IEEE Transactions on Neural Networks14(6) : 1553-1559 (2004).

NO FREE LUNCH WITH THE SANDWICH

Isabelle Rivals and Léon Personnaz

Équipe de Statistique Appliquée,
École Supérieure de Physique et de Chimie Industrielles,
10, rue Vauquelin - 75231 Paris cedex 05 - France.
E-mail: Isabelle.Rivals@espci.fr

***Abstract* – In nonlinear regression theory, the sandwich estimator of the covariance matrix of the model parameters is known as a consistent estimator, even when the parameterized model does not contain the regression. However, in the latter case, we emphasize the fact that the consistency of the sandwich holds only if the inputs of the training set are the values of independent identically distributed random variables. Thus, in the frequent practical modeling situation involving a training set whose inputs are deliberately chosen and imposed by the designer, we question the opportunity to use the sandwich estimator rather than the simple estimator based on the inverse squared Jacobian.**

***Index Terms* – confidence intervals, experimental design, fixed inputs, heteroscedasticiy, i.i.d. random inputs, i.n.i.d. random inputs, least squares estimation, linear Taylor expansion, model misspecification, neural networks, parameter covariance matrix, nonlinear regression, sandwich estimator.**

1. Motivation

In some statistical tests for the comparison between candidate neural models, for the estimation of a confidence interval for the conditional mean of the process output, or for the detection of outliers, an estimate of the covariance matrix of the network parameters is needed. Various estimators have been established, among them:

a) the estimator based on the inverse squared Jacobian (ISJ), which is consistent if the parameterized model contains the regression and if the noise is homoscedastic; this consistency holds, be the input values of the training set fixed, or realizations of independent identically distributed (i.i.d.) random variables, or even realizations of independent not identically distributed (i.n.i.d.) random

variables.

b) the sandwich estimator, which is consistent regardless of whether the noise is homoscedastic or not, if the parameterized model contains the regression. If it does not, the sandwich estimator is proved to be consistent *if the inputs of the training set are the values of i.i.d. random variables*, a fact that is seldom clearly mentioned in the literature.

Thus, we question the opportunity of using the sandwich estimator in frequent practical situations of industrial process modeling, where the input values of the training set are deliberately chosen and imposed by the designer, and hence cannot be considered as realizations of i.i.d. random variables.

In section 2, we summarize the modeling framework of the ISJ estimator, and in section 3, that of the sandwich estimator. In section 4, we consider a model with a single parameter that does not contain the regression, in the case of fixed inputs, and we show analytically that the sandwich estimator is not consistent. In section 5, depending on the designer's choice and control of the inputs, we discuss whether it is appropriate to consider the inputs of the training set as fixed, random i.n.i.d., or random i.i.d. In the simulations of section 6, we compare the ISJ and the sandwich estimators of the variance of a model output, when the parameterized model contains the regression and when it does not, and in various situations concerning the designer's choice and control of the inputs.

2. Modeling framework for the ISJ estimator

This framework considers a scalar output Y which is a random variable[1] depending on a n-input vector $\boldsymbol{x} = [x_1\ x_2\ \dots\ x_n]^T$, which is either fixed or the realization of a random vector $\boldsymbol{X}$. We assume a regression model, so that for any value of $\boldsymbol{x}$:

$$Y = \rho(\boldsymbol{x}) + W \tag{1}$$

where ρ is the unknown regression function, and W is a random variable with zero expectation and finite variance, modeling the unexplainable part of the process output. The goal is to approximate the regression ρ using a parameterized model $\{f(\boldsymbol{x}, \boldsymbol{\theta}), \boldsymbol{x} \in \mathbb{R}^n, \boldsymbol{\theta} \in \mathbb{R}^q\}$, a neural network for instance. If there exists a true parameter value $\boldsymbol{\theta}_t$ of $\boldsymbol{\theta}$ such that $f(\boldsymbol{x}, \boldsymbol{\theta}_t) = \rho(\boldsymbol{x})$ in the input domain of interest, the parameterized model contains the regression, and is said to be true; if not, it is said to be wrong.

A training set $\{\boldsymbol{x}^k, y^k\}_{k=1 \text{ to } N}$ is assumed to be available. A least squares (LS)

---

[1] Notations: we distinguish between random variables and their values (or realizations) by using upper- and lowercase letters; all vectors are column vectors, and are denoted by boldface letters, e.g. the n-vectors $\boldsymbol{x}$ and $\{\boldsymbol{x}^k\}$; matrices are denoted by courier letters, e.g. the (N,q) matrix $\mathtt{z}_N$.

parameter estimate $\boldsymbol{\theta}_N$ minimizes the empirical cost function:

$$l_N(\boldsymbol{\theta}) = \frac{1}{N}\sum_{k=1}^{N}\frac{1}{2}\left(y^k - f\left(\boldsymbol{x}^k, \boldsymbol{\theta}\right)\right)^2 \tag{2}$$

The ISJ estimate of the covariance matrix of $\boldsymbol{\theta}_N$ is given by:

$$\widehat{K(\boldsymbol{\Theta}_N)}_{ISJ} = s_N^2\left(z_N{}^T\, z_N\right)^{-1} \tag{3}$$

where z_N is the (N, q) Jacobian matrix evaluated at $\boldsymbol{\theta}_N$ with elements:

$$\left[z_N\right]_{ki} = \left.\frac{\partial f\left(\boldsymbol{x}^k, \boldsymbol{\theta}\right)}{\partial \theta_i}\right|_{\boldsymbol{\theta}=\boldsymbol{\theta}_N} \tag{4}$$

and s_N^2 is obtained with the residual N-vector $\boldsymbol{r}_N$, whose components are $\left\{r_N^k = y^k - f\left(\boldsymbol{x}^k, \boldsymbol{\theta}_N\right)\right\}$:

$$s_N^2 = \frac{\boldsymbol{r}_N{}^T\,\boldsymbol{r}_N}{N-q} \tag{5}$$

If the parametrized model is true, if the noise is homoscedastic with variance σ^2, and under appropriate regularity conditions, the ISJ covariance estimator corresponding to the estimate (3) is consistent, be the input values of the training set fixed [Seber & Wild 1989], realizations of i.i.d. random variables [White 1982], or of i.n.i.d. random variables [White 1980].

