REGULARIZATION LEARNING OF NEURAL NETWORKS FOR GENERALIZATION

Shotaro Akaho

Mathematical Informatics Section
Electrotechnical Laboratory
1–1–4 Umezono, Tsukuba 305, Japan

Abstract

In this paper, we propose a learning method of neural networks based on the regularization method and analyze its generalization capability. In learning from examples, training samples are independently drawn from some unknown probability distribution. The goal of learning is minimizing the expected risk for future test samples, which are also drawn from the same distribution. The problem can be reduced to estimating the probability distribution with only samples, but it is generally ill-posed. In order to solve it stably, we use the regularization method. Regularization learning can be done in practice by increasing samples by adding appropriate amount of noise to the training samples. We estimate its generalization error, which is defined as a difference between the expected risk accomplished by the learning and the truly minimum expected risk. Assume p-dimensional density function is s-times differentiable for any variable. We show the mean square of the generalization error of regularization learning is given as $Dn^{-2s/(2s+p)}$, where n is the number of samples and D is a constant dependent on the complexity of the neural network and the difficulty of the problem.

1 Introduction

Generalization is one of the most important problems of neural network learning. As it is, even if the learned network is adapted to a given set of training samples well, it often does not fit unknown test samples. In this paper, in order to get the network with small generalization error, we propose a learning method of neural networks based on the regularization method, and analyze its effectiveness.

Neural network learning can be formulated as learning from examples: Assume that training samples are given from environment and each sample $\boldsymbol{z}$ is independently generated according to some unknown probability distribution $P(\boldsymbol{z})$. The goal of learning is to minimize a loss function $Q(\boldsymbol{z}, \alpha)$ not only for training samples but also for test samples, where $Q(\boldsymbol{z}, \alpha)$ belongs to a set of function parameterized by α. For example, in pattern recognition problem, each $\boldsymbol{z}$ denotes a pair of input pattern $\boldsymbol{x}$ and class y. A loss function is provided as $(y - q(\boldsymbol{x}, \alpha))^2$, where $q(\boldsymbol{x}, \alpha)$ denotes a function parameterized by α. Suppose test samples are generated according to the same probability distribution as training samples, the problem is to minimize the *expected risk* defines as

$$I(\alpha) = \int Q(\boldsymbol{z}, \alpha) P(\boldsymbol{z}) \, d\boldsymbol{z}, \tag{1}$$

when the density function $P(\boldsymbol{z})$ is unknown but random independent samples are given[9].

It is known that neural networks with enough number of units has a capability to approximate arbitrary functions[4]. However, because the density function $P(\boldsymbol{z})$ is assumed to be unknown, we cannot minimize the expected risk $I(\alpha)$ just with a small number of training samples. Usually, instead of $I(\alpha)$, *empirical risk* defined as

$$I_{\mathrm{emp}}(\alpha) = \frac{1}{n}\sum_{i=1}^{n} Q(\underset{(i)}{\boldsymbol{Z}}, \alpha). \tag{2}$$

is minimized, where $\underset{(1)}{\boldsymbol{Z}}, \ldots, \underset{(n)}{\boldsymbol{Z}}$ denote training samples.

Two minimizations of (1) and (2) above do not agree with each other in general. As the capability or complexity of a network is getting higher, the capability to describe unknown data becomes lower, because the learned function becomes unstable and overfits just training samples (*over-learning* or *lack of generalization capability*). One method to avoid this problem is to limit a set of realizable functions by restricting the size or structure of a network (*statistical model selection*). Though many methods are proposed from a statistical point of view, they have problems as follows:

1. $I_{\mathrm{emp}}(\alpha)$ is not so good estimate of $I(\alpha)$.

2. Model selection is done for a discrete set of functions, and each function is structurally different from others. Thus usually, we cannot apply a method like the steepest descent method, and model selection needs two steps: first, we train several networks and estimate their generalization capability, and next, we select the best one among them.

3. The method based on AIC and MDL (information criteria) [10][5] is not so robust, because they are essentially parametric.

4. The method based on VC dimension (function complexity) [9][2][1] cannot be used in practice, because the bound of the estimated generalization error is not so tight.

In this paper, we try to avoid model selection procedure. For this purpose, instead of minimizing empirical risk, we consider the minimization of expected risk itself, using $P(\boldsymbol{z})$ estimated with training samples. It is known that the nonparametric estimation of density function from finite number of data is an *ill-posed* problem and the solution is unstable. *Regularization method* is used for stabilizing the solution.

