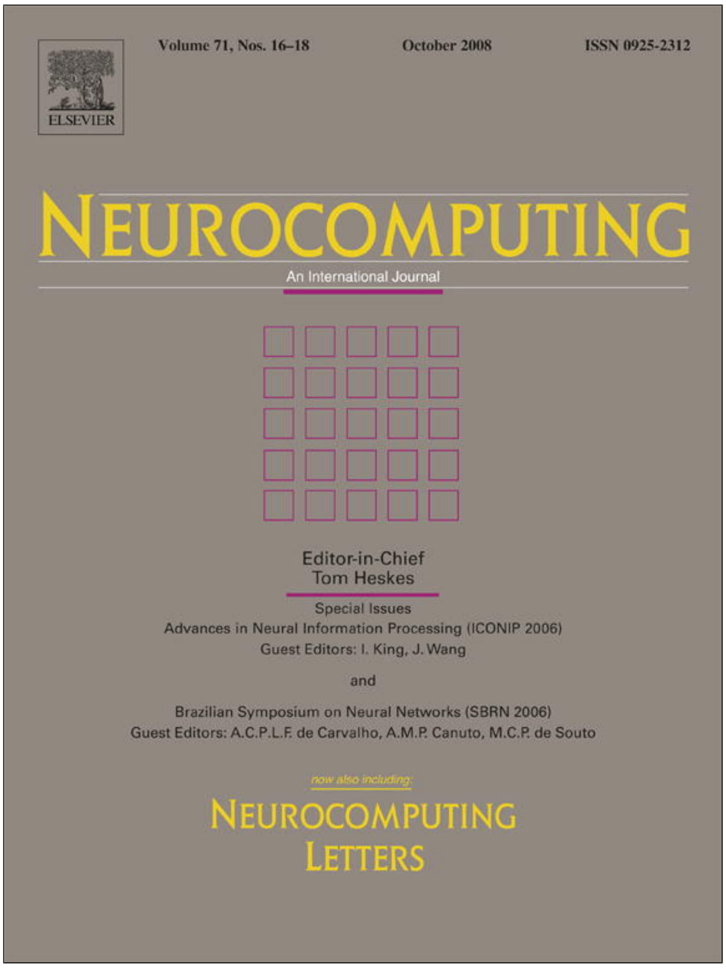

Volume 71, Nos. 16–18 October 2008 ISSN 0925-2312

ELSEVIER

NEUROCOMPUTING
An International Journal

Editor-in-Chief
Tom Heskes

Special Issues
Advances in Neural Information Processing (ICONIP 2006)
Guest Editors: I. King, J. Wang

and

Brazilian Symposium on Neural Networks (SBRN 2006)
Guest Editors: A.C.P.L.F. de Carvalho, A.M.P. Canuto, M.C.P. de Souto

now also including:
NEUROCOMPUTING
LETTERS

Neurocomputing 71 (2008) 3089–3095

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Parameterized cross-validation for nonlinear regression models

Imhoi Koo, Namgil Lee, Rhee Man Kil *

Department of Mathematical Sciences, Korea Advanced Institute of Science and Technology, 373-1 Guseong-dong, Yuseong-gu, Daejeon 305-701, Republic of Korea

ARTICLE INFO

Available online 28 June 2008

Keywords:
Cross-validation
Generalization
Regression
Optimization
VC dimension

ABSTRACT

This paper presents a new method of cross-validation (CV) for nonlinear regression problems. In the conventional CV methods, a validation set, that is, a part of training data is used to check the performance of learning. As a result, the trained regression models cannot utilize the whole training data and obtain the less performance than the expected for the given training data. In this context, we consider to construct the performance prediction model using the validation set to determine the optimal structure for the whole training data. We analyze risk bounds using the VC dimension theory and suggest a parameterized form of risk estimates for the performance prediction model. As a result, we can estimate the optimal structure for the whole training data using the suggested CV method referred to as the parameterize CV (p-CV) method. Through the simulation for function approximation, we have shown the effectiveness of our approach.

1. Introduction

In the learning models, the goal is minimizing the general error for the whole distribution of sample space, not just a set of training samples. This is referred to as the generalization problem. For this problem of learning, there are two constraints, one is the absence of a priori model of data generation and the other is the limited size of data. To handle the first constraint, we need the nonparametric type of estimators such as artificial neural networks in which the training is frequently tackled by the incremental learning algorithms [15,14,7,12,10]: the necessary computational units, usually the kernel functions with locality such as Gaussian kernel functions (GKFs), are recruited in the learning procedure. In this incremental learning, the optimal number of kernel functions should be sought in the sense of minimizing the expected risk (or general error) which is dependent upon the given target function embedded in the samples and the estimation functions (or regression models). However, the expected risk cannot be estimated from the finite number of samples which are usually given for the learning of estimation functions. Instead, we can measure the empirical risk from the limited size of samples. In this regression, the proper network size (or the number of parameters) of learning model is hard to decide since the performance of learning model measured for the whole distribution of samples (not just given samples) should be optimized. For this learning problem, the estimation function is trained to minimize the square errors between the target values and function estimates. However, the minimization of the square errors for training samples does not guarantee to minimize the expected risk. Here, the expected risk can be decomposed by the bias and variance terms of estimation functions. If we increase the number of parameters, the bias term is decreased but the variance term is increased or vice versa. If the number of parameters is so small that the performance is not optimal due to large bias term, it is called the under-fitting of estimation functions. If the number of parameters is too large so that the performance is not optimal due to large variance term, it is called the over-fitting of estimation functions. So, there is a trade-off between the under-fitting and over-fitting of estimation functions.

