Available online at www.sciencedirect.com
SCIENCE DIRECT

NEUROCOMPUTING

ELSEVIER

Neurocomputing 64 (2005) 161–181

www.elsevier.com/locate/neucom

Fast bootstrap methodology for regression model selection

A. Lendasse[a,*], G. Simon[b], V. Wertz[c], M. Verleysen[b,d]

[a]*Helsinki University of Technology, CIS, FI-02015, Finland*
[b]*Université catholique de Louvain, Machine Learning Group–DICE, 3 place du Levant, B-1348 Louvain-la-Neuve, Belgium*
[c]*Université catholique de Louvain, Machine Learning Group–CESAME, 4 av. Georges Lemaître, B-1348 Louvain-la-Neuve, Belgium*
[d]*Université Paris I–Panthéon–Sorbonne, SAMOS-MATISSE, 90 rue de Tolbiac, F-75634 Paris Cedex 13, France*

Available online 21 January 2005

Abstract

Using resampling methods like cross-validation and bootstrap is a necessity in neural network design, for solving the problem of model structure selection. The bootstrap is a powerful method offering a low variance of the model generalization error estimate. Unfortunately, its computational load may be excessive when used to select among neural networks models of different structures or complexities. This paper presents the fast bootstrap (FB) methodology to select the best model structure; this methodology is applied here to regression tasks. The fast bootstrap assumes that the computationally expensive term estimated by the bootstrap, the optimism, is usually a smooth function (low-order polynomial) of the complexity parameter. Approximating the optimism term makes it possible to considerably reduce the necessary number of simulations. The FB methodology is illustrated on multi-layer perceptrons, radial-basis function networks and least-square support vector machines.

Keywords: Model selection; Nonlinear modeling; Bootstrap; Resampling

*Corresponding author.
E-mail addresses: lendasse@cis.hut.fi (A. Lendasse), simon@dice.ucl.ac.be (G. Simon), wertz@auto.ucl.ac.be (V. Wertz), verleysen@dice.ucl.ac.be (M. Verleysen).

0925-2312/$ - see front matter © 2004 Published by Elsevier B.V.
doi:10.1016/j.neucom.2004.11.017

1. Introduction

Model design has raised a considerable research effort since decades, on linear models and nonlinear ones (i.e. artificial neural networks and many others). Model design includes the necessity to compare models (for example of different complexities) in order to select the most appropriate model among a family.

Model structure selection is the problem of choosing a specific model complexity among several possibilities. While effective statistical tests exist to select the complexity of linear models, their extension to nonlinear ones usually relies on approximations not always verified in real situations. For this reason, nonlinear model structure selection often relies on extended experiments repeated for each considered structure complexity. Unfortunately, nonlinear model optimization (i.e. finding the best parameters of a nonlinear model, once the structure—or complexity—is fixed) is in most cases a computationally intensive task in itself. Therefore, repeating this task for various structure complexities often exceeds the acceptable computational load.

Model structure selection necessitates estimating the generalization error of the model for each considered model complexity. The generalization error may be estimated by resampling techniques, such as k-fold cross-validation, leave-one-out and bootstrap. Unfortunately, both k-fold cross-validation and leave-one-out show a large variance in the estimation of the generalization error; their reliable use is thus restricted to problems where a very large number of data is available.

A resampling method that offers a lower variance is the bootstrap [5,8]. However, the bootstrap still necessitates a large number of repetitions to obtain a reliable estimate of the model generalization error, despite the fact that this number is reduced compared to the k-fold cross-validation and the leave-one-out techniques. Its computational load may thus exceed any acceptable level. There is thus a need for methods that approximate the results of the bootstrap with a reduced number of experiments.

In this paper, we present the fast bootstrap (FB) methodology, and apply it to the model structure selection problem. The FB methodology relies on the fact that the computationally intensive part of the bootstrap, the computation of the so-called optimism, usually leads to a very simple curve (with respect to the hyper-parameter to optimize). It will be shown that in many cases the optimism may even be approximated by a linear function of the hyper parameter. Exploiting this simple structure may thus considerably reduce the number of experiments. The use of a statistical test will validate the simple form of the optimism curve, and will help to automatically select its complexity if necessary.

The FB methodology will be applied to supervised regression tasks. The models that are used in this paper are: radial-basis functions networks (RBFN), multi-layer perceptrons (MLP), and least-square support vector machines (LS-SVM). It could be applied to other tasks than function approximation problems, including classification ones.

The paper is organized as follows. Section 2 formulates the model structure selection problem, and introduces resampling techniques. Section 3 describes in more detail the bootstrap method. Section 4 presents the proposed FB methodology, and the statistical test used to assess the complexity of the FB approximation compared to the bootstrap. Section 5 shows experimental results of the FB methodology on two function approximation examples: a toy example, for illustration, and the traditional Santa Fe Laser Data time-series prediction benchmark.

2. Model structure selection

We consider the problem of determining a model which approximates as accurately as possible an unknown function $g(.)$. This approximation is chosen among a set of several possible models. Models in such a set are denoted here by

$$h^q(x, \theta(q)), \tag{1}$$

where q represents the qth element in the set, $\theta(q)$ are the parameters of the qth model and x is the d-dimensional input vector. A time series prediction problem is a particular case of function approximation if nonlinear auto-regressive models (NARX) are used, as described in Section 5.3.

The parameters that define the set of possible models are called hyper-parameters; they are not estimated by the learning algorithm, but by some external procedure as will be described in the next section. These hyper-parameters are most often discrete, as the number of units or neurons in a MLP for example. They can be continuous too, in the case of a model parameter difficult to estimate by standard gradient-based learning procedures. In the case of RBF networks, the hyper-parameters are the number of Gaussian kernels and the widths of these kernels; the first hyper-parameter is discrete and the second one continuous. In this case the model structure selection problem is the selection of the appropriate number of Gaussian kernels and their appropriate widths.

In a typical learning procedure, the $\theta(q)$ parameters are optimized to minimize the approximation error on the learning set; the structure (i.e. the value of the hyper-parameters) is determined as the minimization of the generalization error described below.

The generalization error is defined by

$$E_{\text{gen}}(q, \theta) = \lim_{M \to \infty} \frac{\sum_{i=1}^{M} (h^q(x_i, \theta(q)) - y_i)^2}{M} \tag{2}$$

where x_i are d-dimensional input vectors to the model and y_i the corresponding scalar expected outputs.

According to definition (2), the generalization error is the mean square error of the model, computed on an infinite-size test set. Such set is of course not available; the generalization error has to be approximated. The best model structure q is then the structure that minimizes this approximation of the generalization error.