In section 2, we describe the density approximation method based on the regularization method. Then in section 3, we propose the learning method and analyze its effectiveness. Main theoretical result is theorem 1 in section 3, which valuates the expected error of the estimation of $I(\alpha)$.

2 Nonparametric density approximation

2.1 Regularization method

As noticed in the previous section, the problem of density estimation is ill-posed. The *regularization method* is a general method to solve ill-posed problems nonparametrically[8] and for the estimation of density function $f(t)$ it is formalized as follows.

Consider the problem to solve the following integral equation from n empirical data.

$$Af(t) \equiv \int_{-\infty}^{\infty} u(x-t)f(t)\,dt = F(x), \tag{3}$$

where $u(x)$ is a unit step function. Though we can estimate the cumulative distribution function $F(x)$ stably, only an unstable solution $f(x)$ can be derived by putting it in above equation directly. Now, instead of solving the equation (3), consider the functional minimization problem below:

$$R_\gamma(\hat{f}, F) = \rho^2(A\hat{f}, F) + \gamma\Omega(\hat{f}), \tag{4}$$

where ρ denotes a distance between two functions, Ω denotes a functional called stabilizer that satisfies several conditions such as compactness and positiveness, and γ is a positive parameter. It is known the problem above can be solved stably: $\hat{f}$ converges to the true solution of $Af = F$, when the value γ is decreasing slowly enough in proportion to the approximation accuracy of $F(x)$.

2.2 Regularization method and Parzen method

It is known that the regularization method is equivalent to *Parzen method* (or kernel type density estimation method) for the problem of density estimation[9].

Parzen method is described as follows. Let $\underset{(1)}{X}, \ldots, \underset{(n)}{X}$ be n empirical data from a density function $f(x)$. The approximation of $f(x)$ is given as follows (one dimensional case).

$$f_n(x, a_n) = \frac{a_n}{n}\sum_i K(a_n(x - \underset{(i)}{X})), \tag{5}$$

where $K(x)$ is a function that satisfies several conditions and a_n is a constant.

Remark that $K(x)$ and a_n correspond to stabilizer and regularization parameter respectively in the regularization method.

Roughly speaking, it is a method to shade with $K(x)$ around each sample point. In this method, it is important to determine the shading parameter and to estimate the accuracy of approximation. If we cannot deal with them well, over-learning or over-shading eventually occurs.

In Parzen method, an asymptotically optimum value of a_n can be determined from empirical data [6]. Its detail result is shown in appendix A. The essential points are

- Assume that the density function (p dimensional) is s times partially continuously differentiable.

- Pick $K(x)$ that satisfies some appropriate conditions.

- Then the mean square error of the approximated function can be estimated ((14),(15), theorem 2).

- We can determine an asymptotically optimum value of parameter a_n from that estimation ((22)).

3 Learning algorithm via regularization method

In this section, we propose a learning algorithm using approximated density function, and estimate its capability.

3.1 Estimation of generalization capability

In order to minimize expected risk $I(\alpha)$, we consider the minimization of $\hat{I}(\alpha)$ defined below, instead of $I(\alpha)$.

$$\hat{I}(\alpha) = \int Q(\boldsymbol{z}, \alpha)\hat{P}(\boldsymbol{z})\, d\boldsymbol{z}, \tag{6}$$

where $\hat{P}(\alpha)$ is a density approximated by Parzen method. Since $\hat{I}(\alpha)$ depends upon just empirical data, this minimization is able in theory. Let us estimate how close this value is to the true expected risk under the condition that this minimization has been succeeded. In the next section, we state how this minimization can be done.

We shall say the network has generalization property if the following equation holds for an arbitrary given δ,

$$\mathrm{E}[I(\hat{\alpha}_0) - I(\alpha_0)]^2 < \delta, \tag{7}$$

where $\hat{\alpha}_0$ denotes the α that minimizes $\hat{I}(\alpha)$, and α_0 denotes the α that minimizes $I(\alpha)$.

From theorem 3 in appendix A, we can bound the left hand side of (7) and the following theorem is valid. (see appendix B about its proof).