One of practical methods to cope with the under-fitting or over-fitting problem is the cross-validation (CV) method in which a part of training data referred to as the validation set is used to check the performance of training. However, CV method has the demerits of not using the entire training data: this is critical in the case of small number of data, and even in the case of relatively large number of data, we usually get the less performance than the expected for the given training data. Furthermore, in general, the validation method does not provide the optimality of learning. In this context, we consider to construct the performance prediction model using the validation set to determine the optimal structure for the whole training data. For this problem, we analyze risk bounds using the VC dimension theory [17] and suggest a parameterized form of risk estimates for the performance prediction model. The parameters of performance

* Corresponding author. Tel.: +82 428692736; fax: +82 428695710.
E-mail addresses: imhoi.koo@kiom.re.kr (I. Koo), namgil@kaist.ac.kr (N. Lee), rmkil@kaist.ac.kr (R.M. Kil).

0925-2312/$ - see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.neucom.2008.04.043

prediction model is estimated using the validation set. Then, the optimal structure for the whole training data is determined by the estimated performance prediction model. Finally, the regression model with the optimal structure is trained for the whole training data. As a result, the suggested CV method referred to as the parameterized CV (p-CV) method is able to achieve the maximum performance for the whole training data.

This paper is organized as follows: in Section 2, we suggest the risk bounds of regression models; from the suggested bounds, we provide the risk estimator and describe the suggested p-CV method for regression models in Section 3; to show the effectiveness of the suggested p-CV method, the simulation for function approximation is performed in Section 4; and finally, Section 5 describes the Conclusion.

2. Risk bounds of regression models

Let us consider that the following l samples

$$(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_l, y_l),$$

which are drawn randomly and independently. We assume that the d dimensional input pattern $\mathbf{x}_i \in X$ (an input space, an arbitrary subset of R^d) and the output pattern $y \in Y$ (an output space, an arbitrary subset of R) have the following functional relationship:

$$y(\mathbf{x}) = f_0(\mathbf{x}) + \varepsilon, \tag{1}$$

where $f_0(\mathbf{x})$ represents the target function in the target function space F, and ε represents the randomly drawn noise term with the mean of zero and the variance of σ_ε^2.

Now, let us consider the vector $\mathbf{x}_i$ is drawn independent and identically distributed in accordance with a probability distribution $P(\mathbf{x})$. Then, y_i is drawn from the random trials in accordance with $P(y|\mathbf{x})$ and the probability distribution $P(\mathbf{x}, y)$ defined on $X \times Y$ can be described by

$$P(\mathbf{x}, y) = P(\mathbf{x})P(y|\mathbf{x}). \tag{2}$$

Here, we define a loss function as

$$Q(\mathbf{x}, f_n) = (y(\mathbf{x}) - f_n(\mathbf{x}))^2, \tag{3}$$

where $f_n(\mathbf{x})$ represent the estimation function in F_n, the hypothesis space (or structure) with n parameters.

The goal of learning with the estimation function $f_n(\mathbf{x})$ is minimizing the expected risk $R_\varepsilon(f_n)$ for noisy output $y(\mathbf{x})$ defined by

$$\begin{aligned} R_\varepsilon(f_n) &= E[Q(\mathbf{x}, f_n)] \\ &= \int_{X \times Y} (y(\mathbf{x}) - f_n(\mathbf{x}))^2 \, \mathrm{d}P(\mathbf{x}, y). \end{aligned} \tag{4}$$

Here, we also define the expected risk for noiseless output as $R(f_n)$, that is, $R_\varepsilon(f_n)$ when $y(\mathbf{x}) = f_0(\mathbf{x})$. If ε has normal distribution, the selection of a hypothesis which minimizes the mean square error of (4) is optimal, that is,

$$f_0(\mathbf{x}) = E[y|\mathbf{x}]. \tag{5}$$

However, we cannot estimate (4) in advance. Instead, we get the empirical risk $R_{\varepsilon,\text{emp}}$ for l noisy samples defined by

$$R_{\varepsilon,\text{emp}}(f_n) = \frac{1}{l}\sum_{i=1}^{l} Q(\mathbf{x}_i, f_n) = \frac{1}{l}\sum_{i=1}^{l}(y_i - f_n(\mathbf{x}_i))^2, \tag{6}$$

where $\mathbf{x}_i$ and $f(\mathbf{x}_i)$ represent the ith input and output patterns, respectively. Here, we also define the empirical risk for noiseless output as $R_{\text{emp}}(f_n)$.

For the bounded loss functions, the generalization bounds of the expected risks can be explained by the following theorem:

Theorem 1. *The expected risks for the finite VC dimension h_n of a set of the bounded loss functions $Q(\mathbf{x}, f_n)$, $f_n \in F_n$ have the following upper bounds with a probability at least $1 - 2\delta$:*

$$R_\varepsilon(f_n) \leqslant C_1 r_n + C_2 \varepsilon_n + \sigma_\varepsilon^2, \tag{7}$$

where

$$r_n = \left(\frac{1}{n}\right)^{\alpha}, \tag{8}$$

$$\varepsilon_n = \frac{h_n\left(1 + \ln \frac{2l}{h_n}\right) - \ln \frac{\delta}{2}}{l}, \tag{9}$$

C_1 and α represent constants dependent upon the target and estimation functions, and C_2 represents a constant dependent upon the estimation function.

For the proof of this theorem, refer to Appendix A. The suggested Theorem 1 states that the bounds of expected risks are determined by the approximation error $R(f_n^*)(= C_1 r_n)$ and the regression error $|R(f_n) - R(f_n^*)|(= C_2 \varepsilon_n)$ as illustrated in Fig. 1. It further states that if we increase the number of parameters n, the approximation error is decreased while the regression error can be increased for the given number of samples l since it will increase the VC dimension h_n of a set of loss functions $Q(\mathbf{x}, f_n)$, $f_n \in F_n$. Therefore, we need to make a trade-off between the approximation and regression errors to determine the optimal number of parameters as shown in Fig. 2.

