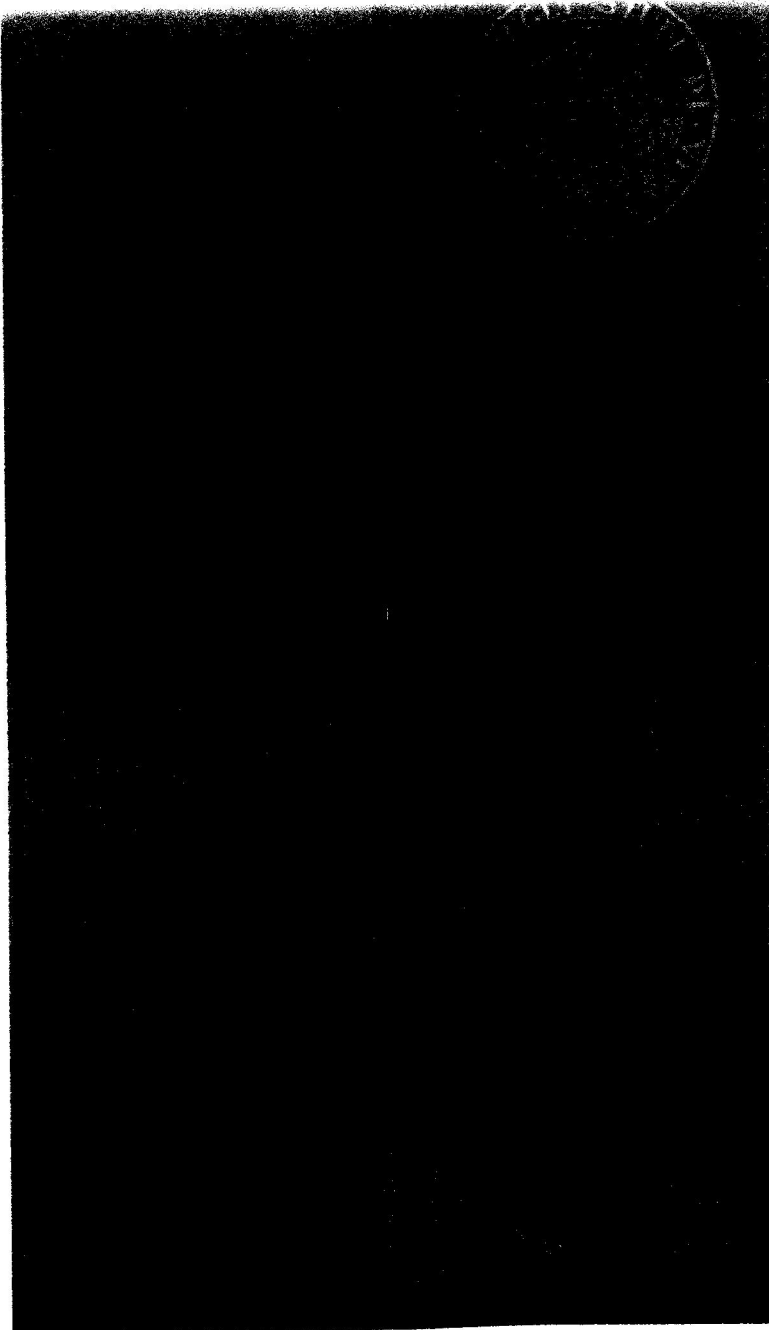


**Dipartimento di Scienze Economiche
dell'Università di Verona**

Via dell'Artigliere 19
37129 Verona
Tel. 045 8028245
Fax 045 8028529



UNIVERSITÀ DI VERONA
DIPARTIMENTO DI SCIENZE ECONOMICHE

*Neural Nearest-Neighbor
Regression for
Time Series Prediction*

MARCO SANDRI
GIOVANNI TONDINI

Novembre 2000

***Neural Nearest-Neighbor Regression For
Time Series Prediction***

MARCO SANDRI

Calcolo Scientifico – S.I.A. – Università di Verona

GIOVANNI TONDINI

Dipartimento di Scienze Economiche, Università di Verona

Abstract

In this note, we describe a forecasting method for time series which represents, from different point of views, an extension of the work of [13], recently published on this journal. Combining some basic notions of the theory of nonlinear dynamical systems and of neural networks, the method leads to the construction of a nonlinear piece-wise model in the reconstructed phase space. A description of the algorithm together with some flexible routines are provided.