In the case of a black-box model such as a neural network, one is usually not directly interested in the particular values of its parameters nor in their covariance; one is rather interested in the output value (the point estimate of the regression) and in the variance of this output $var\left(f(\boldsymbol{x}, \boldsymbol{\Theta}_N)\right)$, for example in order to compute a confidence interval for the regression value $\rho(\boldsymbol{x})$ or to detect outliers. Using the ISJ estimate (3) of the covariance matrix of the parameters, the model output variance at a given input $\boldsymbol{x}$ is estimated with:

$$\widehat{var\left(f(\boldsymbol{x}, \boldsymbol{\Theta}_N)\right)}_{ISJ} = \boldsymbol{z}^T\,\widehat{K(\boldsymbol{\Theta}_N)}_{ISJ}\,\boldsymbol{z} = s_N^2\;\boldsymbol{z}^T\left(z_N{}^T\, z_N\right)^{-1}\boldsymbol{z} \tag{6}$$

where:

$$\boldsymbol{z} = \left.\frac{\partial f(\boldsymbol{x}, \boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\right|_{\boldsymbol{\theta}=\boldsymbol{\theta}_N} \tag{7}$$

However, if the parametrized model is wrong or if the noise is heteroscedastic, the estimator corresponding to the ISJ estimate (3) is not consistent, nor is the output variance estimator based upon it. Like for example in [Tibshirani 1996] and [Anders & Korn 1999], one could hence be tempted to use the sandwich estimate of $K(\boldsymbol{\Theta}_N)$.

3. Modeling framework for the sandwich estimator

We recall the framework of the derivation of this estimator, as described in [White

1989][2]. Both the inputs and the outputs are considered as random. The joint behavior of $\boldsymbol{X}$ and Y is supposed to be described by a joint probability law ν, and an input-output couple is denoted by the (n+1)-vector $\boldsymbol{U} = [\boldsymbol{X}^T\ Y]^T$.

It is assumed that there exists a vector $\boldsymbol{\theta}^*$ (supposed unique for simplicity) that minimizes the theoretical LS cost function:

$$\lambda(\boldsymbol{\theta}) = \int l(\boldsymbol{u}, \boldsymbol{\theta})\, \nu(d\boldsymbol{u}) = \int \frac{1}{2}(y - f(\boldsymbol{x}, \boldsymbol{\theta}))^2\, \nu(d\boldsymbol{u}) \tag{8}$$

The empirical LS cost function (2), defined on a sample $\{\boldsymbol{u}^k\}_{k=1 \text{ to } N}$ of realizations of i.i.d. variables $\{\boldsymbol{U}^k\}_{k=1 \text{ to } N}$, is a realization of the random variable:

$$L_N(\boldsymbol{\theta}) = \frac{1}{N}\sum_{k=1}^{N} l\left(\boldsymbol{U}^k, \boldsymbol{\theta}\right) = \frac{1}{N}\sum_{k=1}^{N} \frac{1}{2}\left(Y^k - f\left(\boldsymbol{X}^k, \boldsymbol{\theta}\right)\right)^2 \tag{9}$$

The estimator $\boldsymbol{\Theta}_N$, whose realization $\boldsymbol{\theta}_N$ minimizes (2), is a strongly consistent estimator of $\boldsymbol{\theta}^*$.

We pose $a^* = E\left[\nabla^2 l(\boldsymbol{U}, \boldsymbol{\theta}^*)\right]$ and $b^* = E\left[\nabla l(\boldsymbol{U}, \boldsymbol{\theta}^*)\, \nabla l(\boldsymbol{U}, \boldsymbol{\theta}^*)^T\right]$, where ∇ and ∇^2 denote the gradient and the Hessian operators, and $c^* = a^{*-1}\, b^*\, a^{*-1}$. The limiting distribution of $\sqrt{N}\,(\boldsymbol{\Theta}_N - \boldsymbol{\theta}^*)$ is $\mathcal{N}(\boldsymbol{0}, c^*)$ [White 1989].

The sandwich estimator of the covariance matrix $K\left(\sqrt{N}\,\boldsymbol{\Theta}_N\right)$ is $C_N = A_N^{-1}\, B_N\, A_N^{-1}$ with:

$$A_N = \frac{1}{N}\sum_{k=1}^{N} \nabla^2 l\left(\boldsymbol{U}^k, \boldsymbol{\Theta}_N\right) \quad ; \quad B_N = \frac{1}{N}\sum_{k=1}^{N} \nabla l\left(\boldsymbol{U}^k, \boldsymbol{\Theta}_N\right) \nabla l\left(\boldsymbol{U}^k, \boldsymbol{\Theta}_N\right)^T \tag{10}$$

The estimator C_N is a strongly consistent estimator of c^*. The sandwich estimator was proved to be consistent regardless of whether the noise is homoscedastic or not, if the parameterized model is true [White 1982, 1989]. If the parameterized model is wrong, the consistency holds only if the inputs of the training set are the values of i.i.d. random variables.