Theorem 1 *Let p-dimensional density function $P(\boldsymbol{z})$ be s times differentiable for any variable, and let the domain region X be bounded. Then, for large n,*

$$\mathrm{E}[I(\hat{\alpha}_0) - I(\alpha_0)]^2 < \overline{D}_1 D_2 n^{-2s/(2s+p)}, \tag{8}$$

holds, where E *denotes the expectation*

$$\mathrm{E}[\,\cdot\,] = \int \cdot\, \prod_j P(\underset{(j)}{\boldsymbol{Z}})\, d\underset{(j)}{\boldsymbol{Z}},$$

$\overline{D}_1$ *denotes*

$$\overline{D}_1 = \sup_{\alpha} D_1(\alpha), \qquad D_1(\alpha) = \int_X Q(\boldsymbol{z}, \alpha)^2\, d\boldsymbol{z}, \tag{9}$$

and the value D_2 depends on $P(\boldsymbol{z}), p$ and s (see appendix B).

Accordingly, the value D_2 corresponds to the difficulty of the problem itself, and the value $\overline{D}_1$ corresponds to the complexity of the set of functions.

For example, consider the case that $Q(\boldsymbol{z}) = (y - q(\boldsymbol{x}, \alpha))^2$ and x_i, y, q are bounded between 0 and 1, which is a typical assumption of neural networks. In this case, $D_1(\alpha)$ is also bounded by 1 for any α.

3.2 A method to minimize the estimated expected risk

Here we consider how to minimize $\hat{I}(\alpha)$. Since $\hat{I}(\alpha)$ includes the calculation of integration and sum, it seems difficult to minimize directly. One simple idea is to increase the number of data by generating samples from the estimated density function $\hat{P}(\boldsymbol{z})$, and to train a network with them. This process can be repeated incrementally. We call these generated samples as “quasi-samples”.

The following algorithm is a possible implementation of this idea. (it is described in one-dimensional notation for simplicity).

```
1: Take $n$ samples $(\underset{(1)}{X},\dots,\underset{(n)}{X})$.
2: Repeat I),II):
    I) Generate some quasi-samples; Each of them is generated by i), ii):
        i) Select one sample $\underset{(i)}{X}$ arbitrary.
        ii) Generate a random sample according to the density $a_nK(a_n(x-\underset{(i)}{X}))$
    II) Train the network slightly with the quasi-samples.
```

Remark. $a_nK(a_n(x-\underset{(i)}{X}))$ does not always satisfy conditions for density function. In this case, it becomes a little more complex to generate quasi-samples and to train the network. Let $f(x)$ be $a_nK(a_n(x-\underset{(i)}{X}))$ for arbitrary selected X_i, and separate $f(x)$ into the positive part and the negative part, namely,

$$f(x) = f^+(x) - f^-(x), \qquad f^+(x), f^-(x) \geq 0. \tag{10}$$

Generate two quasi-samples x^+ according to $f^+(x)/\overline{f^+}$ and x^- according to $f^-(x)/\overline{f^-}$, where $\overline{f^+}$ and $\overline{f^-}$ are the normalization parameters for $f^+(x)$ and $f^-(x)$ respectively. Using x^+ and x^-, train the network to minimize

$$\overline{f^+}Q(x^+,\alpha) - \overline{f^-}Q(x^-,\alpha). \tag{11}$$

This learning becomes equivalent to minimizing $\hat{I}(\alpha)$, as the number of quasi-samples increases.

In the learning for quasi-samples, it is desirable not to learn so much in order to avoid the problem of converging to local minima.

3.3 Experiments

We show the results of a simple numerical experiment.

Samples: Both input x and output y are one-dimensional, x is generated from uniform distribution on $[0,1]$, and y is from constant function $y = 0.5$ with additive normal noise of variance 0.01.

Network size: Feedforward three-layered network with 1,10,1 units in the input layer, the hidden layer, and the output layer respectively.

Learning algorithm: In the regularization learning, we generated 1000 quasi-samples from all training samples and trained the network 100 steps with batch-type back propagation. We call this procedure a "learning-step" and we repeated 50 learning-steps.

In the learning with just training samples, we trained the network (1000×100/ Sample size) steps with batch-type back propagation for each learning-step of regularization learning.

Back propagation used here is the simplest one with no acceleration, and a coefficient of the steepest descent is taken 0.5.

Error estimation: MSE(mean square error) for both training samples and 10000 test samples.

Result: Figure 1 shows the MSE plots for one learning process, where the number of samples is 14.

Figure 2 shows the error plots for several sample sizes. Each error is averaged for the last 20 learning-steps, and further it is averaged for 10 such experiments.