Keywords. Time series, forecasting, neural networks, nearest neighbor

The sections 2.2, 3, 4 of this note are written by Marco Sandri, the sections 1, 2.1, 2.3, 5 are drafted by Giovanni Tondini.

Dipartimento di Scienze Economiche dell'Università di Verona
Via dell'Artigliere, 19
37129 Verona
Tel. 0458028245
Fax 0458028529

Marco Sandri
Calcolo Scientifico – S.I.S.
Tel. 0458028511
Fax 06233220486
E-mail: msandri@univr.it

Giovanni Tondini
Dipartimento di Scienze Economiche
Tel. 045 8028414
Fax 0455028417
E-mail: tondini@chiostro.univr.it

Neural Nearest-Neighbor Regression for Time Series Prediction

Finito di stampare il 13 novembre 2000
dal Centro Stampa dell'Università di Verona

1 Introduction

Jadiz and Riddick [13] provides some flexible routines for implementing the nearest-neighbors approach to time series prediction. In this paper we present some interesting extensions which are based on the theory of nonlinear dynamical systems [14] and on feedforward artificial neural networks (F-ANN)[31].

First, we start dealing with the identification and reconstruction (embedding) of a low-dimensional attractor by the Takens Theorem, that is with the choice of the time lag τ and the embedding dimension d . We then apply a preprocessing (filtering) technique known as Singular Spectrum Analysis (see [1] and [28]), which proved to be quite useful in improving forecasting of noisy data [19]. After identifying in the reconstructed phase space the k closest neighbors of the reference points y_i , we estimate a local neural regression model and predict the state y_{i+T} of the system, where T is the prediction step.

Some papers showed that the proposed technique is able, under proper conditions, to prolong the trajectories of an unknown chaotic low-dimensional dynamical system (see [7] and [8]). This matter is only partially connected to time series forecasting. Experiments conducted on some time series generated by a simple (noisy and noise-free) chaotic model, show that the method is potentially interesting for predicting the future behavior of the process.

Section 2 briefly presents the theoretical notions underlying the proposed method. Section 3 contains the description of the algorithm and the discussion of some applicability aspects. In section 4 the code is described and some practical examples are reported. Moreover, the predictive performances of the methodology applied to some chaotic time series with and without noise are shown. Section 5 concludes.

2 Theoretical background

2.1 Nonlinear dynamics

Let the data generating process (DGP) be measured by the variable $Y_t \in \mathbb{R}$ and let the series $\{y_t : 1 \leq t \leq N\}$ be one of its realizations of length N . Following [13], we denote our data with $\{z_t = (y_t, \mathbf{x}_t) : 1 \leq t \leq N\}$, where $\mathbf{x}_t$ is a vector of covariates. In the field of time series prediction, the elements of $\mathbf{x}_t$ are often lagged values of y_t . This is true

especially when, assuming the existence of an underlying deterministic dynamics in the series, we apply the Takens' Theorem.