Resampling techniques may be used to approximate the generalization error. Several resampling techniques exist:

- The hold-out (HO) consists in removing data from the learning set and keeping them for validation; HO is also called “validation” [11], or “external validation”.

- In Monte-Carlo cross-validation [10] (or simply "cross-validation", CV), several HO partitions between learning and validation sets are randomly and sequentially drawn. The resulting generalization error estimate is the mean of the errors computed on the validation sets.
- The *k*-fold cross-validation [8] consists in dividing the available sample of *N* data into *k* sets of approximately equal size; then a model is learned on *k*-1 sets and independently validated on the remaining one. This operation may be performed *k* times on different splittings between learning and validation sets; the *k* validation results are then averaged to give an approximation of the generalization error of the model.
- Leave-one-out (LOO) [8] is the name given to the *k*-fold cross-validation method when *k* takes its largest possible value $k = N$.
- The bootstrap [5], consists in drawing with replacement sets from the original sample and using these sets to estimate the generalization errors (bootstrap 632 and 632+ are improved versions of the original bootstrap); the bootstrap method will be detailed in Section 3.

All these methods share some asymptotic properties (see for example [16]), but they also differ on the following points:

- LOO is less biased [5,8] but needs a very high number of data to be reliable;
- cross-validation is consistent (i.e. converges to the generalization error when the size of the sample increases) if the size of the validation set grows faster than the size of the learning set (which is hard to expect in practical situations!) [14];
- cross-validation is almost unbiased [5];
- bootstrap is downward biased but has a very low variance [5];
- most recent bootstrap methods (632) are almost unbiased and also have a low variance [5].

In this paper, the bootstrap method is used to exploit the fact that, due to its lower variance property, the number of repetitions needed to obtain a reliable estimate of the generalization error is reduced compared to the *k*-fold cross-validation and the LOO. The bootstrap is described in details in Section 3.

3. Bootstrap

The bootstrap [2,5] is a resampling method that has been developed in order to estimate some statistical parameters (like the mean of a population, its variance, etc.). In the case of model structure selection, the parameter to be estimated is the generalization error (i.e. the average error that the model would make on an infinite size and unknown test set). When using the bootstrap, this error is not estimated directly. Rather the bootstrap estimates the difference between the generalization error and the training error (the latter being called apparent error by Efron). This difference is called the *optimism*. The estimated generalization error will thus be the sum of the training error and of the estimated optimism. The training error is computed using all data from the training set. The optimism is estimated using a resampling technique based on drawing with replacement within the training set. In the following, notation $E_j^{A_l,A_v}$ is used to denote the error made by a model learned on set A_l and tested on set A_v. If these two sets are equal, $E_j^{A_l,A_v}$ denotes a training error; is they are different, it denotes a validation or test error. As the bootstrap methodology requires averaging several estimates, j is used to denote the estimate number. It is important to notice that using different learning sets A_l requires learning several models, while evaluating models on several test sets A_v only requires a single learning (provided that A_l does not change). Finally, let us note that in our model structure selection problem, the bootstrap methodology is used for each of the considered model structures (or model complexities): an estimate of the generalization error is needed for each of the model structures, and the selection of the ''best'' model structure is made according to their minimum.

Following these notations, the Bootstrap method can be decomposed in the following steps:

1. From the initial data set I of size N, one randomly draws N points with replacement. The new set A_j has thus the same size as the initial set. This step is called the resampling.
2. The training of the various model structures q is done on the training set A_j. The learning error on this set is computed:

$$E_j^{A_j,A_j}(q,\theta_j^*(q)) = \frac{\sum_{i=1}^{N}(h^q(x_i^{A_j},\theta_j^*(q)) - y_i^{A_j})^2}{N}, \tag{3}$$

   where h^q is the qth model that is used, $\theta_j^*(q)$ are the model parameters after learning, x_i^{Aj} is the ith input vector from set A_j, y_i^{Aj} the corresponding output and N the number of elements in this set.
3. can also compute the validation error on the initial sample I which now plays the role of the validation set:

$$E_j^{A_j,I}(q,\theta_j^*(q)) = \frac{\sum_{i=1}^{N}(h^q(x_i^{I},\theta_j^*(q)) - y_i^{I})^2}{N}. \tag{4}$$

4. The difference between the learning and validation errors (3) and (4) is computed and defined as the optimism by Efron [5]:

$$optimism_j(q,\theta_j^*(q)) = E_j^{A_j,I}(q,\theta_j^*(q)) - E_j^{A_j,A_j}(q,\theta_j^*(q)). \tag{5}$$

5. Steps 1–4 are repeated J times. The global estimate of the optimism is then calculated as the average of the J values from (5):

$$opti\hat{m}ism(q) = \frac{\sum_{j=1}^{J} optimism_j(q,\theta_j^*(q))}{J}. \tag{6}$$

6. Steps 1–5 result in an estimate of the optimism. To get an estimate of the generalization error of each model structure q, one still has to estimate the apparent error (the original set I is used both for learning and validation):

$$E^{I,I}(q, \theta^*(q)) = \frac{\sum_{i=1}^{N} (h^q(x_i^I, \theta^*(q)) - y_i^I)^2}{N}. \tag{7}$$

7. An approximation of the generalization error is finally obtained by:

$$\hat{E}_{\text{gen}}(q) = \widehat{optimism}(q) + E^{I,I}(q, \theta^*(q)). \tag{8}$$

   $\hat{E}_{\text{gen}}(q)$ is an approximation of the generalization error for each model structure q. The best structure is the one that minimizes this estimate of the generalization error.

4. Fast bootstrap methodology

In this section, an improvement to the bootstrap method is presented. It is called the FB and allows reducing the computational time of the traditional bootstrap [9,15].

Experimental observations have shown that the estimate of the optimism is a simple and smooth function of the hyper-parameters. "Simple and smooth" means that the optimism can be approximated by a low-order function (parameterized by a few parameters) as for example a linear, quadratic or exponential one.

When the hyper-parameter is the number of parameters (or a linear function of it), it has been experimentally observed that the optimism is linear with respect to the number of parameters. This observation will be confirmed by statistical tests.

The linear character of the optimism is also in agreement with the AIC asymptotic criterion that approximates the optimism by a linear function [1]:

$$\widehat{optimism} = 2 \log(N) p E^{I,I}. \tag{9}$$

In (9), p is the number of parameters in the model, N the number of samples in the learning set and $E^{I,I}$ the learning error defined by (7).

Both AIC and bootstrap are methods to estimate the optimism: AIC is an analytical and asymptotic method, while the bootstrap is experimental and remains valid when the number of learning data is smaller.