Hence, with the notations of the previous sections, the sandwich estimate of the covariance matrix $K(\boldsymbol{\Theta}_N)$ is:

$$\widehat{K(\boldsymbol{\Theta}_N)}_{SAN} = \frac{1}{N}\, a_N^{-1}\, b_N\ a_N^{-1} = \frac{1}{N^2}\, a_N^{-1}\left(\sum_{k=1}^{N} \left(r_N^k\right)^2 \boldsymbol{z}_N^k \left(\boldsymbol{z}_N^k\right)^T\right)\ a_N^{-1} \tag{11}$$

where the $\left\{\left(\boldsymbol{z}_N^k\right)^T\right\}$ are the rows of the matrix z_N, and a_N and b_N are the realizations of the random matrices A_N and B_N. The model output variance at $\boldsymbol{x}$ is estimated with:

$$\widehat{var\left(f(\boldsymbol{x}, \boldsymbol{\Theta}_N)\right)}_{SAN} = \boldsymbol{z}^T \widehat{K(\boldsymbol{\Theta}_N)}_{SAN}\, \boldsymbol{z} = \frac{1}{N^2}\, \boldsymbol{z}^T\, a_N^{-1}\left(\sum_{k=1}^{N} \left(r_N^k\right)^2 \boldsymbol{z}_N^k \left(\boldsymbol{z}_N^k\right)^T\right)\ a_N^{-1}\, \boldsymbol{z} \tag{12}$$

---

[2] We try to be as close as possible to White's notations, while sticking to uppercase letters for random variables only, to boldface letters for vectors, and to courier letters for matrices.

4. Example of the non-consistency of the sandwich estimator in the case of a wrong model, and of fixed inputs

We consider a training set obtained on a SISO process simulated with:

$$y^k = \rho + w^k \quad k=1 \text{ to } N \quad \rho = cte \neq 0$$

The noise is homoscedastic with variance σ^2.

The chosen predictive model is linear in x:

$$f(x, \theta) = \theta\, x$$

Thus, this model is wrong. The LS estimate of the parameter θ is:

$$\theta_N = \frac{\sum_{k=1}^{N} x^k\, y^k}{\sum_{k=1}^{N} (x^k)^2}$$

The $\{x^k\}$ are fixed; we further choose them centered, that is $\sum_{k=1}^{N} x^k = 0$. For the sandwich estimator C_N of $var(\sqrt{N}\,\Theta_N)$, we obtain:

$$\begin{cases} A_N = a_N = \frac{1}{N}\sum_{k=1}^{N} (x^k)^2 = \Sigma_{x_N}^2 \\ B_N = \frac{1}{N}\sum_{k=1}^{N} \left(x^k \left(Y^k - x^k\, \Theta_N\right)\right)^2 \end{cases}$$

We have:

$$E(B_N) = \Sigma_{x_N}^2 (\rho^2 + \sigma^2) - \frac{1}{N}\frac{\Sigma_{x_N}^4}{\Sigma_{x_N}^2}\,\sigma^2 \quad \text{with} \quad \Sigma_{x_N}^4 = \frac{1}{N}\sum_{k=1}^{N} (x^k)^4$$

Hence:

$$E(C_N) = \frac{\rho^2 + \sigma^2}{\Sigma_{x_N}^2} - \frac{1}{N}\frac{\Sigma_{x_N}^4}{(\Sigma_{x_N}^2)^3}\,\sigma^2 \tag{13}$$

But the variance of $\sqrt{N}\,\Theta_N$ is given by:

$$var(\sqrt{N}\,\Theta_N) = N\frac{\sum_{k=1}^{N} (x^k)^2\, var(y^k)}{\left(\sum_{k=1}^{N} (x^k)^2\right)^2} = \frac{N\sigma^2}{\sum_{k=1}^{N} (x^k)^2} = \frac{\sigma^2}{\Sigma_{x_N}^2} \tag{14}$$

Hence, $E(C_N)$ given by (13) does not converge to the true value of the parameter variance (14).

We choose the fixed $\{x^k\}$ regularly spaced in $[-\sqrt{3}\,;\sqrt{3}]$, $\rho = 1$ and $\sigma^2 = 1$. The results are summarized in Table 1. As shown above, the sandwich estimator C_N is a not a consistent estimator of the variance of $\sqrt{N}\,\Theta_N$.

This example shows that the property of the sandwich to be consistent, even if the model is wrong, does not hold if the input values of the training set are fixed. We

insist on this point, because it is not clearly mentioned in the literature, see for example [Kauermann & Carroll, 2001], where the inputs are fixed.

N	$var\left(\sqrt{N}\,\Theta_N\right)$	$E(C_N)$
20	0,90	1,73
100	0,98	1,94
1000	1,00	1,99
10 000	1,00	2,00

Table 1. The input values of the training set are fixed, regularly spaced in $\left[-\sqrt{3}\ ;\sqrt{3}\right]$: the sandwich estimator C_N of $var\left(\sqrt{N}\,\Theta_N\right)$ is not consistent.

5. Fixed versus independent identically (or not) distributed random inputs, and experimental context

The consistency of the ISJ and sandwich estimators rely on different assumptions. Be the inputs be fixed, random i.n.i.d., or random i.i.d., the ISJ estimator is consistent if the parameterized model is true and if the noise is homoscedastic. Regardless of whether the noise is homoscedastic or not, the sandwich estimator is consistent even if the model is wrong, but in the latter case, the consistency holds only if the inputs are random i.i.d. variables.