Since each sample includes a noise of variance 0.01, there remains about 0.01 mean square error for test data, even if learning could converge to the optimum function.

In every cases, the learning with just training samples does not seem to have generalization property, overfitting to training samples, while regularization learning vibrates around 0.01 both for training samples and test samples (the reason of vibration is thought to be generating new quasi-samples one after another).

4 Concluding remarks

We proposed a learning method using regularization and analyzed its generalization capability. This method overcomes some drawbacks of model selection, and it also includes another measure of the complexity that does not change so much for the selection of a function class.

The method to add some noise to training samples has been used as a heuristic method in order to reduce generalization error. The present paper provides this kind of methods with theoretical foundation in a general framework, and it can also answer questions such as how large the noise should be.

The field of smoothing spline is deeply related to our method[7][3]. In the present paper, we mainly treated the estimation of density function rather than the estimation of regression function and our main concern is the analysis of generalization capability rather than estimation itself.

The estimation in theorem 1 and theorem 3 is getting worse as the dimension of data increase when differentiability is not high. Thus for using this method, it is important to reduce dimensionality of input data previously by extracting efficient low dimensional features from raw data.

The criterion to estimate the generalization error in this paper is different from Vapnik's one. To show the regularization learning is superior to the learning with just training samples, we must estimate it on the same criterion and solve some real world problems in practice (The experiment we've shown is too simple to show the effectiveness of our method). These tasks are left in future.

Acknowledgment

The author would like to thank K.Tamura, Director of Information Science Division of ETL, for affording an opportunity of this study. He is also deeply indebted to N.Otsu, Director of Machine Understanding Division, H.Asoh, T.Kurita and all other members of Mathematical Informatics Section for their helpful discussions.