Relation (9) only holds when the number p of parameters is equal to the *effective* number of parameters in the model [4]. This is the case for MLP and RBFN models without regularization or early stopping. However, one could also use MLP or RBFN models with an early stopping criterion or models including a regularization term like LS-SVM. In these cases, an extension of (9) using Moody's generalized prediction error (GPE) criterion should be used [12]: in the GPE criterion the number of parameters is replaced by the effective number of parameters (which is smaller than the number of parameters thanks to the use of a regularization term in the cost function).

Therefore, if the hyper-parameter is not the number of parameters but a nonlinear function of the complexity (for example the regularization parameter γ in the LS-SVM), the optimism function linear but a simple form. The methodology presented in this paper will allow determining the form of the optimism function.

The FB works as follows. In the traditional bootstrap, the optimism is a function of the hyper-parameter q (see (6)). Each evaluation of this function necessitates J repetitions (sampling, learning and evaluation). As the function has to be evaluated for Q model complexities, this means that $J.Q$ models have to be learned. The FB will use the simplicity and a priori known form of the optimism function to reduce J and/or Q, and therefore the number of models that have to be learned.

If a polynomial optimism function of low order r is assumed, we know that $r+1$ evaluations are sufficient to determine the optimism function, allowing a huge decrease of Q. This is valid if each evaluation is accurate enough, i.e. if J is sufficiently high. On the contrary, the optimism function can also be approximated with a high number Q of rough evaluations (i.e. a low number J of bootstrap replications). Between these two extremes, both J and Q may be reduced simultaneously to drastically reduce the number $J.Q$ of models while keeping acceptable values for both J and Q.

The experiments detailed in the following section confirm that due to the simple form of the optimism, the latter may be approximated accurately with a strongly reduced number of experiments compared to the original bootstrap. Therefore, the structure selection by traditional bootstrap and FB will lead to similar models, with an important advantage for the FB in terms of computational complexity.

The key point is therefore to find the complexity of these "simple and smooth" low-order functions. For example polynomial functions, of some order, can be used. In this case the question becomes "what is the order of the polynomial approximation of the optimism?" To answer this question the simple analysis of variance (ANOVA) methodology is applied in a polynomial regression framework (see for example [6]): the sum of squared errors (or sum of *residuals*) obtained from two polynomial approximations of different orders are compared. Let H_0 be the null hypothesis of having a low-order model of order κ_0:

$$\hat{v}_i = \alpha_0 + \alpha_1 u_i + \alpha_2 u_i^2 + \cdots + \alpha_{\kappa 0} u_i^{\kappa_0}, \tag{10}$$

where u_i and v_i are, respectively, the input and desired output of the model, and $\hat{v}_i$ the approximation of the latter; H_1 is the alternative hypothesis of having a model of order κ_1 with $\kappa_1 > \kappa_0$. The comparison between the residuals obtained from the two models leads to the following Fisher's statistics:

$$F_{\kappa_1-\kappa_0, K-\kappa_1-1} = \frac{(SR_0 - SR_1)/(\kappa_1 - \kappa_0)}{SR_1/(K-\kappa_1-1)}, \tag{11}$$

where K is the total number of data, and SR_j is the sum of residuals of the model under hypothesis H_j defined as

$$SR_j = \sum_{i=1}^{K} ||v_i - \hat{v}_i||^2. \tag{12}$$

The obtained F-statistics, which follows a Fisher–Snedecor law with $(\kappa_1 - \kappa 0)$ and $(K - \kappa_1 - 1)$ degrees of freedom, is used to test the null hypothesis H_0 'the underlying model is of order κ_0' against H_1 'the underlying model is of order κ_1'.

The methodology used to assess the complexity of the approximation is thus the following:

- choose two model orders κ_0 and κ_1 $(\kappa_1 > \kappa_0)$;
- fit the respective models (10);
- compute the F statistics (11);
- test the result of the F statistics against the 5% confidence level given by the Fisher's tables with $(\kappa_1 - \kappa_0)$ and $(K - \kappa_1 - 1)$ degrees of freedom.

If the test is successful, hypothesis H_0 is accepted, meaning that the order κ_0 of approximation (10) is sufficient.

This methodology is applied in our case to the approximation of the optimism according to the model complexity; in other words, $v = \widehat{optimism}$ and $u = q$. In the experimental section, we will see that in most cases, taking order κ_0 equal to 1 is sufficient.

Note that in some cases, a low order κ_0 will not be found thus invalidating the polynomial hypothesis. In such situations, an ad hoc preprocessing is suggested, as illustrated in Sections 5.2 and 5.3 for the LS-SVM model, where the log of the optimism is taken before using (10).

In practice, the optimism estimate does not have to be computed for very low values of the model complexity q. Indeed, the apparent error for such small values of the complexity will be high compared to the optimism, leading to a high generalization error too.

5. Experimental results

5.1. Nonlinear models used in the experiments

In this section, the FB methodology is applied on two function approximation examples: a toy one-dimensional example used for illustration, and the traditional Santa Fe Laser Data time series prediction benchmark [21]. Both problems are solved by three non-linear models: radial-basis functions networks (RBFN), multi-layer perceptrons (MLP), and least-squares support vector machines (LS-SVM) [20]. Note that although state-of-the-art optimization algorithms have been used for the learning of these models as detailed below, the purpose here was not to perform an extended comparison between, for example, MLP learned with various learning procedures. Other learning procedures without regularization lead to similar conclusions regarding the shape of the optimism curve as estimated by the bootstrap, which is the topic of this paper.

The MLP model [7] used in the experiments is a one hidden layer network with P units or neurons in the hidden layer:

$$h(x) = \sum_{i=1}^{P} w_i \tanh\left(\sum_{j=1}^{d} w_{ij} x_j\right), \tag{13}$$

where x_j are the elements of the d-dimensional input vectors x, and w_i, w_{ij} the elements of the parameter set θ. The hyper-parameter measuring the complexity is the number P of hidden units. One thousand epochs of the Levenberg–Marquardt algorithm implemented in the Matlab® Neural Networks toolbox are used to optimize the parameters; no regularization has been used. The best model has been selected among several runs with different weight initializations.

The RBFN [13] model used here is given by

$$h(x) = \sum_{i=1}^{P} \lambda_i \exp\left(-\frac{\|x - c_i\|^2}{2WSF\sigma_i^2}\right). \tag{14}$$

In this model, the parameter set θ includes the λ_i, c_i and σ_i that are learned by unsupervised and supervised techniques without regularization (see [3] for details). *WSF* is a continuous hyper-parameter that regularizes the widths of the Gaussian kernels (a small *WSF* will lead to narrow local kernels while a large one will smooth the $h(x)$ approximator). The number P of Gaussian kernels is the other hyper-parameter. The FB may be used to estimate the adequate number P of kernels, but not for adjusting the regularization factor *WSF*. As a consequence, the optimism curve resulting from the bootstrap will be drawn for a specific value of *WSF*. A further loop is needed to experimentally select an optimal value for *WSF*; this further arguments the need for a fast procedure inside the loop.

