Estimating the Error at Given Test Input Points for Linear Regression

Masashi Sugiyama

Fraunhofer FIRST
Intelligent Data Analysis Group
Kekuléstr. 7, 12489, Berlin, Germany
and
Department of Computer Science
Tokyo Institute of Technology
2-12-1, O-okayama, Meguro-ku, Tokyo, 152-8552, Japan

email: sugi@cs.titech.ac.jp

ABSTRACT
In model selection procedures in supervised learning, a model is usually chosen so that the expected test error over all possible test input points is minimized. On the other hand, when the test *input* points (without *output* values) are available in advance, it is more effetive to choose a model so that the test error only at the test input points at hand is minimized. In this paper, we follow this idea and derive an estimator of the test error at the given test input points for linear regression. Our estimator is proved to be an unbiased estimator of the test error at the given test input points under certain conditions. Through the simulations with artificial and standard benchmark data sets, we show that the proposed method is successfully applied in test error estimation and is compared favorably to the standard cross-validation and an empirical Bayesian method in ridge parameter selection.

KEY WORDS
Machine Learning, Supervised Learning, Test Error, Expected Test Error, Model Selection, Ridge Regression

1 Introduction

Model selection in supervised learning is usually performed as follows [1, 18, 16, 20, 12, 10, 17]. First, an estimator of the expected test error over all possible test input points (which is often called the generalization error) is derived. Then a model is chosen so that the estimator of the expected test error is minimized. On the other hand, when the test *input* points (without output values) are available in advance, it is natural and more effective to choose the model so that the test error only at the test input points at hand is minimized.

In this paper, we follow this idea and shall derive an estimator of the test error at the given test input points for linear regression. We prove that this estimator is an *unbiased* estimator of the test error at the given test input points under certain conditions.

In experiments, we apply the proposed test error estimator to simple artificial and standard benchmark data sets. The simulation results show that the proposed method can successfully estimate the test error and is compared favorably to the standard cross-validation and an empirical Bayesian method when it is used for the ridge parameter selection.

2 Problem Formulation

In this section, we formulate the problem of estimating the values of a target function at given test input points.

Let us denote the learning target function by $f(\boldsymbol{x})$, which is a real-valued function of d variables defined on the domain $\mathcal{D}$ ($\subset \mathbb{R}^d$). We are given a set of n samples called the *training examples*. A training example consists of a *sample point* $\boldsymbol{x}_i$ in $\mathcal{D}$ and a *sample value* y_i in $\mathbb{R}$. The sample value y_i is degraded by unknown additive noise ϵ_i with mean zero and unknown common variance σ^2.

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

In many learning theories, $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$ are assumed to be drawn independently from a joint probability density function $p(\boldsymbol{x}, y)$ [20, 12, 19, 15], i.e., both the sample points $\{\boldsymbol{x}_i\}_{i=1}^n$ and the noise $\{\epsilon_i\}_{i=1}^n$ are treated as random variables. In contrast, in this paper, we do not treat the sample points $\{\boldsymbol{x}_i\}_{i=1}^n$ as random but we treat them as fixed. We only regard the noise $\{\epsilon_i\}_{i=1}^n$ as random. This may be a key of the following discussion.

We employ the following linear regression model for learning.

$$\hat{f}(\boldsymbol{x}) = \sum_{i=1}^{p} \alpha_i \varphi_i(\boldsymbol{x}), \tag{2}$$

where $\{\alpha_i\}_{i=1}^p$ are parameters to be estimated from training examples and $\{\varphi_i(\boldsymbol{x})\}_{i=1}^p$ are the fixed linearly independent basis functions. Let $\{\hat{\alpha}_i\}_{i=1}^p$ be a linear estimator,

i.e., letting

$$\boldsymbol{y} = (y_1, y_2, \ldots, y_n)^\top, \tag{3}$$

$$\hat{\boldsymbol{\alpha}} = (\hat{\alpha}_1, \hat{\alpha}_2, \ldots, \hat{\alpha}_n)^\top, \tag{4}$$

where $^\top$ denotes the transpose of a vector (or a matrix), we estimate the parameter vector $\hat{\boldsymbol{\alpha}}$ by

$$\hat{\boldsymbol{\alpha}} = \boldsymbol{X}\boldsymbol{y}, \tag{5}$$

where $\boldsymbol{X}$ is an n-dimensional matrix that does not depend on the noise $\{\epsilon_i\}_{i=1}^n$. The matrix $\boldsymbol{X}$, which we call the *learning matrix*, can be any matrix but it usually depends on the training sample points $\{\boldsymbol{x}_i\}_{i=1}^n$. A popular choice of $\boldsymbol{X}$ is the ridge estimation [8] given by