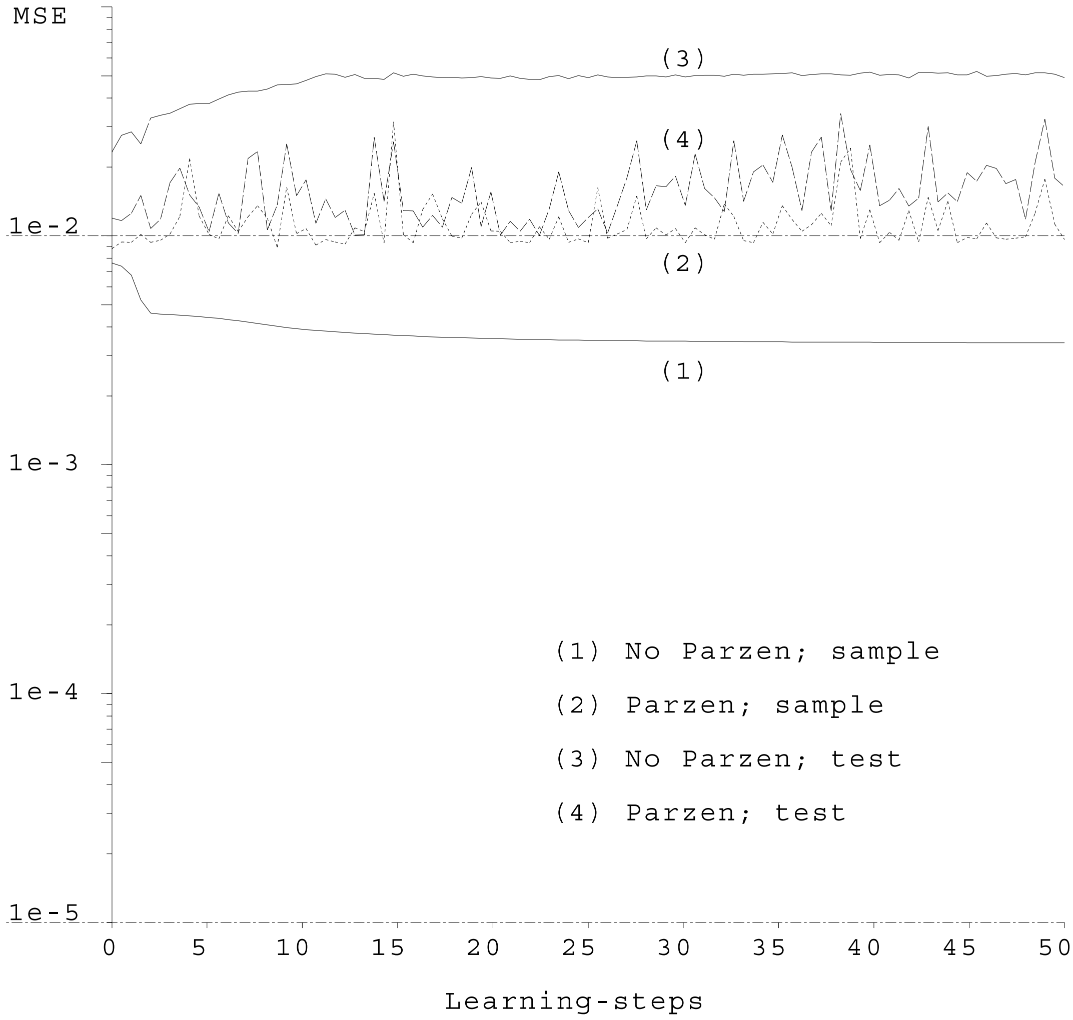

Figure 1: MSE in one learning process both for training samples (#:14) and for test samples(#:10000).

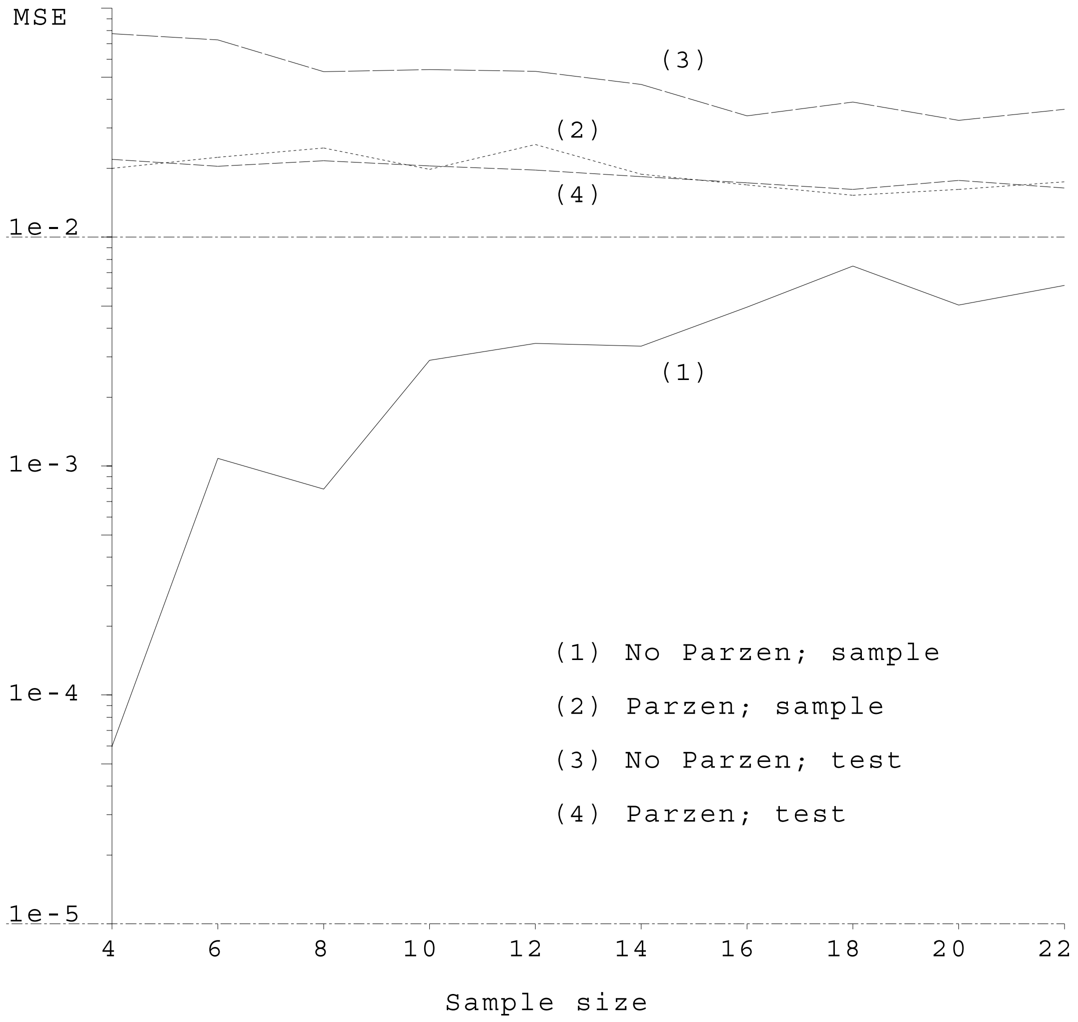

Figure 2: MSE for several sample sizes, each of which is averaged for the last 20 learning steps and further averaged for 10 such experiments.