We assume that $Y_t = \Psi_t + \Xi_t$, where Ξ_t is a purely stochastic component, and Ψ_t is a deterministic process, governed by an evolutionary deterministic law. Nella maggior parte dei casi reali, we do not have direct and complete access to the entire vector of the state variables Γ_t of the true DGP. Typically we observe only one (or a combination) of them through a certain 'viewer' function $h : \mathbb{R}^m \rightarrow \mathbb{R}$, that is: $\Psi_t = h(\Gamma_t)$, where $d\Gamma_t/dt = F(\Gamma_t)$ is the true DGP, and $F : \mathbb{R}^m \rightarrow \mathbb{R}^m$ is the vector field. Under weak regularity conditions on h and F , and for sufficiently large positive integer d ($d \geq 2m + 1$), the Takens' Theorem guarantees that the time evolution of the reconstructed vector:

$$\Psi_t^d = (\Psi_t, \Psi_{t-\tau}, \dots, \Psi_{t-(d-1)\tau}) \in \mathbb{R}^d$$

is diffeomorphically equivalent to the true (and unknown) dynamics of Γ_t [25, 2]. This means that there exists a deterministic map $G : \mathbb{R}^d \rightarrow \mathbb{R}^d$, dependent on d , such that:

$$\Psi_{t+1}^d = G(\Psi_t^d) \quad (1)$$

Because of the unpredictable character of Ξ_t , we should concentrate our forecasting efforts only on the (unknown) Ψ_t component, trying to approximate as better as possible the map G of (1) in the reconstructed space. This approach justifies the choice of setting $\mathbf{x}_t$ as a vector of lagged values of y_t . Of course, the quality of the predictions strongly depends on the signal-to-noise ratio, defined by $s/n = \sigma_\Psi^2 / \sigma_\Xi^2$, where σ_Ψ^2 and σ_Ξ^2 are the variances of Ψ and Ξ respectively. In order to estimate G , many global [17, 29, 27] or local [5, 3, 25], linear [5, 24] or nonlinear [29, 20] models have been proposed. In the present work, we analyse a neural nearest-neighbor forecasting method, that is a nonlinear regression model based on feedforward neural approximators. The aim is to obtain local estimates $\hat{G}_{y_j}$ of G in the neighborhood of y_j , where for neighborhood we mean the k nearest points of y_i in the reconstructed phase space.

2.2 Filtering with SSA

It is worth to point out that all the classical theorems concerning embedding and phase space reconstruction are based on the hypothesis that data are noise-free (see [25]). Only recently some extensions

of geometric time series analysis to genuinely stochastic systems appeared [21, 26]. Moreover, as written above, if we want to predict the future evolution of the series, we must concentrate our efforts only on the deterministic part Ψ of the process. These are the two main reasons which led us to the decision of applying a filter to our data in order to separate as much as possible Ψ from Ξ .

Many are the parametric and nonparametric filtering methods for time series proposed in literature, see for example [23, 4, 11]. An intuitive and easily applicable technique, which shows good denoising properties, is the one based on Singular Spectrum Analysis (SSA) [2, 28]. As pointed out by [16] (see also and [25]), SSA is by itself an alternative (and for some aspects, better) reconstructing method. Thus, it can be easily integrated in the algorithm proposed here.

For simplicity, we suppose that y_t be a unidimensional vector of observations. The generalization to the multidimensional case is straightforward and will be omitted.

The first step in the implementation of SSA is to construct the so-called ‘trajectory matrix’ in the reconstructed space:

$$\mathbf{Y}^d = \frac{1}{\sqrt{N}} \begin{bmatrix} y_1 & \dots & y_d \\ y_2 & \dots & y_{d+1} \\ \vdots & & \vdots \\ y_{N-m+1} & \dots & y_N \end{bmatrix} = \begin{bmatrix} y_1^d \\ y_2^d \\ \vdots \\ y_{N-m+1}^d \end{bmatrix} \quad (2)$$