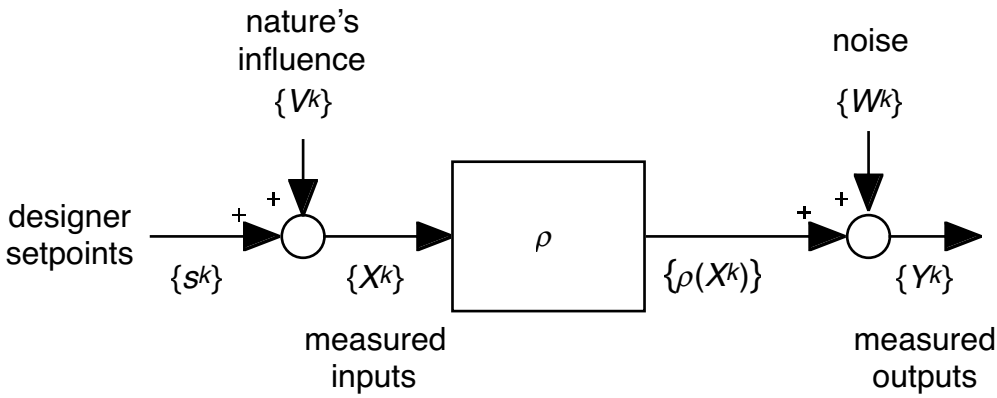

Figure 1. Experimental modeling situations and kind of the training set inputs. The setpoints of the designer are modeled by the fixed $\{s^k\}$. Nature's influence is modeled by the random, i.i.d $\{V^k\}$. The actual, measured process inputs $\{X^k\}$, are a combination of the designer setpoints and of nature's influence. Their respective weights depend on the experimental situation: a) the designer has no control on the inputs, which are collected according to their common distribution during the operation of the process (the $\{X^k = V^k\}$ are random, i.i.d variables) ; b) the designer has complete control on the inputs, and chooses their values deliberately (the $\{x^k = s^k\}$ are fixed) ; c) the designer chooses the inputs deliberately, but does not have complete control on their values (the $\{X^k = s^k + V^k\}$ are random, but i.n.i.d).

We now examine how the inputs should be modeled, according to the design context of the model. As in [White 1989], we assume that the inputs are measured with absolute accuracy[3]. We distinguish between the three following practical situations, as illustrated in Fig. 1:

a) The designer does not choose the training set inputs, which are collected according to their common distribution during the operation of the process. The designer has no control on the values of the inputs, like for an economical process or a natural process (meteorology prediction, snow avalanche forecasting, prediction of the economic growth of a country). The designer collects the input-output couples during the operation of the process, and does not make any further selection among this data for the constitution of the training set. Thus, the input values of the training set are drawn according to the distribution that corresponds to the natural operation of the process. They could have been different while still drawn from the same distribution, and should hence be considered as realizations of identically distributed random variables. In such a case, the designer is interested in the variance of the model output, due not only to the variability of the outputs of the training set, but also to that of its inputs. In this context, the sandwich estimator is consistent, and one can expect it to give correct results for large N, even in the case of a wrong parameterized model or of heteroscedasticity. If the model is true and if the noise is homoscedastic, the ISJ estimator is also consistent.

b) The designer chooses and has complete control on the input values. This situation arises in the context of the development phase of an industrial process, or of laboratory experiments. In such cases, the designer is free to choose the training set inputs, for example according to an experimental design that is optimal in some sense (e.g. that minimizes the width of the confidence intervals for the regression). Then, there is no reason to consider that the inputs of the training set could have been different. The input values of the training set should be considered as fixed, and the designer is interested in the variance of the model output due to the variability of the training set outputs only. In this context, the ISJ estimator as well as the sandwich will give correct results when the model is true, the noise is homoscedastic, and N is large. But both estimators will not be consistent in the case of a wrong model.

Note that our interpretation differs from that of White in [White 1989]. White considers that, even when the designer has complete control on the inputs of the training set, they should still be considered as random, identically distributed variables, with a discrete distribution defined by "the relative frequencies with which different values

---

[3] Input measurement error is taken into account in the so-called "errors-in-variables" models, see for instance [Seber & Wild 1989]

for [the inputs] are set". This would mean that the designer assigns given probabilities to a finite set of input values, and picks the N input values of the training set randomly among them, according to the chosen probabilities. To our knowledge, when the designer has complete control on the inputs, he does not pick them at random, but imposes their values.

c) The designer chooses the inputs, but has not complete control on their values. This intermediary situation arises in the case of a process whose inputs are controlled by the designer, but up to a certain extent. For example, consider a process for which some input temperature is controlled by a regulator. A setpoint for the temperature can be fixed, but, due to disturbances, the actual value of the temperature, which is the real input, is slightly different from the setpoint. The temperature is measured accurately, but it is not set with absolute control. This situation can also arise in the case of an economical or ecological process for which the designer has no control on the inputs, but is free to choose the training set inputs among the numerous available ones, close to values determined by an experimental design. In that case, the inputs should be considered as random, non identically distributed variables. As in the context of fixed inputs, both the ISJ and the sandwich estimators are consistent if the model is true and the noise is homoscedastic; but both are not if the model is wrong.

These situations are summarized in Table 2.

		true parameterized model		wrong model
Inputs	Estimator	homoscedasticity	heteroscedasticity	
fixed or random i.n.i.d.	ISJ	**yes**	**no**	**no**
	SAND	**yes**	**yes**	**no**
random i.i.d.	ISJ	**yes**	**no**	**no**
	SAND	**yes**	**yes**	**yes**

Table 2. Consistency (**yes** or **no**) of the ISJ and sandwich estimators.