The LS-SVM model [17–20] is given in primal weight space by

$$h(x) = \omega^{\mathrm{T}}\varphi(x) + b, \tag{15}$$

where $\varphi(x)$ is a function which maps the input space into a higher dimensional feature space. In LS-SVM for function estimation, the following optimization problem is formulated:

$$\min_{\omega,b,e} J(\omega, b, e) = \frac{1}{2}\omega^{\mathrm{T}}\omega + \gamma\frac{1}{2}\sum_{i=1}^{N} e_i^2, \tag{16}$$

subject to the equality constraints

$$y_i = \omega^{\mathrm{T}}\varphi(x_i) + b + e_i, \quad i = 1, \ldots, N. \tag{17}$$

The parameter set θ consists of ω and b, and the hyper-parameters are the width of the Gaussian kernels (taken to be identical for all kernels) and the γ regularization factor. Solving this optimization problem in dual space leads to finding the α_i and b

coefficients in the following solution:

$$h(x) = \sum_{i=1}^{N} \alpha_i K(x, x_i) + b. \tag{18}$$

The function $\kappa(x, x_i)$ is the kernel defined as the dot product between the $\varphi(x)^{\mathrm{T}}$ and $\varphi(x_i)$ mappings.

LS-SVM can be viewed as a form of parametric ridge regression in the primal space. The training method for the estimation of the ω and b parameters can be found in [19]. Note also that LS-SVM are not only used as supervised learning models; they are also used in the case of unsupervised learning such as nonlinear PCA [19].

5.2. Toy example

The first example used for illustration purposes is a one-dimensional function approximation problem [9]. In this example, 200 inputs x_i have been randomly drawn using a uniform distribution between 0 and 1. The corresponding y_i outputs have been generated according to

$$y_i = \sin(5x_i) + \sin(15x_i) + \sin(25x_i) + \varepsilon_i, \tag{19}$$

where ε_i are uniformly distributed and independently drawn random values in [−0.5, 0.5]. The function without noise and the 200 noisy samples are shown in Fig. 1.

The MLP model has been tested on the toy example for a number of hidden neurons between 1 and 13. Figs. 2 and 3, respectively, show the estimate of the optimism and of the generalization error, both when a full bootstrap and when the FB approximation is used. Both the bootstrap and the FB select an optimal number

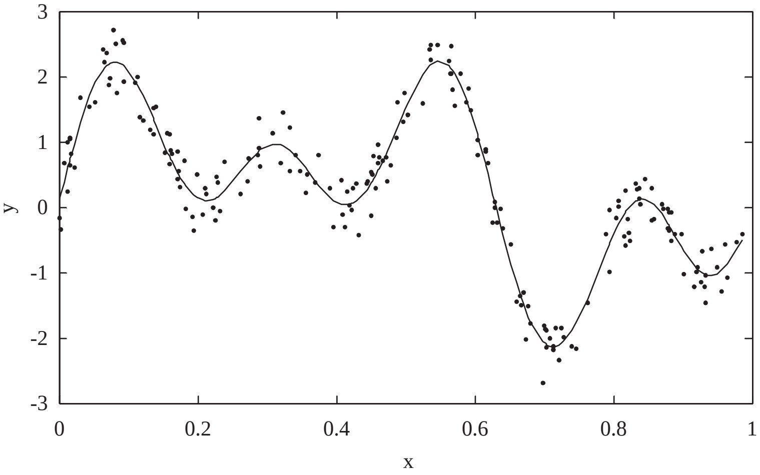

Fig. 1. Toy example: function without noise (solid line) and 200 noisy samples (dots).

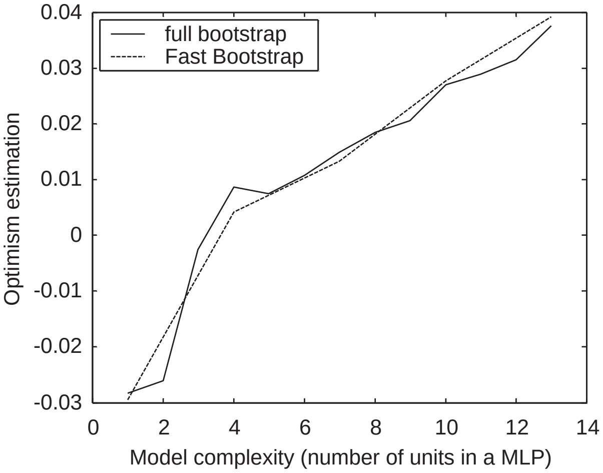

Fig. 2. Estimate of the optimism obtained with a MLP on the toy example.

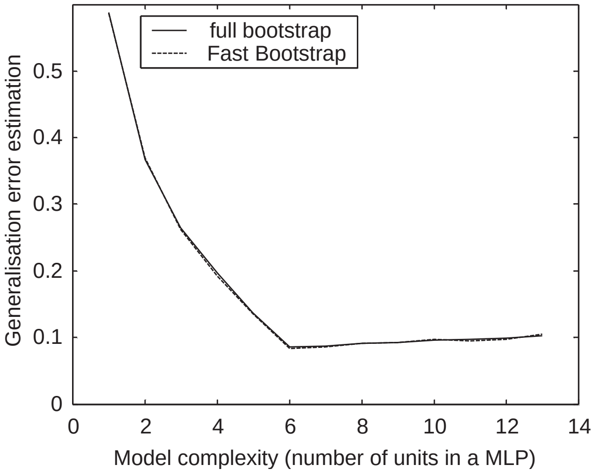

Fig. 3. Estimate of the generalization error obtained with a MLP on the toy example.

of units equal to 6. Table 1 summarizes the experimental conditions, i.e. the number of models, number of bootstrap replications for each model, and the gain obtained by using the FB methodology.

Fishers' statistics has been computed in the FB case. Eq. (11) is used with $\kappa_0 = 1$, $\kappa_1 = 2$ and $K = 5$ to verify that the estimate of the optimism is a linear function of the number of hidden neurons. Table 1 also gives the value of Fisher's test.