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

where λ is a positive scalar called the *ridge parameter*, $\boldsymbol{I}$ denotes the identity matrix, and $\boldsymbol{A}$ is the so-called *design matrix* whose (i,j)-th element is given by

$$\boldsymbol{A}_{i,j} = \varphi_j(\boldsymbol{x}_i). \tag{7}$$

We assume that the design matrix $\boldsymbol{A}$ has rank p and $n > p$.

We are given a set $\{\boldsymbol{t}_i\}_{i=1}^{n_{\boldsymbol{t}}}$ of $n_{\boldsymbol{t}}$ test input points, where $\boldsymbol{t}_i \in \mathcal{D}$. Note that we also treat the test input points $\{\boldsymbol{t}_i\}_{i=1}^{n_{\boldsymbol{t}}}$ as fixed variables. The goal of learning is to accurately estimate $\{f(\boldsymbol{t}_i)\}_{i=1}^{n_{\boldsymbol{t}}}$, which are unknown output values of the target function $f(\boldsymbol{x})$ at the test input points $\{\boldsymbol{t}_i\}_{i=1}^{n_{\boldsymbol{t}}}$. Let us define the expected sum of squared test errors over the noise $\{\epsilon_i\}_{i=1}^n$ by

$$\mathrm{E}_{\boldsymbol{\epsilon}} \sum_{i=1}^{n_{\boldsymbol{t}}} \left(\hat{f}(\boldsymbol{t}_i) - f(\boldsymbol{t}_i) \right)^2, \tag{8}$$

where $\mathrm{E}_{\boldsymbol{\epsilon}}$ denotes the expectation over the noise $\{\epsilon_i\}_{i=1}^n$. For making the following discussions simple, let us define the following quantity.

$$J_{\boldsymbol{t}}[\boldsymbol{X}] = \mathrm{E}_{\boldsymbol{\epsilon}} \sum_{i=1}^{n_{\boldsymbol{t}}} \left(\hat{f}(\boldsymbol{t}_i) - f(\boldsymbol{t}_i) \right)^2 - \sum_{i=1}^{n_{\boldsymbol{t}}} f(\boldsymbol{t}_i)^2, \tag{9}$$

where the second term $\sum_{i=1}^{n_{\boldsymbol{t}}} f(\boldsymbol{t}_i)^2$ is a constant because the learning target function $f(\boldsymbol{x})$ and the test input points $\{\boldsymbol{t}_i\}_{i=1}^{n_{\boldsymbol{t}}}$ are fixed. Therefore, Eq.(9) is essentially the same as the test error given in Eq.(8). From here on, we call Eq.(9) the *test error*. We denote $J_{\boldsymbol{t}}$ as a functional of the learning matrix $\boldsymbol{X}$ since under the above setting, specifying the learned function $\hat{f}$ is equivalent to specifying the learning matrix $\boldsymbol{X}$. In the following, we often omit $\boldsymbol{X}$.

The aim of this paper is to derive an estimator of the above test error (9). In the derivation of the estimator, we assume that the target function $f(\boldsymbol{x})$ is included in the model (2), i.e., $f(\boldsymbol{x})$ is expressed by

$$f(\boldsymbol{x}) = \sum_{i=1}^{p} \alpha_i^* \varphi_i(\boldsymbol{x}), \tag{10}$$

where $\{\alpha_i^*\}_{i=1}^p$ are the unknown true parameters. We should admit that this assumption is rather restrictive. However, we expect that this assumption does not have to be rigorously fulfilled in practice because the proposed method is experimentally shown to work well without the above assumption (see Section 4).