6. Illustrative simulation example

In order to compare the accuracy of the ISJ estimate (6) and of the sandwich estimate (13) of the variance of $f(\boldsymbol{x}, \boldsymbol{\Theta}_N)$, we need a reference estimate that is not model dependent. For a simulated process, it can be obtained with a large number M of realizations of the training set. The i-th LS estimate $f\left(\boldsymbol{x}, \boldsymbol{\theta}_N^{(i)}\right)$ of $\rho(\boldsymbol{x})$ is computed with the i-th training set (i=1 to M), and a good estimate of the variance at input $\boldsymbol{x}$ is computed according to:

$$\frac{1}{M}\sum_{i=1}^{M}\left(f\left(\boldsymbol{x}, \boldsymbol{\theta}_N^{(i)}\right) - \langle f(\boldsymbol{x})\rangle\right)^2 \text{, where } \langle f(\boldsymbol{x})\rangle = \frac{1}{M}\sum_{i=1}^{M} f\left(\boldsymbol{x}, \boldsymbol{\theta}_N^{(i)}\right) \qquad (15)$$

This estimate is not affected by a wrong parametrization, and it is not biased by curvature effects; only a sufficiently large M is needed. In the following, this reference estimate is called the "true variance" (with quotation marks).

We consider three different "distributions" of the inputs of the training set corresponding to the three situations a), c) and b) described in section 5:

– random, i.i.d. inputs (the training set is collected during the natural operation of the process);
– random i.n.i.d. inputs, centered around fixed values (the designer chooses the training set inputs deliberately, but has incomplete control on their values);
– fixed inputs (the designer chooses the training set inputs deliberately, and has complete control on their values).

These situations are considered first in the case of a wrong model, and then in the case of a true one, the noise being homoscedastic in both cases.

6.1 Case of a wrong model

We consider a nonlinear SISO process simulated with:

$$y^k = tanh\left(x^k\right) + 1.5\ exp\left(-8\left(x^k\right)^2\right) + w^k \qquad k=1 \text{ to } N \tag{16}$$

where the $\{w^k\}$ are the values of i.i.d. centered gaussian variables with variance $\sigma^2 = 0.01$, and $N = 100$. The predictive model is a network consisting of a single neuron with hyperbolic tangent activation function:

$$f(x, \theta) = tanh\ (\theta_0 + \theta_1\ x) \tag{17}$$

Thus, this model is wrong.

a) Random i.i.d. inputs

The $\{x^k\}$ are values of i.i.d. random variables uniformly distributed in [–1.5; 1.5]. The "true variance" (15) is computed on $M = 10^4$ training sets *obtained for different realizations of the inputs and of the corresponding outputs*. For each training set, several trainings with the Levenberg-Marquardt algorithm are performed for the estimation of the parameters in order to reach a global minimum. The means and the standard deviations of the ISJ estimate (6) and of the sandwich estimate (12) are estimated with the M sets, in the input domain [–1.5; 1.5]. The results obtained are shown on Fig. 2. Since we are in the conditions of White's theorems, the mean of the sandwich estimate (12) is indeed very close to the "true variance" (15) (Fig. 2b), whereas the mean of the ISJ estimate (6) is not. Note that the standard deviation of the sandwich is much larger than that of the ISJ estimate (Fig. 2c): it is almost twice as large as the mean.

b) Random i.n.i.d. inputs

The $\{x^k\}$ are values of N independent gaussian variables centered around N regularly spaced values in [-1.5; 1.5], with a variance of 10^{-3} around each value. The “true

variance" is estimated with $M = 10^4$ training sets *obtained for different realizations of the inputs and of the corresponding outputs*. The results obtained are shown on Fig. 3. Since the conditions of White's theorems are not met (the inputs are not i.i.d.), the mean sandwich estimate is less close to the "true variance" than in the previous case (Fig. 3b). Moreover, the standard deviation of the sandwich is much larger than that of the ISJ estimate (Fig. 3c).

c) Fixed inputs

The fixed $\{x^k\}$ are regularly spaced in [-1.5; 1.5]. The "true variance" is estimated with $M = 10^4$ training sets *obtained for different realizations of the outputs only*. The results obtained are shown on Fig. 4. Since the conditions of White's theorems are definitely not met (fixed inputs), the mean sandwich estimate is very different from the "true variance" (Fig. 4b). The standard deviation of the sandwich estimator is, again, much larger than that of the ISJ estimator (Fig. 4c).

6.2 Case of a true model

We now consider a nonlinear SISO process simulated with:

$$y^k = tanh\left(x^k\right) + w^k \quad k = 1 \text{ to } N \tag{18}$$

where the $\{w^k\}$ are, again, the values of i.i.d. centered gaussian variables with variance $\sigma^2 = 0.01$. The predictive model is the single neuron with hyperbolic tangent activation function given by (17): this model is true.

a) Random i.i.d. inputs

The results obtained with $N = 100$ are shown on Fig. 5. The model being true, both the ISJ and the sandwich estimator are consistent, and $N = 100$ proves to be large enough to obtain a good precision with both estimators (Fig. 5b). But the standard deviation of the sandwich estimator is larger than that of the ISJ estimator (Fig. 5c). Moreover, with $N = 10$, the sandwich estimator significantly underestimates the output variance, whereas the ISJ estimator does not. The corresponding results, obtained with $M = 10^5$ realizations of the training set, are shown on Fig. 6.

b) Random i.n.i.d. inputs

The model being true, both the ISJ and the sandwich estimator are consistent. Experimentally, with $N = 100$, the means of both the ISJ and the sandwich estimates are close to the "true variance", and the standard error of the sandwich estimator is larger than that of the ISJ (the results are almost identical to those of Fig. 5). But again, with $N = 10$, the sandwich estimator quite underestimates the output variance, whereas the ISJ estimator does not (the results are between those of Fig. 6 and Fig. 7).

