Double SOM for long-term time series prediction

Geoffroy Simon[1], Amaury Lendasse[2], Marie Cottrell[3], Jean-Claude Fort[4], Michel Verleysen[1]

[1,2]Université catholique de Louvain – [1]DICE, [2]CESAME
[1]place du Levant 3, [2] Avenue G. Lemaître 4, B-1348 Louvain-la-Neuve, Belgium
tel : +32 10 47 25 40, fax : +32 10 47 25 98, {simon, verleysen}@dice.ucl.ac.be, lendasse@auto.ucl.ac.be
[3] Université Paris I - Panthéon Sorbonne, UMR CNRS 8595
SAMOS-MATISSE - Rue de Tolbiac, 90 F-75634 Paris cedex 13, France
marie.cottrell@univ-paris1.fr
[4]Université Paul Sabatier Toulouse 3, Lab. Statistiques et Probabilités, CNRS C55830
118 route de Narbonne, F-31062 Toulouse Cedex, France
fort@cict.fr

Keywords: time series forecasting, long-term prediction, self-organizing maps, Santa Fe, electrical load

Abstract **---** Many time series forecasting problems require the estimation of possibly inaccurate, but long-term, trends, rather than accurate short-term prediction. In this paper, a double use of the Self-Organizing Map algorithm makes it possible to build a model for long-term prediction, which is proven to be stable. The method uses the information on the structure of the series when available, by predicting blocs instead of scalar values. It is illustrated on real time series for both scalar and bloc predictions.

1 Introduction

Time series forecasting is a problem encountered in many fields of applications, as finance (returns, stock markets), hydrology (river floods), engineering (electrical consumption), etc. Many methods used traditionally for time series forecasting perform well (depending on the complexity of the problem) on a rather short-term horizon (a few steps of prediction), but are rather poor for longer-term prediction. This is due to the fact that they are usually designed to optimize the performance at short term, their use at longer term being not optimized. However, and despite the fact that long-term predictions in real cases will probably never be very accurate, there is a need in many applications to have insights about the possible structure of the series in the future: are there bounds on the future values, what can we expect in average, are confidence intervals on future values large or narrow, etc.?

Obtaining trends on future values is the purpose of the method proposed in this paper. Simulations, i.e. predictions performed recursively to enlarge the prediction horizon, are the real goal.

In this paper, we will limit the discussion to NARX-like models, i.e. non-linear auto-regressive prediction models, possibly with exogenous inputs, but without moving average terms. NARX is also the type of models for which SOM [1, 2] were used in the past in time series prediction context. For example, [3, 4, 5] used Kohonen maps and some local models to achieve the prediction goal, while [6, 7, 8] used Kohonen maps as global predicting model.

However, most of these methods and their variants are 1) discontinuous at the limits of validity of local models, and 2) not designed specifically to perform long-term prediction. Note that discontinuities precisely prohibit efficient long-term predictions, as injecting predictions in the models will lead to instabilities. In this paper, we will present a global model designed to be used in long-term prediction context, in view of obtaining forecasting trends (means, confidence intervals, bounds, etc.).

The following of this paper is organized as follows. Section 2 sets the basic concepts of time series prediction. Section 3 presents a method for long-term time series forecasting, based on a double use of the SOM algorithm. Structured series, for example hourly values with a daily quasi-periodicity, will be exploited by the method. Section 4 gives a sketch of the proof of the method stability, and section 5 presents experiments performed on two time series, the Santa Fe A series and a problem of electrical load forecasting.

2 Time series prediction

The classical problem of time series prediction is defined as follows:

$$\hat{y}_{t+1} = f\left(y_t, y_{t-1}, \ldots, y_{t-p+1}\right). \qquad (1)$$

In equation (1), series y is supposed to be known until time t, $\hat{y}_{t+1}$ is the estimation of the series at time t+1, vector

$$Y_t = \left[y_t\ y_{t-1} \ldots y_{t-p+1}\right] \qquad (2)$$

is the *regressor* at time t, and f(.) is the model used for

prediction. If $f(.)$ is linear, model (1) is AR (Auto-Regressive); if $f(.)$ is non-linear, the model is NAR (Nonlinear AR). Note that exogenous variables may be added to the past values of the series in regressor (2); in this case, the models become respectively ARX and NARX. The methodology presented in this paper is non-linear (NAR), and may be extended straightforward to NARX models. However in order to simplify the presentation, only NAR models will be considered here.

