert space that includes the learning target function $f(\boldsymbol{x})$. $\mathcal{S}_K$ is the subspace spanned by reproducing kernels centered on training sample points, i.e., $\mathcal{S}_K$ is spanned by $\{K(\boldsymbol{x}, \boldsymbol{x}_i)\}_{i=1}^{\ell}$, where $\ell$ is the number of training examples. $g$ is the orthogonal projection of $f$ onto $\mathcal{S}_K$, i.e., the best approximation to $f$ in $\mathcal{S}_K$. $\hat{f}$ is a learning result function searched in the subspace $\mathcal{S}_K$. Let $J_G$ be the generalization error between $\hat{f}$ and $f$, and $J'_G$ be the generalization error between $\hat{f}$ and $g$. (a) Setting of the original SIC proposed by Sugiyama and Ogawa (2001). It was shown that when $\mathcal{S}_K = \mathcal{H}$, SIC is an unbiased estimator of $J_G$ with finitely many samples. $\mathcal{S}_K = \mathcal{H}$ implies that a RKHS $\mathcal{H}$ whose dimension is at most $\ell$ ($< \infty$) is considered. (b) Setting of this paper. We consider the case that $\mathcal{S}_K \subset \mathcal{H}$, which allows any RKHS $\mathcal{H}$ including infinite dimensional ones, and we show that SIC is an unbiased estimator of $J'_G$ with finite samples.

do not span the whole RKHS, SIC is an unbiased estimator of an essential part of the generalization error (see Figure 1b). This implies that it is possible to use any, possibly infinite dimensional, RKHSs that give rise to rich function classes commonly used, e.g., in Gaussian processes, support vector machines or boosting. The result is obtained under the assumption that the basis functions used for regression are the reproducing kernels centered on training sample points (i.e., kernel regression models). We further show that when the kernel matrix is invertible, SIC can be expressed by a much simpler form, making its computation stable.

The rest of this paper is organized as follows. In Section 2, the regression problem is formulated. In Section 3, the kernel regression model and learning methods are introduced. In Section 4, the derivation of the original subspace information criterion (SIC) is reviewed following Sugiyama and Ogawa (2001). In Section 5, we extend SIC such that infinite dimensional RKHSs are allowed. An efficient expression of SIC and discussions on the generalization measure are also given. Section 6 is devoted to computer simulations for experimentally investigating the usefulness of the proposed method. Finally, Section 7 gives concluding remarks and future prospects. The nomenclature used in this article is summarized in Table 1.

Table 1: Nomenclature

| | |
|---|---|
| $f(\boldsymbol{x})$ | Learning target function |
| $\mathcal{D}$ | Domain of input $\boldsymbol{x}$ |
| $n$ | Dimension of input space $\mathcal{D}$ |
| $\mathcal{H}$ | Reproducing kernel Hilbert space that includes $f(\boldsymbol{x})$ |
| $K(\cdot,\cdot)$ | Reproducing kernel of $\mathcal{H}$ |
| $\boldsymbol{x}_i$ | Training sample point |
| $y_i$ | Training sample value: $y_i = f(\boldsymbol{x}_i) + \epsilon_i$ |
| $\epsilon_i$ | Noise included in $y_i$ |
| $\sigma^2$ | Noise variance |
| $(\boldsymbol{x}_i, y_i)$ | Training example |
| $\ell$ | Number of training examples |
| $\mathcal{S}_K$ | Subspace of $\mathcal{H}$ spanned by $\{K(\boldsymbol{x},\boldsymbol{x}_i)\}_{i=1}^{\ell}$ |
| $g(\boldsymbol{x})$ | Orthogonal projection of $f(\boldsymbol{x})$ onto $\mathcal{S}_K$ |
| $\boldsymbol{\epsilon}$ | Noise vector: $\boldsymbol{\epsilon} = (\epsilon_1, \epsilon_2, \ldots, \epsilon_\ell)^\top$ |
| $\boldsymbol{z}$ | Noiseless sample value vector: $\boldsymbol{z} = (f(\boldsymbol{x}_1), f(\boldsymbol{x}_2), \ldots, f(\boldsymbol{x}_\ell))^\top$ |
| $\boldsymbol{y}$ | Noisy sample value vector: $\boldsymbol{y} = (y_1, y_2, \ldots, y_\ell)^\top$ |
| $\top$ | Transpose of a matrix or vector |
| $\mathrm{E}_{\boldsymbol{\epsilon}}$ | Expectation over noise |
| $\hat{f}(\boldsymbol{x})$ | Kernel regression model: $\hat{f}(\boldsymbol{x}) = \sum_{i=1}^{\ell} \alpha_i K(\boldsymbol{x},\boldsymbol{x}_i)$ |
| $\alpha_i$ | Parameter in kernel regression model $\hat{f}(\boldsymbol{x})$ |
| $\boldsymbol{\alpha}$ | Parameter vector: $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_\ell)^\top$ |
| $\boldsymbol{\alpha}^*$ | (Quasi) true parameter vector |
| $\hat{\boldsymbol{\alpha}}$ | Estimated parameter vector |
| $\boldsymbol{X}$ | Learning matrix that gives $\hat{\boldsymbol{\alpha}}$: $\hat{\boldsymbol{\alpha}} = \boldsymbol{X}\boldsymbol{y}$ |
| $\hat{\boldsymbol{\alpha}}_u$ | Unbiased estimate of true parameter vector: $\mathrm{E}_{\boldsymbol{\epsilon}}\hat{\boldsymbol{\alpha}}_u = \boldsymbol{\alpha}^*$ |
| $\boldsymbol{X}_u$ | Learning matrix that gives $\hat{\boldsymbol{\alpha}}_u$: $\hat{\boldsymbol{\alpha}}_u = \boldsymbol{X}_u\boldsymbol{y}$ |
| $\lambda$ | Ridge parameter |
| $\boldsymbol{K}$ | Kernel matrix: $\boldsymbol{K}_{i,j} = K(\boldsymbol{x}_i, \boldsymbol{x}_j)$ |
| $\boldsymbol{I}$ | Identity matrix |
| $\langle\cdot,\cdot\rangle_{\mathcal{H}}$ | Inner product in $\mathcal{H}$ |
| $\lVert\cdot\rVert_{\mathcal{H}}$ | Norm in $\mathcal{H}$ |
| $\langle\cdot,\cdot\rangle$ | Euclidean inner product in $\mathbb{R}^\ell$ |
| $\lVert\cdot\rVert$ | Euclidean norm in $\mathbb{R}^\ell$ |
| $\lVert\cdot\rVert_{\boldsymbol{K}}$ | Weighted norm by $\boldsymbol{K}$: $\lVert\cdot\rVert_{\boldsymbol{K}}^2 = \langle\boldsymbol{K}\cdot,\cdot\rangle$ |
| $\mathrm{tr}\,(\cdot)$ | Trace of a matrix |
| $J_G$ | Generalization error: $J_G = \mathrm{E}_{\boldsymbol{\epsilon}}\lVert\hat{f} - f\rVert_{\mathcal{H}}^2$ |
| $J'_G$ | Essential part of $J_G$: $J'_G = \mathrm{E}_{\boldsymbol{\epsilon}}\lVert\hat{f} - g\rVert_{\mathcal{H}}^2$ |
| $\dagger$ | Moore-Penrose generalized inverse |

## 2. Problem Formulation

Let us discuss the regression problem of approximating a target function from a set of $\ell$ *training examples.* Let $f(\boldsymbol{x})$ be a *learning target function* of $n$ variables defined on a subset $\mathcal{D}$ of the $n$-dimensional Euclidean space $\mathbb{R}^n$. The training examples consist of *sample points* $\boldsymbol{x}_i$ in $\mathcal{D}$ and corresponding *sample values* $y_i$ in $\mathbb{R}$:

$$\{(\boldsymbol{x}_i, y_i) \mid y_i = f(\boldsymbol{x}_i) + \epsilon_i\}_{i=1}^{\ell},$$

where $y_i$ is degraded by unknown additive noise $\epsilon_i$. We assume that $\epsilon_i$ is independently drawn from a distribution with mean zero and variance $\sigma^2$. The purpose of regression is to obtain the optimal approximation $\hat{f}(\boldsymbol{x})$ to the learning target function $f(\boldsymbol{x})$ that minimizes a *generalization error.*

In this paper, we assume that the unknown learning target function $f(\boldsymbol{x})$ belongs to a specified *reproducing kernel Hilbert space* (RKHS) $\mathcal{H}$ (see e.g., Aronszajn, 1950; Wahba, 1990; Vapnik, 1998; Cristianini and Shawe-Taylor, 2000). The *reproducing kernel* of a functional Hilbert space $\mathcal{H}$, denoted by $K(\boldsymbol{x}, \boldsymbol{x}')$, is a bivariate function defined on $\mathcal{D} \times \mathcal{D}$ that satisfies the following conditions:

- For any fixed $\boldsymbol{x}'$ in $\mathcal{D}$, $K(\boldsymbol{x}, \boldsymbol{x}')$ is a function of $\boldsymbol{x}$ in $\mathcal{H}$.
- For any function $f$ in $\mathcal{H}$ and for any $\boldsymbol{x}'$ in $\mathcal{D}$, it holds that

  $$\langle f(\cdot), K(\cdot, \boldsymbol{x}')\rangle_{\mathcal{H}} = f(\boldsymbol{x}'),$$

  where $\langle \cdot, \cdot \rangle_{\mathcal{H}}$ stands for the inner product in $\mathcal{H}$.

In previous work (Sugiyama and Ogawa, 2001), it was assumed that $\{K(\boldsymbol{x}, \boldsymbol{x}_i)\}_{i=1}^{\ell}$ span the whole RKHS $\mathcal{H}$ (see Figure 1a). This holds only if $\dim \mathcal{H} \leq \ell$ $(< \infty)$. In contrast, now we remove this restriction on the dimension of the RKHS $\mathcal{H}$. Thus, the dimension could be infinity and we can treat a rich class of function spaces such as the Gaussian RKHS (see Figure 1b).

We measure the generalization error of $\hat{f}(\boldsymbol{x})$ by

$$J_G = \mathrm{E}_{\boldsymbol{\epsilon}} \|\hat{f} - f\|_{\mathcal{H}}^2, \tag{1}$$

where $\mathrm{E}_{\boldsymbol{\epsilon}}$ denotes the expectation over the noise and $\|\cdot\|_{\mathcal{H}}$ is the norm in the RKHS $\mathcal{H}$. Using the RKHS norm for error measure is rather common in the field of function approximation (e.g., Daubechies, 1992; Donoho and Johnstone, 1994; Donoho, 1995). A brief discussion on this generalization measure is given in Section 5.3.

As can be seen from Eq.(1), the generalization error $J_G$ includes the unknown learning target function $f(\boldsymbol{x})$ so it can not be directly calculated. The aim of this paper is to give an estimator of Eq.(1) that can be calculated without using $f(\boldsymbol{x})$.

## 3. Regression Model and Learning Methods

In this section, we describe our regression model and examples of the learning methods that can be dealt with.

### 3.1 Kernel Regression Model with Linear Learning Methods

We will employ the following kernel regression model $\hat{f}(\boldsymbol{x})$:

$$\hat{f}(\boldsymbol{x}) = \sum_{i=1}^{\ell} \alpha_i K(\boldsymbol{x}, \boldsymbol{x}_i), \tag{2}$$

where $\{\alpha_i\}_{i=1}^{\ell}$ are parameters to be estimated from training examples, and $K(\cdot, \cdot)$ is the reproducing kernel of $\mathcal{H}$. The reason why the above kernel regression model is chosen will be explained in Section 5.1.

Let us denote the estimated parameters by $\{\hat{\alpha}_i\}_{i=1}^{\ell}$. In this article, we focus on the case that the estimated parameters are given by linear combinations of sample values $\{y_i\}_{i=1}^{\ell}$. More specifically, letting

$$\begin{aligned} \boldsymbol{y} &= (y_1, y_2, \ldots, y_\ell)^\top, \\ \hat{\boldsymbol{\alpha}} &= (\hat{\alpha}_1, \hat{\alpha}_2, \ldots, \hat{\alpha}_\ell)^\top, \end{aligned} \tag{3}$$

where $^\top$ denotes the transpose of a vector (or a matrix), we consider the case that the estimated parameter vector $\hat{\boldsymbol{\alpha}}$ is given by

$$\hat{\boldsymbol{\alpha}} = \boldsymbol{X}\boldsymbol{y},$$

where $\boldsymbol{X}$ is an $\ell$-dimensional matrix that does not depend on the noise $\{\epsilon_i\}_{i=1}^{\ell}$. The learning matrix $\boldsymbol{X}$, which we call the *learning matrix*, can be any matrix, but it is usually determined on the basis of a prespecified learning criterion. In the rest of this section, examples of learning matrices are described.

### 3.2 Kernel Ridge Regression

Kernel ridge regression determines the parameter vector $\boldsymbol{\alpha}$ so that the following regularized training error is minimized (Saunders et al., 1998; Cristianini and Shawe-Taylor, 2000):

$$\hat{\boldsymbol{\alpha}} = \underset{\boldsymbol{\alpha}}{\operatorname{argmin}} \left[ \sum_{i=1}^{\ell} \left( \sum_{j=1}^{\ell} \alpha_j K(\boldsymbol{x}_i, \boldsymbol{x}_j) - y_i \right)^2 + \lambda\, \boldsymbol{\alpha}^\top \boldsymbol{K} \boldsymbol{\alpha} \right], \tag{4}$$

where $\lambda$ is a positive scalar called the *ridge parameter*, and $\boldsymbol{K}$ is the so-called kernel matrix, i.e., the $(i, j)$-th element of $\boldsymbol{K}$ is given by

$$\boldsymbol{K}_{i,j} = K(\boldsymbol{x}_i, \boldsymbol{x}_j). \tag{5}$$

A minimizer of Eq.(4) is given by the following learning matrix:

$$\boldsymbol{X} = (\boldsymbol{K} + \lambda \boldsymbol{I})^{-1}, \tag{6}$$

where $\boldsymbol{I}$ denotes the identity matrix. Note that Bayesian learning with a particular Gaussian process prior agrees with kernel ridge regression (see e.g., Williams and Rasmussen, 1996; Williams, 1998; Cristianini and Shawe-Taylor, 2000).

Instead of $\boldsymbol{\alpha}^\top \boldsymbol{K}\boldsymbol{\alpha}$, any regularizers of the form $\boldsymbol{\alpha}^\top \boldsymbol{T}\boldsymbol{\alpha}$ can be used in a similar fashion, where $\boldsymbol{T}$ is an $\ell$-dimensional symmetric positive semi-definite matrix. In this case, the learning matrix is given by

$$\boldsymbol{X} = (\boldsymbol{K}^2 + \lambda \boldsymbol{T})^\dagger \boldsymbol{K}, \tag{7}$$

where $^\dagger$ denotes the *Moore-Penrose generalized inverse* (see e.g., Albert, 1972; Hunter, 2000). In the following, we may refer to the above generalized form as kernel ridge regression.

It should be noted that, according to the representer theorem (Kimeldorf and Wahba, 1970), a minimizer of the regularized training error in the RKHS $\mathcal{H}$ can be expressed in the form of Eq.(2). Therefore, using kernel regression models does not impose any restriction on the choice of basis functions in the regression model.

### 3.3 Kernel Subset Regression

Let $S$ be a subset of indices $\{1, 2, \ldots, \ell\}$. Kernel subset regression determines the parameter vector $\boldsymbol{\alpha}$ so that the training error is minimized in a subset of basis functions specified by $S$, i.e.,

$$\begin{aligned} \hat{\boldsymbol{\alpha}} &= \operatorname*{argmin}_{\boldsymbol{\alpha}} \left[ \sum_{i=1}^{\ell} \left( \sum_{j \in S} \alpha_j K(\boldsymbol{x}_i, \boldsymbol{x}_j) - y_i \right)^2 \right] \\ &\qquad\qquad\qquad \text{subject to } \alpha_j = 0 \text{ for all } j \notin S. \end{aligned} \tag{8}$$

A minimizer of Eq.(8) is given by the following learning matrix:

$$\boldsymbol{X} = \boldsymbol{K}_S^\dagger,$$

where $\boldsymbol{K}_S$ is equal to $\boldsymbol{K}$ but the $j$-th column is zero for all $j \notin S$.