Appendix A Optimum parameter in Parzen method

In parzen method, an asymptotically optimum a_n can be obtained from empirical data[6]. We summarize the result below.

Consider the density function of p-dimensional random variable $\boldsymbol{x} = (x_1, \ldots, x_p)$. Here we assume the density function $f(\boldsymbol{x})$ belongs to G_s which is a set of functions that is s times differentiable for all x_i. (Multi-dimensional) Parzen method is defined as approximating $f(\boldsymbol{x})$ from n sample data $\underset{(1)}{\boldsymbol{X}}, \ldots, \underset{(n)}{\boldsymbol{X}}$ with

$$f_n(\boldsymbol{x}, a_n) = \frac{1}{n} \sum_{j=1}^{n} \prod_{i=1}^{p} a_n K(a_n(x_i - \underset{(j)}{X_i})). \tag{12}$$

Now consider the estimation of the following expression,

$$U(\boldsymbol{x}, a_n) = \mathrm{E}[f_n(\boldsymbol{x}, a_n) - f(\boldsymbol{x})]^2, \tag{13}$$

where E denotes the expectation: $\mathrm{E}[\,\cdot\,] = \int \cdot \prod_j f(\underset{(j)}{\boldsymbol{X}})\, d\underset{(j)}{\boldsymbol{X}}$.

A class of kernel function is assumed to belong to H_s that is a set of functions $K(x)$ which satisfy (14), (15) below

$$K(x) = K(-x), \qquad \int K(x)\, dx = 1, \qquad \sup |K(x)| < \infty, \tag{14}$$

$$\int x^i K(x)\, dx = 0 \quad (i = 1, \cdots, s-1), \qquad \alpha = \int x^s K(x)\, dx \neq 0, \qquad \int x^s |K(x)|\, dx < \infty. \tag{15}$$

For example, normal distribution function belongs to H_2.

The following theorem is valid.

Theorem 2 *If $f(x) \in G_s$ and $K(x) \in H_s$,*

$$U(\boldsymbol{x}, a_n) \simeq \frac{a_n^p}{n} \|K\|^{2p} f(\boldsymbol{x}) + a_n^{-2s} \frac{\alpha^2}{(s!)^2} (\sum_{i=1}^{p} \partial_i^s f(\boldsymbol{x}))^2, \tag{16}$$

where

$$\partial_i^s f(\boldsymbol{x}) = \frac{\partial^s f(\boldsymbol{x})}{\partial {x_i}^s}, \qquad \|K\|^2 = \int K(x)^2\, dx, \tag{17}$$

and $a \simeq b$ denotes $a/b \to 1$ $(n \to \infty)$.

From this theorem, the optimum $a_n (= a_n^0)$ that minimizes $U(\boldsymbol{x}, a_n)$ is obtained as below, which depends upon $f(\boldsymbol{x})$.

$$a_n^0 = C(\boldsymbol{x}) n^\gamma, \qquad \gamma = \frac{1}{2s+p}, \qquad C(\boldsymbol{x}) = \left(\frac{2s\alpha^2}{(s!)^2 p \|K\|^{2p}} \frac{(\sum_i \partial_i^s f(\boldsymbol{x}))^2}{f(\boldsymbol{x})} \right)^\gamma. \tag{18}$$

Substituting a_n with this value, we have $U(\boldsymbol{x}, a_n^0) = O(n^{-2s/(2s+p)})$.