From now on, we drop the superscript d whenever possible. Using the well known Singular Value Decomposition (SVD), the rectangular matrix $\mathbf{Y}$ can be decomposed as $\mathbf{Y} = \mathbf{S}\mathbf{\Sigma}\mathbf{C}^T$, where $\mathbf{S}$ is a matrix whose columns $\mathbf{s}_i$ are the eigenvectors of the symmetric matrix $\mathbf{Y}\mathbf{Y}^T$; $\mathbf{C}$ is a matrix whose columns $\mathbf{c}_i$ are the eigenvectors of the covariance matrix $\mathbf{Y}^T\mathbf{Y}$; $\mathbf{\Sigma}$ is a diagonal matrix whose elements λ_i are the positive square roots of the eigenvalues of $\mathbf{Y}^T\mathbf{Y}$. The $\mathbf{s}_i$ and $\mathbf{c}_i$ vectors are also called *singular vectors* and the elements of $\mathbf{\Sigma}$ (considered in descending order) are called *singular values* of $\mathbf{Y}$.

Because singular vectors are orthogonal, the $\mathbf{c}_i$ vectors can be used as an orthonormal basis of the space $\mathbb{R}^d$ on which d -dimensional points $\mathbf{z}_i^d$ can be projected. Those vectors are sometimes called ‘empirical orthogonal functions’ (EOFs). The columns of the matrix $\mathbf{A} \equiv \mathbf{Z}\mathbf{C}$ are called *principal components* (PCs) and represent the coefficients of projection of vectors $\mathbf{z}_i^d$ onto the base $\{\mathbf{c}_i\}_{i=1}^p$. Finally, the singular values λ_i can be ordered as $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_m \geq 0$.

If we consider a trajectory exploring on average an ellipsoid of dimension d , the $\mathbf{c}_i$ vectors correspond to the directions of the principal axes of the ellipsoid, and the associated λ_i^2 values correspond to the lengths of the axes. Each singular value λ_i is equal to the variance of the i -th principal component.

Recalling our basic assumption that $\{y_t\}$ is the result of a deterministic DGP contaminated by an observational white noise, than, if the series is stationary and sufficiently long, the covariance matrix $\mathbf{Y}^T\mathbf{Y}$ can be approximated as follows:

$$\mathbf{Y}^T\mathbf{Y} = \mathbf{\Psi}^T\mathbf{\Psi} + (\sigma_{\Xi}^2/d)\mathbf{I} \quad (3)$$

where σ_{Ξ}^2 denotes the variance due to noise, $\mathbf{I}$ is the $(d \times d)$ identity matrix and $\mathbf{\Psi}$ is the trajectory matrix of the deterministic part Ψ_t . Clearly, in this case the singular vectors $\mathbf{c}_i$ of the perturbed trajectory matrix will be the same as those of the unperturbed one, whereas, owing to noise, the singular values will be uniformly increased by an amount σ_{Ξ}^2/d .

For any given value of the window length d and the level of noise σ_{Ξ}^2 , the signal-to-noise ratio (s.n.r.) associated with each direction - measured by the quantity $d\lambda_i^2/\sigma_{\Xi}^2$ - will decrease as the order of the corresponding singular value increases and clearly noise can entirely dominate signal for ‘higher order directions’. While, in general, the singular values of a noise-free series will be uniformly declining with the order i , in the presence of sufficiently strong white noise we should observe a plateau - the ‘noise floor’ - in the spectrum. If this is the case, only $p \leq d$ singular values will be above the noise floor.

2.3 Neural regression models

The choice of a feed-forward artificial neural network (F-ANN) is motivated by two properties. First, they can approximate virtually any function of interest to any desired degree of accuracy, provided many sufficiently hidden units are available. In other words, this kind of neural network represents a class of universal approximators [12]. This property makes neural models very appealing in nonlinear regression. Secondly, [31] demonstrates that, under mild smoothness conditions on f , feedforward neural networks are consistent estimators of f . In the specific context of chaotic dynamics, neural networks are found to be competitive with the best of the approximation methods in the

construction of a nonlinear map from a given time series [30] as well as in recovering the derivatives of a nonlinear map [6].