## 4. Subspace Information Criterion: Review

Subspace information criterion (SIC) proposed by Sugiyama and Ogawa (2001) is an unbiased estimator of the generalization error $J_G$ defined by Eq.(1). In this section, we review the original SIC that is applicable when $\{K(\boldsymbol{x}, \boldsymbol{x}_i)\}_{i=1}^{\ell}$ span the whole RKHS $\mathcal{H}$.

When the functions $\{K(\boldsymbol{x}, \boldsymbol{x}_i)\}_{i=1}^{\ell}$ span the whole space $\mathcal{H}$, the learning target function $f(\boldsymbol{x})$ is expressed as

$$f(\boldsymbol{x}) = \sum_{i=1}^{\ell} \alpha_i^* K(\boldsymbol{x}, \boldsymbol{x}_i),$$

where the true parameters $\{\alpha_i^*\}_{i=1}^{\ell}$ are unknown.[1] Letting

$$\boldsymbol{\alpha}^* = (\alpha_1^*, \alpha_2^*, \ldots, \alpha_\ell^*)^\top,$$

[1] When $\{K(\boldsymbol{x}, \boldsymbol{x}_i)\}_{i=1}^{\ell}$ are linearly dependent, $\{\alpha_i^*\}_{i=1}^{\ell}$ are not determined uniquely. In this case, we adopt the minimum norm one given by $\boldsymbol{\alpha}^* = \boldsymbol{K}^\dagger (f(\boldsymbol{x}_1), f(\boldsymbol{x}_2), \ldots, f(\boldsymbol{x}_\ell))^\top$ (see Corollary 2 and its proof given in Appendix B for detail).

we can express the generalization error $J_G$ as

$$\begin{aligned} J_G &= \mathrm{E}_{\boldsymbol{\epsilon}}\|\hat{f} - f\|_{\mathcal{H}}^2 \\ &= \mathrm{E}_{\boldsymbol{\epsilon}}\|\sum_{i=1}^{\ell}(\hat{\alpha}_i - \alpha_i^*)K(\cdot, \boldsymbol{x}_i)\|_{\mathcal{H}}^2 \\ &= \mathrm{E}_{\boldsymbol{\epsilon}}\sum_{i,j=1}^{\ell}(\hat{\alpha}_i - \alpha_i^*)(\hat{\alpha}_j - \alpha_j^*)\langle K(\cdot, \boldsymbol{x}_j), K(\cdot, \boldsymbol{x}_i)\rangle_{\mathcal{H}} \\ &= \mathrm{E}_{\boldsymbol{\epsilon}}\sum_{i,j=1}^{\ell}(\hat{\alpha}_i - \alpha_i^*)(\hat{\alpha}_j - \alpha_j^*)K(\boldsymbol{x}_i, \boldsymbol{x}_j) \\ &= \mathrm{E}_{\boldsymbol{\epsilon}}\langle \boldsymbol{K}(\hat{\boldsymbol{\alpha}} - \boldsymbol{\alpha}^*), \hat{\boldsymbol{\alpha}} - \boldsymbol{\alpha}^*\rangle, \end{aligned}$$

where the inner product $\langle\cdot,\cdot\rangle$ in the last equation is the ordinary Euclidean inner product in $\mathbb{R}^{\ell}$, i.e., $\langle \boldsymbol{\alpha}_a, \boldsymbol{\alpha}_b\rangle = \boldsymbol{\alpha}_b^{\top}\boldsymbol{\alpha}_a$. For convenience, let us define the weighted norm in $\mathbb{R}^{\ell}$:

$$\|\boldsymbol{\alpha}\|_{\boldsymbol{K}}^2 = \langle \boldsymbol{K}\boldsymbol{\alpha}, \boldsymbol{\alpha}\rangle.$$

Then $J_G$ is expressed as

$$J_G = \mathrm{E}_{\boldsymbol{\epsilon}}\|\hat{\boldsymbol{\alpha}} - \boldsymbol{\alpha}^*\|_{\boldsymbol{K}}^2.$$

It is known that $J_G$ can be decomposed into the *bias* and *variance* (see e.g., Geman et al., 1992; Heskes, 1998):

$$J_G = \|\mathrm{E}_{\boldsymbol{\epsilon}}\hat{\boldsymbol{\alpha}} - \boldsymbol{\alpha}^*\|_{\boldsymbol{K}}^2 + \mathrm{E}_{\boldsymbol{\epsilon}}\|\hat{\boldsymbol{\alpha}} - \mathrm{E}_{\boldsymbol{\epsilon}}\hat{\boldsymbol{\alpha}}\|_{\boldsymbol{K}}^2. \tag{9}$$

Let us define the noiseless sample value vector $\boldsymbol{z}$ and the noise vector $\boldsymbol{\epsilon}$ by

$$\begin{aligned} \boldsymbol{z} &= (f(\boldsymbol{x}_1), f(\boldsymbol{x}_2), \ldots, f(\boldsymbol{x}_{\ell}))^{\top}, \\ \boldsymbol{\epsilon} &= (\epsilon_1, \epsilon_2, \ldots, \epsilon_{\ell})^{\top}. \end{aligned} \tag{10}$$

Then the (noisy) sample value vector $\boldsymbol{y}$ defined by Eq.(3) is expressed as

$$\boldsymbol{y} = \boldsymbol{z} + \boldsymbol{\epsilon}.$$

Recalling that the mean noise $\mathrm{E}_{\boldsymbol{\epsilon}}\boldsymbol{\epsilon}$ is zero and the noise covariance matrix is given by $\mathrm{E}_{\boldsymbol{\epsilon}}\boldsymbol{\epsilon}\boldsymbol{\epsilon}^{\top} = \sigma^2\boldsymbol{I}$, we can express the variance term $\mathrm{E}_{\boldsymbol{\epsilon}}\|\hat{\boldsymbol{\alpha}} - \mathrm{E}_{\boldsymbol{\epsilon}}\hat{\boldsymbol{\alpha}}\|_{\boldsymbol{K}}^2$ in Eq.(9) as

$$\begin{aligned} \mathrm{E}_{\boldsymbol{\epsilon}}\|\hat{\boldsymbol{\alpha}} - \mathrm{E}_{\boldsymbol{\epsilon}}\hat{\boldsymbol{\alpha}}\|_{\boldsymbol{K}}^2 &= \mathrm{E}_{\boldsymbol{\epsilon}}\|\boldsymbol{X}\boldsymbol{y} - \mathrm{E}_{\boldsymbol{\epsilon}}\boldsymbol{X}\boldsymbol{y}\|_{\boldsymbol{K}}^2 \\ &= \mathrm{E}_{\boldsymbol{\epsilon}}\|\boldsymbol{X}(\boldsymbol{z} + \boldsymbol{\epsilon}) - \boldsymbol{X}\boldsymbol{z}\|_{\boldsymbol{K}}^2 \\ &= \mathrm{E}_{\boldsymbol{\epsilon}}\|\boldsymbol{X}\boldsymbol{\epsilon}\|_{\boldsymbol{K}}^2 \\ &= \sigma^2\mathrm{tr}\left(\boldsymbol{K}\boldsymbol{X}\boldsymbol{X}^{\top}\right), \end{aligned} \tag{11}$$

where $\mathrm{tr}\,(\cdot)$ denotes the trace of a matrix, i.e., the sum of diagonal elements. Eq.(11) implies that the variance term $\mathrm{E}_{\boldsymbol{\epsilon}}\|\hat{\boldsymbol{\alpha}} - \mathrm{E}_{\boldsymbol{\epsilon}}\hat{\boldsymbol{\alpha}}\|_{\boldsymbol{K}}^2$ in Eq.(9) can be calculated if the noise variance $\sigma^2$ is available. When $\sigma^2$ is unknown, one of the practical methods for estimating $\sigma^2$ is given as follows (see e.g., Wahba, 1990):

$$\hat{\sigma}^2 = \frac{\sum_{i=1}^{\ell}\left(\hat{f}(\boldsymbol{x}_i) - y_i\right)^2}{\ell - \mathrm{tr}\,(\boldsymbol{K}\boldsymbol{X})} = \frac{\|\boldsymbol{K}\boldsymbol{X}\boldsymbol{y} - \boldsymbol{y}\|^2}{\ell - \mathrm{tr}\,(\boldsymbol{K}\boldsymbol{X})}, \tag{12}$$

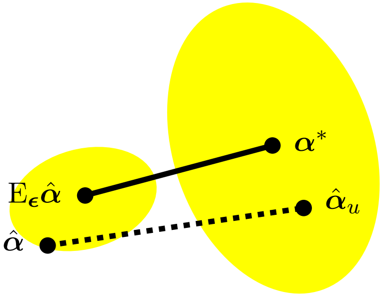

$\boldsymbol{\alpha}^*$
$\mathrm{E}_{\boldsymbol{\epsilon}}\hat{\boldsymbol{\alpha}}$
$\hat{\boldsymbol{\alpha}}_u$
$\hat{\boldsymbol{\alpha}}$

Figure 2: Basic idea of SIC. The bias term $\|\mathrm{E}_{\boldsymbol{\epsilon}}\hat{\boldsymbol{\alpha}} - \boldsymbol{\alpha}^*\|_{\boldsymbol{K}}^2$ (depicted by the solid line) can be roughly estimated by $\|\hat{\boldsymbol{\alpha}} - \hat{\boldsymbol{\alpha}}_u\|_{\boldsymbol{K}}^2$ (depicted by the dotted line).

where $\|\cdot\|$ is the ordinary Euclidean norm in $\mathbb{R}^\ell$.

On the other hand, the bias term $\|\mathrm{E}_{\boldsymbol{\epsilon}}\hat{\boldsymbol{\alpha}}-\boldsymbol{\alpha}^*\|_{\boldsymbol{K}}^2$ in Eq.(9) is totally inaccessible since both $\mathrm{E}_{\boldsymbol{\epsilon}}\hat{\boldsymbol{\alpha}}$ and $\boldsymbol{\alpha}^*$ are unavailable. The key idea of SIC is to assume that a learning matrix $\boldsymbol{X}_u$ that gives an unbiased estimate $\hat{\boldsymbol{\alpha}}_u$ of the unknown true parameter vector $\boldsymbol{\alpha}^*$ is available:

$$\mathrm{E}_{\boldsymbol{\epsilon}}\hat{\boldsymbol{\alpha}}_u = \boldsymbol{\alpha}^*, \tag{13}$$

where

$$\hat{\boldsymbol{\alpha}}_u = \boldsymbol{X}_u\boldsymbol{y}.$$

Using the unbiased estimate $\hat{\boldsymbol{\alpha}}_u$, we can roughly estimate the bias term $\|\mathrm{E}_{\boldsymbol{\epsilon}}\hat{\boldsymbol{\alpha}} - \boldsymbol{\alpha}^*\|_{\boldsymbol{K}}^2$ in Eq.(9) by $\|\hat{\boldsymbol{\alpha}} - \hat{\boldsymbol{\alpha}}_u\|_{\boldsymbol{K}}^2$ (see Figure 2). More specifically, we have

$$\begin{aligned}
\|\mathrm{E}_{\boldsymbol{\epsilon}}\hat{\boldsymbol{\alpha}} - \boldsymbol{\alpha}^*\|_{\boldsymbol{K}}^2 &= \|\hat{\boldsymbol{\alpha}} - \hat{\boldsymbol{\alpha}}_u\|_{\boldsymbol{K}}^2 - \|\hat{\boldsymbol{\alpha}} - \hat{\boldsymbol{\alpha}}_u\|_{\boldsymbol{K}}^2 + \|\mathrm{E}_{\boldsymbol{\epsilon}}\hat{\boldsymbol{\alpha}} - \boldsymbol{\alpha}^*\|_{\boldsymbol{K}}^2 \\
&= \|\hat{\boldsymbol{\alpha}} - \hat{\boldsymbol{\alpha}}_u\|_{\boldsymbol{K}}^2 - \|\mathrm{E}_{\boldsymbol{\epsilon}}(\hat{\boldsymbol{\alpha}} - \hat{\boldsymbol{\alpha}}_u) - \mathrm{E}_{\boldsymbol{\epsilon}}(\hat{\boldsymbol{\alpha}} - \hat{\boldsymbol{\alpha}}_u) + \hat{\boldsymbol{\alpha}} - \hat{\boldsymbol{\alpha}}_u\|_{\boldsymbol{K}}^2 \\
&\quad + \|\mathrm{E}_{\boldsymbol{\epsilon}}(\hat{\boldsymbol{\alpha}} - \hat{\boldsymbol{\alpha}}_u)\|_{\boldsymbol{K}}^2 \\
&= \|\hat{\boldsymbol{\alpha}} - \hat{\boldsymbol{\alpha}}_u\|_{\boldsymbol{K}}^2 - \|\mathrm{E}_{\boldsymbol{\epsilon}}(\hat{\boldsymbol{\alpha}} - \hat{\boldsymbol{\alpha}}_u)\|_{\boldsymbol{K}}^2 \\
&\quad -2\langle \boldsymbol{K}\mathrm{E}_{\boldsymbol{\epsilon}}(\hat{\boldsymbol{\alpha}} - \hat{\boldsymbol{\alpha}}_u), -\mathrm{E}_{\boldsymbol{\epsilon}}(\hat{\boldsymbol{\alpha}} - \hat{\boldsymbol{\alpha}}_u) + \hat{\boldsymbol{\alpha}} - \hat{\boldsymbol{\alpha}}_u\rangle \\
&\quad -\| - \mathrm{E}_{\boldsymbol{\epsilon}}(\hat{\boldsymbol{\alpha}} - \hat{\boldsymbol{\alpha}}_u) + \hat{\boldsymbol{\alpha}} - \hat{\boldsymbol{\alpha}}_u\|_{\boldsymbol{K}}^2 + \|\mathrm{E}_{\boldsymbol{\epsilon}}(\hat{\boldsymbol{\alpha}} - \hat{\boldsymbol{\alpha}}_u)\|_{\boldsymbol{K}}^2 \\
&= \|\hat{\boldsymbol{\alpha}} - \hat{\boldsymbol{\alpha}}_u\|_{\boldsymbol{K}}^2 + 2\langle \boldsymbol{K}\mathrm{E}_{\boldsymbol{\epsilon}}(\hat{\boldsymbol{\alpha}} - \hat{\boldsymbol{\alpha}}_u), \mathrm{E}_{\boldsymbol{\epsilon}}(\hat{\boldsymbol{\alpha}} - \hat{\boldsymbol{\alpha}}_u) - (\hat{\boldsymbol{\alpha}} - \hat{\boldsymbol{\alpha}}_u)\rangle \\
&\quad -\|\mathrm{E}_{\boldsymbol{\epsilon}}(\hat{\boldsymbol{\alpha}} - \hat{\boldsymbol{\alpha}}_u) - (\hat{\boldsymbol{\alpha}} - \hat{\boldsymbol{\alpha}}_u)\|_{\boldsymbol{K}}^2.
\end{aligned} \tag{14}$$

However, the second and third terms in the last equation of Eq.(14) are still inaccessible since $\mathrm{E}_{\boldsymbol{\epsilon}}(\hat{\boldsymbol{\alpha}} - \hat{\boldsymbol{\alpha}}_u)$ is unknown, so we will average them out over the noise. Then the second term vanishes:

$$\mathrm{E}_{\boldsymbol{\epsilon}}\langle \boldsymbol{K}\mathrm{E}_{\boldsymbol{\epsilon}}(\hat{\boldsymbol{\alpha}} - \hat{\boldsymbol{\alpha}}_u), \mathrm{E}_{\boldsymbol{\epsilon}}(\hat{\boldsymbol{\alpha}} - \hat{\boldsymbol{\alpha}}_u) - (\hat{\boldsymbol{\alpha}} - \hat{\boldsymbol{\alpha}}_u)\rangle = 0,$$

and the third term is reduced to

$$\begin{aligned}
\mathrm{E}_{\boldsymbol{\epsilon}}\left(\|\mathrm{E}_{\boldsymbol{\epsilon}}(\hat{\boldsymbol{\alpha}} - \hat{\boldsymbol{\alpha}}_u) - (\hat{\boldsymbol{\alpha}} - \hat{\boldsymbol{\alpha}}_u)\|_{\boldsymbol{K}}^2\right) &= \mathrm{E}_{\boldsymbol{\epsilon}}\left(\|\mathrm{E}_{\boldsymbol{\epsilon}}(\boldsymbol{X} - \boldsymbol{X}_u)\boldsymbol{y} - (\boldsymbol{X} - \boldsymbol{X}_u)\boldsymbol{y}\|_{\boldsymbol{K}}^2\right) \\
&= \mathrm{E}_{\boldsymbol{\epsilon}}\left(\|(\boldsymbol{X} - \boldsymbol{X}_u)\boldsymbol{z} - (\boldsymbol{X} - \boldsymbol{X}_u)(\boldsymbol{z} + \boldsymbol{\epsilon})\|_{\boldsymbol{K}}^2\right) \\
&= \mathrm{E}_{\boldsymbol{\epsilon}}\left(\|(\boldsymbol{X} - \boldsymbol{X}_u)\boldsymbol{\epsilon}\|_{\boldsymbol{K}}^2\right) \\
&= \sigma^2 \mathrm{tr}\left(\boldsymbol{K}(\boldsymbol{X} - \boldsymbol{X}_u)(\boldsymbol{X} - \boldsymbol{X}_u)^\top\right).
\end{aligned}$$

Consequently we have the subspace information criterion (SIC) (Sugiyama and Ogawa, 2001):

$$\mathrm{SIC} = \|\hat{\boldsymbol{\alpha}} - \hat{\boldsymbol{\alpha}}_u\|_{\boldsymbol{K}}^2 - \sigma^2 \mathrm{tr}\left(\boldsymbol{K}(\boldsymbol{X} - \boldsymbol{X}_u)(\boldsymbol{X} - \boldsymbol{X}_u)^\top\right) + \sigma^2 \mathrm{tr}\left(\boldsymbol{K}\boldsymbol{X}\boldsymbol{X}^\top\right). \tag{15}$$

The name *subspace information criterion* (SIC) came from the fact that it was first introduced for selecting subspace models. The first two terms in SIC are estimates of the bias term and the last term corresponds to the variance term. It was shown by Sugiyama and Ogawa (2001) that Eq.(15) is an unbiased estimator of the generalization error $J_G$, i.e., $\mathrm{E}_{\boldsymbol{\epsilon}}\mathrm{SIC} = J_G$.