3 Unbiased Estimator of Test Error

In this section, we derive an estimator of the test error (9).

In many model selection methods proposed so far [20, 12, 19, 15], the expected test error over all possible test samples (which is often referred to as the generalization error) is estimated based on the empirical error (or the training error) because the empirical error converges to the expected test error in the large sample limit [12]. However, the empirical error may not converge to the error at the given test input points. Therefore, in this paper, we do not resort to the empirical error, but we shall directly estimate the test error (9).

Let $\boldsymbol{A}_{\boldsymbol{t}}$ be the design matrix for the test input points $\{\boldsymbol{t}_i\}_{i=1}^{n_{\boldsymbol{t}}}$, i.e., the (i,j)-th element of $\boldsymbol{A}_{\boldsymbol{t}}$ is given by

$$[\boldsymbol{A}_{\boldsymbol{t}}]_{i,j} = \varphi_j(\boldsymbol{t}_i). \tag{11}$$

Let $\langle \cdot, \cdot \rangle$ and $\| \cdot \|$ be the inner product and the norm, respectively. Let $\mathrm{tr}\,(\cdot)$ be the trace of a matrix, and let † be the Moore-Penrose generalized inverse of a matrix [3, 9]. Then we have the following theorem.

Theorem 1 *For any learning target function $f(\boldsymbol{x})$ of the form (10), any training sample points $\{\boldsymbol{x}_i\}_{i=1}^n$, any test input points $\{\boldsymbol{t}_i\}_{i=1}^{n_{\boldsymbol{t}}}$, and any learning matrix $\boldsymbol{X}$, it holds that*

$$J_{\boldsymbol{t}}[\boldsymbol{X}] = \mathrm{E}_{\boldsymbol{\epsilon}} \Big[\|\boldsymbol{A}_{\boldsymbol{t}}\boldsymbol{X}\boldsymbol{y}\|^2 - 2\langle \boldsymbol{A}_{\boldsymbol{t}}\boldsymbol{X}\boldsymbol{y}, \boldsymbol{A}_{\boldsymbol{t}}\boldsymbol{A}^\dagger \boldsymbol{y} \rangle + 2\hat{\sigma}^2 \mathrm{tr}\left(\boldsymbol{A}_{\boldsymbol{t}}\boldsymbol{X}(\boldsymbol{A}_{\boldsymbol{t}}\boldsymbol{A}^\dagger)^\top \right) \Big], \tag{12}$$

where

$$\hat{\sigma}^2 = \frac{\|\boldsymbol{A}\boldsymbol{A}^\dagger \boldsymbol{y} - \boldsymbol{y}\|^2}{n-p}. \tag{13}$$

A proof of the above theorem is given in Appendix A. Let us call the quantity inside the bracket of Eq.(12) an Unbiased Points-of-interest-error Estimator (UPE):

$$\mathrm{UPE}[\boldsymbol{X}] = \|\boldsymbol{A}_{\boldsymbol{t}}\boldsymbol{X}\boldsymbol{y}\|^2 - 2\langle \boldsymbol{A}_{\boldsymbol{t}}\boldsymbol{X}\boldsymbol{y}, \boldsymbol{A}_{\boldsymbol{t}}\boldsymbol{A}^\dagger \boldsymbol{y} \rangle + 2\hat{\sigma}^2 \mathrm{tr}\left(\boldsymbol{A}_{\boldsymbol{t}}\boldsymbol{X}(\boldsymbol{A}_{\boldsymbol{t}}\boldsymbol{A}^\dagger)^\top \right). \tag{14}$$

Then Theorem 1 shows that the above UPE is an unbiased estimator of the test error over the noise $\{\epsilon_i\}_{i=1}^n$:

$$\mathrm{E}_{\boldsymbol{\epsilon}} \mathrm{UPE} = J_{\boldsymbol{t}}. \tag{15}$$

Note that the above $\hat{\sigma}^2$ is an unbiased estimator of the noise variance σ^2. In practice, another efficient noise variance estimator (see e.g., [4, 20, 21]) may also be employed in Eq.(14). However, in the following, we only use the unbiased estimator (13) for simplicity.