3. p-CV method for regression models

For the practical application of the suggested theorem, we consider the expected risk bounds in more simplified form. First, let us approximate ε_n of (9) as the following form:

$$\varepsilon_n \approx (1 + \ln 2)\left(\frac{h_n}{l}\right)^{\beta}, \tag{10}$$

where β is set to $\frac{1}{2}$.

This implies that the upper bound of the expected risk $R_u(f_n)$ can be described as

$$R_u(f_n) \approx C_1\left(\frac{1}{n}\right)^{\alpha} + C_2'\left(\frac{h_n}{l}\right)^{\beta} + \sigma_\varepsilon^2, \tag{11}$$

where $C_2' = (1 + \ln 2)C_2$.

In general, the VC dimension of regression models can be bounded by the following form:

$$h_n \leqslant O(n^b), \tag{12}$$

where b represents a constant dependent upon the type of regression models. If the regression model is linear such as perceptron, the value of b is determined by 1. In the large class of

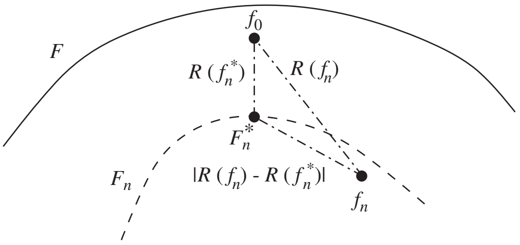

Fig. 1. The geometrical interpretation of $R(f_n)$: the target function f is in the function space F while the optimal hypothesis f_n^* and the estimated hypothesis f_n belong to the nth hypothesis F_n. Usually, F_n is the subspace of F. The expected risk $R(f_n)$ is always less than two distances, $R(f_n^*)$ and $|R(f_n) - R(f_n^*)|$ due to the triangular inequality.

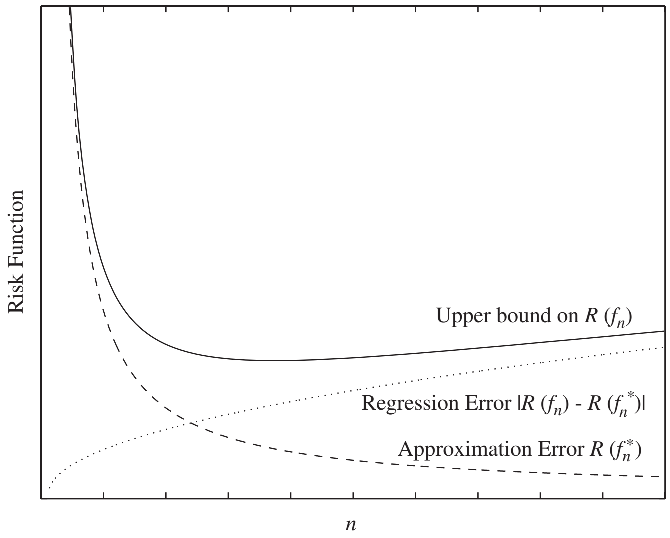

Fig. 2. The plot of the upper bound on $R(f_n)$ versus n for the fixed l: as n is increasing, the approximation error $R(f_n^*)$ is decreasing, on the other hand, the regression error $|R(f_n) - R(f_n^*)|$ is increasing. The expected risk $R(f_n)$ is always less than or equal to the sum of these two errors.

neural networks including the sigmoidal, radial basis functions, and sigma–pi networks, the value of b is larger than 1, but does not exceed 4 since the VC dimension exists within the following range [9,16]:

$$O(n \log m) \leqslant h_n \leqslant O(n^2 m^2), \tag{13}$$

where m represents the number of hidden units. Here, in general, $m \leqslant n$.

Summarizing (11)–(13), we suggest the risk estimator $\hat{R}_\varepsilon(f_n)$ as

$$\hat{R}_\varepsilon(f_n) = C_1 \left(\frac{1}{n}\right)^\alpha + C_2' \frac{n^{\beta'}}{\sqrt{l}} + \theta, \tag{14}$$

where C_1, C_2', α, $\beta'(= b\beta)$, and θ are coefficients to be estimated. Here, the bias term θ is included to compensate for the bound of $R_\varepsilon(f_n)$ derived from the triangular inequality and the noise variance.

From the functional form of $\hat{R}_\varepsilon(f_n)$, the optimal number of parameters n^* can be determined by the coefficients, C_1, C_2', α, β', and the number of training samples l. The optimal number of parameters n^* can be determined by searching for the value of n such that $\partial \hat{R}_\varepsilon / \partial n = 0$, that is,

$$n^* \approx \left(\frac{C_1 \alpha}{C_2' \beta'}\right)^{1/(\alpha+\beta')} l^{1/2(\alpha+\beta')}. \tag{15}$$

The optimal number of parameters determined by the above equation shows us that (1) n^* is increasing as l is increasing and (2) the rate of the increment is mainly dependent upon α and β'. For instance, let us consider the structure of learning model with $\alpha = 1$ and $\beta' = \frac{1}{2}$. Then, the optimal number $n^* \propto l^{1/3}$.

The suggested true risk bounds can be a useful guideline for searching for the optimal structure of learning model. One way of optimizing the learning model is to select the optimal structure minimizing the predicted risk bounds. From this point of view, we consider the following p-CV procedure for searching for the optimal network structure of regression models:

3.1. *Procedure of the p-CV for regression models*

Step 1: Divide the given samples into two sets, the training and validation sets.