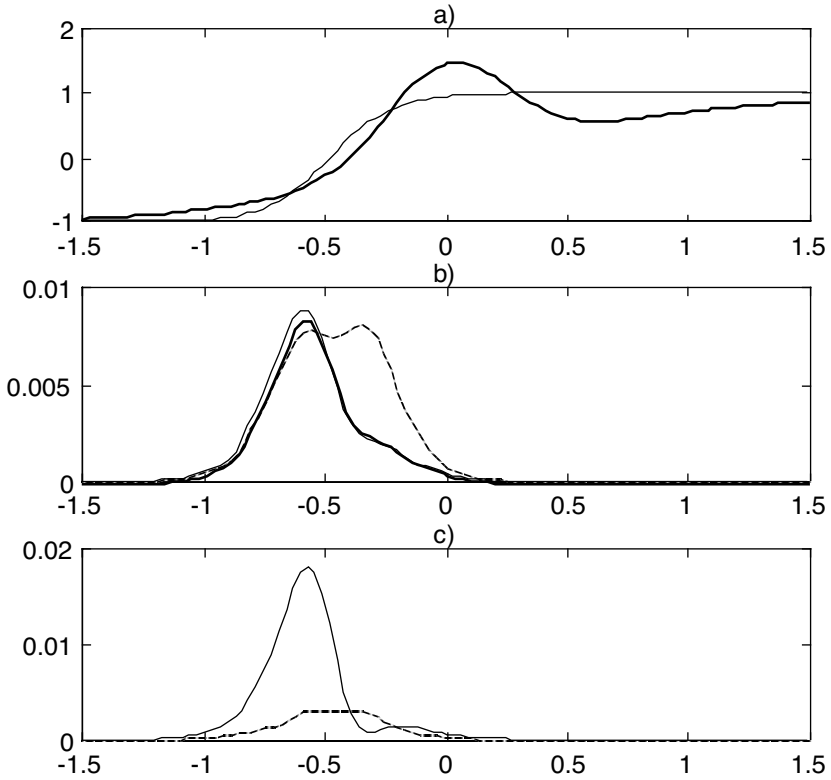

Figure 2. Wrong model, random i.i.d. inputs, $N = 100$: a) regression (thick line), mean output estimate (thin line); b) "true variance" of $f(x, \boldsymbol{\Theta}_N)$ (15) (thick line), mean ISJ variance estimate (6) (dotted line), mean sandwich variance estimate (12) (thin line); c) standard errors of the ISJ (dotted line), and of the sandwich (thin line) variance estimators.

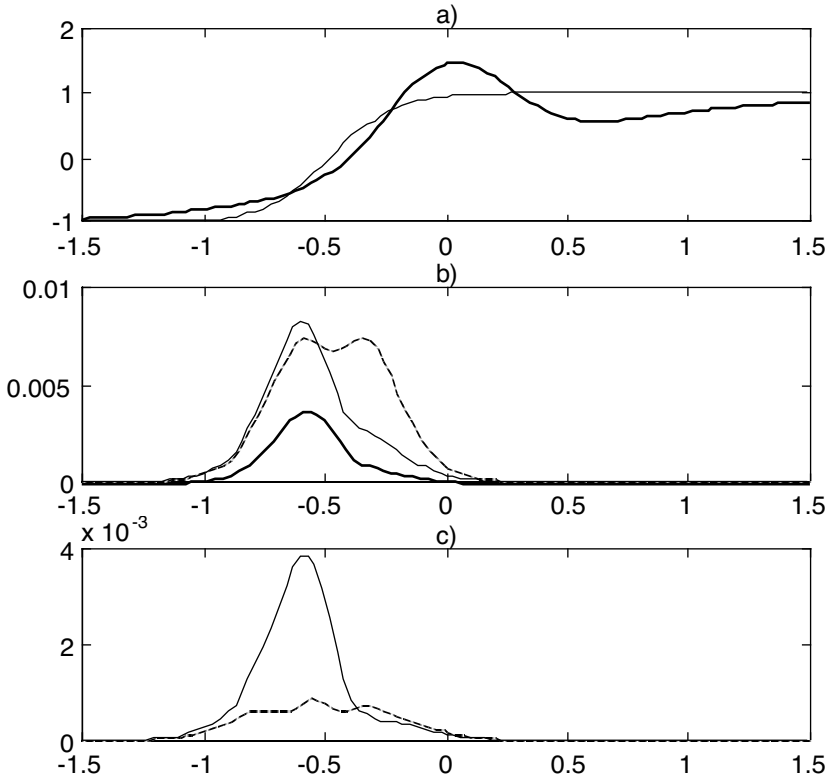

Figure 3. Wrong model, random i.n.i.d. inputs, $N = 100$: same caption as in Fig. 2.