The RBFN model has been tested on the same example. The FB has been used to optimize the number of kernels in the model, in the [10,19] range. The bootstrap

Table 1
Experimental conditions and results of Fisher's test for the three models applied to the toy example

MLP	Number of hidden neurons	Bootstrap replications	Number of experiments	Gain	$F_{1,2}$
Bootstrap	1–13 by steps of 1	100	1300		
Fast Bootstrap	1–13 by steps of 3	10	50	96.2%	3.2545
RBFN	Number of kernels	Bootstrap replications	Number of experiments	Gain	$F_{1,1}$
Bootstrap	10–19 by steps of 1	100	1000		
Fast Bootstrap	10–19 by steps of 3	10	40	96%	16.3085
LS-SVM	Regularization parameter γ	Bootstrap replications	Number of experiments	Gain	$F_{1,7}$
Bootstrap	5–50 by steps of 0.1	100	45100		
Fast Bootstrap	5–50 by steps of 5	10	100	99.8%	0 *

The 0 value marked by * for the $F_{1,7}$ Fisher's test in the LS-SVM case reaches the numerical limit of the computer calculation.

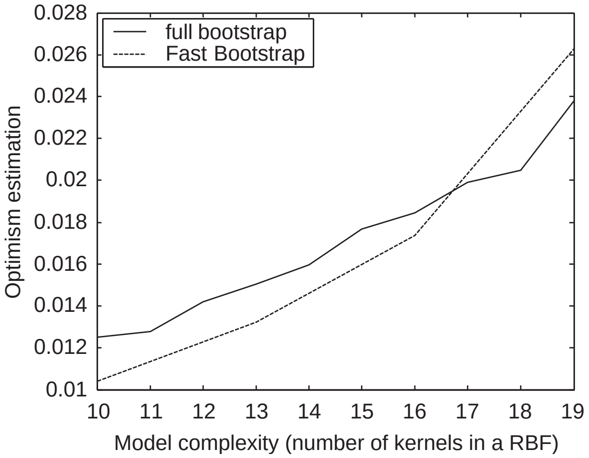

Fig. 4. Estimate of the optimism obtained with a RBFN on the toy example.

selects an optimum number of kernels equal to 17, while the FB gives a close estimate equal to 16.

Simulations conditions are summarized in Table 1. Figs. 4 and 5, respectively, show the estimate of the optimism and of the generalization error, both when a full bootstrap and when the FB approximation is used.

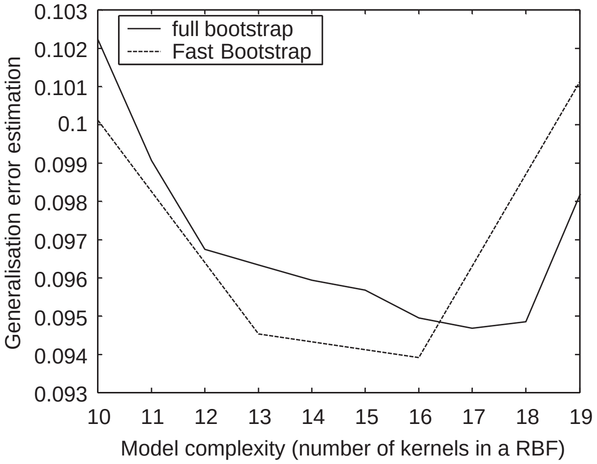

Fig. 5. Estimate of the generalization error obtained with a RBFN on the toy example.

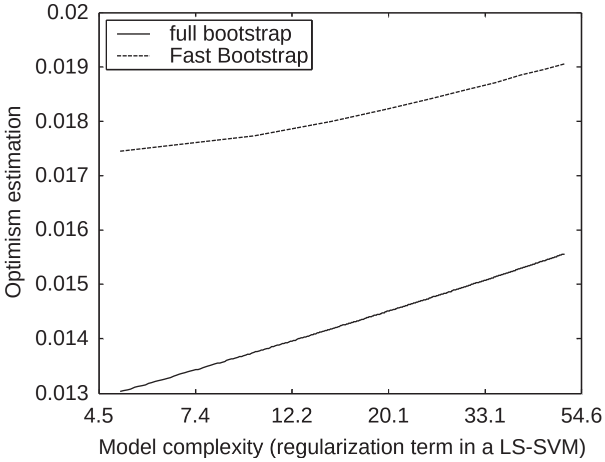

Fig. 6. Estimate of the optimism obtained with a LS-SVM on the toy example; the solid line has been artificially shifted for illustration purposes (see text for details).

Fishers' statistics has been computed in the FB case. Eq. (11) is used with $\kappa_0 = 1$, $\kappa_1 = 2$ and $K = 6$ to verify that the estimate of the optimism is a linear function of the number of kernels.

Finally, the LS-SVM has been applied to the same problem. The regularization parameter γ is the hyper-parameter to be optimized in the [5–50] range. Simulations conditions are summarized in Table 1. Figs. 6 and 7, respectively, show the estimate of the optimism and of the generalization error, both when a full bootstrap and when the FB approximation is used. Note that the Y-scale of these figures has been

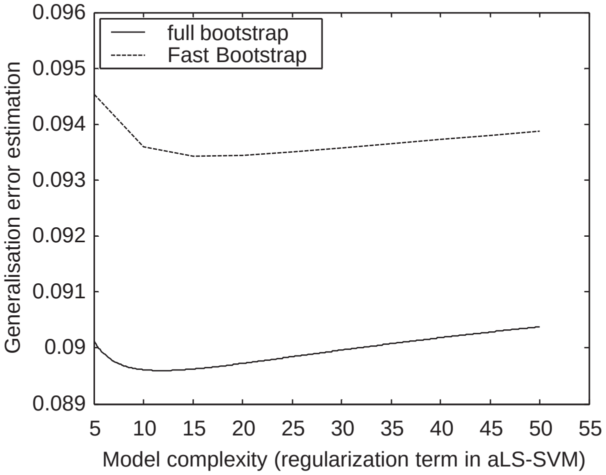

Fig. 7. Estimate of the generalization error obtained with a LS-SVM on the toy example; the solid line has been artificially shifted for illustration purposes (see text for details).

adapted to show more clearly the shape of the curves; indeed, in both figures, the bootstrap and FB curves are in fact very close. Therefore, the two full bootstrap curves (solid lines) have been artificially shifted downwards by subtracting a fixed 0.005 value, for illustration purposes only. The optimal regularization parameter γ is found to be equal to 12 by the bootstrap method, and to 15 by the FB one.

Fishers' statistics has been computed in the FB case. Fig. 6 suggests that the optimism estimate is a linear function of the regularization parameter logarithm (note the logarithmic X-scale). Therefore, in order to perform Fisher's test, the logarithm of the regularization parameter is taken. Eq. (11) is used with $\kappa_0 = 1$, $\kappa_1 = 2$ and $K = 10$ to verify that the optimism estimate is a linear function of the regularization parameter logarithm.