SIC requires a learning matrix $\boldsymbol{X}_u$ that gives an unbiased estimate $\hat{\boldsymbol{\alpha}}_u$ of the true parameter $\boldsymbol{\alpha}^*$. When $\{K(\boldsymbol{x}, \boldsymbol{x}_i)\}_{i=1}^{\ell}$ span the whole RKHS $\mathcal{H}$, such $\boldsymbol{X}_u$ exists and is given as follows (Sugiyama and Ogawa, 2001):

$$\boldsymbol{X}_u = \boldsymbol{K}^\dagger. \tag{16}$$

However, obtaining $\boldsymbol{X}_u$ when $\{K(\boldsymbol{x}, \boldsymbol{x}_i)\}_{i=1}^{\ell}$ do not span the whole RKHS $\mathcal{H}$ is still an open problem. In the following section, we will therefore aim to solve this problem.

## 5. Subspace Information Criterion for Infinite Dimensional RKHSs

In this section, we extend the range of application of SIC to the case when $\{K(\boldsymbol{x}, \boldsymbol{x}_i)\}_{i=1}^{\ell}$ do not span the whole RKHS $\mathcal{H}$. This extension allows us to use even infinite dimensional RKHSs.

### 5.1 Extending SIC to the Case When $\{K(\boldsymbol{x}, \boldsymbol{x}_i)\}_{i=1}^{\ell}$ Do Not Span $\mathcal{H}$

In Section 3, we decided to use the kernel regression model without any further discussion. Now, we first explain the reason why the kernel regression model is chosen. For this purpose, let us consider an ordinary linear regression model:

$$\hat{f}(\boldsymbol{x}) = \sum_{i=1}^{p} \alpha_i \varphi_i(\boldsymbol{x}), \tag{17}$$

where $\{\varphi_i(\boldsymbol{x})\}_{i=1}^{p}$ are functions in the RKHS $\mathcal{H}$ and $p$ denotes the number of basis functions in the linear regression model.

Let $\mathcal{S}$ be a subspace spanned by $\{\varphi_i(\boldsymbol{x})\}_{i=1}^{p}$. Since the learning target function $f(\boldsymbol{x})$ does not generally lie in the subspace $\mathcal{S}$, $f(\boldsymbol{x})$ can be decomposed into two elements:

$$f(\boldsymbol{x}) = g(\boldsymbol{x}) + h(\boldsymbol{x}),$$

where $g(\boldsymbol{x})$ is the orthogonal projection of $f(\boldsymbol{x})$ onto the subspace $\mathcal{S}$ and $h(\boldsymbol{x})$ is the orthogonal projection of $f(\boldsymbol{x})$ onto the orthogonal complement of $\mathcal{S}$. Then the generalization error can be expressed as

$$\begin{aligned} \mathrm{E}_{\boldsymbol{\epsilon}}\|\hat{f} - f\|_{\mathcal{H}}^2 &= \mathrm{E}_{\boldsymbol{\epsilon}}\|\hat{f} - g\|_{\mathcal{H}}^2 - 2\mathrm{E}_{\boldsymbol{\epsilon}}\langle \hat{f} - g, h\rangle_{\mathcal{H}} + \|h\|_{\mathcal{H}}^2 \\ &= \mathrm{E}_{\boldsymbol{\epsilon}}\|\hat{f} - g\|_{\mathcal{H}}^2 + \|h\|_{\mathcal{H}}^2. \end{aligned}$$

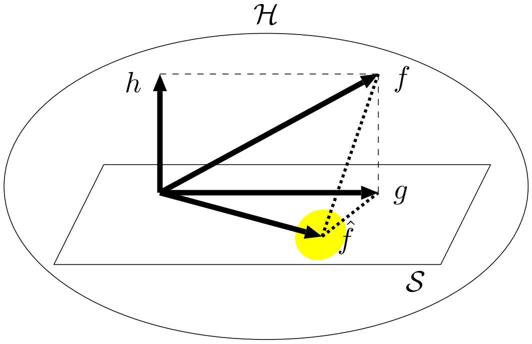


Figure 3: $\mathcal{H}$ is an RKHS that includes the learning target function $f$. $f$ can be uniquely decomposed into $g$ and $h$, where $g$ is included in the subspace $\mathcal{S}$ and $h$ is orthogonal to $\mathcal{S}$. Since the learning result function $\hat{f}$ is searched in the subspace $\mathcal{S}$, the component $h$ is essentially irrelevant. Therefore, we do not have to investigate the generalization error $\mathrm{E}_{\boldsymbol{\epsilon}}\|\hat{f}-f\|_{\mathcal{H}}^2$ itself, but it is sufficient to investigate only $\mathrm{E}_{\boldsymbol{\epsilon}}\|\hat{f}-g\|_{\mathcal{H}}^2$.

Since the second term $\|h\|_{\mathcal{H}}^2$ is irrelevant to $\hat{f}$, we ignore it and focus only on the first term $\mathrm{E}_{\boldsymbol{\epsilon}}\|\hat{f}-g\|_{\mathcal{H}}^2$ (see Figure 3). Let us denote the first term by $J'_G$:

$$J'_G = \mathrm{E}_{\boldsymbol{\epsilon}}\|\hat{f}-g\|_{\mathcal{H}}^2. \tag{18}$$

If we regard $g(\boldsymbol{x})$ as the learning target function, then the setting seems exactly the same as that of Section 4. Therefore, we may apply SIC and obtain an unbiased estimator of $J'_G$. However, the problem is that we need a learning matrix $\boldsymbol{X}_u$ that gives an unbiased estimate $\hat{\boldsymbol{\alpha}}_u$ of the true parameter vector $\boldsymbol{\alpha}^*$. Here, 'true parameter vector' indicates the parameter vector in $g(\boldsymbol{x})$, not in $f(\boldsymbol{x})$ because $g(\boldsymbol{x})$ is now regarded as the learning target function:[2]

$$g(\boldsymbol{x}) = \sum_{i=1}^{p} \alpha_i^* \varphi_i(\boldsymbol{x}). \tag{19}$$

If the training examples $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^{\ell}$ are sampled from $g(\boldsymbol{x})$, then Eq.(16) can be straightforwardly used for obtaining an unbiased estimate of $\boldsymbol{\alpha}^*$. However, in reality, the training examples are sampled from $f(\boldsymbol{x})$. This is the difference from the previous section.

For resolving this problem, we give the following theorem.

**Theorem 1** *Let $g(\boldsymbol{x})$ be the orthogonal projection of $f(\boldsymbol{x})$ onto the subspace $\mathcal{S}$, expressed by Eq.(19). A learning matrix $\boldsymbol{X}_u$ that gives an unbiased estimate of the true parameter vector $\boldsymbol{\alpha}^*$ exists if and only if the subspace $\mathcal{S}$ satisfies*

$$\mathcal{S} \subset \mathcal{S}_K, \tag{20}$$

*where $\mathcal{S}_K$ denotes a subspace spanned by $\{K(\boldsymbol{x}, \boldsymbol{x}_i)\}_{i=1}^{\ell}$.*

[2] When $\{\varphi_i(\boldsymbol{x})\}_{i=1}^{p}$ are linearly dependent, we again adopt the minimum norm parameter vector as the true parameter vector $\boldsymbol{\alpha}^*$.

A proof of Theorem 1 is provided in Appendix A. Theorem 1 means that even when $\{K(\boldsymbol{x}, \boldsymbol{x}_i)\}_{i=1}^{\ell}$ do not span the whole RKHS $\mathcal{H}$, SIC can be exactly applied if and only if the subspace $\mathcal{S}$ is included in $\mathcal{S}_K$. This is the reason why we adopted the kernel regression model given by Eq.(2), which yields $\mathcal{S} = \mathcal{S}_K$. For the kernel regression model (2), we have the following corollary.

**Corollary 2** *For an arbitrarily chosen RKHS $\mathcal{H}$ and the kernel regression model given by Eq.(2), a learning matrix $\boldsymbol{X}_u$ that gives an unbiased estimate $\hat{\boldsymbol{\alpha}}_u$ of the true parameter vector $\boldsymbol{\alpha}^*$ (corresponding to $g(\boldsymbol{x})$) is given by*

$$\boldsymbol{X}_u = \boldsymbol{K}^{\dagger}. \tag{21}$$

A proof of Corollary 2 is provided in Appendix B. Eq.(21) is exactly equivalent to Eq.(16). Therefore, the above corollary shows that, as long as we are concerned with kernel regression, SIC is applicable irrespective of the choice of the RKHS $\mathcal{H}$. If $\{K(\boldsymbol{x}, \boldsymbol{x}_i)\}_{i=1}^{\ell}$ span the whole RKHS $\mathcal{H}$, SIC is an unbiased estimator of the generalization error $J_G$. Otherwise SIC is an unbiased estimator of $J'_G$, which is an essential part of the generalization error $J_G$ (see Figure 1 again):

$$\mathrm{E}_{\boldsymbol{\epsilon}}\mathrm{SIC} = J'_G.$$

On the other hand, for an ordinary linear regression model given by Eq.(17) such that the condition (20) does not hold, an unbiased estimate of the true parameter vector $\boldsymbol{\alpha}^*$ can not be generally obtained. In this case, a consistent estimate of $\boldsymbol{\alpha}^*$ can be obtained for a particular generalization measure, and accordingly SIC is a consistent estimator of the generalization error (Sugiyama and Ogawa, 2002a).

### 5.2 An Efficient Expression of SIC When Kernel Matrix $\boldsymbol{K}$ Is Invertible

Now we show that when the kernel matrix $\boldsymbol{K}$ defined by Eq.(5) is invertible, SIC can be computed much simpler.

Substituting Eq.(21) into Eq.(15), SIC is expressed as

$$\mathrm{SIC} = \|\hat{\boldsymbol{\alpha}}\|_{\boldsymbol{K}}^2 - 2\langle \boldsymbol{K}\hat{\boldsymbol{\alpha}}, \boldsymbol{K}^{\dagger}\boldsymbol{y}\rangle + \|\boldsymbol{K}^{\dagger}\boldsymbol{y}\|_{\boldsymbol{K}}^2 + 2\sigma^2 \mathrm{tr}\left(\boldsymbol{K}\boldsymbol{X}\boldsymbol{K}^{\dagger}\right) - \sigma^2 \mathrm{tr}\left(\boldsymbol{K}^{\dagger}\right).$$

Since the third and fifth terms are irrelevant to $\boldsymbol{X}$, they can be safely ignored. When $\boldsymbol{K}^{-1}$ exists, a practical expression of SIC (denoted by $\mathrm{SIC}_{\mathrm{e}}$, SIC essential) for kernel regression is given by

$$\begin{aligned} \mathrm{SIC}_{\mathrm{e}} &= \|\hat{\boldsymbol{\alpha}}\|_{\boldsymbol{K}}^2 - 2\langle \boldsymbol{K}\hat{\boldsymbol{\alpha}}, \boldsymbol{K}^{-1}\boldsymbol{y}\rangle + 2\sigma^2 \mathrm{tr}\left(\boldsymbol{K}\boldsymbol{X}\boldsymbol{K}^{-1}\right) \\ &= \boldsymbol{y}^{\top}\boldsymbol{X}^{\top}\boldsymbol{K}\boldsymbol{X}\boldsymbol{y} - 2\boldsymbol{y}^{\top}\boldsymbol{X}\boldsymbol{y} + 2\sigma^2 \mathrm{tr}\left(\boldsymbol{X}\right). \end{aligned} \tag{22}$$

Similarly, when $\boldsymbol{K}^{-1}$ exists, $J'_G$ defined by Eq.(18) is expressed as

$$\begin{aligned} J'_G &= \mathrm{E}_{\boldsymbol{\epsilon}}\|\hat{\boldsymbol{\alpha}} - \boldsymbol{\alpha}^*\|_{\boldsymbol{K}}^2 \\ &= \mathrm{E}_{\boldsymbol{\epsilon}}\|\hat{\boldsymbol{\alpha}}\|_{\boldsymbol{K}}^2 - 2\mathrm{E}_{\boldsymbol{\epsilon}}\langle \boldsymbol{K}\hat{\boldsymbol{\alpha}}, \boldsymbol{\alpha}^*\rangle + \|\boldsymbol{\alpha}^*\|_{\boldsymbol{K}}^2 \\ &= \mathrm{E}_{\boldsymbol{\epsilon}}\|\hat{\boldsymbol{\alpha}}\|_{\boldsymbol{K}}^2 - 2\mathrm{E}_{\boldsymbol{\epsilon}}\langle \boldsymbol{K}\hat{\boldsymbol{\alpha}}, \boldsymbol{K}^{-1}\boldsymbol{z}\rangle + \|\boldsymbol{\alpha}^*\|_{\boldsymbol{K}}^2 \\ &= \mathrm{E}_{\boldsymbol{\epsilon}}\|\hat{\boldsymbol{\alpha}}\|_{\boldsymbol{K}}^2 - 2\mathrm{E}_{\boldsymbol{\epsilon}}\langle \hat{\boldsymbol{\alpha}}, \boldsymbol{z}\rangle + \|\boldsymbol{\alpha}^*\|_{\boldsymbol{K}}^2, \end{aligned}$$

where $\boldsymbol{z}$ is defined by Eq.(10). Since

$$\mathrm{E}_{\boldsymbol{\epsilon}}\mathrm{SIC}_{\mathrm{e}} = \mathrm{E}_{\boldsymbol{\epsilon}}\|\hat{\boldsymbol{\alpha}}\|_{\boldsymbol{K}}^2 - 2\mathrm{E}_{\boldsymbol{\epsilon}}\langle\hat{\boldsymbol{\alpha}}, \boldsymbol{z}\rangle,$$

we have

$$\mathrm{E}_{\boldsymbol{\epsilon}}\mathrm{SIC}_{\mathrm{e}} + (\text{constant}) = J'_G.$$

This implies that $\mathrm{SIC}_{\mathrm{e}}$ is essentially equivalent to SIC. However, Eq.(22) has the excellent property that $\boldsymbol{K}^{-1}$ is no longer needed. This will highly contribute to the stability of computation since the matrix inversion can become unstable if the matrix is ill-conditioned. Note that when $\dim\mathcal{H} < \ell$, $\boldsymbol{K}$ is always singular and $\boldsymbol{K}^\dagger$ may not be erased as in Eq.(22).