In some forecasting problems, it is interesting to predict several values of the series in one bloc, rather than a single $\hat{y}_{t+1}$ scalar value. For example, in an electrical load forecasting problem, hourly values to predict have a structure that makes natural to predict 24 values (one day) in a single pass. In such case, problem (1) becomes

$$\left[\hat{y}_{t+k} \quad \hat{y}_{t+k-1} \quad \cdots \quad \hat{y}_{t+1}\right] = f\left(y_t, y_{t-1}, \ldots, y_{t-p+1}\right). \quad (3)$$

Model $f(.)$ has now a vector output. Size p of the regressor is not necessarily equal to the forecasting horizon k. Nevertheless, in many cases, p will be a multiple of k; this will be assumed in the following again for simplicity reasons, while this is absolutely not a necessary condition for the proposed method. In our electrical load forecasting example, k=24, while the regressor could for example include the last three days values (p=72).

This paper will not address the problem of choosing optimal values for k and p; even for non-linear models $f(.)$, there is a vast literature on the topic (see for example [9] and [10]). If the quality of forecasting is chosen as criterion, extensive cross-validation may for example be used to choose optimal values for k and p, even if this way of working is computationally very intensive.

3 Double SOM for long-term prediction

3.1 Definitions

Based on time series y_t, we define the *series of deformations* as

$$d_t = y_{t+k} - y_y \,. \quad (4)$$

In problems where k=1, d_t is simply the difference between two successive values of the series. In our electrical load example where k=24, d_t represents the difference between the consumptions taken at the same hour on two consecutive days.

Similarly to (2), we also define a regressor in the deformation space as

$$D_t = \left[d_t \, d_{t-1} \ldots d_{t-p+1}\right] \quad (5)$$

3.2 SOM in the regressor space

Each time we have a regressor Y_t, the characterization of model $f(.)$ begins by looking in the past how the series evolves right after such regressor. In other words, we look in the past what are the deformations associated to regressors similar to Y_t. Of course, there is no chance to find exactly the same regressor Y_t in the past of the series: the past regressors will thus be arranged in classes (using a one-dimensional Kohonen string). Gathering regressors into classes also offers the advantage of performing local averages, which will reduce overfitting.

A Kohonen string with n_r centroids (or codevectors) A_i is thus organized in the space of regressors; each regressor Y_t is associated to a centroid $A_{i(t)}$ according to the nearest neighbor rule. In our electrical load example, assuming that the regressor includes the values of the last three days of the series, the dimension of centroids A_i is 72.

3.3 SOM in the deformation space

Once the Kohonen map in the regressor space has been formed, and a centroid $A_{i(t)}$ related to each regressor Y_t, we are looking for the way how, in the past, the series has evolved from any of the regressors associated to $A_{i(t)}$. This evolution is characterized by the deformations D_t associated to these regressors. We are thus looking for the statistical law of deformations D_t conditional to class i. To estimate these laws, we will proceed in two steps. First, classes are created in the deformation space, similarly to the ones created in the regressor space. A Kohonen string with n_d centroids B_j is thus organized in the space of deformations; each deformation D_t is associated to a centroid $B_{j(t)}$ according to the nearest neighbor rule. In our electrical load example and in the same conditions as above, the dimension of centroids B_j is 72 too. Secondly, the empirical law of deformations conditional to each class i is computed, as detailed in the next subsection.

3.4 Transition table

A so-called *transition table* T of size $n_r \times n_d$ is defined by

$$T[i,j] = P\left(B_j \middle| A_i\right), \quad (6)$$

the empirical probability that deformation D_t is associated to centroid B_j when the corresponding regressor Y_t is associated to centroid A_i. The sum of terms on each line i of the table is thus equal to 1, this line representing the empirical law μ_i of deformations conditional to class i.