Without going into details (see [31] for a wide and rigorous treatment of the statistic aspects of F-ANN), a F-ANN is made by a set of elementary units (the neurons) connected together according to an architecture organized by layers: an input layer, one or more hidden layers and an output layer. In our algorithm, input patterns are the k d -dimensional nearest neighbors of $\mathbf{y}_i$, $\{\mathbf{y}_{n_j}\}_{j=1}^k$, where d is the embedding dimension of the reconstructed space. Input units send information to the hidden units using a linear transformation determined by their connection strength γ_{rs} . Each hidden unit performs a nonlinear transformation on its total input, producing an output using a sigmoid function $K(\cdot)$, called activation function, which usually is the same for all the hidden units. The output $\hat{\mathbf{y}}_{i+T} \in \mathbb{R}^d$ of a 1-layer F-ANN can be represented as

$$\hat{\mathbf{y}}_{i+T} = \sum_{r=1}^h \beta_r \cdot K\left(\sum_{s=1}^d \gamma_{rs} y_{is}\right) + b_r \quad (4)$$

where y_{is} is the s component of $\mathbf{y}_i$ ($y_{is} = y_{i-(s-1)\tau}$), and β_r , with $\beta_r \in \mathbb{R}^h$, are the weights connecting the output of the hidden units to the output units, h is the number of hidden units, and b_r is the bias of the hidden units. The k output vectors of the net are then compared to the points $\{\mathbf{y}_{n_j+T}\}_{j=1}^k$ which represent the T -ahead evolutions of the system starting from the neighbor $\mathbf{y}_i$, $\{\mathbf{y}_{n_j}\}_{j=1}^k$. The β and γ parameters are estimated minimizing the error function:

$$\sum_{j=1}^k (\mathbf{y}_{n_j+T} - \hat{\mathbf{y}}_{n_j+T})^2$$

3 The algorithm

The proposed forecasting algorithm can be summarized as follows:

1. Starting from the observed time series $\{\mathbf{y}_t\}_{t=1}^N$, use the method of delay (MOD) to build the $L = N - d + 1$ points $\mathbf{y}_t \in \mathbb{R}^d$ in the reconstructed phase space, where $\mathbf{y}_t = \{\mathbf{y}_t, \mathbf{y}_{t+\tau}, \mathbf{y}_{t+2\tau}, \dots, \mathbf{y}_{t+(d-1)\tau}\}$, d is the embedding dimension and τ is the time delay.
2. Apply SSA on the trajectory matrix $\mathbf{Y}$ and take the first p principal components, that is consider the first p columns of the matrix

YC. Call $\bar{\mathbf{y}}_i$ the p -dimensional points of the filtered phase space obtained in this manner.

3. Take the last point of the reconstructed trajectories $\bar{\mathbf{y}}_L$.
4. Look for the k closest points to $\bar{\mathbf{y}}_L$, $\{\bar{\mathbf{y}}_{n_j}\}_{j=1}^k$, i.e. find the k points which minimize the distance from the reference point $\bar{\mathbf{y}}_L$; this set of points represents the total input of the neural network.
5. Select the successors $\{\bar{\mathbf{y}}_{n_j+1}\}_{j=1}^k$, i.e. find the k points in the phase space which correspond to the evolutions of the dynamical system starting from the set of points elected in step (4); this set of points is the total output of the neural network.
6. Locally estimate the map G fitting a neural model $\hat{G}_{\bar{\mathbf{y}}_L}$ to the two sets of points in $\mathbb{R}^d$ calculated in step 4 and 5.
7. Put $\bar{\mathbf{y}}_L$ in the input of the network and calculate $\hat{\mathbf{y}}_{L+T} = \hat{G}_{\bar{\mathbf{y}}_L}(\bar{\mathbf{y}}_L)$; this is the prediction of the state of the system after a time of length T .

The parameters involved in the algorithm are: the number h of hidden units of the F-ANN, the time delay τ , the embedding dimension d and the number of nearest neighbors k .