Since $f(\boldsymbol{x})$ is assumed to be unknown, we must estimate $C(\boldsymbol{x})$ in some way. The method applied here is to approximate $f(\boldsymbol{x})$ with $f_n(\boldsymbol{x})$. For this purpose, consider two sequences $\{\tau_{n,i}\}, \{b_n\}$.

$$\{\tau_{n,0}\} : O(n^\gamma) \qquad \{\tau_{n,i}\} : O(n^{\gamma^2}) \quad (i = 1, \cdots, p), \tag{19}$$

$$\{b_n\} : b_n \to 0 \quad (n \to \infty), \quad nb_n \geq C > 0. \tag{20}$$

We approximate $C(\boldsymbol{x})$ with those sequences as follows.

$$\hat{C}(\boldsymbol{x}) = \left(\frac{2s\alpha^2}{(s!)^2 p \|K\|^{2p}} \frac{(\sum_i \partial_i^s f_n(\boldsymbol{x}, \tau_{n,i}))^2 + b_n}{|f_n(\boldsymbol{x}, \tau_{n,0})| + b_n} \right)^{\gamma}. \tag{21}$$

Then an approximation of a_n^0 is obtained as below,

$$\hat{a_n^0}(\boldsymbol{x}) = \hat{C}(\boldsymbol{x}) n^{\gamma}, \tag{22}$$

and the following theorem is obtained.

Theorem 3 *Under the condition of theorem 2 and moreover let $K(x)$ possess bounded, integrable s-th derivatives and*

$$\int x^s K^{(s)}(x)\, dx < \infty. \tag{23}$$

If $K^*(x)$ below admits nonincreasing and integrable majorant $K_0(x)$ on $[0, \infty)^p$,

$$K^*(x) \equiv p \prod_{j=1}^{p} K(x_j) + \sum_{i=1}^{p} (\prod_{\substack{j=1 \\ j \neq i}}^{p} K(x_j)) K^{(1)}(x_i) x_i, \tag{24}$$

then the following equation is valid for $\hat{a_n^0}$ defined as (22).

$$U(\hat{a_n^0}) \simeq U(a_n^0) = O(n^{-2s/(2s+p)}). \tag{25}$$

Appendix B Proof of theorem 1

The following lemma is valid.

Lemma 1 *Let p-dimensional density function $P(\boldsymbol{z})$ be s times differentiable for any variable and let the domain region X be bounded. Then for large n,*

$$\mathrm{E}[\hat{I}(\alpha) - I(\alpha)]^2 \leq \frac{1}{4} D_1(\alpha) D_2 n^{-2s/(2s+p)}, \tag{26}$$

is valid, where E denotes the expectation

$$\mathrm{E}[\,\cdot\,] = \int \cdot \prod_j P(\underset{(j)}{\boldsymbol{Z}})\, d\underset{(j)}{\boldsymbol{Z}},$$

and $D_1(\alpha)$ denotes

$$D_1(\alpha) = \int_X Q(\boldsymbol{z}, \alpha)^2\, d\boldsymbol{z}, \tag{27}$$

and the value D_2 depends on $P(\boldsymbol{z}), p$ and s.

Proof of lemma 1 Let $P(\boldsymbol{z})$ have a bounded domain region, namely, a set $\{\,\boldsymbol{z} \mid P(\boldsymbol{z}) > 0\}$ belongs to a bounded set X. Denote

$$V(\alpha) = \mathrm{E}[\hat{I}(\alpha) - I(\alpha)]^2 = \mathrm{E}[\int_X Q(\boldsymbol{z},\alpha)(\hat{P}(\boldsymbol{z}) - P(\boldsymbol{z}))d\boldsymbol{z}]^2. \tag{28}$$

Then, from Cauchy-Schwartz's inequality and exchanging operation of integrations,

$$V(\alpha) \le \int_X Q(\boldsymbol{z},\alpha)^2\,d\boldsymbol{z} \int_X \mathrm{E}[\hat{P}(\boldsymbol{z}) - P(\boldsymbol{z}))]^2\,d\boldsymbol{z}. \tag{29}$$

The second factor of right hand side can be estimated by (18), theorem 2 and theorem 3. Thus

$$V(\alpha) \le \frac{1}{4} D_1(\alpha) D_2 n^{-2s/(2s+p)} \tag{30}$$

is obtained, where

$$D_1(\alpha) = \int_X Q(\boldsymbol{z},\alpha)^2\,d\boldsymbol{z} \tag{31}$$

$$D_2 = 4(p+2s)\int_X (\frac{\|K\|^{2p}}{2s} P(\boldsymbol{z}))^{2s\gamma} (\frac{1}{s!\sqrt{p}} \sum_i \partial_i^s P(\boldsymbol{z}))^{2p\gamma}\,d\boldsymbol{z}. \tag{32}$$

Q.E.D.