This transition table justifies the use of Kohonen one-dimensional strings to create classes (in the regressor and deformation spaces), instead of simple vector quantization methods or Kohonen two-dimensional maps. The use of Kohonen maps is justified by the fact that the transition table will be organized: for example, as adjacent values on a specific row i will correspond to close centroids B_j in the deformation space, the corresponding probabilities given by (6) will be similar in most cases. A graph of the table values will also illustrate the behavior of the time series. As each entry (row or column) on the table corresponds to one of the regressor or deformation spaces, one-dimensional SOMs are preferred too (compared to

two-dimensional maps). An example of such a table will be provided in section 5. The classes resulting of the SOM algorithm will also be shown both in the regressor and deformation spaces. It is then possible to observe the code vectors represented in their respective classes.

3.5 Prediction

The organization of the SOM strings in the regressor and deformation spaces, and the evaluation of the transition table, constitute the modeling of the past behavior of the series. Next, a prediction may be performed as follows:

- regressor Y_t at time t is built;
- centroid $A_{i(t)}$ in the regressor space is identified;
- a deformation D_j is drawn randomly, according to the empirical law μ_i of probabilities $T[i, j]$;
- Y_t and D_j are summed according to

$$\left[y_{t+k}\ y_{t+k-1}\cdots y_{t+k-p+1}\right] = Y_t + D_t \qquad (7)$$

- the part $[y_{t+k}\ \ y_{t+k-1}\ \ \ldots\ \ y_{t+1}]$ extracted from vector (7) constitutes the prediction as defined by (3).

In our electrical load example, again assuming a regressor with three-days values, the vector defined by (7) has a dimension of 72, while the extracted prediction has dimension 24.

3.6 Long-term prediction and trends

As with any other prediction model, it is possible, in order to perform long-term prediction, to inject recursively the prediction(s) (7) in models (1) or (3). Of course, injecting predictions in the right members of models (1) and (3) may lead to rapid divergence, if the predictions are not accurate enough. One advantage of the proposed method is that any prediction remains in a limited domain, as proven in section 4. Therefore it cannot diverge, unlike other prediction models used for long-term forecasting. As mentioned in the introduction, this is precisely the aim of the method proposed in this paper: there is no argument to claim that it will perform better than any other forecasting method at horizon 1 (y_{t+1}) or even at horizon k (y_{t+k}). However, when the whole procedure from sections 3.2 to 3.5 is repeated by injecting predictions to obtain long-term forecasting, the structure itself of the method, together with its proof of stability, ensures a reasonable behavior of the long-term prediction. Note that when k>1, injecting predictions in (3) means to inject k predicted values, to obtain another set of k new predictions.

Trends are the ultimate goal of the method. Indeed the whole procedure can be repeated several times (Monte-Carlo procedure), leading to different forecasting curves, or simulations, due to the random choice in step 3.5. The different curves obtained may be seen as various instances of possible forecastings, and their trends, mean, standard deviations, etc. as global characteristics of the series in the future. This will be illustrated in section 5, where the method will be applied to two time series, respectively with k=1 and k>1.

4 Method Stability: Sketch of Proof

The predictions obtained by the model described in Section 3 should ideally be confined in the initial space defined by the learning data set. In that case, the series of predicted values Y_t is said to be stable. Otherwise, if the series tends to infinity or otherwise diverges, it is said to be unstable. Stability is naturally a necessary condition to obtain a valid long-term prediction.

The following of this section summarizes the proof of the stability of the method. The proof consists in two steps: it is first shown that the series generated by the model is a Markov chain; secondly, it is demonstrated that this particular type of Markov chain is stable.

To prove that the series is a Markov chain, we consider Y_0=x, the initial regressor of the series Y_t, and C_0 the corresponding SOM class in the regressor space. The deformation that is applied to Y_0 is D_0. Then the next values of the series are given by $Y_1 = Y_0 + D_0$, $Y_2 = Y_0 + D_0 + D_1$, …, with D_0, D_1, … drawn randomly from the transition table for classes C_0, C_1, … respectively.

The series Y_t is therefore a Markov chain, homogeneous in time (the transition laws are not time dependant), irreducible and defined over a numerable set (the initial Y_t are in finite number, and so are the deformations).

To show the stability of this Markov chain and thus the existence of a stationary law, Foster's criterion [11] is applied. A necessary and sufficient condition for an irreducible chain to be ergodic is that there exists a positive function $g(.)$, a positive ε and a finite set A such that:

$$\begin{aligned} &\forall x \notin A : E\left(g\left(Y_{t+1}\right) \mid Y_t = x\right) - g(x) \le \varepsilon \\ &\forall x \in A : E\left(g\left(Y_{t+1}\right) \mid Y_t = x\right) < \infty \end{aligned} \quad . \qquad (8)$$

We use $g(.) = ||.||^2$ in (8). Since the Markov chain is homogenous, it is sufficient to observe transition D_0 from Y_0 to Y_1. The demonstration is done for a two-dimensional case but can be generalized easily to other dimensions.