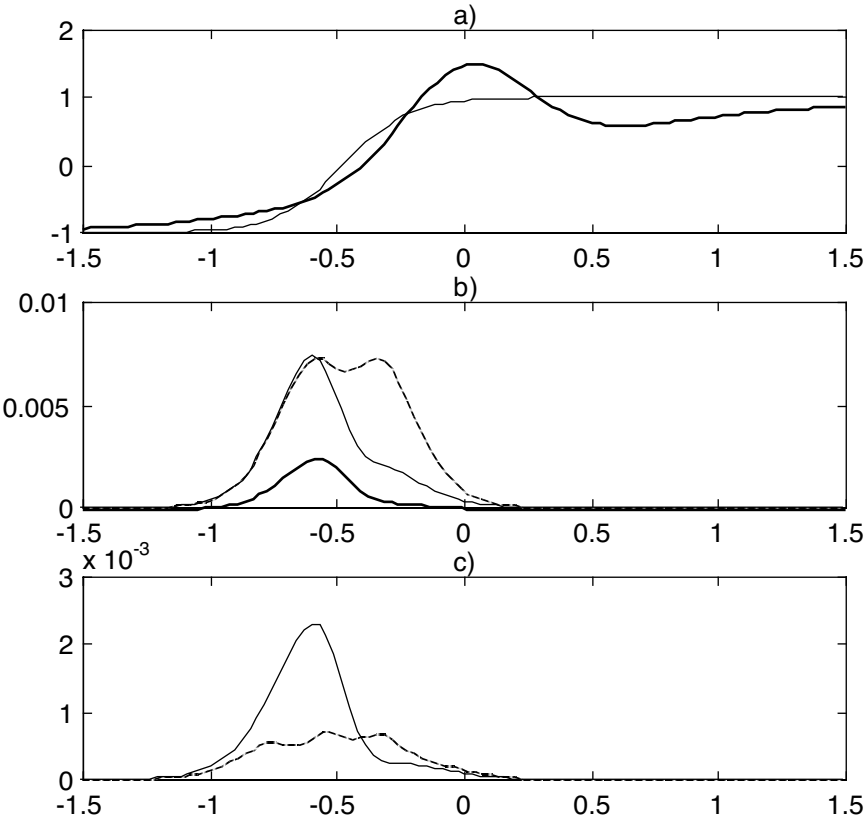

Figure 4. Wrong model, fixed inputs, $N = 100$: same caption as in Fig. 2.

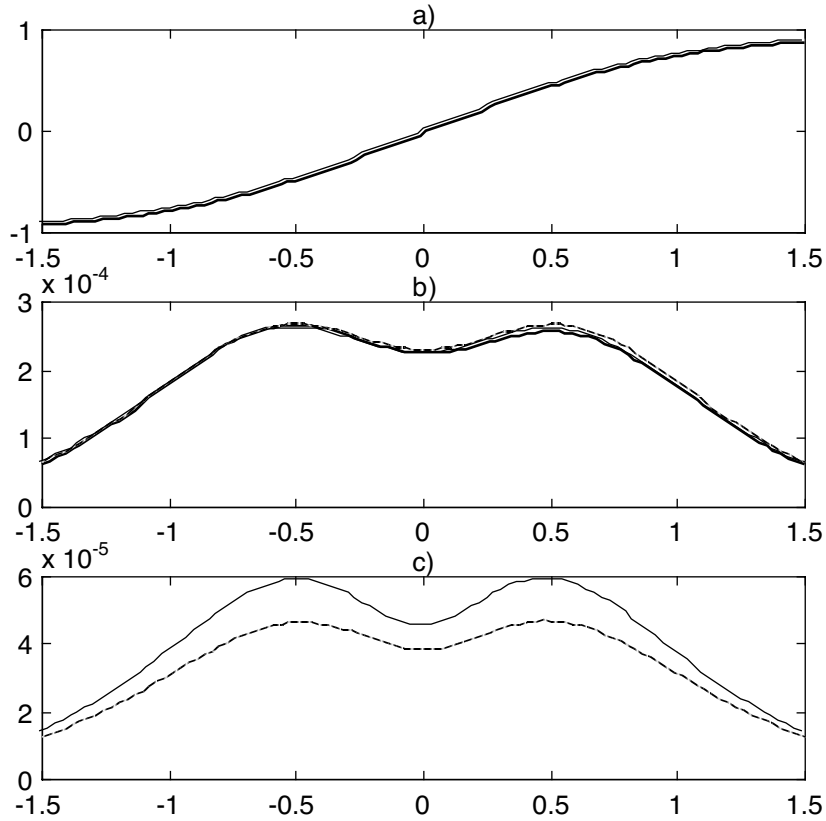

Figure 5. True model, random i.i.d. inputs, $N = 100$: same caption as in Fig. 2.

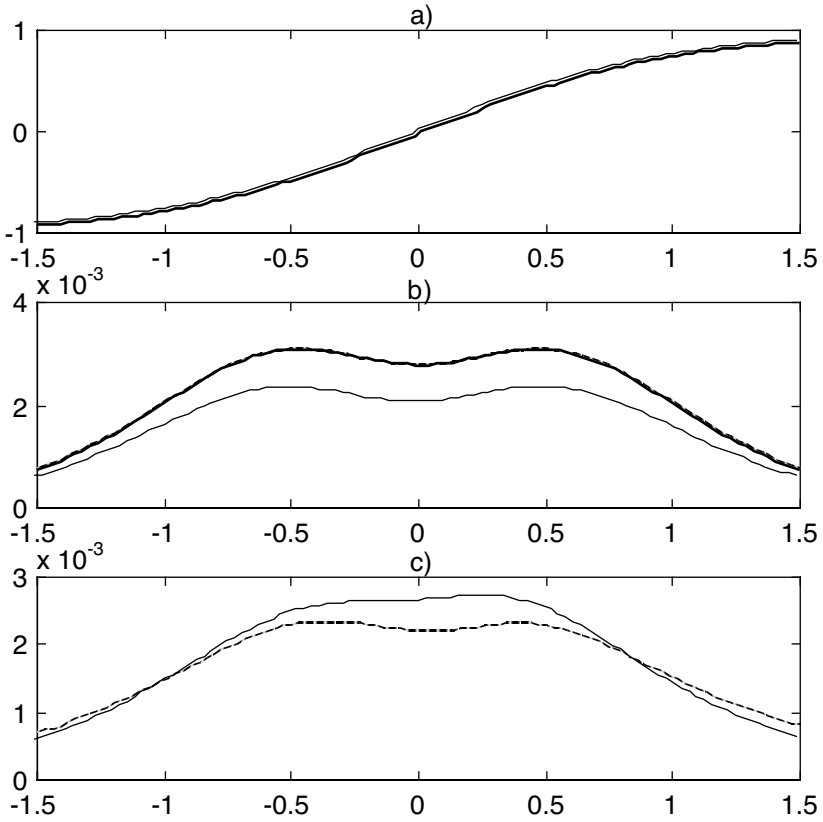

Figure 6. True model, random i.i.d. inputs, $N = 10$: same caption as in Fig. 2.