Step 2: Estimate the empirical risks using the validation set for various values of n.

Step 3: The coefficients of $\widehat{R}_\varepsilon(f_n)$ are estimated using the values of $R_{\varepsilon,\text{emp}}(f_n)$ as the target values:

(1) First, the value of α is estimated from the log–log plot of $R_{\varepsilon,\text{emp}}(f_n)$ versus n: the value of α is determined by the gradient of the log–log plot for the smaller range of n/l, for instance, less than 0.1. One way of identifying this range is by comparing the gradients of empirical risks for the training and test patterns, that is, by selecting the region where two gradients has similar values.

(2) Once the value of α is determined, the value of β' is searched between $\frac{1}{2}$ and 2 where the coefficients C_1, C_2', and θ of (14) is minimized in the sense of minimizing the sum of the square error of risk functions over n, that is,

$$E_\text{R} = \frac{1}{N} \sum_{n=1}^{N} (R_{\varepsilon,\text{emp}}(f_n) - \widehat{R}_\varepsilon(f_n))^2, \tag{16}$$

where N represents the number of parameters.

(3) After we determine the initial values of these coefficients, the fine tuning of these coefficients is carried out by applying the gradient descent method minimizing (16).

Step 4: Determine the optimal number of parameters (or computational units) n^* that minimizes the measure of the true risk estimate for all of the samples using the estimated formulas of risk bounds, that is, n^* of (15).

Step 5: Train the learning model with n^* using all of the samples.

In the suggested p-CV method, the optimal structure of the regression model is determined by predicting the model's performance. For the decision of the optimal size of network structure, it is important how well the risk estimator $\hat{R}_\varepsilon(f_n)$ is fitted with the expected risk $R_\varepsilon(f_n)$. In this sense, we need large number of validation samples. However, if we set the number of validation samples too large, it becomes difficult to train the regression models due to small number of training samples. From this point of view, we suggest to use the half of the given samples as the validation set. With this setting, we will show the accuracy of the risk estimator $\hat{R}_\varepsilon(f_n)$ through the simulation for function approximation explained in the next section.

4. Simulation

To show the validity of the suggested theorems, we performed experiments for the regression of data generated from the following two-dimensional function:

$$f(x_1, x_2) = 0.4x_1 \sin(4\pi x_1) + 0.6x_2 \cos(6\pi x_2), \tag{17}$$

where the values of x_1 and x_2 were selected between 0 and 1.

For this regression, we chose the learning model with the GKFs. The training of this learning model was done by the incremental learning algorithm [10] in which the necessary GKFs were recruited at the positions where the learning model made large estimation errors. In this model, the training was composed of two learning procedures, the procedure of recruiting the necessary number of GKFs and the procedure of parameter estimation associated with GKFs. The goal of the recruiting procedure is to recruit the necessary number of kernel functions. Through this procedure, we adjusted the shape parameters and the positions of the kernel functions representing the reference points of the learning samples. After the recruiting procedure, the weight parameters associated with the kernel functions were adjusted in such a way as to minimize the mean squared error between the desired and actual outputs.

To train this learning model, a set of 1000 training samples were generated randomly from (17) and they were applied to the incremental learning algorithm in which the number of kernel functions n was increased from 10 to 300. Test sets that were not overlapped with the training sets, a set of 1000 samples were also generated randomly from (17). In the training samples, one half of the samples (500 samples) were used as a validation set and another half of the samples (500 samples) were used to train the learning model. After training, we measured the empirical risk $R_{\text{emp}}(f_n)$ for the given number of kernel functions n using the validation set. From the values of $R_{\text{emp}}(f_n)$, the coefficients of the risk estimator $\widehat{R}(f_n)$ were adjusted. Here, the value of α indicated the convergence speed of the learning model, which was dependent upon the smoothness of the target function and the regression model. If α was greater than or equal to 1, it was referred to as the fast rate of approximation. Usually, the value of α was determined between 0 and 1. The range of β' was between $\frac{1}{2}$ and 2 due to the range of the VC dimension of artificial neural networks as described in (13). After the estimation of these coefficients of $\widehat{R}(f_n)$, the optimal number of kernel functions n^* could be estimated by locating the minimum point of (14), that is, n^* of (15). The estimated results of n^* for 500 and 1000 training samples were given by 206 and 278, respectively. These results could be considered as reasonable answers for n^* considering the trends of the measured risks as shown in Figs. 3 and 4.

From the estimation of $\widehat{R}(f_n)$, the confidence intervals can be estimated by the following equation [11]:

$$|R(f_n) - R_{\text{emp}}(f_n)| \leqslant t_{\delta,l-1}\sqrt{\frac{\widehat{k}_n - 1}{l}}\widehat{R}(f_n), \tag{18}$$

where $t_{\delta,l-1}$ represents a critical point value of student t-distribution with $l-1$ degrees of freedom and $\widehat{k}_n$ represents a constant calculated by

$$\widehat{k}_n = \frac{1}{l}\sum_{j=1}^{l} e_n^4(\mathbf{x}_j) \Big/ \left(\frac{1}{l}\sum_{k=1}^{l} e_n^2(\mathbf{x}_k)\right)^2, \tag{19}$$

where $e_n(\mathbf{x}_j)$ represents the error for the jth sample in the test set.