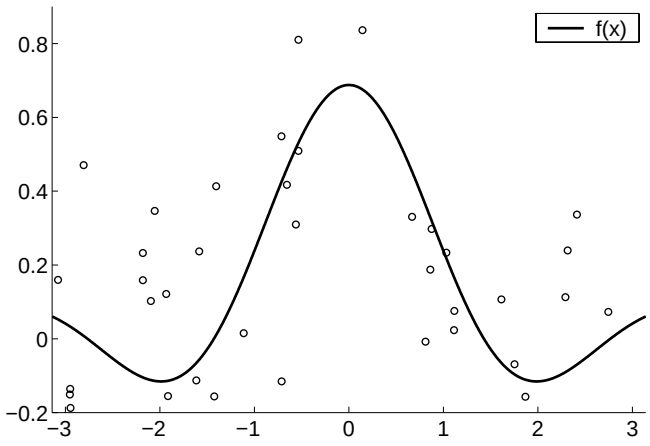

Figure 1. Target function $f(x)$.

4 Simulations

In this section, we empirically evaluate the performance of the proposed test error estimator UPE.

4.1 Artificial Illustrative Data Sets

First, we perform a simple artificial simulation for just illustrating how the proposed test error estimator works.

Let the dimension d of the input vector be 1. We use the linear regression model (2) for learning. Let the number p of basis functions be 10, and let the basis functions $\{\varphi_i(x)\}_{i=1}^{10}$ be

$$\varphi_i(x) = \exp\left(-\frac{(x-s_i)^2}{2}\right), \tag{16}$$

where $\{s_i\}_{i=1}^{10}$ are the template points located at regular intervals in $[-\pi, \pi]$. We use the sinc-like function depicted in Figure 1 as the target function $f(x)$, which is created by the least-squares estimation using the above basis functions $\{\varphi_i(x)\}_{i=1}^{10}$ and the samples taken from the sinc function: $\{(s_i, \operatorname{sinc} s_i)\}_{i=1}^{10}$. Note that the above $f(x)$ is included in our regression model.

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

$$(n, \sigma^2) = (100, 0.01), (50, 0.01), (100, 0.09), (50, 0.09). \tag{17}$$

Let the number $n_{\boldsymbol{t}}$ of test input points be 50, and the test input points $\{t_i\}_{i=1}^{50}$ are also independently drawn from the uniform distribution on $(-\pi, \pi)$.

The simulations are repeated 100 times for each (n, σ^2) in Eq.(17), randomly drawing the sample points $\{x_i\}_{i=1}^{n}$, noise $\{\epsilon_i\}_{i=1}^{n}$, and test input points $\{t_i\}_{i=1}^{50}$ from scratch in each trial. Note that in theory, we fixed the training sample points $\{x_i\}_{i=1}^{n}$ and the test input points $\{t_i\}_{i=1}^{50}$, and we only changed the noise $\{\epsilon_i\}_{i=1}^{n}$ (see Section 2). However, in this experiment, we also change the training sample points $\{x_i\}_{i=1}^{n}$ and test input points $\{t_i\}_{i=1}^{50}$ because we want to investigate whether the proposed method works irrespective of the choice of the training and test sets. For this reason, we measure the performance by the following single-trial test error in simulations.

$$J = \sum_{i=1}^{n_{\boldsymbol{t}}} \left(\hat{f}(\boldsymbol{t}_i) - f(\boldsymbol{t}_i)\right)^2 - \sum_{i=1}^{n_{\boldsymbol{t}}} f(\boldsymbol{t}_i)^2. \tag{18}$$

The parameters $\{\alpha_i\}_{i=1}^{10}$ in the regression model are determined by the ridge estimation, i.e., the learning matrix $\boldsymbol{X}$ is given by Eq.(6). The performance of the proposed test error estimator UPE is investigated as a function of the ridge parameter λ using the following values:

$$\lambda \in \{10^{-6}, 10^{-5}, 10^{-4}, \ldots, 10^{2}\}. \tag{19}$$

Figure 2 depicts the values of J (top) and UPE (bottom) as a function of the ridge parameter λ in Eq.(19). The horizontal axis denotes the values of λ in log-scale. In order to clearly compare the mean curves, the mean of J is also drawn in the bottom graph by the dashed line. Note that the values are negative since a positive constant is subtracted (see Eq.(18)). The graphs show that for all 4 cases in Eq.(17), the proposed estimator gives reasonably accurate estimates of the single-trial test error J.