Before going in further details, let us remark that class C_0 covers less than a half plane. It is thus included in a cone with vertex A and delimited by the normalized vectors $a_1$and a_2 (see Fig. 1). There are two possibilities: either $a_1$and a_2 form an acute angle, either an obtuse one, as shown in Figs. 1a and 1b respectively.

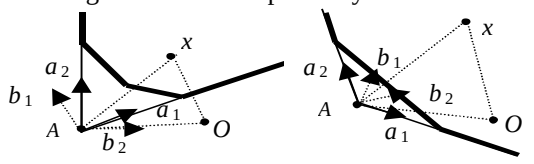

Fig. 1 Notations and conventions for the proof of stability; see text for details. Left: Fig 1a the class is included into a zone with acute angle; right: Fig 1b the class is included into a zone with obtuse angle.

Before applying Foster's criterion, note that the three following geometrical properties can be proven:
1. Denoting

$$lim_{x\to\infty} \frac{x}{\|x\|} \cdot a_i = \delta_i \,, \qquad (9)$$

we have δ_1 and δ_2 both positive in the acute angle case, while either δ_1 or δ_2 is positive for an obtuse angle.
2. We define b_1 such that the angle between a_1 and b_1 is $\pi/2$. Similarly b_2 is defined such that its angle with a_2 is also $\pi/2$. Then, for both the acute and obtuse angle cases, we have

$$inf_{x\in C} \frac{\overrightarrow{Ax}}{\|x\|} \cdot b_i > 0 \,. \qquad (10)$$

3. Assume that (which can easily be proved numerically):

$$E_{\mu_0}(D_0) \cdot a_1 < 0 \text{ and } E_{\mu_0}(D_0) \cdot a_2 < 0 \,, \qquad (11)$$

where μ_0 is the empirical low corresponding to class C_0 in the transition table. Then as shown in Fig. 2, we have:

$$E_{\mu_0}(D_0) \cdot b_i < 0 \qquad (12)$$

for either i=1 or i=2 in case of an acute angle (Fig. 2a) or for both of i=1 and i=2 for the obtuse case (Fig. 2b).

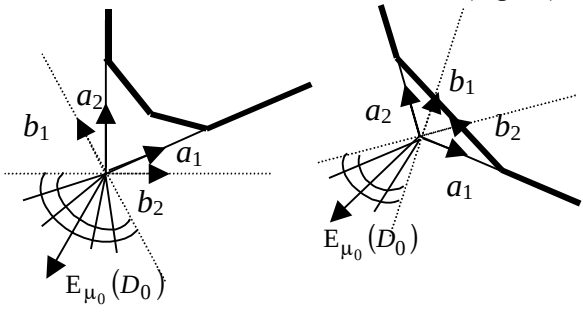

Fig. 2 Third geometrical property, see text for details. Left: Fig 2a the acute angle case; right: Fig 2b the obtuse angle case.

Now we can apply Foster's criterion. Considering class C_0 and the corresponding transition law, with $g(x) = \|x\|^2$, we have:

$$\begin{aligned} E(g(Y_1)|Y_0 = x) - g(x) &= E(g(Y_0 + D_0)|Y_0 = x) - g(x) \\ &= E\left(\|Y_0 + D_0\|^2 | Y_0 = x\right) - \|x\|^2 \\ &= 2\|x\| \left[\frac{x\, E_{\mu_0}(D_0)}{\|x\|} + \frac{E_{\mu_0}\left(\|D_0\|^2\right)}{2\|x\|} \right] \end{aligned} \qquad (13)$$

At this point, using geometrical properties 1., 2. and 3., it can be shown that there exist a positive real number η such that :

$$\frac{x\, E_{\mu_0}(D_0)}{\|x\|} + \frac{E_{\mu_0}\left(\|D_0\|^2\right)}{2\|x\|} < -\eta < 0 \qquad (14)$$

when $\|x\|$ is large enough. We thus have

$$E(g(Y_1)|Y_0 = x) - g(Y_0) < -2\|x\|\eta \qquad (15)$$

which tends to minus infinity for $\|x\|$ tending to infinity. Thus, finally, Foster's criterion can be applied since, for Y_0=x large enough, we obtain

$$E(g(Y_1)|Y_0 = x) - g(Y_0) \to -\infty \,. \qquad (16)$$

Thus the chain Y_t is ergodic, and the transition law is stationary.