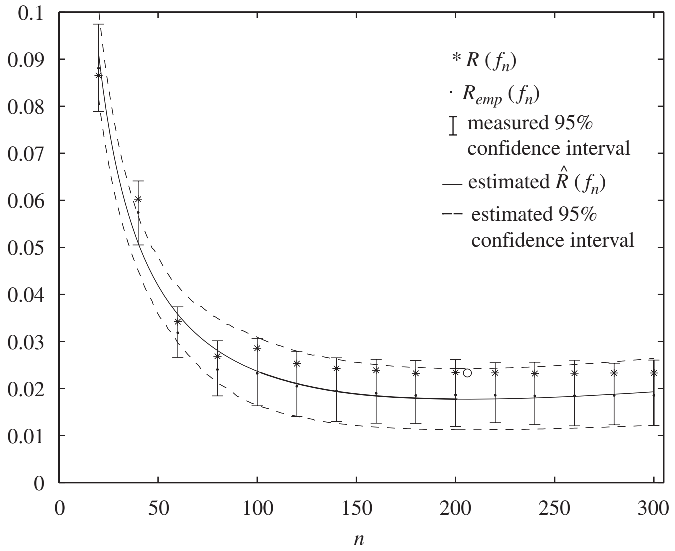

Fig. 3. The plot of measured and estimated risks, and confidence intervals versus number of kernel functions n in the case of the number of training samples 500. The circle represents the true risk at $n^* = 206$.

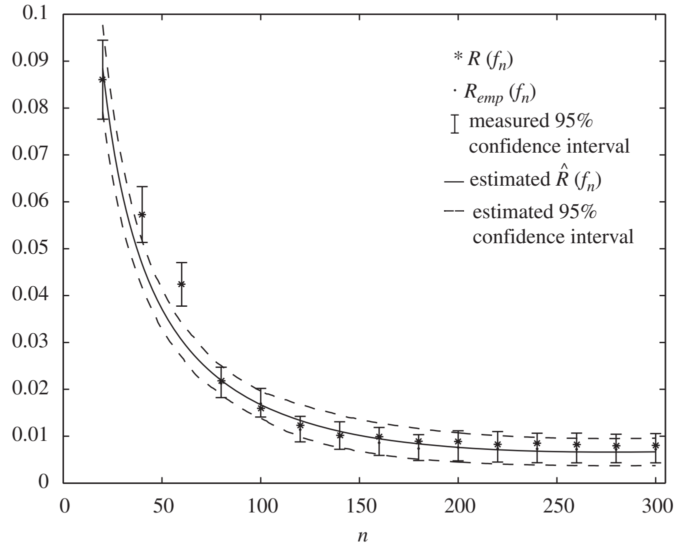

Fig. 4. The plot of measured and estimated risks, and confidence intervals versus number of kernel functions n in the case of the number of training samples 1000. The circle represents the true risk at $n^* = 278$.

After the adaptation of coefficients associated with the true risk estimate and confidence intervals for the training set of 500 samples and the validation set of 500 samples, we compared the predicted results of risk estimates and estimated confidence intervals with the measured performance using the validation set of 500 samples. These results were plotted in Fig. 3. We also compared the predicted results of risk estimates and estimated confidence intervals with the measured performance using the test set of 1000 samples when we train the estimation function using all of training samples. These results were plotted in Fig. 4. In both figures, the risk estimates were drawn by a solid line, while the estimated confidence intervals were drawn by a dashed line. We compared the results of performance prediction with the risks calculated by the numerical integration of loss functions and the 95% confidence intervals using the bootstrap percentiles [5]. The risk estimates were well fitted to the measured risks and the estimated confidence intervals were well fitted to the measured confidence intervals. In this case, the averages of measured risks were mostly within the estimated confidence intervals. In the case of the number of training samples 1000, there were some differences in the range of n between 40 and 60 as shown in Fig. 4. This might be mainly due to the inconsistency of the learning algorithm we used, that was, for the range of n between 40 and 60, the learning algorithm might not find the good parameters of the regression model while in the other ranges, especially for the larger values of n, the learning algorithm might find the good parameters of the regression model. For the purpose of optimizing the structure of regression models, performance fitting for larger values of n could be the major factor for finding the solution. As a whole, these simulation results showed us that the suggested functional form of true risk bounds and confidence intervals were in accordance with the measured data. As a result of this simulation, the optimal number of kernel functions for all training samples (= 1000 training samples) was estimated as 278, the true risk estimate was 0.0066, and the estimated 95% confidence interval was 0.0066 ± 0.0029 in which the measured risk $R_{\varepsilon}(f_n) = 0.0086$ was surely included as shown in Fig. 4.

To show the effectiveness of the suggested p-CV method in determining the optimal structure of regression models, we also conducted simulations for function approximation using the p-CV, 5-fold CV, and 10-fold CV methods. For this simulation, we