We also attempted similar simulations with different basis functions and different target functions. The results are almost identical to the above case, so we omit the graphs.

4.2 DELVE Data Sets

For the above simple artificial data sets, we found that the proposed test error estimator is reasonably accurate. Here we apply the proposed estimator to standard benchmark data sets, and evaluate whether this good property can be carried over to practical problems. We will use 5 practical data sets provided by DELVE [14]: *Boston*, *Bank-8fm*, *Bank-8nm* *Kin-8fm*, and *Kin-8nm*. The *Boston* data set includes 506 samples with 13-dimensional input and 1-dimensional output data. The other data sets include 8192 samples which consist of 8-dimensional input and 1-dimensional output data.

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 50 randomly selected samples $\{(\boldsymbol{t}_i, u_i)\}_{i=1}^{50}$ from the rest are used for testing. Let the number p of basis functions be 50, and let the basis functions $\{\varphi_i(\boldsymbol{x})\}_{i=1}^{50}$ be

$$\varphi_i(\boldsymbol{x}) = \exp\left(-\frac{\|\boldsymbol{x}-\boldsymbol{x}_i\|^2}{2}\right), \tag{20}$$

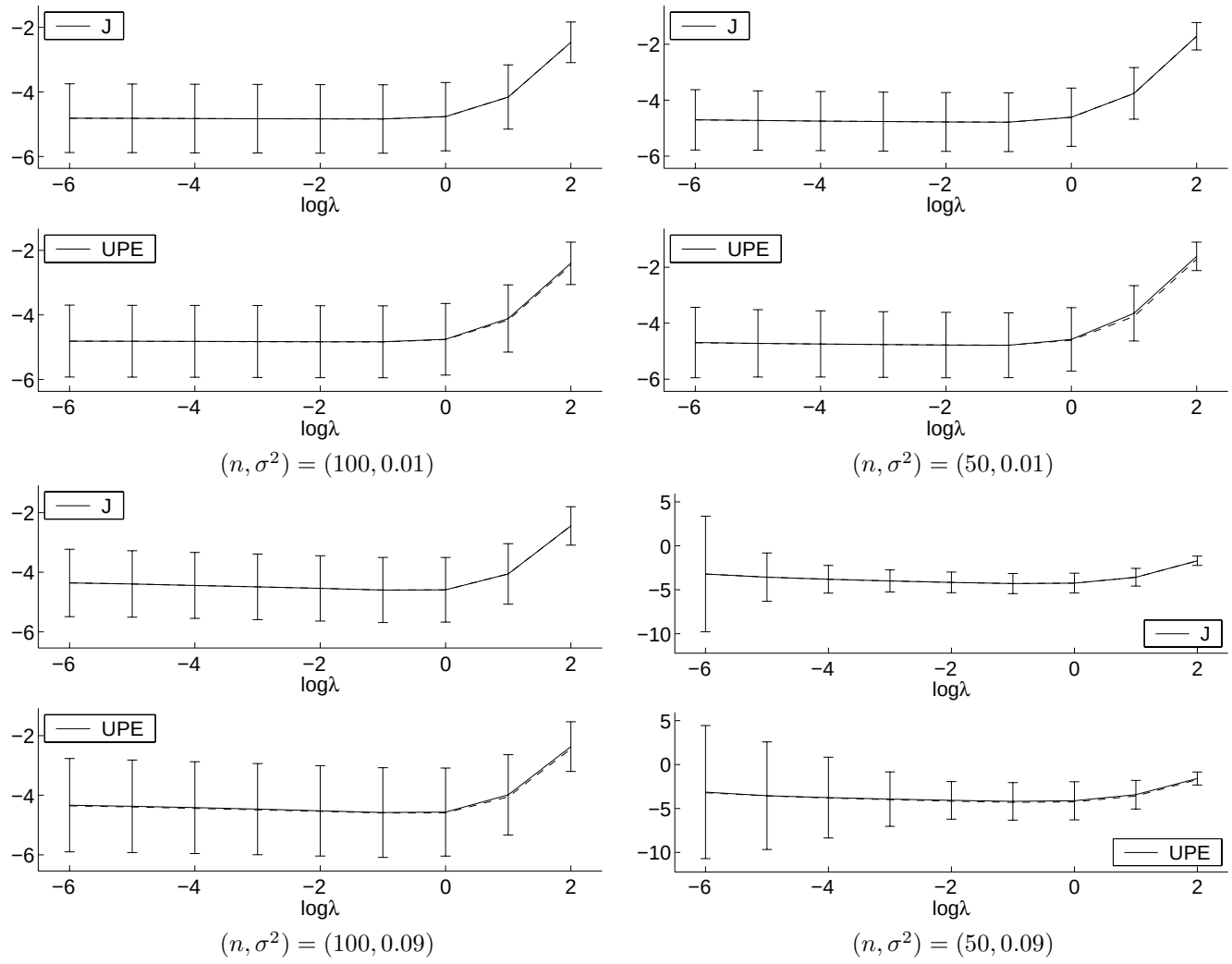

Figure 2. Values of the single-trial test error J (top) and UPE (bottom) as a function of the ridge parameter λ. The horizontal axis denotes the values of λ in log-scale. In order to clearly compare the mean curves, the mean of J is also drawn in the bottom graph by the dashed line. Note that the values are negative since a positive constant is subtracted (see Eq.(18)).

i.e., the first 50 samples $\{\boldsymbol{x}_i\}_{i=1}^{50}$ are used as the template points. The ridge estimation, whose learning matrix is given by Eq.(6), is again used for learning. We determine the value of the ridge parameter λ by the proposed UPE, the standard leave-one-out cross-validation (LOOCV) [20, 13], or an empirical Bayesian method (EB) [2]. Note that LOOCV gives an almost unbiased estimate of the expected test error [11, 15]. The ridge parameter λ is chosen from the following values:

$$\lambda \in \{10^{-7}, 10^{-6}, 10^{-5}, \ldots, 10^{7}\}. \tag{21}$$