5.3. Santa fe laser data

The Santa Fe Laser Data time-series [21] has been obtained from a far-infrared-laser in a chaotic state. This time-series has become a well-known benchmark in time series prediction since the Santa Fe competition in 1991. It includes 1000 points, and is illustrated in Fig. 8.

The prediction task may be expressed as a function approximation problem using model (20):

$$\hat{x}(t+1) = h^q(x(t), x(t-1), x(t-2), x(t-3), x(t-4), x(t-5), \theta(q)). \tag{20}$$

In (20), $\theta(q)$ are the parameters of the MLP, RBFN or LS-SVM model used in the experiments. The largest time lag (t-5) has been chosen according to published results on this benchmark [21]. This model is a nonlinear auto-regressive one (NAR) [11];

the FB methodology can be extended in a straightforward manner to NARX models (including exogenous variables).

For this example, only the FB methodology has been used. Indeed in a real experimental setting, only the FB will be used, Fisher's test being checked to verify the hypothesis made to allow the reduction in the number of experiments.

The MLP model has been tested on the SantaFe Laser Data time series for a number of hidden neurons between 1 and 10. Figs. 9 and 10, respectively, show the estimate of the optimism and of the generalization error when the FB approximation

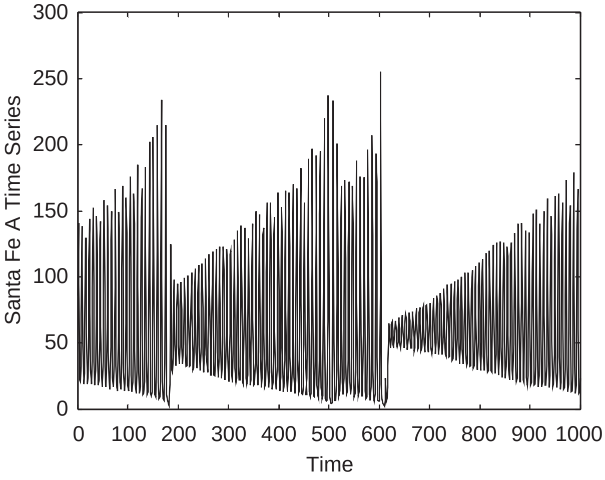

Fig. 8. SantaFe Laser Data time series.

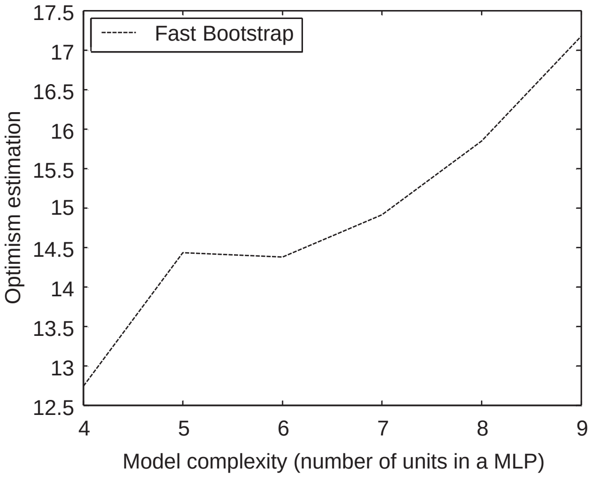

Fig. 9. Estimate of the optimism obtained with a MLP on the SantaFe Laser Data time series.

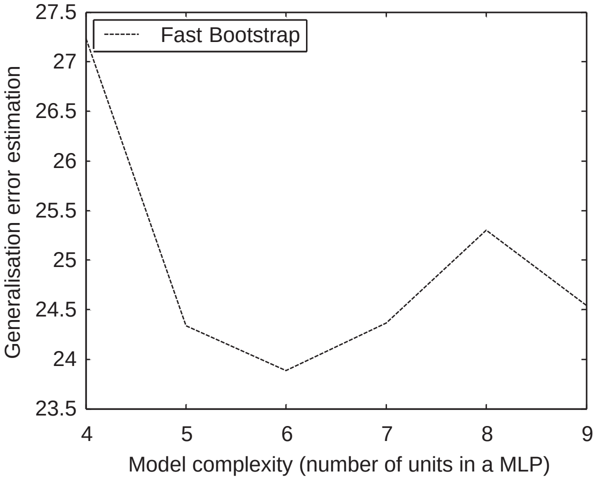

Fig. 10. Estimate of the generalization error obtained with a MLP on the SantaFe Laser Data time series.

Table 2
Experimental conditions and results of Fisher's test for the three models applied to the SantaFe Laser Data time series

MLP	Number of hidden neurons	Bootstrap replications	Number of experiments	$F_{1,3}$
Fast Bootstrap	4–9 by steps of 1	10	60	0.2002
RBFN	Number of kernels	Bootstrap replications	Number of experiments	$F_{1,2}$
Fast Bootstrap	60–140 by steps of 20	20	100	1.6472
LS-SVM	Regularization parameter γ	Bootstrap replications	Number of experiments	$F_{1,16}$
Fast Bootstrap	15–105 by steps of 5	10	190	0*

The 0 value marked by * for the $F_{1,7}$ Fisher's test in the LS-SVM case reaches the numerical limit of the computer calculation.

is used. Table 2 summarizes the experimental conditions, i.e. the number of models, number of bootstrap replications for each model, and the result of Fisher's test. Eq. (11) is used with $\kappa_0 = 1$, $\kappa_1 = 2$ and $K = 4$ to verify that the estimate of the optimism is a linear function of the number of hidden neurons.

The RBFN model has been tested on the same example. The FB has been used to optimize the number of kernels in the model, in the [60,140] range. Simulations conditions are summarized in Table 2. Figs. 11 and 12, respectively, show the

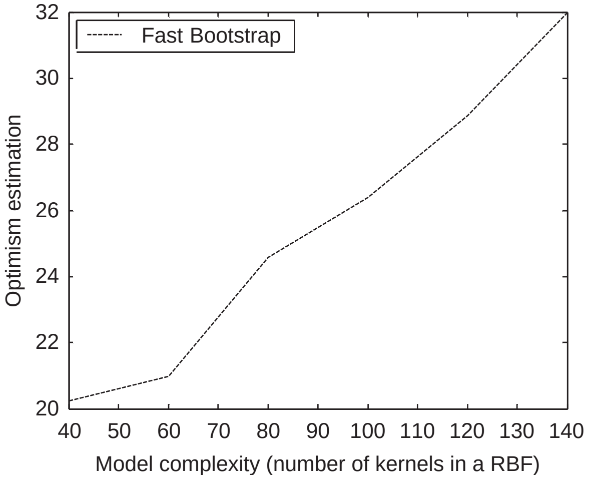

Fig. 11. Estimate of the optimism obtained with a RBFN on the SantaFe Laser Data time series.