5 Experimental Results

The method proposed in this paper is illustrated here on two time series. The first one is the SantaFe A series: a laser series presented in [9]. In this case the forecasting horizon k is equal to 1. The second series represents the hourly electrical load in Poland from 1989 to 1996. The forecasting horizon is equal to 24 (corresponding to one day) in this second data set.
The model (i.e. the double SOM string and the transition table) is trained on a learning set. The optimal number of parameters (the numbers n_r and n_d of centroids in each SOM) is chosen in order to maximize the performances on a validation set, according to criterion

$$e_{MSE} = \sum_{Y_t \in ValidSet} \left(Y_{t+1} - \hat{Y}_{t+1}\right)^2 \,. \qquad (17)$$

Finally, this optimal model is applied on a test set in order to evaluate the performances on new data.
Before using this model for simulation, a new learning is done with a new learning set obtained from the reassembled learning and validation sets. This new learning is only performed for the model with optimal n_r and n_d.

5.1 SantaFe A Times Series

The initial data set has been divided in three subsets: the learning set with 6000 data, the validation set with 2000 data, and test set with 100 data. The regressors Y_t are obtained as

$$Y_t = [y_t y_{t-1} y_{t-2} y_{t-3} y_{t-5} y_{t-6}]. \qquad (18)$$

This choice is here made a priori, according to previous experience on this series [9]; as mentioned in section 2, the purpose here is not to discuss about the structure of the regressor.
Note that the best neural network models described in [9] do not predict much more than 30 data, making a 100-data test set a "long-term" forecasting.
In this simulation, Kohonen strings of 1 to 200 centroids in each space have been used; all 40 000 possible models have been tested. The best model among them has 179 centroids in the regressor space and 161 centroids in the deformation space.
After learning this model on both the learning and validation sets, 1000 simulations were performed on a 100-steps horizon. Then the mean and confidence interval at 95% level were computed, giving information on the time series trends.

Fig. 3. presents the mean of the 1000 simulations compared to the true values, together with the confidence interval at 95% level. Fig 4. shows a zoom on the first 30 values. In Fig. 5, we can see 100 simulations for the same 30 values. Note the stability obtained through the replications. Fig. 6 shows the code vectors and regressors (resp. deformations) in each class. For simplicity, those curves come from a simpler model, with $n_r = 6$ and $n_d = 8$, its transition table being shown in Table 1.

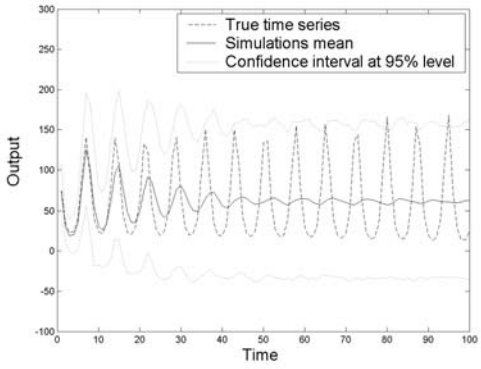

Fig. 3 Comparison between the mean of the 1000 simulations and the true values, together with confidence intervals at 95% level.

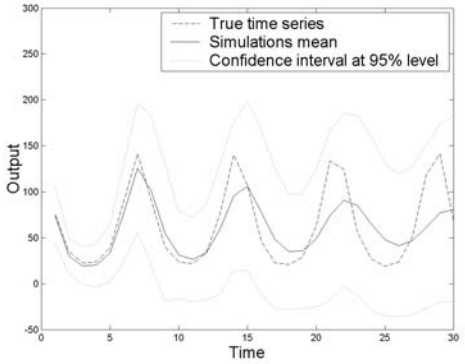

Fig. 4 Comparison for the first 30 values between the mean of the 1000 simulations and the true values, together with confidence intervals at 95% level.