The simulation is repeated 1000 times for each data set, randomly selecting the training samples $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^{100}$ and the test samples $\{(\boldsymbol{t}_i, u_i)\}_{i=1}^{50}$ from scratch in each trial (i.e., sampling without replacement). In this simulation, we evaluate the performance of each method by the following single-trial test error.

$$J = \sum_{i=1}^{50} \left(\hat{f}(\boldsymbol{t}_i) - u_i \right)^2. \tag{22}$$

Note that we do not subtract $\sum_{i=1}^{50} u_i^2$ from the above error as did in Eq.(18) because we later normalize the values of the test error.

The single-trial test errors obtained after model selection with UPE, LOOCV, or EB are summarized in Table 1. The table describes the mean and standard deviation of the normalized test error, where the values of the test error are normalized so that the mean test error obtained by the optimal ridge parameter is one. The results of the best method and all other methods with significant difference (99% t-test [7]) are described in bold face. The table shows that the proposed method is compared favorably to other methods.

Table 1. Mean and standard deviation of the normalized test error for the proposed UPE, the leave-one-out cross-validation (LOOCV), and the empirical Bayesian method (EB). The results of the best method and all other methods with no significant difference (99% t-test) are described in bold face.

Data Set	UPE	LOOCV	EB
Boston	$\mathbf{1.17 \pm 0.54}$	1.26 ± 0.58	1.39 ± 0.59
Bank-8fm	$\mathbf{1.07 \pm 0.29}$	1.11 ± 0.32	$\mathbf{1.09 \pm 0.31}$
Bank-8nm	$\mathbf{1.09 \pm 0.51}$	$\mathbf{1.12 \pm 0.56}$	1.18 ± 0.60
Kin-8fm	$\mathbf{1.06 \pm 0.32}$	1.17 ± 0.36	1.68 ± 0.48
Kin-8nm	$\mathbf{1.11 \pm 0.27}$	$\mathbf{1.09 \pm 0.24}$	1.15 ± 0.24

5 Conclusions and Future Prospects

In this paper, we derived an unbiased estimator of the test error, and experimentally showed that this estimator can be successfully applied to the ridge parameter selection. In theory, we assumed that the learning target function is included in the model. Although this assumption is rather restrictive, experimental results show that this assumption does not have to be satisfied rigorously in practice. How the slight violation of the assumption affects the accuracy of the test error estimator should be elucidated in the future.