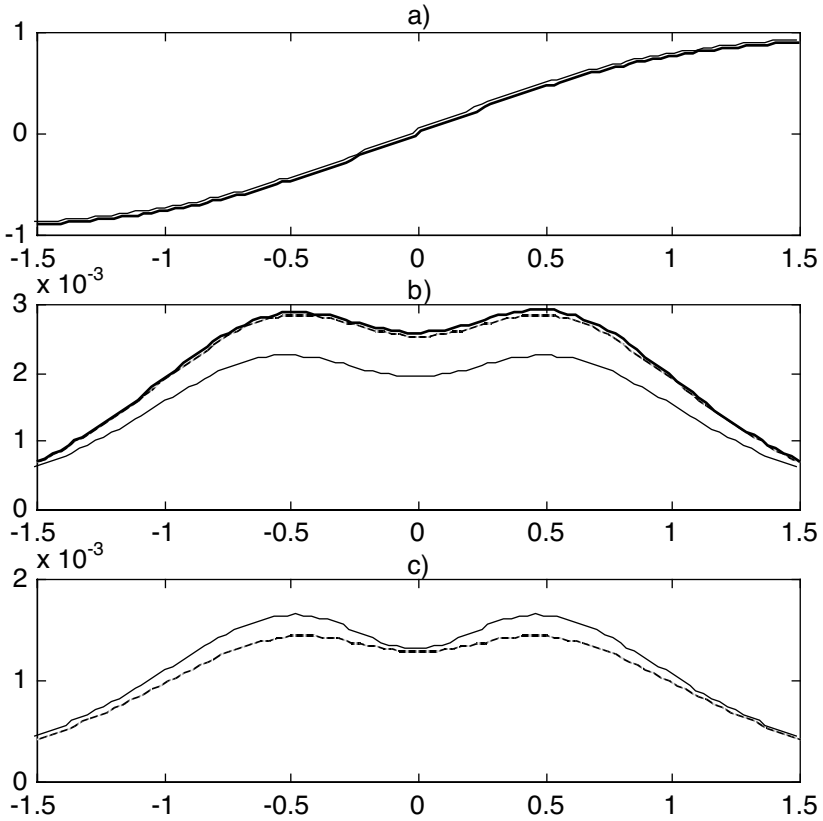

Figure 7. True model, fixed inputs, $N = 10$: same caption as in Fig. 2.

c) Fixed inputs

With $N = 100$, both estimators give good mean results, but the standard error of the sandwich estimator is still larger than that of the ISJ estimator (the results are almost identical to those of Fig. 5). And again, with $N = 10$, the sandwich estimator underestimates the output variance, whereas the ISJ estimator does not, as shown on Fig. 7 ($M = 10^5$).

The fact that the sandwich estimator is biased for small N, and has a larger standard error than the ISJ estimator in the case of a true model and fixed inputs has been proved in [Kauerman & Carroll 2001], for linear and generalized linear models.

7. Conclusion

Let us conclude regarding the opportunity of using the sandwich estimate of the covariance matrix of the parameters rather than the simple ISJ estimate, in the homoscedastic case. This question arises when estimating the variance of the output of a neural network, for example in order to build a confidence interval for the regression, or for the detection of outliers.

In the case where the designer collects the training set input values during the natural operation of the process (i.i.d. random inputs):

a) The sandwich estimator shows an advantage over the ISJ estimator when the parameterized model is wrong, for large N. But, in that case, a confidence interval for the regression is anyway meaningless, since the estimator of the regression is biased. Moreover, when N is large, it should be easy to find a model that contains the regression using a constructive procedure and statistical tests [Rivals & Personnaz, 2003].
b) If the model is true, both the ISJ and the sandwich estimators are consistent. But, in our simulations, we observed that the variance of the ISJ estimator is smaller than that of the sandwich, even for large N; for small N, we also observed a larger bias of the sandwich. However, this remains to be investigated mathematically.

In the case where the designer chooses the training set inputs and imposes their values quite accurately (fixed or i.n.i.d. random inputs):

a) If the model is wrong, both the ISJ and the sandwich estimators are not consistent;
b) If the model is true, both the ISJ and the sandwich estimators are consistent; but, as for i.i.d. inputs, the variance of the ISJ estimator is smaller than that of the sandwich, and the sandwich has a significant bias for small N. This was shown in [Kauerman & Carroll 2001], at least for linear and generalized linear models .

Thus, even in the case of a true model, be the inputs fixed of random i.i.d, there is no real advantage in using the sandwich estimator.

However, it is still interesting to use the sandwich estimator in the case where heteroscedasticity is suspected and where the available information about the noise is not sufficient to perform weighted least squares, or precisely in order to test the homoscedasticity of the noise.

Acknowledgements

We thank the anonymous reviewers, whose constructive comments decisively contributed to renew the matter discussed in this paper.

References

Anders U. & Korn O. (1999). Model selection in neural networks. Neural Networks **12**, 309-323.

Kauerman G. & Carroll R. J. (2001). A note on the efficiency of sandwich covariance matrix estimation. Journal of the American Statistical Association **96**, 1387-1396.

Rivals I. & Personnaz L. (2003). Neural network construction and selection in nonlinear modeling. IEEE Transactions on Neural Networks **14**, pp. 804-819.

Seber G. A. F. & Wild C. (1989). Nonlinear regression. New York: Wiley.

Tibshirani R. J. (1996). A comparison of some error estimates for neural models. *Neural Computation* **8**, 152-163.

White H. (1980). Nonlinear regression on cross-section data. Econometrica 48, 721-746.

White H. (1982). Maximum likelihood estimation of misspecified models. Econometrica **50**, 1-25.

White H. (1989). Learning in artificial neural networks: a statistical perspective. Neural Computation **1**, 425-464.