Proof of theorem 1 It is easy to show the inequality

$$|I(\hat{\alpha}_0) - I(\alpha_0)| < |I(\hat{\alpha}_0) - \hat{I}(\hat{\alpha}_0)| + |I(\alpha_0) - \hat{I}(\alpha_0)|, \tag{33}$$

where $\hat{\alpha}_0$ is the α that minimizes $\hat{I}(\alpha)$ and α_0 is the α that minimizes $I(\alpha)$.

On the other hand, if the inequality $0 \le x < y+z$ is satisfied, $x^2 < (y+z)^2 \le 2(y^2+z^2)$ holds in general. Thus the following inequality is valid.

$$\mathrm{E}[I(\hat{\alpha}_0) - I(\alpha_0)]^2 \le 2\left(\mathrm{E}[\hat{I}(\alpha_0) - I(\alpha_0)]^2 + \mathrm{E}[\hat{I}(\hat{\alpha}_0) - I(\hat{\alpha}_0)]^2\right). \tag{34}$$

Consequently from lemma 1, we have

$$\mathrm{E}[I(\hat{\alpha}_0) - I(\alpha_0)]^2 \le 4 \sup_\alpha V(\alpha). \tag{35}$$

The theorem is proved. Q.E.D.

In lemma 1, we assumed that the domain region is bounded. Let us consider the case that it is not bounded. Even in this case, we can always take a bounded X which satisfies for any β,

$$\int_{X^c} |\hat{P}(\boldsymbol{z})| + P(\boldsymbol{z})\,d\boldsymbol{z} = O(n^\beta). \tag{36}$$

Since X depends on n in general, the value

$$\int_X Q(\boldsymbol{z},\alpha)^2\,d\boldsymbol{z} \tag{37}$$

becomes $g(n;\beta)$, which is a function of n and β. Take X so that $g(n;\beta)$ is as small as possible. Let us assume Q is bounded on X^c. Then

$$\begin{aligned} V(\alpha) &\le \mathrm{E}[\int_X Q(\boldsymbol{z})|P(\boldsymbol{z}) - \hat{P}(\boldsymbol{z})|\,d\boldsymbol{z} + Q_{\max} n^\beta]^2 \\ &\le 2\,\mathrm{E}[\int_X Q(\boldsymbol{z})|P(\boldsymbol{z}) - \hat{P}(\boldsymbol{z})|]^2\,d\boldsymbol{z} + 2Q_{\max}^2 n^{2\beta} \qquad (38) \\ &= O(g(n) n^{-2s/(2s+p)} + n^{2\beta}) \qquad (39) \end{aligned}$$

As β is decreasing, $g(n;\beta)$ decreases. Hence, let β_0 be a solution of the following equation,

$$2\beta = \log_n g(n;\beta) - \frac{2s}{2s+p}, \tag{40}$$

$V(\alpha)$ is bounded by

$$V(\alpha) \leq O(n^{2\beta_0}). \tag{41}$$

References

[1] E.B. Baum and D. Haussler: What size net gives valid generalization? *Neural Computation*, Vol. 1, pp. 151–160, 1989.

[2] A. Blumer, A. Ehrenfeucht, D. Haussler, and M.K. Warmuth: Learnability and the Vapnik-Chervonenkis dimension. *J. of the Assoc. for Comp. Machinery*, pp. 929–965, 1989.

[3] P. Craven and G. Wahba: Smoothing noisy data with spline functions. *Numerische Mathematik*, Vol. 31, pp. 377–403, 1979.

[4] K. Hornik, M. Stinchcombe, and H. White: Multilayer feedforward networks are universal approximators. *Neural Networks*, Vol. 2, pp. 359–366, 1989.

[5] T. Kurita: An attempt on model selection for neural networks. In *IEICE Technical Report* **PRU**89–16, 1989. In Japanese.

[6] E.A. Nadaraya: *Nonparametric estimation of probability densities and regression curves.* Kluwer Academic Publishers, 1989.

[7] T. Poggio: Networks for approximation and learning. *Proc. IEEE*, Vol. 78, No. 9, pp. 1481–1496, 1990.

[8] A.N. Tikhonov and V.Ya. Arsenin: *Solutions of Ill-posed Problems.* Winston, Washington, 1977.

[9] V.A. Vapnik: *Estimation of Dependences Based on Empirical Data.* Springer-Verlag, 1984.

[10] K. Yamanishi: Learning non-parametric-densities using finite-dimensional parametric hypotheses. In *Proc. of ALT '91*, pp. 175–186, 1991.