An interesting related topic in this line of research is to directly estimate the output values at the test input points at hand, which is referred to as the *transductive inference* [19]. Although an interesting transductive inference method has been proposed recently [5], its performance heavily depends on a certain tuning parameter, which should be chosen by hand. On the other hand, the proposed test error estimator is applicable to any linear estimations. Our prospecting and challenging future work is to obtain a new linear estimation method that is suitable for the challenging scenario of transductive inference.

Acknowledgements

The author would like to thank Koji Tsuda and Bernhard Schölkopf for their comments when he visited Max Planck Institute for Biological Cybernetics. The discussions with them motivated him to write the current paper. Special thanks also go to Klaus-Robert Müller for his useful comments. The author also acknowledges the Alexander von Humboldt Foundation for partial financial support.

A Proof of Theorem 1

Let

$$\boldsymbol{\alpha}^* = (\alpha_1^*, \alpha_2^*, \ldots, \alpha_p^*)^\top, \tag{23}$$

$$\boldsymbol{z} = (f(\boldsymbol{x}_1), f(\boldsymbol{x}_2), \ldots, f(\boldsymbol{x}_n))^\top, \tag{24}$$

$$\boldsymbol{z_t} = (f(\boldsymbol{t}_1), f(\boldsymbol{t}_2), \ldots, f(\boldsymbol{t}_{n_t}))^\top, \tag{25}$$

$$\hat{\boldsymbol{z}}_{\boldsymbol{t}} = (\hat{f}(\boldsymbol{t}_1), \hat{f}(\boldsymbol{t}_2), \ldots, \hat{f}(\boldsymbol{t}_{n_t}))^\top, \tag{26}$$

$$\boldsymbol{\epsilon} = (\epsilon_1, \epsilon_2, \ldots, \epsilon_n)^\top. \tag{27}$$

Since $\boldsymbol{z_t}$ and $\hat{\boldsymbol{z}}_{\boldsymbol{t}}$ are given by

$$\boldsymbol{z_t} = \boldsymbol{A_t}\boldsymbol{\alpha}^*, \tag{28}$$

$$\hat{\boldsymbol{z}}_{\boldsymbol{t}} = \boldsymbol{A_t}\hat{\boldsymbol{\alpha}} = \boldsymbol{A_t}\boldsymbol{X}\boldsymbol{y}, \tag{29}$$

we have

$$\begin{aligned} J_{\boldsymbol{t}} &= \mathrm{E}_{\boldsymbol{\epsilon}}\|\hat{\boldsymbol{z}}_{\boldsymbol{t}} - \boldsymbol{z_t}\|^2 - \|\boldsymbol{z_t}\|^2 \\ &= \mathrm{E}_{\boldsymbol{\epsilon}}\|\hat{\boldsymbol{z}}_{\boldsymbol{t}}\|^2 - 2\mathrm{E}_{\boldsymbol{\epsilon}}\langle\hat{\boldsymbol{z}}_{\boldsymbol{t}}, \boldsymbol{z_t}\rangle \\ &= \mathrm{E}_{\boldsymbol{\epsilon}}\|\boldsymbol{A_t}\boldsymbol{X}\boldsymbol{y}\|^2 - 2\mathrm{E}_{\boldsymbol{\epsilon}}\langle\boldsymbol{A_t}\boldsymbol{X}\boldsymbol{y}, \boldsymbol{A_t}\boldsymbol{\alpha}^*\rangle. \end{aligned} \tag{30}$$

On the other hand, it holds that

$$\boldsymbol{z} = \boldsymbol{A}\boldsymbol{\alpha}^*, \tag{31}$$

so we have

$$\boldsymbol{A}^\dagger\boldsymbol{z} = \boldsymbol{A}^\dagger\boldsymbol{A}\boldsymbol{\alpha}^* = \boldsymbol{\alpha}^*. \tag{32}$$