One of the most important and unresolved problems in practical applications of F-ANN is how to determine the appropriate complexity h of the network in order to avoid overfitting and to improve generalization. Roughly speaking, the F-ANN should have a sufficiently high degree of complexity, that is a sufficient capacity to approximate the unknown function $G_{\mathbf{y}}$. In the meantime, it should not be too much complex because it could overfit data loosing the capacity to generalize. The two main approaches to controlling complexity are model selection (pruning, bootstrap estimate of the generalization error) and regularization (weight decay, early stopping [22], etc.). In our forecasting procedure, overfitting is not a crucial matter because data are always filtered before model estimation. Therefore, we prefer to base the choice of h on the minimization of the in-sample normalized mean square prediction error (NMSPE) for different degrees of complexity.

Some authors [9, 14, 10] argue that an 'optimal' embedding can only be defined relative to a specific purpose for which the embedding is used. Nevertheless, the usual autocorrelation function, the time delay mutual information and a visual inspection of delay representations with various lags provide useful information about reasonable delay times. For highly sampled flow data, small delays yield delay

vectors which are all concentrated around the diagonal in the reconstructed space and thus the structure perpendicular to the diagonal is almost invisible. On the converse, large delays lead to vectors whose components are almost uncorrelated and seemingly randomly distributed.

The false nearest neighbors procedure is a popular and intuitive method which can give guidance about the proper dimension of reconstruction [15]. It examines the fraction of nearest neighbors as a function of the embedding dimension to determine the necessary global dimension to unfold an attractor. Thus the minimum embedding dimension is found when most of the nearest neighbors do not move apart significantly in the next higher dimensional embedding. Because prediction is the final purpose of the embedding step of our algorithm, we prefer to choose d together with the number k of nearest neighbors. Following [3], we divide the available training data into a learning and a validation set and compute the NMSPE on the validation set for different values of k and d . The optimal values of the parameters are those corresponding to the minimum prediction error.

The problem concerning the choice of the optimal number p of principal components, which allow to eliminate as much as possible the noise from the series with a minimal distortion of the underlying dynamics, is a hard and substantially unresolved question. Some authors [28] suggest to evaluate p by bootstrap methods, while others [18] propose a 'best-prediction' method which is strictly related to our forecasting problem under many aspects. For simplicity, here we prefer to adopt the approach of [1] where the optimal p is defined as the number of singular values above the noise floor.

4 Code and examples

A description of the main commands is given in appendix.

In order to show the use of the provided code, we consider three examples which are more and more close to a real situation.

In the first case we suppose to be in a noise-free environment and to have a series containing all the variables which completely define the state of the system. These are two rather unrealistic conditions but they greatly simplify our problem: we can avoid the embedding and filtering steps. The algorithm reduces to finding the k nearest neighbors of the point $\bar{y}_L$ and fitting a local neural model $\hat{G}_{\bar{y}_L}$.

We start generating a 3-dimensional trajectory matrix of a Lorenz dynamical model by our `integrate` command (a fixed-stepsize third-order Runge-Kutta method). The required arguments are: (1) a string containing the name of an ODE file, (2) a vector of system parameters, (3) a vector containing the initial condition, (4) the stepsize, (5) the transient time, (6) the length of the output series.

```
lor=integrate('lorenz',[16 45.92 4],[10,10,10],.01,1000,10000);
```

After splitting the series in two parts, a learning (`insamp`) and a testing (`outsamp`) set, we apply the `nnpredic` command: the number k of nearest neighbors is 100, the prediction step T is 25, the local model is a F-ANN with a direct prediction method and a number of units in the hidden layer equal to 10.

```
nlearn=length(lor)-ntest;
insamp=lor(1:nlearn,:); outsamp=lor(nlearn+1:end,:);
preds=nnpredic(insamp,outsamp,100,25,'neural','direct',10);
```