Although we do not need to invert the matrix $\boldsymbol{K}$, the inversion of an $\ell$-dimensional matrix is still needed when we are concerned with, e.g., kernel ridge regression (6) or Gaussian processes. Generally it requires $\mathcal{O}(\ell^3)$ scalar multiplications, which may be infeasible when the number $\ell$ of training examples is very large. In such cases, efficient calculation methods of matrix inversion, e.g., conjugate gradient inversion (Gibbs, 1997; Gibbs and MacKay, 1997) which iteratively approximates the matrix inversion with $\mathcal{O}(\ell^2)$ scalar multiplications, would be extremely useful. Furthermore, it is practically efficient to keep the number of kernel functions reasonable. For this purpose, methods such as clustering (Jain and Dubes, 1988), self-organizing map (Kohonen, 1995), principal component analysis (Jolliffe, 1986), and sparse greedy matrix approximation method (Smola and Schölkopf, 2000) may be useful.

## 5.3 Discussion of the Generalization Measure

We defined the generalization measure by Eq.(1). Here we discuss the relation between our generalization measure and the conventional generalization measures, and the dependency between the shape of the reproducing kernel and the generalization measure.

### 5.3.1 RKHS Based Generalization Measure and Conventional Generalization Measure

Using the RKHS norm for error measure is fairly common in the field of function approximation (e.g., Daubechies, 1992; Donoho and Johnstone, 1994; Donoho, 1995). The advantage of using Eq.(1) is that any RKHS norms can be employed as the generalization error, e.g., the Sobolev norm (Wahba, 1990) or weighted norms in the frequency space (Smola et al., 1998; Girosi, 1998).

On the other hand, the following generalization measure is used in the VC learning theory (Vapnik, 1998):[3]

$$\int_{\mathcal{D}} \left(\hat{f}(\boldsymbol{x}) - f(\boldsymbol{x})\right)^2 p(\boldsymbol{x})d\boldsymbol{x}, \tag{23}$$

where $p(\boldsymbol{x})$ is the probability density function of unseen test input points. It may seem that our generalization measure (1) can not take $p(\boldsymbol{x})$ into account as was done in Eq.(23). However, if $p(\boldsymbol{x})$ is known or can be estimated, it is possible to incorporate $p(\boldsymbol{x})$ by defining

[3] Note that in the book by Vapnik (1998), the generalization measure is defined more generally, using an arbitrary loss function. Here, we chose the square loss function for simplicity.

the inner product in the RKHS $\mathcal{H}$ as follows.

$$\langle f, g \rangle_{\mathcal{H}} = \int_{\mathcal{D}} f(\boldsymbol{x}) g(\boldsymbol{x}) p(\boldsymbol{x}) d\boldsymbol{x}. \tag{24}$$

Tsuda et al. (2002) proposed utilizing unlabeled samples (i.e., sample points without sample values) for estimating $p(\boldsymbol{x})$. However, further work is needed to apply this idea to the current setting.

In asymptotic statistical learning theories, the following generalization measure (or conceptually similar one) is often used (e.g., Akaike, 1974; Amari et al., 1992; Murata et al., 1994; Watanabe, 2001):

$$\mathrm{E}_{\{\boldsymbol{x}_i\}_{i=1}^{\ell}} \mathrm{E}_{\boldsymbol{\epsilon}} \int_{\mathcal{D}} \left( \hat{f}(\boldsymbol{x}) - f(\boldsymbol{x}) \right)^2 p(\boldsymbol{x}) d\boldsymbol{x}, \tag{25}$$

where $\mathrm{E}_{\{\boldsymbol{x}_i\}_{i=1}^{\ell}}$ denotes the expectation over training sample points $\{\boldsymbol{x}_i\}_{i=1}^{\ell}$. The notable difference is that expectation over training sample points $\{\boldsymbol{x}_i\}_{i=1}^{\ell}$ is *not* taken in our generalization measure (1), while it is taken in Eq.(25). This difference may stem from the difference in the purpose of model selection. If the purpose is to find the universally best model that provides the optimal generalization capability for all possible training sets on average, the average over training sets should be taken as in Eq.(25). This standpoint emphasizes that the model is universal and should not be altered according to the training sets. However, the universal model does not necessarily provide better generalization capability for the training set at hand. On the other hand, if it is allowed to change models adaptively depending on the training set, it is preferable not to take the average over training sets because by not taking the average, we can find the model that provides the optimal generalization capability for the training set at hand. In this article, we are taking the latter standpoint of *data-dependent* model selection, and the expectation over training sample points $\{\boldsymbol{x}_i\}_{i=1}^{\ell}$ is not taken. However, as can be seen from Eq.(1), we are still taking the expectation over the noise. In pursuit of the development of fully data-dependent model selection methods, we would ultimately like to not even take an expectation over the noise. This, however, is beyond the scope of this paper.

Finally in most of the statistical learning methods, the training sample points $\{\boldsymbol{x}_i\}_{i=1}^{\ell}$ are assumed to be independently drawn from $p(\boldsymbol{x})$, while we do not assume this condition in the current paper. Therefore, we can even deal with training sample points $\{\boldsymbol{x}_i\}_{i=1}^{\ell}$ sampled deterministically or drawn from another probability density function. This is advantageous especially in time-series analysis or in active learning scenarios (e.g., Fedorov, 1972; MacKay, 1992b; Cohn et al., 1996; Fukumizu, 2000; Sugiyama and Ogawa, 2000) because training sample points $\{\boldsymbol{x}_i\}_{i=1}^{\ell}$ are designed by users so they are no longer subject to $p(\boldsymbol{x})$.

### 5.3.2 Shape of Reproducing Kernel and Generalization Measure

An RKHS $\mathcal{H}$ is specified by a set of functions that span the function space and the inner product. This means that the shape of the reproducing kernel depends on the definition of the inner product (and also the generalization measure). This does not cause any problems when the reproducing kernels centered on training sample points span the whole RKHS (or equivalently the learning target function is included in the span of $\{K(\boldsymbol{x}, \boldsymbol{x}_i)\}_{i=1}^{\ell}$) because the regression function is essentially determined by the set of functions that span the function space. Therefore, we can use any norm as the generalization measure without changing the regression model (Sugiyama and Ogawa, 2002a). On the other hand, when the learning target function is not included in the span of $\{K(\boldsymbol{x}, \boldsymbol{x}_i)\}_{i=1}^{\ell}$, we can not determine both the shape of the reproducing kernel (so the regression function) and the generalization measure at the same time. Therefore, we are urged to choose either the following two scenarios.

The first scenario is to design the shape of the reproducing kernel as desired. Then the generalization measure is implicitly specified because the inner product is specified once the shape of the reproducing kernel is determined. The characteristics of the generalization measure can be interpreted by expanding the kernel function onto basis functions in the RKHS. For example, in the case of a Gaussian kernel, the generalization measure penalizes high frequency components (see e.g., Smola et al., 1998; Girosi, 1998, for details).

The second scenario is to specify the inner product (or the generalization measure) as desired. Then the shape of the reproducing kernel is accordingly determined. For example, if the function space is spanned by linearly independent functions $\{\phi_i(\boldsymbol{x})\}_{i=1}^{\infty}$ and the inner product is determined so that $\{\phi_i(\boldsymbol{x})\}_{i=1}^{\infty}$ form an orthonormal basis, then the reproducing kernel is given by

$$K(\boldsymbol{x}, \boldsymbol{x}') = \sum_{i=1}^{\infty} \phi_i(\boldsymbol{x})\phi_i(\boldsymbol{x}')$$

if it converges. However, we still require more general methods to explicitly express the reproducing kernels from given inner product, e.g., Eq.(24). Further investigation is needed to actually choose the second scenario.

For this reason, we will follow the first scenario and use, e.g., a Gaussian kernel in our computer simulations of next section, bearing in mind that high frequency components are penalized in the generalization measure.

## 6. Computer Simulations

In this section, the effectiveness of the proposed model selection method is investigated through computer simulations. As stated in Section 1, we are interested in observing whether the proposed SIC works well when the number of training examples is small. For this reason, our simulations are mainly focused on relatively small sample cases.[4]

### 6.1 Illustrative Examples

First, we consider artificial examples for illustrating how SIC works.

#### 6.1.1 Setting

For illustrating purpose, let the dimension $n$ of input vector be 1. We use the Gaussian RKHS with width $c = 1$, which may be one of the standard RKHSs (see e.g., Vapnik, 1998;

[4] In this simulation, the matrix inversion is calculated in a straightforward fashion since we focus on rather small sample cases. In large sample cases, the calculation of matrix inversion is time-consuming so efficient calculation methods of matrix inversion would be extremely useful (e.g., Gibbs, 1997; Gibbs and MacKay, 1997; Smola and Schölkopf, 2000)

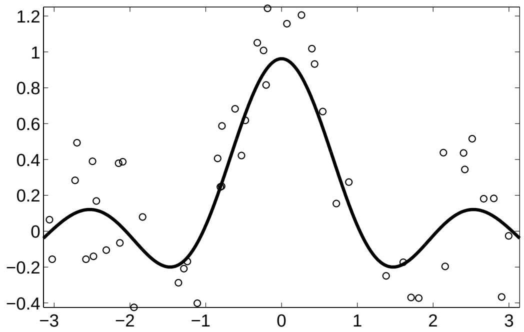


Figure 4: Target function and 50 training examples with noise variance $\sigma^2 = 0.09$.

Schölkopf et al., 2000):

$$K(x, x') = \exp\left(-\frac{(x - x')^2}{2c^2}\right). \tag{26}$$

The learning target function $f(x)$ is created as follows. We sample the following sinc function at 100 equally spaced template points $\{s_m\}_{m=1}^{100}$ in $[-\pi, \pi]$:

$$\text{sinc } x,$$

and obtain noiseless sample values $\{t_m\}_{m=1}^{100}$. We have chosen the sinc function because it is simple and often used as an illustrative regression example (e.g., Vapnik, 1998; Schölkopf et al., 2000). Using $\{(s_m, t_m)\}_{m=1}^{100}$ as training examples, we create the learning target function $f(x)$ by kernel ridge regression (7) with $\boldsymbol{T} = \boldsymbol{I}$ and $\lambda = 0.1$. The obtained $f(x)$ is illustrated in Figure 4.

The training set $\{(x_i, y_i)\}_{i=1}^{\ell}$ is created using this target function as follows. The sample points $\{x_i\}_{i=1}^{\ell}$ are independently drawn from the uniform distribution on $(-\pi, \pi)$. The sample values $\{y_i\}_{i=1}^{\ell}$ are created as $y_i = f(x_i) + \epsilon_i$, where the noise $\{\epsilon_i\}_{i=1}^{\ell}$ is independently drawn from the normal distribution with mean zero and variance $\sigma^2$. We consider the following four cases as the number $\ell$ of training examples and the noise variance $\sigma^2$:

$$(\ell, \sigma^2) = (100, 0.01), (50, 0.01), (100, 0.09), (50, 0.09), \tag{27}$$

i.e., we investigate the cases with small/large samples and small/large noise levels. An example of the training set is also illustrated in Figure 4. Kernel ridge regression (7) with $\boldsymbol{T} = \boldsymbol{I}$ is used for learning.

Note that the above setting is common in Sections 6.1.2 and 6.1.3.

### 6.1.2 Generalization Error Estimation

As proved in Section 5, the mean SIC is exactly unbiased if the noise variance is known. In this simulation, we will experimentally investigate how robust the unbiasedness of SIC is when the noise variance is estimated from training examples. The variance of SIC will also be experimentally evaluated.

The generalization error estimation performance of SIC is investigated as a function of the ridge parameter $\lambda$, using the following values:

$$\lambda \in \{10^{-3}, 10^{-2.5}, 10^{-2}, \ldots, 10^{3}\}. \tag{28}$$

SIC is calculated by Eq.(22), where the noise variance $\sigma^2$ is estimated by Eq.(12). The goodness of the learning result is measured by

$$\text{Error} = \|\hat{\boldsymbol{\alpha}}\|_{\boldsymbol{K}}^2 - 2\langle \hat{\boldsymbol{\alpha}}, \boldsymbol{z} \rangle, \tag{29}$$

where $\hat{\boldsymbol{\alpha}}$ denotes the parameter vector of the learned function at a given ridge parameter, and $\boldsymbol{z}$ is defined by Eq.(10). Note that the above error measure is essentially equivalent to the RKHS generalization measure $\|\hat{f} - f\|_{\mathcal{H}}^2$ since

$$\begin{aligned}
\|\hat{f} - f\|_{\mathcal{H}}^2 &= \|\hat{f} - g\|_{\mathcal{H}}^2 + \|g - f\|_{\mathcal{H}}^2 \\
&= \|\hat{\boldsymbol{\alpha}} - \boldsymbol{\alpha}^*\|_{\boldsymbol{K}}^2 + \|g - f\|_{\mathcal{H}}^2 \\
&= \|\hat{\boldsymbol{\alpha}}\|_{\boldsymbol{K}}^2 - 2\langle \boldsymbol{K}\hat{\boldsymbol{\alpha}}, \boldsymbol{\alpha}^* \rangle + \|\boldsymbol{\alpha}^*\|_{\boldsymbol{K}}^2 + \|g - f\|_{\mathcal{H}}^2 \\
&= \|\hat{\boldsymbol{\alpha}}\|_{\boldsymbol{K}}^2 - 2\langle \boldsymbol{K}\hat{\boldsymbol{\alpha}}, \boldsymbol{K}^{-1}\boldsymbol{z} \rangle + \|\boldsymbol{\alpha}^*\|_{\boldsymbol{K}}^2 + \|g - f\|_{\mathcal{H}}^2 \\
&= \|\hat{\boldsymbol{\alpha}}\|_{\boldsymbol{K}}^2 - 2\langle \hat{\boldsymbol{\alpha}}, \boldsymbol{z} \rangle + (\text{constant}).
\end{aligned}$$

The simulations are repeated 100 times for each $(\ell, \sigma^2)$ in Eq.(27), randomly drawing the sample points $\{x_i\}_{i=1}^{\ell}$ and noise $\{\epsilon_i\}_{i=1}^{\ell}$ from scratch in each trial.

Figure 5 displays the values of the true error (29) and SIC as a function of the ridge parameter $\lambda$ for each $(\ell, \sigma^2)$ in Eq.(27). The horizontal axis denotes the values of $\lambda$ in log-scale. From the top, the four curves denote the mean error with error bar, the mean SIC with error bar, the absolute difference between the mean error and the mean SIC, and the absolute difference between the standard deviations of the error and SIC. The error bar in the first and second rows of the figure denotes the standard deviation over 100 trials.