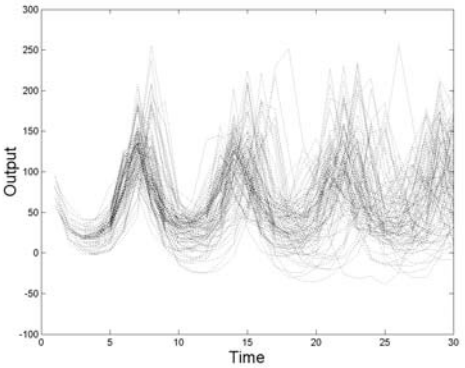

Fig. 5 100 simulations picked out at random from the 1000 simulations made for the Santa Fe A long-term forecasting.

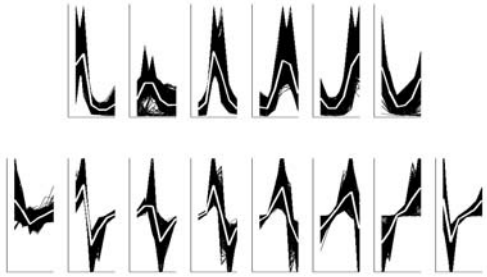

Fig. 6 The code vectors and associated curves in the regressor (top) and deformation (bottom) spaces. The code vectors are represented in white as 6-dimensional vectors (according to Eq. (18)). Regressors (resp. deformations) belonging to each class are shown on black. (A simpler model was chosen with $n_r = 6$ and $n_d = 8$).

0.121	0	0	0	0	0	0.226	0.653
0.668	0.302	0.002	0	0.001	0.003	0.019	0.005
0.049	0.545	0.406	0	0	0	0	0
0.025	0	0.304	0.543	0.127	0.001	0	0
0.002	0	0	0	0.508	0.476	0.014	0
0.060	0	0	0	0.001	0.337	0.561	0.041

Table 1 Example of transition table, here with $n_r = 6$ and $n_d = 8$ as in Fig. 6. Note that for each line, the sum of the transition values equal one.

5.2 Electrical load

For our second example, we use the Polish electrical load time series. This series contains hourly values from 1989 to 1996. This is an illustration of the case $k>1$, since it seems natural to predict the 24 next values in one step (the next day).

Ignoring the best regressor, different combinations were tried. The results concern regressors constructed with the daily 24 values for a few past days. More precisely, we take today's 24 values, plus yesterday, two, six and seven days ago. The regressors are thus of dimension 120. Again, this choice is made a priori since founding the optimal regressor is not our goal here.

In this second example, the learning set contains 2000 data of dimension 120, the validation set has 800 data, and test set 200 data.

Tested models have 5 to 200 centroids in each space, by steps of 5. The best model has 160 and 140 centroids in the regressor and deformation spaces respectively and is trained again on the joined learning and validation sets. Then, 1000 simulations are performed, and the mean and confidence interval at 95% level computed.

Fig. 7. presents the mean of the 1000 simulations compared to the true values, together with the confidence interval at 95% level, zoomed on the first 240 values (10 days). In Fig. 8, we can see the first 1500 predicted values (thus more than two months), compared to the true values (confidence interval have been removed for clarity). Long-term simulation stability is highlighted in Fig. 9.

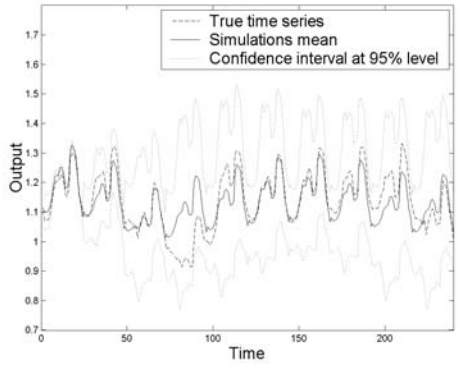

Fig. 7 Comparison between the mean of the 1000 simulations and the true values, together with confidence intervals at 95% level. Here we can see the 240 first values (10 days).