The below commands yield a plot of the true trajectory together with the corresponding predicted values (see Fig. 2):

```
preds=preds(1:end-25,:); outsamp=outsamp(26:end,:);
plot3(preds(:,1),preds(:,2),preds(:,3),'.',...
      outsamp(:,1),outsamp(:,2),outsamp(:,3)));
```

In the second example, we suppose to measure only one of the three noise-free state variables describing the system. Therefore, we must first apply the `MOD` to reconstruct the underlying attractor. The `rebuild` command yields the trajectory matrix in the reconstructed phase space. Fig. 3 plots the cross-validation NMSPE vs. d and k and suggests an embedding dimension equal to 3 together with $k \approx 80$.

```
x=lor(:,1);
m=3; tau=5;
y=rebuild(x,m,tau);
ntest=length(y)-ntest;
insamp=y(1:nlearn,:); outsamp=y(nlearn+1:end,:);
preds=nnpredic(insamp,outsamp,80,25,'neural','direct',10);
preds=preds(1:end-25,:); outsamp=outsamp(26:end,:);
plot3(preds(:,1),preds(:,2),preds(:,3),'.',...
      outsamp(:,1),outsamp(:,2),outsamp(:,3)));
```

In the final example we consider a quite realistic case: only one noisy state variable is available. Therefore, we need to apply the algorithm as a whole.

```
noisyx=x+2*randn(size(x));
m=10; tau=5; p=3;
ssaplot(noisyx,m,tau,[1:4],'all');
```

The `ssaplot` command gives different plots of the spectrum of singular values (see Fig. 4) and suggest to take three principal components in the SSA filter.

```
filtx=(rc(noisyx,m,tau,p))';
y=rebuild(filtx,p,tau);
insamp=y(1:nlearn,:); outsamp=y(nlearn+1:end,:);
preds=nnpredic(insamp,outsamp,80,25,'neural','direct',10);
preds=preds(1:end-25,:); outsamp=outsamp(26:end,:);
n=length(preds);
rbx=rebuild(x,p,tau);
rbx=rbx(nlearn+26:end,:);
plot(1:n,preds(:,3),'.',1:n,rbx(:,3));
disp(nmse(preds(:,3),rbx(:,3)));
```

Fig. 5 shows the plot of the true noise-less series and of the corresponding predicted values. The NMSPEs for linear, quadratic and neural models are 0.64, 0.55 and 0.24, respectively.

5 Summary

The simulations of the above section suggest that neural nearest-neighbors prediction models can outperform the corresponding linear or quadratical models. Therefore, although they impose a heavier computational burden, neural models should be preferred to or, at least, should be used in combination with other forecasting models.

References

- [1] D. S. Broomhead and G. P. King. Extracting qualitative dynamics from experimental data. *Physica D*, 20:217–236, 1986.

- [2] D. S. Broomhead and G. P. King. On the qualitative analysis of experimental dynamical systems. In S. Sarkar, editor, *Nonlinear phenomena and chaos*. Adam Hilger, Bristol, 1986.
- [3] M. Casdagli. Nonlinear prediction of chaotic time series. *Physica D*, 35:335–356, 1989.
- [4] D. Donoho. Nonlinear wavelet methods for recovery of signals, densities, and spectra from indirect and noisy data. In I. Daubechies, editor, *Different Perspectives on Wavelets, Proceeding of Symposia in Applied Mathematics*. Amer. Math. Soc., Providence, 1993.
- [5] J. D. Farmer and J. J. Sidorowich. Exploiting chaos to predict the future and reduce noise. In Y.-C. Lee, editor, *Evolution, Learning and Cognition*, pages 277–330. World Scientific, Singapore, 1989.
- [6] A. R. Gallant and H. White. On learning the derivatives of an unknown mapping with multi-layer feedforward networks. *Neural Networks*, 5:129–138, 1992.
- [7] S. Golia and M. Sandri. Resampling chaotic time series. *Phys. Rev. Lett.*, 78(22):4197–4200, 1997.
- [8] S. Golia and M. Sandri. A resampling algorithm for chaotic time series. *Statistics and Computing*, Special Issue on chaos:forthcoming, 2001.
- [9] P. Grassberger, T. Schreiber, and C. Schaffrath. Nonlinear time sequence analysis. *Int. J. of Bifurcation and Chaos*, 1(3):521–547, 1991.
- [10] R. Hegger, H. Kantz, and T. Schreiber. Practicable implementation of nonlinear time series methods. *to be published*, 1998.
- [11] S. Heiler. A survey on nonparametric time series analysis. Technical report, University of Konstanz - Center of Finance and Econometrics, 1999.
- [12] K. Hornik, M. Stinchcombe, and H. White. Universal approximation of an unknown mapping and its derivatives using feedforward networks. *Neural Networks*, 3:551–560, 1990. [Reprinted in White (1992)].
- [13] T. Jaditz and L. Riddick. Time-series near-neighbor regression. *Studies in Nonlinear Dynamics and Econometrics*, 4(1):35–44, 2000.