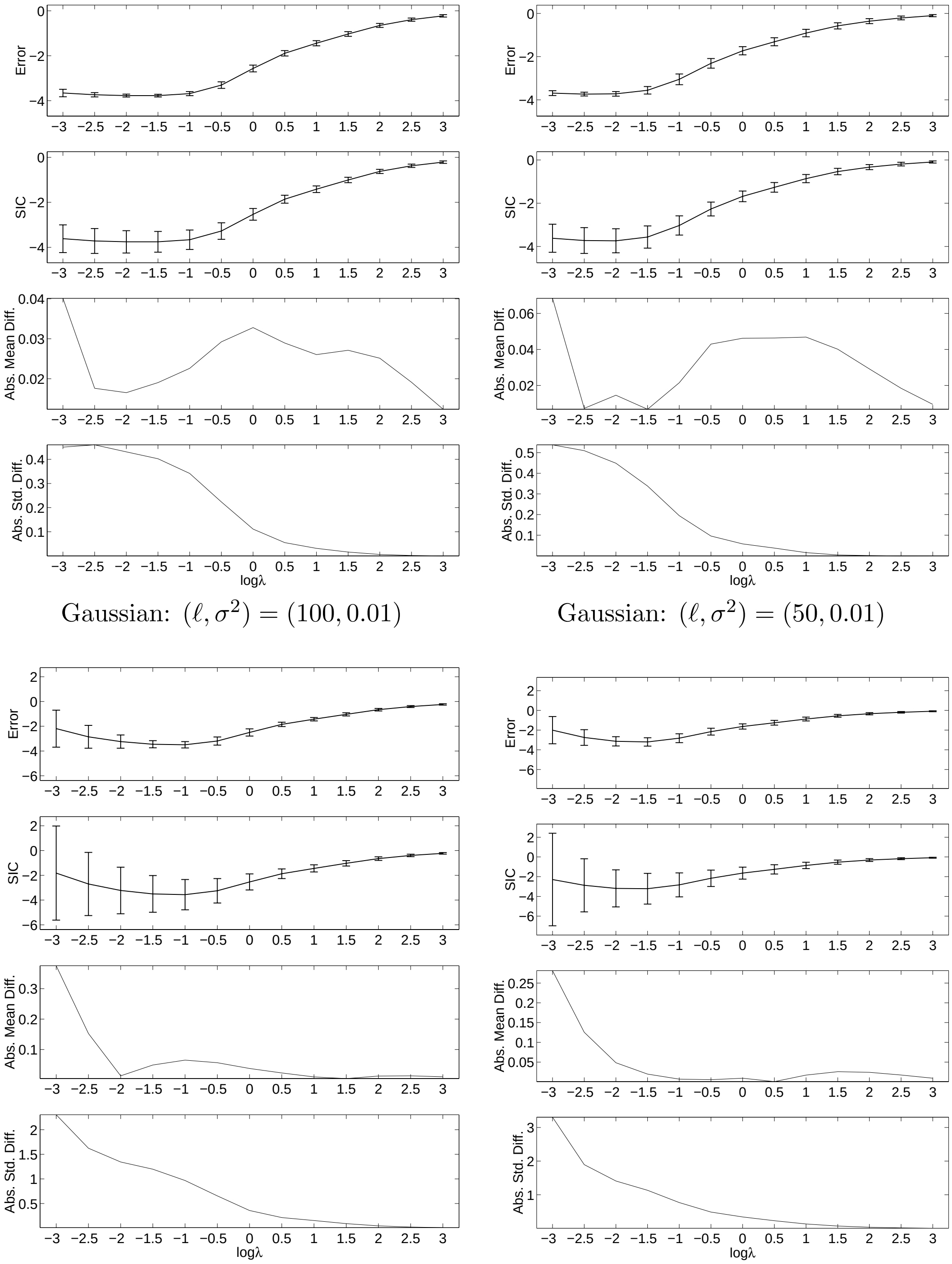
When $(\ell, \sigma^2) = (100, 0.01)$, the mean SIC seems to capture the mean error very well (see the first and second rows of the figure). Indeed, the absolute difference between the mean error and the mean SIC depicted in the third row shows that the difference is reasonably small. Therefore, the mean SIC can be regarded as an good estimator of the mean error. The second row of the figure also shows that the error bar of SIC is reasonably small for middle/large $\lambda$. Indeed it is almost the same as that of the true error, as can be observed from the absolute difference in the standard deviation depicted in the fourth row of the figure. However, as $\lambda$ gets small, the error bar of SIC tends to become slightly larger, which may be caused by the fact that the uncertainty of the estimated parameter vector $\hat{\boldsymbol{\alpha}}$ increases large as $\lambda$ decreases.

When $(\ell, \sigma^2) = (50, 0.01)$, the simulation results bear a close resemblance to the case of $(\ell, \sigma^2) = (100, 0.01)$. This means that even when the number of training examples decreases, the error estimation performance of SIC is still maintained.

When $(\ell, \sigma^2) = (100, 0.09)$, the absolute mean difference (see the third row of the figure) is still small so the mean SIC is a good estimator of the mean error even when the noise level increases. However, the error bars of SIC become large (see the second row) compared with the case of $(\ell, \sigma^2) = (100, 0.01)$. Finally when $(\ell, \sigma^2) = (50, 0.09)$, the simulation results show the same trend as the case of $(\ell, \sigma^2) = (50, 0.01)$.

We also perform the same simulation for the sinc kernel with $\Omega = 2.5$:

$$K(x, x') = \frac{\Omega}{\pi} \text{sinc}\left( \frac{\Omega}{\pi}(x - x') \right).$$



Gaussian: $(\ell, \sigma^2) = (100, 0.01)$

Gaussian: $(\ell, \sigma^2) = (50, 0.01)$

Gaussian: $(\ell, \sigma^2) = (100, 0.09)$

Gaussian: $(\ell, \sigma^2) = (50, 0.09)$

Figure 5: Values of the true error (29) and SIC for Gaussian kernel. The horizontal axis denotes the value of $\lambda$ in log-scale. Shown from the top are the mean error with error bar, the mean SIC with error bar, the absolute difference between the mean error and the mean SIC, and the absolute difference between the standard deviations of the error and SIC.

The simulation results are depicted in Figure 6, showing a similar tendency to the case of the Gaussian kernel. Furthermore, we performed the same simulation for several other kernels including regularized trigonometric polynomial kernel, regularized polynomial kernel, and Laplacian kernel (see e.g., Vapnik, 1998). The results are qualitatively the same as those of the Gaussian and sinc kernel cases so they are omitted.

From the above simulation results, the unbiasedness of SIC is shown to be unchanged irrespective of the number of training examples, the noise level, and the choice of RKHSs. However, as $\lambda$ gets small, the error bar of SIC tends to become larger, especially for high noise level ($\sigma^2 = 0.09$).

Therefore, it is extremely important to investigate how this large error bar of SIC affects the ridge parameter selection, which is the primal interest of the following simulations.

### 6.1.3 Ridge Parameter Selection

Now we experimentally investigate how SIC works in ridge parameter selection.

The ridge parameter is selected from Eq.(28). The goodness of the selection is evaluated by the test error at 1000 randomly created test points $\{(x'_i, y'_i)\}_{i=1}^{1000}$:

$$\text{Test error} = \frac{1}{1000}\sum_{i=1}^{1000}\left(\hat{f}(x'_i) - f(x'_i)\right)^2.$$

As ridge parameter selection criteria, the following three methods are compared.

**SIC:** SIC is calculated by Eq.(22), where the noise variance $\sigma^2$ is estimated by Eq.(12).

**Leave-one-out cross-validation (CV):** The leave-one-out error is calculated efficiently in closed-form (see e.g., Wahba, 1990; Orr, 1996):

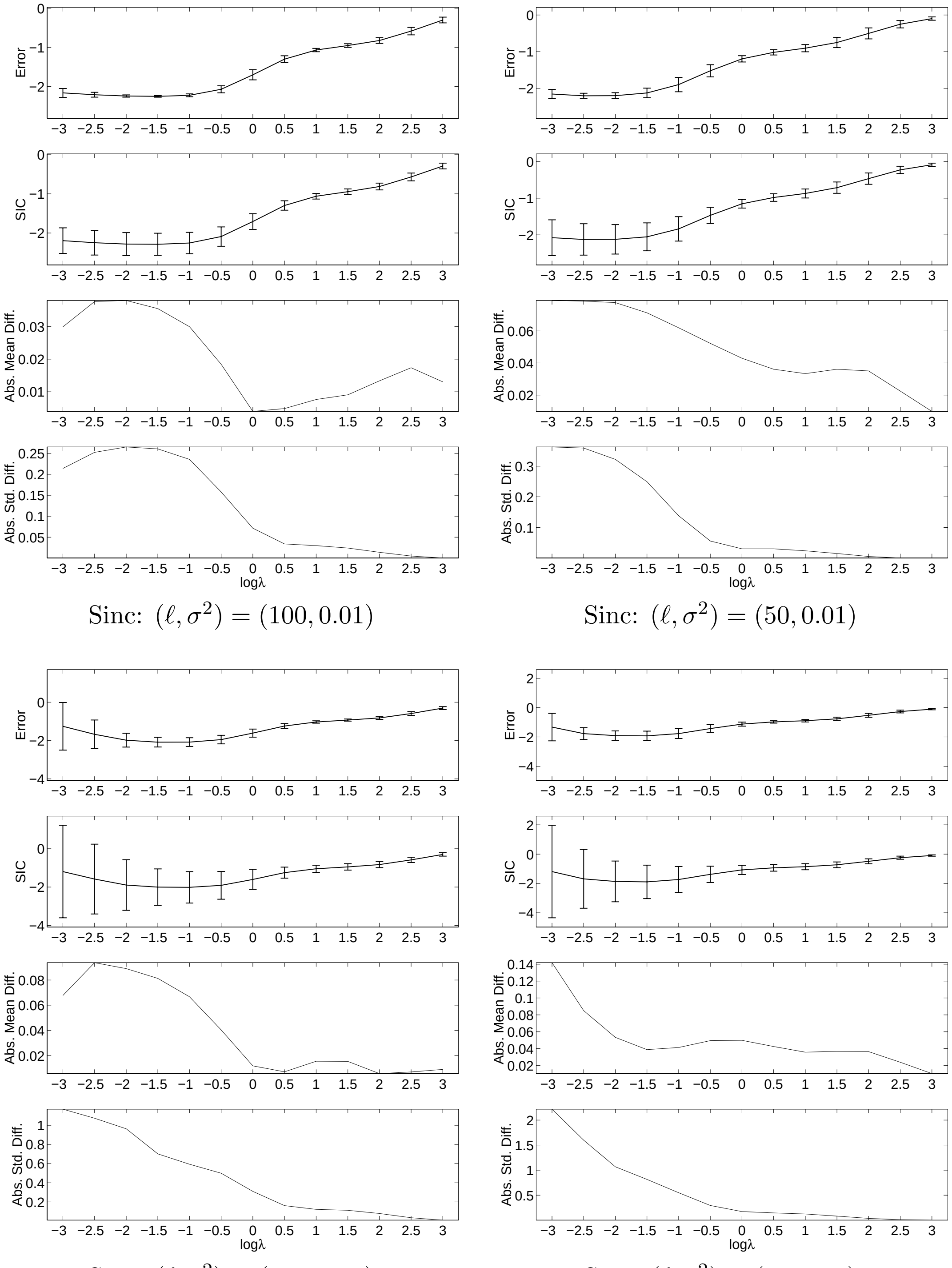
$$\frac{1}{\ell}\|(\mathrm{diag}\,(\boldsymbol{I} - \boldsymbol{K}\boldsymbol{X}))^{-1}(\boldsymbol{I} - \boldsymbol{K}\boldsymbol{X})\boldsymbol{y}\|^2,$$

where $\mathrm{diag}\,(\boldsymbol{I} - \boldsymbol{K}\boldsymbol{X})$ is the same size and has the same diagonal as $(\boldsymbol{I} - \boldsymbol{K}\boldsymbol{X})$ but is zero on the off-diagonal elements.

**Akaike's Bayesian information criterion (ABIC) (Akaike, 1980):** ABIC, which is a so-called empirical Bayesian method (see also Schwarz, 1978; MacKay, 1992a; Watanabe, 2001), determines the ridge parameter so that its likelihood is maximized.

The simulation is repeated 100 times, randomly drawing the sample points $\{x_i\}_{i=1}^{\ell}$, noise $\{\epsilon_i\}_{i=1}^{\ell}$, and the test sample points $\{x'_i\}_{i=1}^{1000}$ from the scratch in each trial.

In Figure 7, the test error by each method is depicted. The left graphs depict the distributions of the test error with standard box plot while the right graph shows the test error for every trial as a scatter plot. The box plot notation specifies marks at 95, 75, 50, 25, and 5 percentiles from the top. 'OPT' indicates the test error obtained by the optimal ridge parameter, i.e., we actually calculated the test error for each $\lambda$ in Eq.(28) and selected the one that minimizes the test error. In the scatter plot, a circle denotes the test errors by SIC vs. CV while a cross denotes the test errors by SIC vs. ABIC. Plot symbols in the



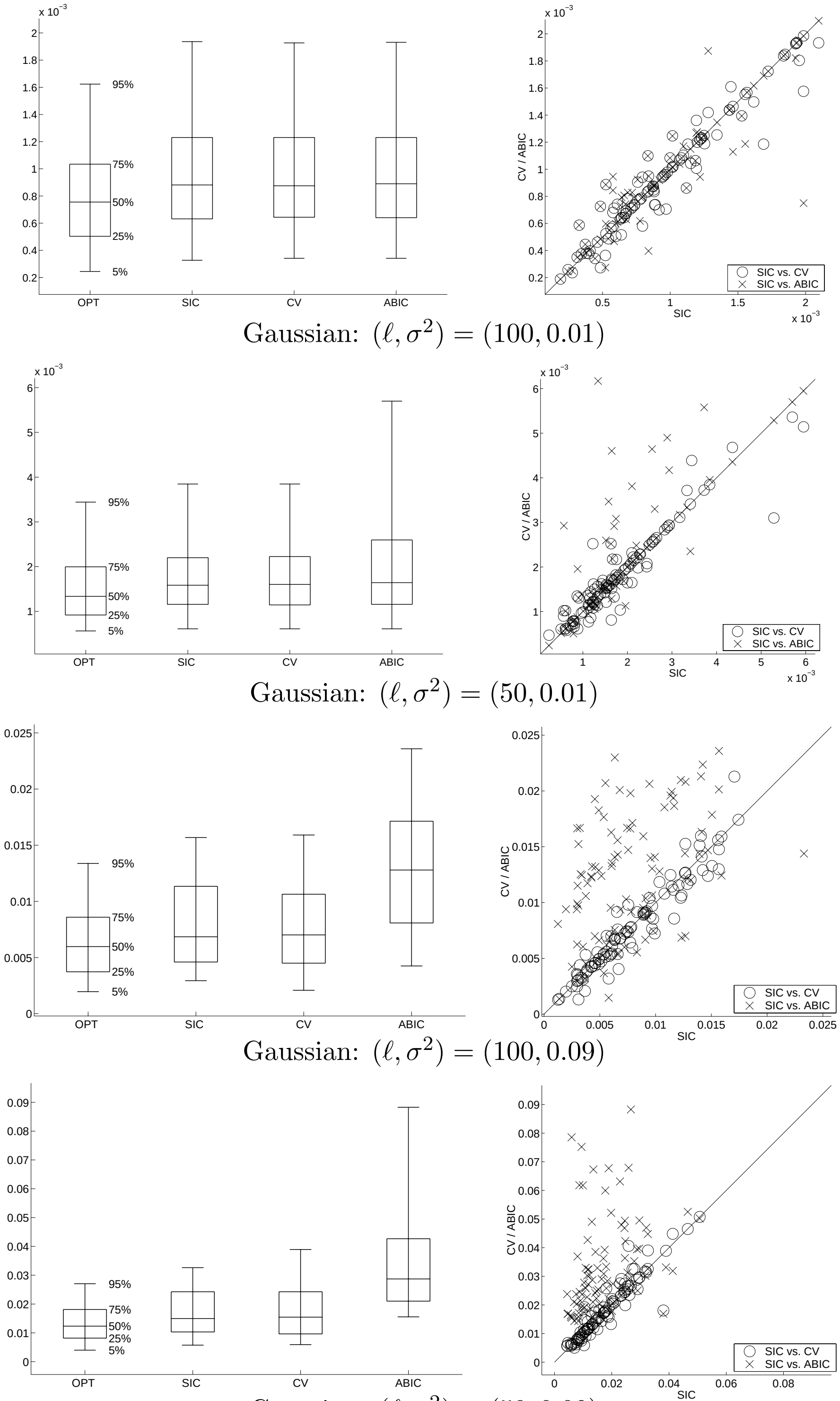
Sinc: $(\ell, \sigma^2) = (100, 0.01)$

Sinc: $(\ell, \sigma^2) = (50, 0.01)$

Sinc: $(\ell, \sigma^2) = (100, 0.09)$

Sinc: $(\ell, \sigma^2) = (50, 0.09)$

Figure 6: Values of true error and SIC for sinc kernel.



Gaussian: $(\ell, \sigma^2) = (100, 0.01)$

Gaussian: $(\ell, \sigma^2) = (50, 0.01)$

Gaussian: $(\ell, \sigma^2) = (100, 0.09)$

Gaussian: $(\ell, \sigma^2) = (50, 0.09)$

Figure 7: Test error for Gaussian kernel. The box plot notation specifies marks at 95, 75, 50, 25, and 5 percentiles of values. 'OPT' in the left graphs indicates the test error obtained by the optimal ridge parameter. In the right graphs, a circle denotes the test errors by SIC vs. CV while a cross denotes the test errors by SIC vs. ABIC.