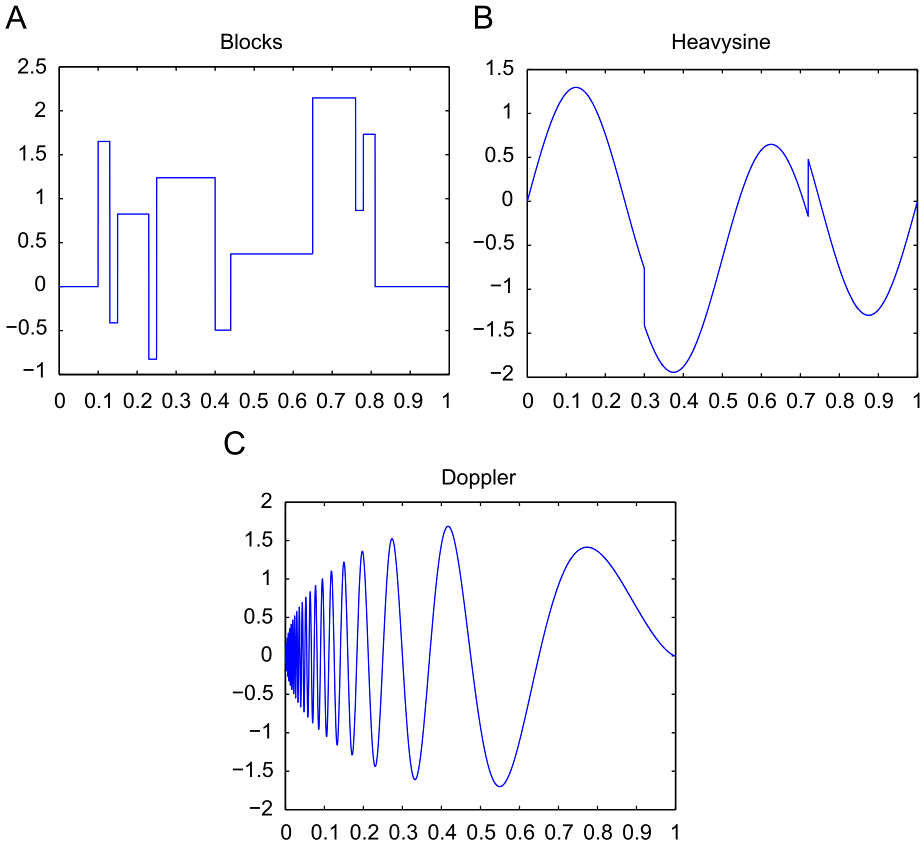

Fig. 5. Target functions from Donoho and Johnstone [4]: (A)–(C) represent the Blocks, Heavysine, and Doppler functions, respectively.

selected the benchmark data given by Donoho and Johnstone (D–J) [4]: they are Blocks, Heavysine, and Doppler functions, as illustrated in Fig. 5. We also used the same learning model as described in the previous simulation. For each target function, a set of 200 training samples was randomly generated to train the learning model, and the number of kernel functions n of the learning model was increased from 1 to 100. A set of 2000 samples was also randomly generated from the same target function as a test set. The outputs of both training samples and test samples were corrupted by the noise terms that were generated randomly from a normal distribution with a mean of zero and a standard deviation of $\sigma_\varepsilon = 0.3$.

For the given set of training samples, each CV method estimated the risk estimator $\widehat{R}(f_n)$ for each n and selected the optimal n^* such that

$$n^* = \arg\min_n \widehat{R}(f_n). \qquad (20)$$

Then, the risks $R(f_{n^*})$ of the selected model f_{n^*} were evaluated on the test samples and the results were compared with the minimum risk, that is, $\min_n R(f_n)$. Quantitatively, we considered the risk ratio r_R as the log ratios of two risks $R(f_{n^*})$ and $\min_n R(f_n)$, that is,

$$r_\mathrm{R} = \log_{10} \frac{R(f_{n^*})}{\min_n R(f_n)}. \qquad (21)$$

This ratio shows how the risk for the selected n^* is close to the optimal risk. If this value is close to 0, the risk for the selected n^* is near the optimal risk. After the whole procedures were repeated 30 times, the simulation results for the means and standard deviations of the risk ratios are described in Table 1 where the boldfaced number represents the smallest mean value of the risk ratios for each target function. As shown in these simulation results, the suggested p-CV method outperforms the 5-CV or 10-CV method from a view point of risk ratios while the 5-CV and 10-CV methods shows the similar performances. These simulation results imply that the performance prediction using the suggested p-CV method can be more effective in regression models compared to the conventional CV methods.

Table 1
Means and standard deviations of risk ratios (21) for the p-CV, 5-fold CV, and 10-fold CV methods

Target functions	p-CV		5-Fold CV		10-Fold CV	
	Mean	S. dev.	Mean	S. dev.	Mean	S. dev.
Blocks	**0.0026**	0.0047	0.0072	0.0066	0.0072	0.0089
Heavysine	**0.0115**	0.0149	0.0259	0.0340	0.0138	0.0177
Doppler	**0.0078**	0.0107	0.0142	0.0157	0.0201	0.0249

In summary, through the simulation for function approximation, we have shown that the suggested risk estimator is able to capture the learning curve, that is, the risks versus the number of parameters and the suggested p-CV method is able to provide the appropriate network size of learning models for the whole training data. As a result, we have shown that the suggested p-CV method outperforms the 5-CV or 10-CV method from a view point of risk ratios for the regression of D–J benchmark data. Furthermore, even if the number of learning samples is changed, we can easily estimate the optimal network size using the risk estimator obtained from the previous training.

5. Conclusion

We have suggested a new method of optimizing the network structure for the regression models. The risk bounds are investigated from the view points of function approximation theories and also VC dimension theories. The suggested risk bounds can provide a useful guideline for the optimization of regression models. We also suggested the optimization method using the suggested performance prediction method referred to as the p-CV method. Through the simulation for function approximation, we have shown that the suggested p-CV method can be more effective to select the optimal size of regression models compared to the conventional CV methods. The suggested approach can be easily applied to the optimization of the structure (or network size) of various types of learning models including artificial neural networks in the sense of minimizing the expected risks.

Appendix A

Let us consider the case of a regression problem solved by the learning model with the bounded loss function defined by the error square, that is, $0 \leqslant Q(\mathbf{x}, f_n) \leqslant B$. For this regression problem, Vapnik [17] suggested the following indicator function:

$$I(Q(\mathbf{x}, f_n), \theta) = \begin{cases} 1 & \text{if } Q(\mathbf{x}, f_n) \geqslant \theta, \\ 0 & \text{otherwise,} \end{cases} \tag{A.1}$$

where θ represents the threshold value between 0 and B. Here, the VC dimension of a set of real functions $Q(\mathbf{x}, f_n)$, $f_n \in F_n$ is defined to be the VC dimension of the set of corresponding indicator functions of (A.1) with $f_n \in F_n$ and $\theta \in (0, B)$.