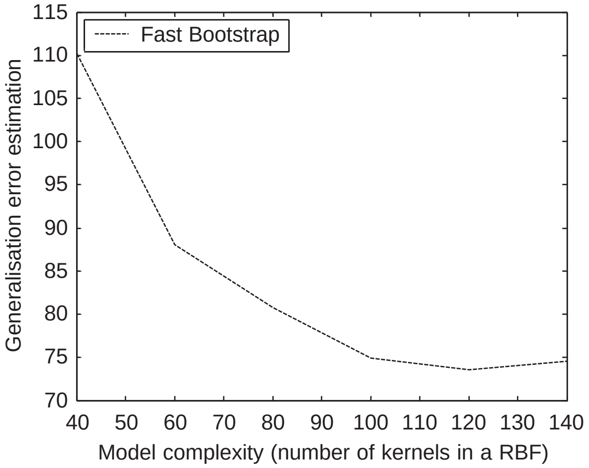

Fig. 12. Estimate of the generalization error obtained with a RBFN on the SantaFe Laser Data time series.

estimate of the optimism and of the generalization error when the FB approximation is used. Fishers' statistics has been computed: Eq. (11) is used with $\kappa_0 = 1$, $\kappa_1 = 2$ and $K = 5$ to verify that the estimate of the optimism is a linear function of the number of kernels.

Finally, the LS-SVM has been applied to the same problem. LS-SVM better scale to high dimensional input spaces than RBFN and MLP and on this problem a largest time lag (*t*-50) can be used [19]. Nevertheless, in order to compare the results between RBFN, MLP and LS-SVM, the same largest time lag (*t*-5) is used.

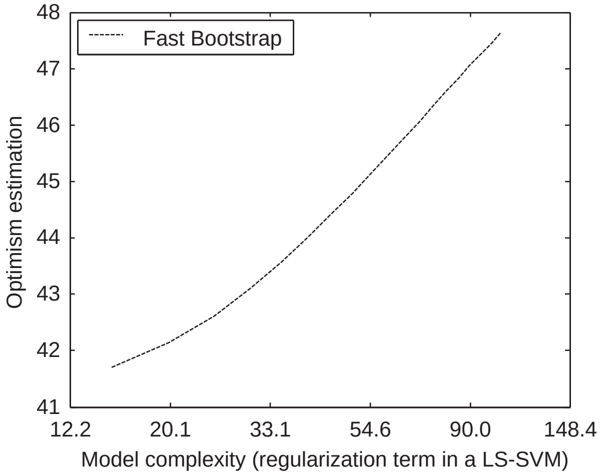

Fig. 13. Estimate of the optimism obtained with a LS-SVM on the SantaFe Laser Data time series.

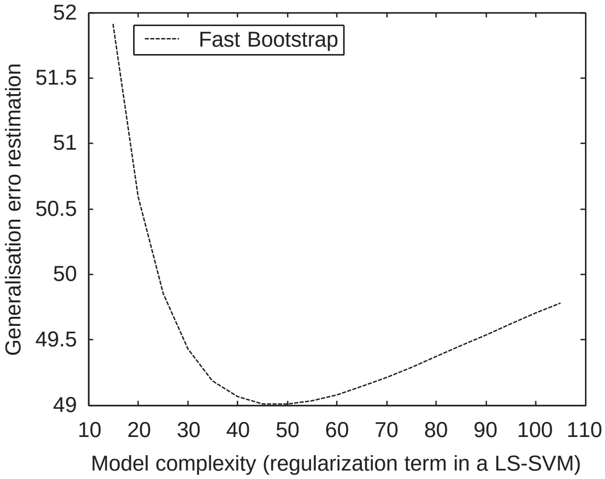

Fig. 14. Estimate of the generalization error obtained with a LS-SVM on the SantaFe Laser Data time series.

The regularization parameter γ is the hyper-parameter to optimize in the [25–1000] range. Simulations conditions are summarized in Table 2. Figs. 13 and 14, respectively, show the estimate of the optimism and of the generalization error when the FB approximation is used. Fig. 14 clearly suggests that the optimism estimate is a linear function of the regularization parameter logarithm (note the logarithmic X-scale). Therefore in order to perform Fisher's test, the logarithm of the regularization parameter is taken. Eq. (11) is used with $\kappa_0 = 1$, $\kappa_1 = 2$ and $K = 10$ to verify that the optimism estimate is a linear function of the regularization parameter logarithm.

6. Conclusion

Neural networks, and more generally linear and nonlinear models, require both parameter optimization and model structure selection in order to perform their regression or classification task. While the former is usually achieved through a standard or specialized optimization procedure on real parameters using a learning set, the discrete nature of model structure makes the use of validation sets unavoidable. In order to avoid the dependency on a specific choice of these validations sets, resampling is necessary.

The drawback of most resampling procedures (k-fold cross-validation, bootstrap, etc.) is their huge requirements in terms of computational load. Learning, sometimes slow in itself, is nested in simulations loops often making the total computation time prohibitive. There is thus a need for accelerated resampling procedures implementing an effective compromise between accuracy and computational load.

In this paper, the FB procedure is presented. It is shown experimentally that the computationally intensive term of the bootstrap, the optimism, often takes a simple form with regards to the model complexity parameter. This property is exploited through a dramatic decrease of the number of experiments needed for the optimism estimation. The smoothness of the optimism estimation can also be exploited to estimate the complexity parameter by gradient-based search, which is usually difficult or impossible with other resampling procedures, producing irregular estimates. A statistical test is used to verify a posteriori the validity of the hypothesis made in the suggested approximation; if the test fails, it is always possible to increase the number of experiments. Such accelerated procedure makes it possible to combine the power of the bootstrap resampling and the constraints of real applications.

Acknowledgements

Michel Verleysen is Senior Research Associate of the Belgian National Fund for Scientific Research (FNRS). G. Simon is funded by the Belgian F.R.I.A. Part the work of V. Wertz is supported by the Interuniversity Attraction Poles (IAP), initiated by the Belgian Federal State, Ministry of Sciences, Technologies and Culture. Part the work of A. Lendasse is supported by the project New Information Processing Principles, 44886, of the Academy of Finland. The scientific responsibility rests with the authors.

References

[1] H. Akaike, Information theory and an extension of the maximum likelihood principle, in: Proceedings of second International Symposium on information Theory, Budapest, 1973, pp. 267–81.