Table 2: P-values for Gaussian kernel. 'T' and 'Wilcoxon' denote t-test and Wilcoxon signed rank test, respectively. The difference between two methods is regarded as significant if $p < 0.05$, and as very significant if $p < 0.01$.

| Methods | SIC vs. CV | | SIC vs. ABIC | |
|---|---|---|---|---|
| Test | T | Wilcoxon | T | Wilcoxon |
| $\ell = 100$, $\sigma^2 = 0.01$ | $7.45 \times 10^{-1}$ | $2.74 \times 10^{-1}$ | $8.87 \times 10^{-1}$ | $5.78 \times 10^{-1}$ |
| $\ell = 50$, $\sigma^2 = 0.01$ | $9.97 \times 10^{-1}$ | $7.31 \times 10^{-1}$ | $1.64 \times 10^{-1}$ | $4.95 \times 10^{-4}$ |
| $\ell = 100$, $\sigma^2 = 0.09$ | $8.04 \times 10^{-1}$ | $1.66 \times 10^{-1}$ | $2.01 \times 10^{-9}$ | $6.96 \times 10^{-12}$ |
| $\ell = 50$, $\sigma^2 = 0.09$ | $8.50 \times 10^{-1}$ | $2.85 \times 10^{-1}$ | $5.00 \times 10^{-12}$ | $1.31 \times 10^{-15}$ |

upper-left area mean that SIC outperforms CV or ABIC, while the plot symbols in the lower-right area denote the opposite.

When $(\ell, \sigma^2) = (100, 0.01)$, the box plot shows that SIC works well in all percentiles, and it is comparable to CV and ABIC. The right scatter plot shows that almost all circles and crosses are plotted near the diagonal line, so in accordance with the box plots, the performance of SIC is comparable to CV and ABIC. In order to investigate whether the difference in the test error is significant or not, we perform two kinds of hypothesis tests (see e.g., Henkel, 1979). One is the *t-test*, which compares the means of two samples under the assumption that they are drawn from the normal distribution with the same (but unknown) variance. The other is the *Wilcoxon signed rank test*, which is a non-parametric test so it can be applied to any distribution. The p-values are described in Table 2. Let us regard the difference in the test error as significant if $p < 0.05$, and as very significant if $p < 0.01$. The p-values do not indicate that both the difference between SIC and CV and the difference between SIC and ABIC are significant. Therefore, SIC works as well as CV and ABIC.

When $(\ell, \sigma^2) = (50, 0.01)$, SIC and CV maintain a comparably good performance even when the number of training examples is decreased (cf. Table 2). On the other hand, the performance of ABIC is degraded in the 75 and 95 percentiles. This can also be observed from the scatter plot, i.e., some crosses are plotted in the upper-left area. Although the t-test does not reveal this significant difference between SIC and ABIC, the Wilcoxon test does so. In this case, the Wilcoxon test is considered to be more reliable than the t-test since the distribution of the test error by ABIC deviates from the normal distribution, as can be observed from the box plot (e.g., 95 percentile tail is much longer than 5 percentile tail).

When $(\ell, \sigma^2) = (100, 0.09)$, the performance of SIC is slightly worse than that of CV in 5 and 75 percentiles, while SIC and CV perform equally when $(\ell, \sigma^2) = (100, 0.01)$. This implies that the large error bar of SIC shown in Figure 5 slightly degrades the ridge parameter selection performance. However, the p-values described in Table 2 do not indicate that the difference between SIC and CV is significant. Therefore, the large variance of SIC when the noise level is high may be permissible. On the other hand, the performance of ABIC is significantly degraded in all percentiles, and most of the crosses are plotted in the upper-left area in the scatter plot (cf. p-values in Table 2).

Finally, when $(\ell, \sigma^2) = (50, 0.09)$, SIC works excellently, although the performance of SIC is slightly degraded when compared with the case of $(\ell, \sigma^2) = (50, 0.01)$. This may be again caused by the large error bar of SIC shown in Figure 5. However, compared with CV, SIC is shown to work reasonably well since SIC outperforms CV in 95 percentile and works equally in other percentiles (although the p-values described in Table 2 do not reveal the difference between SIC and CV as significant). On the other hand, ABIC again gives large errors and the difference between SIC and ABIC is very significant (cf. Table 2).

The above simulation results show that, for the toy artificial data, SIC can be successfully applied to ridge parameter selection. Especially, it is notable that the good performance of SIC is maintained even when the number $\ell$ of training examples decreases. However, as expected, the performance of SIC is degraded as the noise level increases (see the cases with $\sigma^2 = 0.09$). Nevertheless, the performance of SIC is still comparable to CV and is slightly better than ABIC.

We also performed similar simulations for several other learning target functions and several RKHSs, and observed that the simulation results bear a close resemblance to the above findings (therefore they are omitted).

### 6.2 Real Data Sets

Now we apply SIC to real data sets, and evaluate its practical usefulness. We use 10 data sets provided by DELVE (Rasmussen et al., 1996): *Abalone, Boston, Bank-8fm, Bank-8nm, Bank-8fh, Bank-8nh, Kin-8fm, Kin-8nm, Kin-8fh*, and *Kin-8nh*.

*Abalone* data set includes 4177 samples, each of which consists of 9 physical measurements. The task is to estimate the last attribute (the age of abalones) from the rest. The first attribute is qualitative (male/female/infant) so it is ignored, i.e., 7-dimensional input and 1-dimensional output data is used. For convenience, every attribute is normalized in $[0, 1]$. 100 randomly selected samples $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^{100}$ are used for training, and the other 4077 samples $\{(\boldsymbol{x}'_i, y'_i)\}_{i=1}^{4077}$ are used for testing. The reason why only 100 samples are used for training is that we are interested in the performance in the small sample case, as stated in Section 1. The test error is measured by

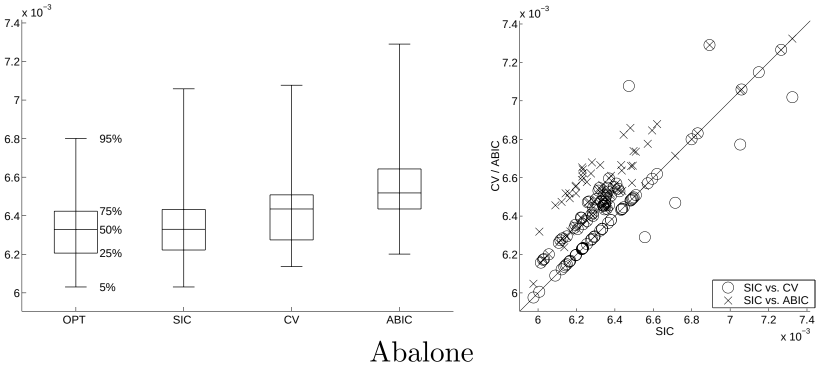
$$\text{Test error} = \frac{1}{4077} \sum_{i=1}^{4077} \left( \hat{f}(\boldsymbol{x}'_i) - y'_i \right)^2, \tag{30}$$

where $\{(\boldsymbol{x}'_i, y'_i)\}_{i=1}^{4077}$ denote the test samples. A Gaussian kernel (26) with width $c = 1$ is employed and kernel ridge regression (7) with $\boldsymbol{T} = \boldsymbol{I}$ is used for learning. The ridge parameter $\lambda$ is selected from the following values.

$$\lambda \in \{10^{-3}, 10^{-2}, 10^{-1}, \ldots, 10^{3}\}.$$

Again we compare SIC with CV and ABIC (see Section 6.1.3 for detail). The simulation is repeated 100 times, randomly selecting the training set $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^{100}$ from scratch in each trial (i.e., sampling without replacement). Note that the test set $\{(\boldsymbol{x}'_i, y'_i)\}_{i=1}^{4077}$ also varies in each trial.

In Figure 8, the test error by each method is displayed. The left graph depicts the distribution of the test error with standard box plot while the right graph shows the test



Abalone

Figure 8: Simulation results with Abalone data sets (Gaussian kernel). The box plot notation specifies marks at 95, 75, 50, 25, and 5 percentiles of values. ‘OPT’ in the left graph indicates the test error obtained by the optimal ridge parameter. In the right graph, a circle denotes the test errors by SIC and CV while a cross denotes the test errors by SIC and ABIC.

error in every trial using a scatter plot (cf. Section 6.1.3 for a detailed discussion of this plot type).

The box plot in Figure 8 shows that SIC works fairly well up to the 75 percentile when compared with OPT, and is slightly better than CV and ABIC for all percentiles. The scatter plot also shows that SIC tends to work slightly better than CV and ABIC, although CV outperforms SIC in 4 trials. Again we test whether the difference in the test error is significant or not by t-test and Wilcoxon signed rank test (see Section 6.1.3 for detail). The p-values are described in Table 3. The table shows that the p-values of the Wilcoxon test are very small, so SIC is significantly different from both CV and ABIC. On the other hand, the p-value of the t-test does not say the difference between SIC and CV is significant. However, the distributions of errors by SIC, CV, and ABIC are not symmetric (i.e., 95 percentile tail is longer than 5 percentile tail, see the box plot in Figure 8), Therefore, in this case, the t-test may not be reliable.

A similar simulation is also performed for the *Boston* data set, which includes 506 samples, each consisting of 13-dimensional input and 1-dimensional output data. Every attribute is again normalized in $[0, 1]$. 100 randomly chosen samples are used for training and the other 406 samples are used for testing. The test error is measured by a similar manner to Eq.(30). The simulation results for the *Boston* data set are depicted in Figure 9. The box plot shows that SIC works excellently for all percentiles when compared with OPT, and SIC outperforms CV and ABIC. The scatter plot shows that SIC always provides equal or smaller test error than CV and ABIC. The p-values in Table 3 also confirm that SIC is significantly different from CV and ABIC.

The *Bank* data family consists of four different data sets. They are labeled as ‘fm’, ‘nm’, ‘fh’, and ‘nh’, where ‘f’ or ‘n’ signifies ‘fairly linear’ or ‘non-linear’, respectively, and ‘m’ or ‘h’ signifies ‘medium unpredictability/noise’ or ‘high unpredictability/noise’, respectively. Each of the 4 data sets includes 8192 samples, consisting of 8-dimensional input and 1-

Table 3: P-values for real data sets (Gaussian kernel). 'T' and 'Wilcoxon' denote t-test and Wilcoxon signed rank test, respectively. The difference between two methods is regarded as significant if $p < 0.05$, and as very significant if $p < 0.01$.

| Methods | SIC vs. CV | | SIC vs. ABIC | |
|---|---|---|---|---|
| Test | T | Wilcoxon | T | Wilcoxon |
| Abalone | $1.97 \times 10^{-1}$ | $9.48 \times 10^{-6}$ | $2.14 \times 10^{-4}$ | $2.62 \times 10^{-16}$ |
| Boston | $3.49 \times 10^{-11}$ | $2.52 \times 10^{-8}$ | $2.33 \times 10^{-39}$ | $4.01 \times 10^{-18}$ |
| Bank-8fm | $1.00 \times 10^{-15}$ | $1.15 \times 10^{-11}$ | $2.13 \times 10^{-23}$ | $2.53 \times 10^{-14}$ |
| Bank-8nm | $9.81 \times 10^{-2}$ | $6.20 \times 10^{-7}$ | $7.68 \times 10^{-6}$ | $4.26 \times 10^{-9}$ |
| Bank-8fh | $4.34 \times 10^{-1}$ | $1.07 \times 10^{-1}$ | $2.29 \times 10^{-1}$ | $2.98 \times 10^{-1}$ |
| Bank-8nh | $1.15 \times 10^{-1}$ | $6.07 \times 10^{-3}$ | $9.02 \times 10^{-11}$ | $1.10 \times 10^{-7}$ |
| Kin-8fm | $7.94 \times 10^{-3}$ | $6.10 \times 10^{-5}$ | $4.75 \times 10^{-58}$ | $1.25 \times 10^{-17}$ |
| Kin-8nm | $8.84 \times 10^{-5}$ | $1.61 \times 10^{-8}$ | $2.30 \times 10^{-25}$ | $5.97 \times 10^{-17}$ |
| Kin-8fh | $4.26 \times 10^{-4}$ | $4.87 \times 10^{-6}$ | $5.53 \times 10^{-32}$ | $2.63 \times 10^{-17}$ |
| Kin-8nh | $7.66 \times 10^{-2}$ | $5.48 \times 10^{-3}$ | $3.60 \times 10^{-1}$ | $6.60 \times 10^{-4}$ |

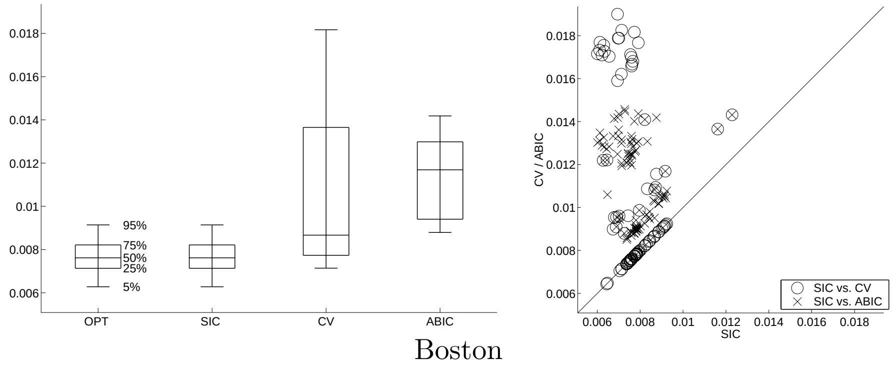
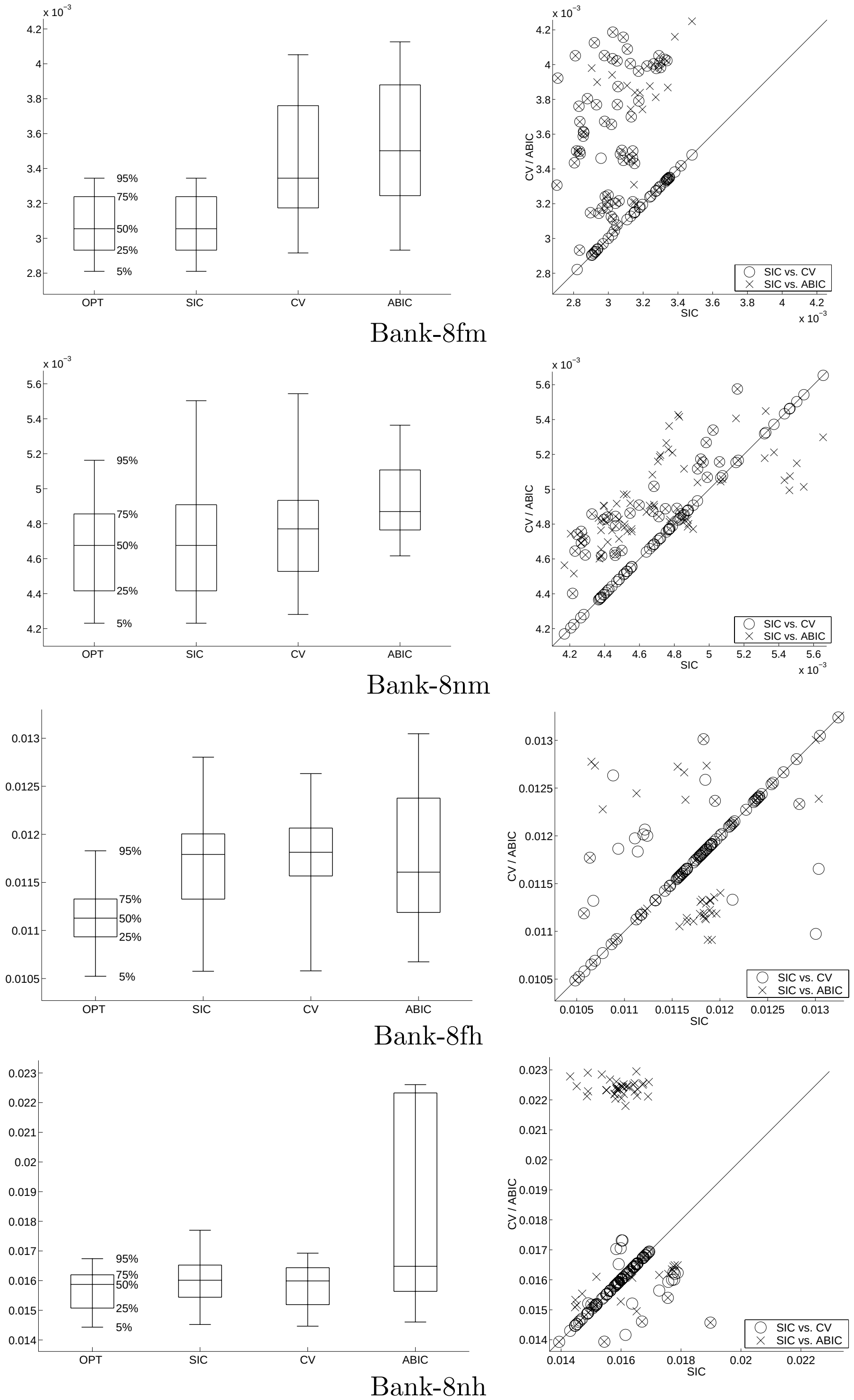


Figure 9: Simulation results with Boston data sets (Gaussian kernel).

dimensional output data. Every attribute is again normalized in $[0, 1]$. 100 randomly selected samples are used for training and the other 8092 samples are used for testing. The test error is measured in a similar manner to Eq.(30). The simulation results for the *Bank* data sets are depicted in Figure 10.