Let us also define the risk function $R(f_n)$ as the risk function for noiseless output $y(\mathbf{x})$, that is, $y(\mathbf{x}) = f_0(\mathbf{x})$ in (1). Then, the generalization bounds of the regression model with the finite VC dimension can be suggested as follows [1,17]:

Lemma A1. *For the finite VC dimension h_n of a set of the bounded loss functions $Q(\mathbf{x}, f_n)$, $f_n \in F_n$ and $l \geqslant 8B/\varepsilon_n$, the following inequality holds with a probability at least $1-\delta$:*

$$|R(f_n) - R_{\mathrm{emp}}(f_n)| \leqslant \varepsilon_n \quad \text{for any } f_n \in F_n, \tag{A.2}$$

where $R_{\mathrm{emp}}(f_n)$ represents the empirical risk for noiseless samples and

$$\varepsilon_n = \frac{2B}{\ln 2} \frac{h_n\left(1 + \ln \frac{2l}{h_n}\right) - \ln \frac{\delta}{2}}{l}. \tag{A.3}$$

Lemma A1 states that the generalization bound ε_n is mainly dependent upon h_n/l. If the ratio of h_n/l is large, the generalization bound is large, or vice versa. The required number of samples l in Lemma A1 can be achieved when $l \geqslant h_n$ and $h_n \geqslant 2$. This condition may be suited to most learning cases. From Lemma A1, the generalization bounds of the expected risks for the bounded loss functions can be explained by the following theorem:

Theorem A1. *The expected risks for the finite VC dimension h_n of a set of the bounded loss functions $Q(\mathbf{x}, f_n)$, $f_n \in F_n$ have the following upper bounds with a probability at least $1-2\delta$:*

$$R_{\varepsilon}(f_n) \leqslant C_1 r_n + C_2 \varepsilon_n + \sigma_{\varepsilon}^2, \tag{A.4}$$

where

$$r_n = \left(\frac{1}{n}\right)^{\alpha}, \tag{A.5}$$

$$\varepsilon_n = \frac{h_n\left(1 + \ln \frac{2l}{h_n}\right) - \ln \frac{\delta}{2}}{l}, \tag{A.6}$$

C_1 and α represent constants dependent upon the target and estimation functions, and C_2 represents a constant dependent upon the estimation function.

Proof. The risk function $R(f_n)$ can be rewritten as

$$R(f_n) = R(f_n) - R(f_n^*) + R(f_n^*), \tag{A.7}$$

where f_n^* represents the optimal estimation function in F_n.

From the triangular inequality, the following inequality holds:

$$R(f_n) \leqslant |R(f_n) - R(f_n^*)| + R(f_n^*). \tag{A.8}$$

The first term of the right hand side of (A.8) represents the distance between two risk functions, $R(f_n)$ and $R(f_n^*)$, in other words, the risk function of f_n, which is the estimated function and the risk function of f_n^*, which is the optimal estimation function. This distance is referred to as the regression error. The second term of the righthand side of (A.8) represents the risk function of f_n^* referred to as the approximation error.

Let us first consider the regression error $|R(f_n) - R(f_n^*)|$ of inequality (A.8). In the case of empirical risk minimization, the following conditions hold:

$$R(f_n^*) \leqslant R(f_n) \quad \text{and} \tag{A.9}$$

$$R_{\mathrm{emp}}(f_n^*) \geqslant R_{\mathrm{emp}}(f_n). \tag{A.10}$$

According to Lemma 1, with a probability at least $1-\delta$,

$$|R(f_n) - R_{\mathrm{emp}}(f_n)| \leqslant \varepsilon_n \quad \text{for any } f_n \in F_n, \tag{A.11}$$

where

$$\varepsilon_n = \frac{2B}{\ln 2} \frac{h_n\left(1 + \ln \frac{2l}{h_n}\right) - \ln \frac{\delta}{2}}{l}. \tag{A.12}$$

Therefore, under the conditions of (A.9) and (A.10), the following inequality is satisfied with a probability at least $1-2\delta$,

$$\begin{aligned} |R(f_n) - R(f_n^*)| &\leqslant |R(f_n) - R(f_n^*) + R_{\mathrm{emp}}(f_n^*) - R_{\mathrm{emp}}(f_n)| \\ &\leqslant |R(f_n) - R_{\mathrm{emp}}(f_n)| + |R(f_n^*) - R_{\mathrm{emp}}(f_n^*)| \\ &\leqslant 2\varepsilon_n. \end{aligned} \tag{A.13}$$

The second term, the approximation error $R(f_n^*)$ of inequality (A.8) can be described by the function approximation theories. According to Lorentz's work [13], if n represents the degree of polynomials of the estimation function, the true risk of f_n^* is bounded by

$$\sqrt{R(f_n^*)} \leqslant r_n' \propto \left(\frac{1}{n}\right)^{s/m}, \tag{A.14}$$

where s represents the number of existing derivatives and m represents the dimension of input space of the target function.

In the case of the estimation function with various types of nonlinear kernel functions such as sigmoid or GKFs, r_n' can be described [2,3,6,8] by

$$r_n' \propto \left(\frac{1}{n}\right)^{\alpha'}, \tag{A.15}$$

where α' is a constant dependent upon the target and estimation functions. For instance, if the kernel functions are sigmoid functions, α' is estimated as $\frac{1}{2}$ under the following smoothness constraint of the target function [2]:

$$\int_{-\infty}^{+\infty} |w||F(w)|\,\mathrm{d}w < \infty, \tag{A.16}$$

where $F(w)$ represents the Fourier transform of the target function.