[2] D. Anguita, A. Boni, S. Ridella, Evaluating the generalization ability of support vector machines through the bootstrap, Neural Process Lett 11 (1) (2000) 51–58.

[3] N. Benoudjit, M. Verleysen, On the kernel widths in radial-basis function networks, Neural Process. Lett. 18 (2) (2003) 139–154.

[4] C.M. Bishop, Neural Networks for Pattern Recognition, Oxford University Press, Oxford, 1995.

[5] B. Efron, R.J. Tibshirani, An Introduction to the Bootstrap, Chapman & Hall, New York, 1993.

[6] W.H. Greene, Econometric Analysis, third ed., Prentice-Hall, Englewood Cliffs, NJ, 1997.

[7] S. Haykin, Neural Networks: A Comprehensive Foundation, Prentice-Hall, Upper Saddle River, NJ, 1998.

[8] R. Kohavi, A study of cross-validation and bootstrap for accuracy estimation and model selection, in: Proceedings of the 14th International Joint Conference on A.I., Montréal, 1995 2; pp. 1137–1143.

[9] A. Lendasse, G. Simon, V. Wertz, M. Verleysen, Fast bootstrap for least-square support vector machines, in: Proceedings of the European Symposium on Artificial Neural Networks ESANN'2004, Bruges, April 2004, pp. 525–530.

[10] A. Lendasse, V. Wertz, M. Verleysen, Model selection with cross-validations and bootstraps—application to time series prediction with RBFN models, in: O. Kaynak, E. Alpaydin, E. Oja, L. Xu (Eds.), Artificial Neural Networks and Neural Information Processing—ICANN/ICONIP 2003, Springer-Verlag Lecture Notes in Computer Science 2714, Berlin, 2003, pp. 573–580.

[11] L. Ljung, System Identification—Theory for the User, second ed., Prentice-Hall, Upper Saddle River, NJ, 1999.

[12] J.E. Moody, The effective Number of Parameters: An analysis of generalization and regularization in nonlinear learning systems, Adv. Neural Inform. Process. Systems (1992) 847–854.

[13] T. Poggio, F. Girosi, Networks for approximation and learning, Proc. IEEE 78 (9) (1990) 1481–1497.

[14] J. Shao, D. Tu, The Jackknife and Bootstrap, Series in Statistics, Springer-Springer, New York, 1995.

[15] G. Simon, A. Lendasse, M. Verleysen, 2003. Bootstrap for Model Selection: Linear Approximation of the Optimism, in: J. Mira, J.R. Alvarez (eds.), Computational Methods in Neural Modeling, Springer Lecture Notes in Computer Science, vol. 2686, Berlin, pp. I182–I189.

[16] M. Stone, An asymptotic equivalence of choice of model by cross-validation and Akaike's criterion, J. Roy. Statist. Soc. B 39 (1977) 44–47.

[17] J.A.K. Suykens, J. De brabanter, L. Lukas, J. Vandewalle, Weighted least squares support vector machines: robustness and sparse approximation, Neurocomputing, Special Issue on fundamental and information processing aspects of neurocomputing 48 (1–4) (2002) 85–105.

[18] J.A.K. Suykens, L. Lukas, J. Vandewalle, Sparse least squares support vector machine classifiers, in: Proceedings of the European Symposium on Artificial Neural Networks ESANN'2000, Bruges, 2000, pp. 37–42.

[19] J.A.K. Suykens, T. Van Gestel, J. De Brabanter, B. De Moor, J. Vandewalle, Least squares support vector machines, World Scientific, Singapore, 2002, ISBN 981-238-151-1.

[20] J.A.K. Suykens, J. Vandewalle, Training multilayer perceptron classifiers based on modified support vector method, IEEE Trans. Neural Networks, 10 (4) (1999) 907–911.

[21] A.S. Weigend, N.A. Gershenfeld, Time Series Prediciton: Forecasting the Future and Understanding the Past, Addison-Wesley, Reading, MA, 1994.

Amaury Lendasse was born in 1972 in Tournai, Belgium. He received the M.S. degree in mechanical engineering from the Universitée catholique de Louvain (Belgium) in 1996, M.S. in control in 1997 and Ph.D. in 2003 from the same University. Currently he is a Postdoc researcher in the Neural Network Research Center in the Helsinki University of Technology in Finland. He is author or co-author of about 50 scientific papers in international journals and books or communications to conferences with reviewing committee. His research concerns time series prediction, Kohonen maps, nonlinear projections and nonlinear approximation.

Geoffroy Simon was born in 1978 in Belgium. He received the M.S. degree in Computer Sciences from the Facultés Universitaires Notre Dame de la Paix (Namur, Belgium) in 2002. He is now working as Ph. D. student at the Microelectronic Laboratory of the Electrical Engineering Departement of the Université catholique de Louvain (UCL), and he is also a member of the Machine Learning Group at the UCL. His work is funded by a grant from the Belgian FRIA. His research topics cover artificial neural networks, nonlinear statistics and self-organization applied to time-series forecasting problems.

Vincent Wertz holds a degree in mathematical engineering and a Ph.D. in applied sciences, both from the Université catholique de Louvain. He has held various research positions at this university and is now professor in the department of mathematical engineering. His main research interests are in the areas of identification and control, with a recent emphasis on nonlinear techniques (neural networks, fuzzy control). He is also actively involved in pedagogical reforms at the school of engineering, where a PBL approach has been introduced in the first years of the curriculum.

Michel Verleysen was born in 1965 in Belgium. He received the M.S. and Ph.D. degrees in electrical engineering from the Université catholique de Louvain (Belgium) in 1987 and 1992, respectively. He was an Invited Professor at the Swiss E.P.F.L. (Ecole Polytechnique Fédérale de Lausanne, Switzerland) in 1992, at the Université d'Evry Val d'Essonne (France) in 2001, and at the Université ParisI-Panthéon-Sorbonne in 2002, 2003 and 2004. He is now a Senior Research Associate of the Belgian F.N.R.S. (Fonds National de la Recherche Scientique) and Lecturer at the Université catholique de Louvain. He is editor-in-chief of the Neural Processing Letters journal and chairman of the annual ESANN conference (European Symposium on Artificial Neural Networks); he is associate editor of the IEEE Trans. on Neural Networks journal, and member of the editorial board and program committee of several journals and conferences on neural networks and learning. He is author or co-author of about 160 scientific papers in international journals and books or communications to conferences with reviewing committee. He is the co-author of the scientific popularization book on artificial neural networks in the series "Que Sais-Je?", in French. His research interests artificial neural networks, self-organization, time-series forecasting, nonlinear statistics, adaptive signal processing, and electronic implementations of neural and biomedical systems.