For the *Bank-8fm* data set, SIC works excellently for all percentiles, and it outperforms CV and ABIC. The p-values described in Table 3 confirm that SIC is significantly different from CV and ABIC. The scatter plot shows that SIC always provides equal or smaller test error than CV and ABIC.

For the *Bank-8nm* data set, SIC performs slightly better than CV in 25 and 50 percentiles. The Wilcoxon test described in Table 3 shows that the difference between SIC and



Bank-8fm

Bank-8nm

Bank-8fh

Bank-8nh

Figure 10: Simulation results with real data sets (Gaussian kernel).

CV is very significant, while the t-test does not say the difference is significant. However, in this case, the distributions of the test error are not symmetric so the t-test may not be reliable. On the other hand, SIC outperforms ABIC up to 75 percentile, while ABIC outperforms SIC at 95 percentile. This can also be observed in the scatter plot, showing that some crosses are plotted in the lower-right area.

For the *Bank-8fh* data set, all 3 methods do not seem to work well. SIC is better than CV for 25 percentile while CV is better than SIC for 95 percentile. SIC is better than ABIC for 5, 75, and 95 percentiles while ABIC is better than SIC for 25 and 50 percentiles. The p-values described in Table 3 do not say both the difference between SIC and CV and the difference between SIC and ABIC are significant. Note that ABIC once gave the error of 0.033, but this plot is omitted in the scatter plot since it heavily degrades the whole graph.

For the *Bank-8nh* data set, SIC is comparable with CV for 5, 50, and 75 percentiles, although 25 and 95 percentiles are worse than those of CV. The Wilcoxon test described in Table 3 says that SIC is significantly different from CV while t-test does not say the difference is significant. On the other hand, ABIC sometimes gives very large errors as can be seen from the scatter plot, although ABIC sometimes outperforms SIC.

The simulation results for the *Bank* data family show that when the unpredictability/noise is medium (signified by 'm'), SIC works very well. However, for the data sets with high unpredictability/noise (signified by 'h'), the performance of SIC tends to be degraded. This may be caused by the large variance of SIC in the case of high noise level (see Section 6.1.2). Although this drawback is to be improved in the future, it may be permissible given the facts that all the 3 methods do not work well for the *Bank-8fh* data set and SIC outperforms ABIC for the *Bank-8nh* data set.

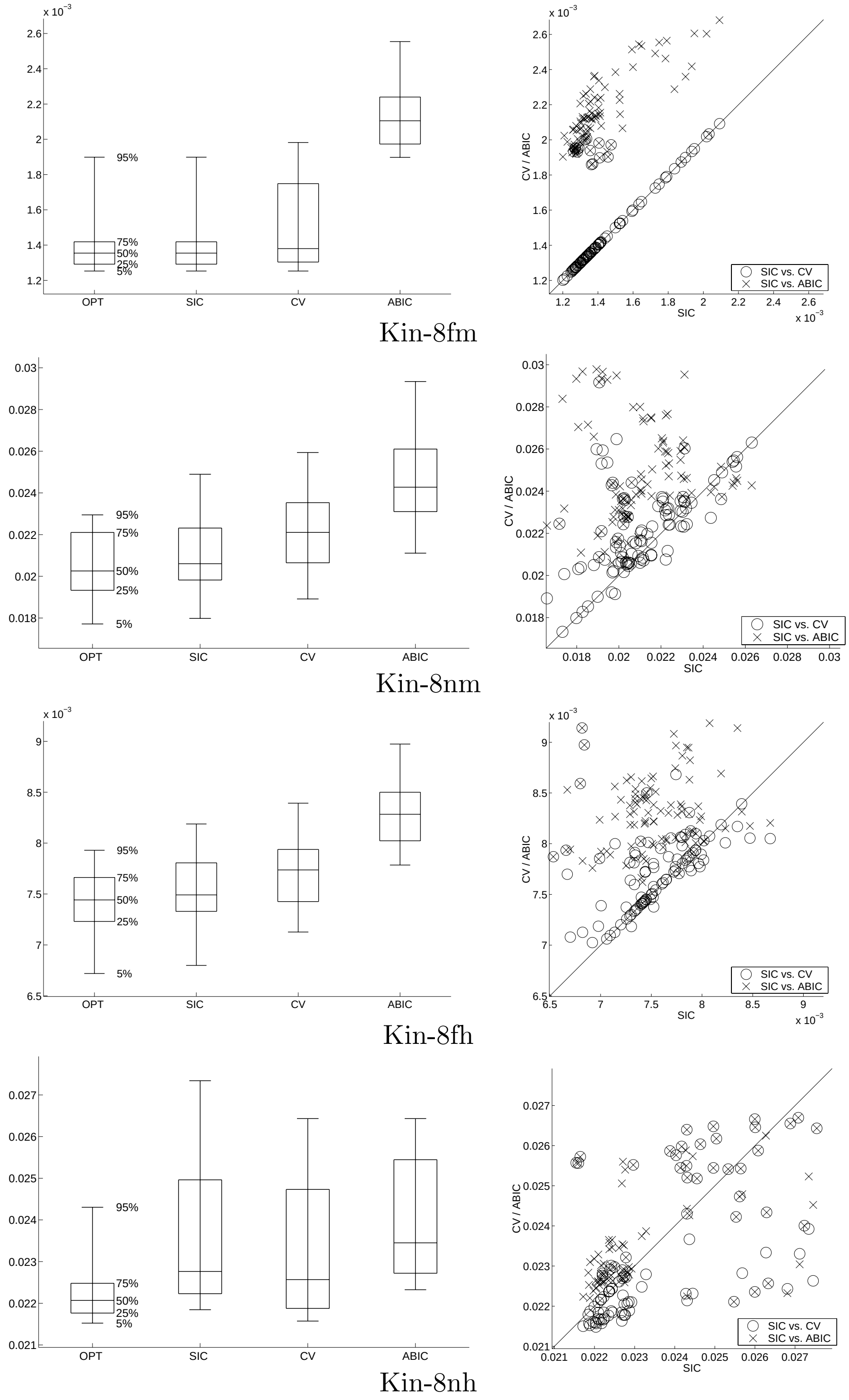
The *Kin* data family also consists of four different data sets labeled as 'fm', 'nm', 'fh', and 'nh'. Each of the 4 data sets includes 8192 samples, consisting of 8-dimensional input and 1-dimensional output data. We normalize every attribute in $[0, 1]$. 100 randomly selected samples are used for training and the other 8092 samples are used for testing. The test error is measured by a similar manner to Eq.(30). The simulation results for the *Kin* data sets are depicted in Figure 11.

For the *Kin-8fm* data set, SIC shows an outstanding performance for all percentiles. 5, 25, and 50 percentiles of SIC are comparable with those of CV, and SIC outperforms CV for 75 and 95 percentiles. This can also be observed from the scatter plot, showing that almost all circles are on the diagonal line but some are plotted in the upper-left area. The p-values described in Table 3 say that SIC is significantly different from CV. On the other hand, ABIC does not work properly and it provides large test errors. The p-values described in Table 3 say that the difference between SIC and ABIC is very significant.

For the *Kin-8nm* data set, SIC works reasonably well, although 95 percentile is rather large compared with OPT. For all percentiles, SIC outperforms CV and ABIC, and the difference is shown to be very significant by the statistical tests (see Table 3). Even so, the scatter plot shows that CV and ABIC can occasionally improve upon SIC.

The results for the *Kin-8fh* data set are similar to those of the *Kin-8nm* data set.

Finally, for the *Kin-8nh* data set, all the 3 methods do not seem to work well. CV outperforms SIC for all percentiles and SIC outperforms ABIC up to 75 percentile, although 95 percentile of SIC is worse than that of ABIC. The Wilcoxon test says that CV and ABIC are significantly different from SIC, while the t-test does not say the difference is significant.



Kin-8fm

Kin-8nm

Kin-8fh

Kin-8nh

Figure 11: Simulation results with real data sets (Gaussian kernel).

In this case, the Wilcoxon test may be reliable because of the asymmetry of the distributions (see the box plot). Therefore, CV may outperform SIC. In contrast, SIC may be different from ABIC but we can not judge which is better from the statistical test.

The simulation results for the *Kin* data family show that for the *Kin-8fm*, *Kin-8nm*, and *Kin-8fh* data sets, SIC works fairly well and it tends to outperform CV and ABIC. However, for the *Kin-8nh* data sets, SIC does not work properly. This may be again caused by the large variance of SIC in the case of high noise level (see Section 6.1.2). However, it should be noted that in this case, even CV and ABIC do not work well.

The above simulation results[5] for real data sets imply that SIC should be considered as an important practical model selection criterion for choosing the ridge parameter, although the performance can be degraded when the noise level is very high.

## 7. Conclusions and Future Prospects

The paper studied model selection based on a generalization of SIC to the case that reproducing kernels centered on training sample points do not span the whole reproducing kernel Hilbert space (RKHS). This extension allows an efficient model selection even in infinite dimensional RKHSs. The SIC based generalization error estimation is applicable under the assumptions that (a) the learning target function belongs to a specified RKHS. (b) the kernel regression model is employed, and (c) the generalization error is measured by the expected squared norm in the RKHS. Extensive simulation studies showed that SIC outperformed leave-one-out cross-validation and an empirical Bayesian method for most of the data sets. Therefore, SIC may be considered as one of the practical model selection criteria for choosing the ridge parameter.

On the other hand, there is still plenty of room for further investigation and improvements. In the following, we describe possible future directions.

The unbiasedness of SIC is theoretically guaranteed if the noise variance is known. Even when the noise variance is unknown, the unbiasedness of SIC is still theoretically maintained if an unbiased estimator of the noise variance is available (Sugiyama and Ogawa, 2001). In this article, we used a biased estimator of the noise variance given by Eq.(12), and experimentally confirmed that SIC still stays almost unbiased even if a biased noise variance estimator is used. Future studies to theoretically investigate this finding are needed.

As discussed in Section 5.3.2, the generalization measure and the shape of the reproducing kernel generally relate to each other. If the shape of the reproducing kernel is designed as desired, the generalization measure is implicitly fixed so it could be different from a desired cost function. On the other hand, if the generalization measure is specified, it is not straightforward to obtain an explicit expression of the reproducing kernel. In the simulation studies carried out in Section 6, we took the former standpoint and mainly used the Gaussian kernel. In theory, the generalization error is measured by the RKHS norm which penalizes high frequency components in the Gaussian kernel case, while the experimental performance is measured by the test error. Although this is inconsistent, clearly the RKHS norm based generalization measure, which is well approximated by SIC, is highly correlated to the true test error. Similar situations can also be found in, e.g., predictive training er-

[5] We also performed similar simulations with several other RKHSs. The results were comparable with the Gaussian kernel case, so they are omitted.

ror based methods (Mallows, 1964, 1973) where the model is chosen so that the error at training sample points is minimized, Kullback-Leibler divergence based methods (Akaike, 1974; Takeuchi, 1976; Sugiura, 1978; Konishi and Kitagawa, 1996, see also Murata et al., 1994) where the model is selected so that the Kullback-Leibler divergence is minimized, or empirical Bayesian methods (Schwarz, 1978; Akaike, 1980; MacKay, 1992a; Watanabe, 2001) where hyper-parameters are determined so that their likelihood is maximized. In those cases, the criteria are different from the test error but they are correlated to the test error. Furthermore, even the expected prediction error (23), which is one of the standard error measures in statistical model selection, is slightly different from the test error, e.g., Eq.(30). An interesting direction of research is therefore to find further alternative error measures that are well correlated to the test error and at the same time, that can be estimated robustly. For example, under transductive settings (i.e., the cases where the test input points are known in advance), devising a method for obtaining an RKHS whose norm directly evaluates the error at test input points may be promising.

Experimentally SIC is shown to work well in most of the cases. However, its performance along with CV can be degraded for a too high noise level. For SIC, this may be caused by the fact that SIC is derived as an *exact* unbiased estimator of (an essential part of) the generalization error but the variance of SIC is not taken into account. A possible fix of this instability is to add a small bias to SIC for stabilization, e.g., along the lines of Sugiyama and Ogawa (2001) or Tsuda et al. (2002). However, it remains to be investigated whether these or some alternative strategy will also be successful for infinite dimensional RKHSs.

We focused on the case that the parameters in the kernel regression model are estimated linearly. On the other hand, practically useful estimation methods such as robust regression (Huber, 1981), support vector regression (e.g., Vapnik, 1995; Schölkopf et al., 1998; Burges, 1998; Cristianini and Shawe-Taylor, 2000; Schölkopf et al., 2000; Müller et al., 2001; Schölkopf and Smola, 2002), and sparse regression (Williams, 1995; Tibshirani, 1996; Chen et al., 1998) are non-linear. Tsuda et al. (2002) showed that the idea of SIC still plays an important role even for non-linear estimation methods. The extension carried out in that paper allowed us to approximately apply SIC to sparse regression. However, the unbiasedness of SIC is no longer maintained. Therefore, further research is needed to extend SIC such that non-linear estimation methods can be dealt with in a theoretically rigorous fashion.

Finally, throughout this paper, we assumed that the target function belongs to a specified RKHS. In experiments with real data sets, we observed that SIC works properly even when the target function does not exactly lie in the specified RKHS (i.e., unrealizable case). Although this is surely a useful property in practice, it still remains open how to devise a method for optimally determining the appropriate RKHS, e.g., the kernel type and width.

## Acknowledgments

The authors would like to thank Dr. Akira Tanaka for readily showing us a preprint of his paper (Tanaka et al., 2002). It greatly motivated us to conduct the current research. Our special thanks go to Dr. Bernd Droge of the SFB 373 (Berlin) for valuable discussions, comments on the manuscript, and for pointing to some highly useful statistics references.

We thank Dr. Pavel Laskov for finding an error in the proof of Theorem 1 in the earlier version of the manuscript. Discussion with Dr. Gunnar Rätsch, Dr. Motoaki Kawanabe, Dr. Koji Tsuda, Dr. Hidemitsu Ogawa, and Dr. Shun-ichi Amari was very fruitful. We also would like to thank the anonymous reviewers for their valuable comments, that helped to improve the manuscript. M. S. thanks Fraunhofer FIRST for warm hospitality during his research stay in Berlin. M. S. also acknowledges partial financial support from MEXT, Grants-in-Aid for Scientific Research, 14380158 and 14780262. K.-R. M. acknowledges partial financial support from DFG under contracts JA 379/91 and MU 987/11 and the BMBF under contract FKZ 01IBB02A.