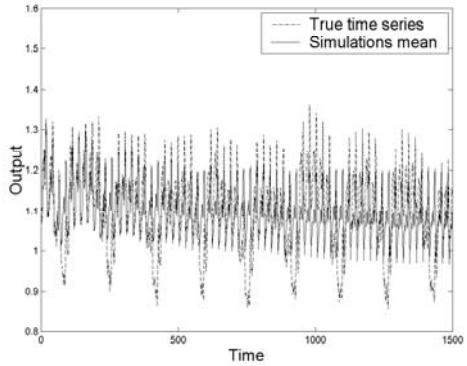

Fig. 8 Comparison between the mean of the 1000 simulations and the true values. Here we can see the 1500 first predicted values (more than 2 months).

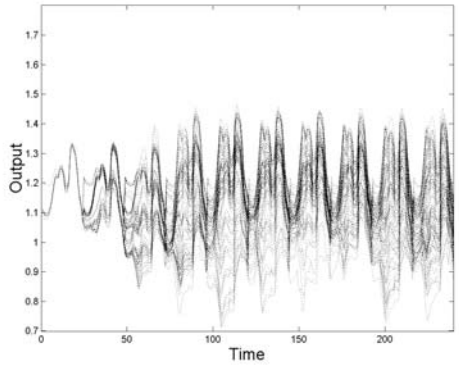

Fig. 9 As in Fig. 5, 100 simulations picked out at random from the 1000 simulations. Here again, note the regularity obtained for the different replications.

Conclusion

In this paper, a new specific method for achieving long term forecasting using double SOM is presented. The main argument proving the stability of the method, and thus its validity in long-term prediction context, is presented, and a sketch of its prove is given.

The long-term forecasting capacities of the method are highlighted for a classical benchmark as the Santa Fe A time series as well as for a real-world application, an electrical load time series.

Acknowledgements

We would like to thank Professor Osowski from Warsaw Technical University for providing us the Polish Electrical Consumption data used in our second example. G. Simon is funded by the Belgian F.R.I.A. M. Verleysen is Senior Research Associate of the Belgian F.N.R.S. The work of A. Lendasse is supported by the Interuniversity Attraction Poles (IAP), initiated by the Belgian Federal State, Ministry of Sciences, Technologies and Culture. The scientific responsibility rests with the authors.

References

[1] T. Kohonen, "Self-organising Maps", *Springer Series in Information Sciences*, Vol. 30, Springer, Berlin, 1995.

[2] M. Cottrell, J. C. Fort, G. Pagès , "Theoretical aspects of the SOM algorithm", *Neurocomputing*, 21, p. 119-138, 1998.

[3] J. Walter, H. Ritter, K. Schulten, "Non-linear prediction with self-organising maps", in *Proc. of IJCNN*, San Diego, CA, 589-594, July 1990.

[4] J. Vesanto, "Using the SOM and Local Models in Time-Series Prediction", In *Proc. of WSOM'97*, Espoo, Finland, pp. 209-214, 1997.

[5] T. Koskela, M. Varsta, J. Heikkonen, and K. Kaski, "Recurrent SOM with Local Linear Models in Time Series Prediction", in *Proc. of ESANN'98*, pp. 167-172, D-Facto, Brussels, 1998.

[6] M. Cottrell, E. de Bodt, Ph. Grégoire, "Simulating Interest Rate Structure Evolution on a Long Term Horizon: A Kohonen Map Application", in *Proc. of Neural Networks in The Capital Markets*, Californian Institute of Technology, World Scientific Ed., Pasadena, 1996.

[7] M. Cottrell, B. Girard, P. Rousset, "Forecasting of curves using a Kohonen classification", *Journal of Forecasting*, Vol. 17, pp. 429-439, 1998.

[8] A. Lendasse, M. Verleysen, E. de Bodt, M. Cottrell, Ph. Grégoire, "Forecasting Time-Series by Kohonen Classification", in *Proc. of ESANN'98*, pp. 221-226, D Facto, Brussels, 1998.

[9] A. S. Weigend and N.A. Gershenfeld, "*Times Series Prediction: Forcasting the future and Understanding the Past*", Addison-Wesley Publishing Company, 1994.

[10] A. Lendasse, J. Lee, V. Wertz, M. Verleysen, "Forecasting electricity consumption using nonlinear projection and self-organizing maps", *Neurocomputing*, Vol. 48, Nos. 1-4, pp. 299-311, 2002.

[11] G. Fayolle, V. A. Malyshev, M. V. Menshikov, "*Topics in constructive theory of countable Markov chains*", Cambridge University Press, 1995.