Therefore, we have

$$\begin{aligned} J_{\boldsymbol{t}} &= \mathrm{E}_{\boldsymbol{\epsilon}}\|\boldsymbol{A_t}\boldsymbol{X}\boldsymbol{y}\|^2 - 2\mathrm{E}_{\boldsymbol{\epsilon}}\langle\boldsymbol{A_t}\boldsymbol{X}\boldsymbol{y}, \boldsymbol{A_t}\boldsymbol{A}^\dagger\boldsymbol{z}\rangle \\ &= \mathrm{E}_{\boldsymbol{\epsilon}}\|\boldsymbol{A_t}\boldsymbol{X}\boldsymbol{y}\|^2 - 2\mathrm{E}_{\boldsymbol{\epsilon}}\langle\boldsymbol{A_t}\boldsymbol{X}\boldsymbol{y}, \boldsymbol{A_t}\boldsymbol{A}^\dagger\boldsymbol{y}\rangle \\ &\quad +2\mathrm{E}_{\boldsymbol{\epsilon}}\langle\boldsymbol{A_t}\boldsymbol{X}\boldsymbol{y}, \boldsymbol{A_t}\boldsymbol{A}^\dagger\boldsymbol{\epsilon}\rangle \end{aligned} \tag{33}$$

Since it is known that $\hat{\sigma}^2$ given by Eq.(13) is an unbiased estimator of σ^2 [6], the last term in Eq.(33) yields

$$\begin{aligned} &2\mathrm{E}_{\boldsymbol{\epsilon}}\langle\boldsymbol{A_t}\boldsymbol{X}\boldsymbol{y}, \boldsymbol{A_t}\boldsymbol{A}^\dagger\boldsymbol{\epsilon}\rangle \\ &\quad= 2\mathrm{E}_{\boldsymbol{\epsilon}}\langle\boldsymbol{A_t}\boldsymbol{X}\boldsymbol{z}, \boldsymbol{A_t}\boldsymbol{A}^\dagger\boldsymbol{\epsilon}\rangle + 2\mathrm{E}_{\boldsymbol{\epsilon}}\langle\boldsymbol{A_t}\boldsymbol{X}\boldsymbol{\epsilon}, \boldsymbol{A_t}\boldsymbol{A}^\dagger\boldsymbol{\epsilon}\rangle \\ &\quad= 2\sigma^2\mathrm{tr}\left(\boldsymbol{A_t}\boldsymbol{X}(\boldsymbol{A_t}\boldsymbol{A}^\dagger)^\top\right) \\ &\quad= 2\mathrm{E}_{\boldsymbol{\epsilon}}\hat{\sigma}^2\mathrm{tr}\left(\boldsymbol{A_t}\boldsymbol{X}(\boldsymbol{A_t}\boldsymbol{A}^\dagger)^\top\right), \end{aligned} \tag{34}$$

which concludes the proof. ■

References

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

[2] 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.

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

[4] A. Buja, T. Hastie, and R. Tibshirani. Linear smoothers and additive models. *The Annals of Statistics*, 17:453–555, 1989.

[5] O. Chapelle, V. N. Vapnik, and J. Weston. Transductive inference for estimating values of functions. In S. A. Solla, T. K. Leen, and K.-R. Müller, editors, *Advances in Neural Information Processing Systems 12*, pages 421–427. MIT Press, 2000.

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

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

[8] A. E. Hoerl and R. W. Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(3):55–67, 1970.

[9] 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.

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

[11] A. Luntz and V. Brailovsky. On estimation of characters obtained in statistical procedure of recognition. *Technicheskaya Kibernetica*, 3, 1969. in Russian.

[12] 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.

[13] M. J. L. Orr. Introduction to radial basis function networks. Technical report, Center for Cognitive Science, University of Edinburgh, 1996.

[14] 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.

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

[16] 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.

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

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

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

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

[21] Z. Zhang. Parameter estimation techniques: A tutorial with application to conic fitting. *Image and Vision Computing*, 15(1):59–76, 1997.