## Appendix A. Proof of Theorem 1

First, let us introduce the notion of the *Neumann-Schatten product* (Schatten, 1970). For any fixed $g$ in a Hilbert space $\mathcal{H}_1$ and any fixed $f$ in a Hilbert space $\mathcal{H}_2$, the Neumann-Schatten product of $f$ and $g$, denoted by $(f \otimes \overline{g})$, is an operator from $\mathcal{H}_1$ to $\mathcal{H}_2$ that satisfies for any $h$ in $\mathcal{H}_1$

$$(f \otimes \overline{g})\, h = \langle h, g \rangle f.$$

When $\mathcal{H}_1$ and $\mathcal{H}_2$ are both the Euclidean spaces, $(f \otimes \overline{g})$ is expressed as

$$(f \otimes \overline{g}) = f g^\top.$$

Using the above Neumann-Schatten product, let us define an operator $A$ from the RKHS $\mathcal{H}$ to $\mathbb{R}^\ell$ as follows.

$$A = \sum_{i=1}^{\ell} \left( \boldsymbol{e}_i^{(\ell)} \otimes \overline{K(\cdot, \boldsymbol{x}_i)} \right),$$

where $\boldsymbol{e}_i^{(\ell)}$ is the $i$-th standard basis in $\mathbb{R}^\ell$, i.e., it is the $\ell$-dimensional vector with the $i$-th element 1 and others 0. Note that the property of the reproducing kernel implies

$$Af = (f(\boldsymbol{x}_1), f(\boldsymbol{x}_2), \ldots, f(\boldsymbol{x}_\ell))^\top.$$

Let $A^*$ be the adjoint operator of $A$. Notice that $A^*$ is expressed as

$$A^* = \sum_{i=1}^{\ell} \left( K(\cdot, \boldsymbol{x}_i) \otimes \overline{\boldsymbol{e}_i^{(\ell)}} \right).$$

Let us assume that there exists a learning matrix $\boldsymbol{X}_u$ that gives an unbiased estimate $\hat{\boldsymbol{\alpha}}_u$ of the unknown $\boldsymbol{\alpha}^*$. Then, recalling Eq.(13), we have

$$\boldsymbol{\alpha}^* = \mathrm{E}_{\boldsymbol{\epsilon}} \hat{\boldsymbol{\alpha}}_u = \mathrm{E}_{\boldsymbol{\epsilon}} \boldsymbol{X}_u \boldsymbol{y} = \mathrm{E}_{\boldsymbol{\epsilon}} \boldsymbol{X}_u (Af + \boldsymbol{\epsilon}) = \boldsymbol{X}_u A f. \tag{31}$$

On the other hand, using the operator $B$ defined by

$$B = \sum_{i=1}^{p} \left( \boldsymbol{e}_i^{(p)} \otimes \overline{\varphi_i(\cdot)} \right),$$

it holds that

$$B^*\boldsymbol{\alpha}^* = \sum_{i=1}^{p} \langle \boldsymbol{\alpha}^*, \boldsymbol{e}_i^{(p)} \rangle \varphi_i(\cdot) = \sum_{i=1}^{p} \alpha_i^* \varphi_i(\cdot) = g(\cdot) = P_{\mathcal{S}} f, \tag{32}$$

where $P_{\mathcal{S}}$ denotes the orthogonal projection operator onto the subspace $\mathcal{S}$. From Eqs.(31) and (32), we have

$$B^* \boldsymbol{X}_u A f = P_{\mathcal{S}} f.$$

Since $f$ is not specified, $\boldsymbol{X}_u$ must satisfy the above equation for all $f$ in $\mathcal{H}$, i.e., it yields

$$B^* \boldsymbol{X}_u A = P_{\mathcal{S}}. \tag{33}$$

It is known that Eq.(33) has a solution if and only if the following condition holds (see e.g., Albert, 1972; Hunter, 2000):

$$B^*(B^*)^- P_{\mathcal{S}} A^- A = P_{\mathcal{S}}, \tag{34}$$

where $B^-$ is the so-called *equation solving generalized inverse* that satisfies $BB^-B = B$. Using $P_{\mathcal{S}} = B^*(B^*)^\dagger$, Eq.(34) is expressed as

$$\begin{aligned} P_{\mathcal{S}} &= B^*(B^*)^- B^*(B^*)^\dagger A^- A \\ &= B^*(B^*)^\dagger A^- A \\ &= P_{\mathcal{S}} A^- A. \end{aligned} \tag{35}$$

Since $P_{\mathcal{S}} = P_{\mathcal{S}}^*$, Eq.(35) is equivalent to

$$P_{\mathcal{S}} = A^*(A^-)^* P_{\mathcal{S}}. \tag{36}$$

Now we show Eq.(36) holds if and only if $\mathcal{S} \subset \mathcal{S}_K$. When Eq.(36) holds, $\mathcal{S} \subset \mathcal{S}_K$ is satisfied since the range of $A^*$ is equivalent to $\mathcal{S}_K$. Conversely, if $\mathcal{S} \subset \mathcal{S}_K$, it holds that

$$P_{\mathcal{S}} = P_{\mathcal{S}_K} P_{\mathcal{S}} = A^*(A^*)^\dagger P_{\mathcal{S}}.$$

Then we have

$$\begin{aligned} A^*(A^-)^* P_{\mathcal{S}} &= A^*(A^-)^* A^*(A^*)^\dagger P_{\mathcal{S}} \\ &= (AA^-A)^*(A^*)^\dagger P_{\mathcal{S}} \\ &= A^*(A^*)^\dagger P_{\mathcal{S}} \\ &= P_{\mathcal{S}}, \end{aligned}$$

which concludes the proof. ■

## Appendix B. Proof of Corollary 2

If Eq.(34) holds, a solution of Eq.(33) is given as follows (see e.g., Albert, 1972; Hunter, 2000):

$$\boldsymbol{X}_u = (B^*)^\dagger P_{\mathcal{S}} A^\dagger.$$

If we take $p = \ell$ and $\varphi_i(\boldsymbol{x}) = K(\boldsymbol{x}, \boldsymbol{x}_i)$ for $i = 1, 2, \ldots, \ell$, it holds that $B = A$ and $P_{\mathcal{S}} = P_{\mathcal{S}_K} = A^{\dagger}A$. Then we have

$$\boldsymbol{X}_u = (A^*)^{\dagger}A^{\dagger}AA^{\dagger} = (A^*)^{\dagger}A^{\dagger} = (AA^*)^{\dagger} = \boldsymbol{K}^{\dagger}, \tag{37}$$

where the last equation follows from

$$\begin{aligned} AA^* &= \sum_{i=1}^{\ell}\left(\boldsymbol{e}_i^{(\ell)} \otimes \overline{K(\cdot, \boldsymbol{x}_i)}\right)\sum_{j=1}^{\ell}\left(K(\cdot, \boldsymbol{x}_j) \otimes \overline{\boldsymbol{e}_j^{(\ell)}}\right) \\ &= \sum_{i,j=1}^{\ell}\langle K(\cdot, \boldsymbol{x}_j), K(\cdot, \boldsymbol{x}_i)\rangle_{\mathcal{H}}\left(\boldsymbol{e}_i^{(\ell)} \otimes \overline{\boldsymbol{e}_j^{(\ell)}}\right) \\ &= \sum_{i,j=1}^{\ell} K(\boldsymbol{x}_i, \boldsymbol{x}_j)\boldsymbol{e}_i^{(\ell)}\boldsymbol{e}_j^{(\ell)\top} \\ &= \boldsymbol{K}. \end{aligned}$$

Eq.(37) proves Corollary 2. ■

## References

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

H. Akaike. Likelihood and the Bayes procedure. In N. J. Bernardo, M. H. DeGroot, D. V. Lindley, and A. F. M. Smith, editors, *Bayesian Statistics*, pages 141–166, Valencia, 1980. University Press.

A. Albert. *Regression and the Moore-Penrose Pseudoinverse.* Academic Press, New York and London, 1972.

S. Amari, N. Fujita, and S. Shinomoto. Four types of learning curves. *Neural Computation*, 4(4):605–618, 1992.

N. Aronszajn. Theory of reproducing kernels. *Transactions of the American Mathematical Society*, 68:337–404, 1950.

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

O. Bunke and B. Droge. Bootstrap and cross-validation estimates of the prediction error for linear regression models. *Annals of Statistics*, 12:1400–1424, 1984.

C. J. C. Burges. A tutorial on support vector machines for pattern recognition. *Knowledge Discovery and Data Mining*, 2(2):121–167, 1998.

S. S. Chen, D. L. Donoho, and M. A. Saunders. Atomic decomposition by basis pursuit. *SIAM Journal on Scientific Computing*, 20(1):33–61, 1998.

V. Cherkassky, X. Shao, F. M. Mulier, and V. N. Vapnik. Model complexity control for regression using VC generalization bounds. *IEEE Transactions on Neural Networks*, 10 (5):1075–1089, 1999.

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.

N. Cristianini and J. Shawe-Taylor. *An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods*. Cambridge University Press, Cambridge, UK, 2000.

I. Daubechies. *Ten Lectures on Wavelets*. Society for Industrial and Applied Mathematics, Philadelphia and Pennsylvania, 1992.

L. Devroye, L. Györfi, and G. Lugosi. *A Probabilistic Theory of Pattern Recognition*. Number 31 in Applications of mathematics. Springer, New York, 1996.

D. L. Donoho. De-noising by soft thresholding. *IEEE Transactions on Information Theory*, 41(3):613–627, 1995.

D. L. Donoho and I. M. Johnstone. Ideal spatial adaptation via wavelet shirinkage. *Biometrika*, 81:425–455, 1994.

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.

S. Geman, E. Bienenstock, and R. Doursat. Neural networks and the bias/variance dilemma. *Neural Computation*, 4(1):1–58, 1992.

M. N. Gibbs. *Bayesian Gaussian Processes for Regression and Classification*. PhD thesis, Cambridge University, 1997.

M. N. Gibbs and D. J. C. MacKay. Efficient implementation of Gaussian processes, 1997. Available electronically at `ftp://www.inference.phy.cam.ac.uk/pub/www/mng10/GP/gpros.ps.gz`.

F. Girosi. An equivalence between sparse approximation and support vector machines. *Neural Computation*, 10(6):1455–1480, 1998.

R. E. Henkel. *Tests of Significance*. SAGE Publication, Beverly Hills, 1979.

T. Heskes. Bias/variance decompositions for likelihood-based estimators. *Neural Computation*, 10(6):1425–1433, 1998.

P. J. Huber. *Robust Statistics*. John Wiley, New York, 1981.

J. J. Hunter. A survey of generalized inverses and their use in stochastic modeling. *Research Letters in the Information and Mathematical Sciences*, 1(1):25–36, 2000.

A. K. Jain and R. C. Dubes. *Algorithms for Clustering Data.* Prentice Hall, 1988.

I. T. Jolliffe. *Principal Component Analysis.* Springer-Verlag, New York, 1986.

G. S. Kimeldorf and G. Wahba. A correspondence between Bayesan estimation on stochastic processes and smoothing by splines. *Annals of Mathematical Statistics*, 41(2):495–502, 1970.

T. Kohonen. *Self-Organizing Maps.* Springer, Berlin, 1995.

S. Konishi and G. Kitagawa. Generalized information criteria in model selection. *Biometrika*, 83:875–890, 1996.

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.

C. L. Mallows. Choosing a subset regression. *Presented at the Central Regional Meeting of the Institute of Mathematical Statistics*, 1964.

C. L. Mallows. Some comments on $C_P$. *Technometrics*, 15(4):661–675, 1973.

K.-R. Müller, S. Mika, G. Rätsch, K. Tsuda, and B. Schölkopf. An introduction to kernel-based learning algorithms. *IEEE Transactions on Neural Networks*, 12(2):181–201, 2001.

N. Murata, S. Yoshizawa, and S. Amari. Network information criterion — Determining the number of hidden units for an artificial neural network model. *IEEE Transactions on Neural Networks*, 5(6):865–872, 1994.

R. Nishii. Asymptotic properties of criteria for selection of variables in multiple regression. *Annals of Statistics*, 12:758–765, 1984.

M. J. L. Orr. Introduction to radial basis function networks. Technical report, Center for Cognitive Science, University of Edinburgh, 1996. Available electronically at `http://www.anc.ed.ac.uk/~mjo/papers/intro.ps.gz`.

C. E. Rasmussen, R. M. Neal, G. E. Hinton, D. van Camp, M. Revow, Z. Ghahramani, R. Kustra, and R. Tibshirani. The DELVE manual, 1996. Available electronically at `http://www.cs.toronto.edu/~delve/`.

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

J. Rissanen. Stochastic complexity. *Journal of the Royal Statistical Society, Series B*, 49 (3):223–239, 1987.

J. Rissanen. Fisher information and stochastic complexity. *IEEE Transactions on Information Theory*, IT-42(1):40–47, 1996.

C. Saunders, A. Gammerman, and V. Vovk. Ridge regression learning algorithm in dual variables. In *Proceedings of the 15th International Conference on Machine Learning*, pages 515–521, 1998.

R. Schatten. *Norm Ideals of Completely Continuous Operators*. Springer-Verlag, Berlin, 1970.

B. Schölkopf, C. J. C. Burges, and A. J. Smola, editors. *Advances in Kernel Methods: Support Vector Machines*. The MIT Press, Cambridge, MA, 1998.

B. Schölkopf, A. Smola, R. Williamson, and P. Bartlett. New support vector algorithms. *Neural Computation*, 12(5):1207–1245, 2000.

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

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

R. Shibata. An optimal selection of regression variables. *Biometrika*, 68(1):45–54, 1981.

A. J. Smola and B. Schölkopf. Sparse greedy matrix approximation for machine learning. In *Proceedings of ICML-2000, International Conference on Machine Learning*, pages 911–918, 2000.

A. J. Smola, B. Schölkopf, and K.-R. Müller. The connection between regularization operators and support vector kernels. *Neural Networks*, 11(4):637–649, 1998.

N. Sugiura. Further analysis of the data by Akaike's information criterion and the finite corrections. *Communications in Statistics: Theory and Methods*, 7(1):13–26, 1978.

M. Sugiyama, D. Imaizumi, and H. Ogawa. Subspace information criterion for image restoration — Optimizing parameters in linear filters. *IEICE Transactions on Information and Systems*, E84-D(9):1249–1256, 2001.

M. Sugiyama and H. Ogawa. Incremental active learning for optimal generalization. *Neural Computation*, 12(12):2909–2940, 2000.

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, 2002a.

M. Sugiyama and H. Ogawa. Theoretical and experimental evaluation of subspace information criterion. *Machine Learning, Special Issue on New Methods for Model Selection and Model Combination*, 48(1/2/3):25–50, 2002b.

M. Sugiyama and H. Ogawa. A unified method for optimizing linear image restoration filters. *Signal Processing*, 82(11):1773–1787, 2002c.

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

A. Tanaka, H. Imai, and M. Miyakoshi. Choosing the parameter of image restoration filters by modified subspace information criterion. *IEICE Transactions on Fundamentals*, E85-A (5):1104–1110, 2002.

R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society, Series B*, 58(1):267–288, 1996.

K. Tsuda, M. Sugiyama, and K.-R. Müller. Subspace information criterion for non-quadratic regularizers — Model selection for sparse regressors. *IEEE Transactions on Neural Networks*, 13(1):70–80, 2002.

V. Vapnik and O. Chapelle. Bounds on error expectation for support vector machines. *Neural Computation*, 12(9):2013–2036, 2000.

V. N. Vapnik. *Estimation of Dependencies Based on Empirical Data.* Springer-Verlag, New York, 1982.

V. N. Vapnik. *The Nature of Statistical Learning Theory.* Springer-Verlag, Berlin, 1995.

V. N. Vapnik. *Statistical Learning Theory.* John Wiley & Sons, Inc., New York, 1998.

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

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

C. K. I. Williams. Prediction with Gaussian processes: From linear regression to linear prediction and beyond. In M. I. Jordan, editor, *Learning in Graphical Models*, pages 599–621. The MIT Press, Cambridge, 1998.

C. K. I. Williams and C. E. Rasmussen. Gaussian processes for regression. In D. S. Touretzky, M. C. Mozer, and M. E. Hasselmo, editors, *Advances in Neural Information Processing Systems*, volume 8, pages 514–520. The MIT Press, 1996.

P. M. Williams. Bayesian regularization and pruning using a Laplace prior. *Neural Computation*, 7(1):117–143, 1995.

K. Yamanishi. A decision-theoretic extension of stochastic complexity and its application to learning. *IEEE Transactions on Information Theory*, IT-44(4):1424–1439, 1998.