Here, the expected risks for noisy output $y(\mathbf{x})$ can be redescribed by the following equation:

$$\begin{aligned} R_\varepsilon(f_n) &= E[(y(\mathbf{x}) - f_n(\mathbf{x}))^2] \\ &= E[(y(\mathbf{x}) - f_0(\mathbf{x}) + f_0(\mathbf{x}) - f_n(\mathbf{x}))^2] \\ &= E[(y(\mathbf{x}) - f_0(\mathbf{x}))^2] + 2E[(y(\mathbf{x}) - f_0(\mathbf{x}))(f_0(\mathbf{x}) - f_n(\mathbf{x}))] \\ &\quad + E[(f_0(\mathbf{x}) - f_n(\mathbf{x}))^2] \\ &= \sigma_\varepsilon^2 + R(f_n) \end{aligned} \tag{A.17}$$

due to (1) and (5).

Therefore, from (A.17), (A.8), and (A.11)–(A.15), the following inequality is satisfied with a probability at least $1-2\delta$:

$$R_\varepsilon(f_n) \leqslant C_1 \left(\frac{1}{n}\right)^{\alpha} + C_2 \frac{h_n\left(1 + \ln \frac{2l}{h_n}\right) - \ln \frac{\delta}{2}}{l} + \sigma_\varepsilon^2, \tag{A.18}$$

where C_1 and $\alpha (= 2\alpha')$ represent constants dependent upon the target and estimation functions, and $C_2 (= 4B/\ln 2)$ represents a constant dependent upon the estimation function. □

References

[1] M. Anthony, N. Biggs, Computational Learning Theory, Cambridge University Press, Cambridge, 1992.

[2] A. Barron, Universal approximation bounds for superpositions of a sigmoid functions, IEEE Trans. Inf. Theory 39 (3) (1993) 930–945.

[3] L. Breiman, Hanging hyperplanes for regression, classification and function approximation, IEEE Trans. Inf. Theory 39 (3) (1993) 999–1013.

[4] D. Donoho, I. Johnstone, Adapting to unknown smoothness via wavelet shrinkage, J. Am. Stat. Assoc. 90 (432) (1995) 1200–1224.

[5] B. Efron, R. Tibshirani, An Introduction to the Bootstrap, Chapman and Hall, London, 1993.

[6] F. Girosi, G. Anzellotti, Rate of convergence for radial basis functions and neural networks, in: Artificial Neural Networks for Speech and Vision, Chapman & Hall, London, 1993.

[7] E.J. Hartman, J.D. Keeler, J.M. Kowalski, Layered neural networks with Gaussian hidden units as universal approximations, Journal Neural Comput. 2 (2) (1990) 210–215.

[8] L. Jones, A simple lemma on greedy approximation in Hilbert space and convergence rates for projection pursuit regression, Ann. Stat. 20 (1) (1992) 608–613.

[9] M. Karpinski, A. Macintyre, Polynomial bounds for vc dimension of sigmoidal neural networks, in: Proceedings of the 27th Annual ACM Symposium on Theory of Computing, 1995, pp. 200–208.

[10] R.M. Kil, Function approximation based on a network with kernel functions of bounds and locality: an approach of non-parametric estimation, ETRI J. 15 (2) (1993) 35–51.

[11] I. Koo, R.M. Kil, Confidence intervals for the risks of regression models, in: Lecture Notes in Computer Science, vol. 4232, 2006, pp. 755–764.

[12] S. Lee, R.M. Kil, A Gaussian potential function network with hierarchically self-organizing learning, Neural Networks 4 (2) (1991) 207–224.

[13] G. Lorentz, Approximation of Functions, Chelsea Publishing Co., New York, 1986.

[14] J. Moody, C.J. Darken, Fast learning in networks of locally-tuned processing units, Neural Comput. 1 (2) (1989) 281–294.

[15] M. Niranjan, F. Fallside, Neural networks and radial basis functions in classifying static speech patterns, Technical Report CUED/F-INFENG/TR22, Cambridge University, 1988.

[16] A. Sakurai, Polynomial bounds for the vc-dimension of sigmoidal, radial basis function, and sigma-pi networks, in: Proceedings of the World Congress on Neural Networks, vol. 1, 1995, pp. 58–63.

[17] V. Vapnik, Statistical Learning Theory, Wiley, New York, 1998.

Imhoi Koo received the M.S. and Ph.D. degrees in Applied Mathematics, from the Korea Advanced Institute of Science and Technology (KAIST) Daejeon, Korea, in 2001 and 2007, respectively. Currently, he is a postdoc researcher in the Department of Medical Research, Korea Institute of Oriental Medicine (KIOM). His research focuses on the mathematical analysis for machine learning, particularly including model selection, regularization methods, and feature selection. Recently, he is working on developing biological data mining technologies.

Namgil Lee received his M.S. degree in Applied Mathematics from the Korea Advanced Institute of Science and Technology (KAIST) in 2006. Currently, he is working toward the Ph.D. degree in Mathematics at KAIST. His research interests include machine learning, kernel methods, statistical learning theory, and analysis and prediction of nonlinear time series.

Rhee Man Kil received the Ph.D. degree from the University of Southern California in 1991, and currently an associate professor of the Department of Mathematical Sciences, Korea Advanced Institute of Science and Technology. His general research interests lie in theories and applications of machine learning and his current interests focus on model selection in regression, noise-robust feature extraction, and binaural speech segregation and recognition.