- [14] H. Kantz and T. Schreiber. *Nonlinear Time Series Analysis*. Cambridge Univ. Press, Cambridge, UK, 1997.
- [15] M. B. Kennel, R. Brown, and H. D. I. Abarbanel. Determining embedding dimension for phase-space reconstruction using a geometrical construction. *Phys. Rev. A*, 45:3403–3411, 1992.
- [16] D. Kugiumtzis and N. Christophersen. State space reconstruction: Method of delays vs singular spectrum approach. Research report 236, University of Oslo - Dept. of Informatics, 1997.
- [17] A. Lapedes and R. Farber. Nonlinear signal processing using neural networks: Prediction and signal modeling. Technical Report LA-UR-87-2662, Los Alamos National Laboratories, Los Alamos, New Mexico, 1987.
- [18] F. Lisi. Statistical dimension estimation in singular spectrum analysis. *Journal of the Italian Statistical Society*, 5(2):203–209, 1996.
- [19] F. Lisi, O. Nocolis, and M. Sandri. Combining singular-spectrum analysis and neural networks for time series forecasting. *Neural Processing Letters*, 2(4):6–10, 1995.
- [20] J. E. Moody and C. Darken. Learning with localized receptive fields. In D. Touretzky, G. Hinton, and T. Sejnowski, editors, *Proceedings of the 1988 Connectionist Models Summer School*, pages 133–143, San Mateo, 1988. (Pittsburg 1988), Morgan Kaufmann.
- [21] M. R. Muldoon, D. S. Broomhead, J. P. Huke, and R. Hegger. Delay embedding in the presence of dynamical noise. *Dyn. and Stab. Syst.*, 13:175, 1998.
- [22] L. Prechelt. Automatic early stopping using cross validation: Quantifying the criteria. *Neural Networks*, 9(3):457–462, 1996.
- [23] T. Sauer. A noise reduction method for signals from nonlinear systems. *Physica D*, 58:193–201, 1992.
- [24] T. Sauer. Time series prediction using delay coordinate embedding. In A. S. Weigend and N. A. Gershenfeld, editors, *Time Series Prediction: Forecasting the Future and Understanding the Past*. Addison-Wesley, 1993.
- [25] T. Sauer, J. A. Yorke, and M. Casdagli. Embedology. *J. Stat. Phys.*, 65:579–616, 1991.
- [26] J. Stark. Delay reconstruction: Dynamics vs. statistics. In A. Mees, editor, *Nonlinear Dynamics and Statistics*, pages 81–103. Birkhauser, Basel, 2000.
- [27] H. Tong. *Non-linear Time Series, A Dynamical System Approach*. Clarendon Press Oxford, Oxford, UK, 1990.
- [28] R. Vautard, P. Yiou, and M. Ghil. Singular-spectrum analysis: A toolkit for short noisy chaotic signals. *Physica D*, 58:95–126, 1992.
- [29] E. A. Wan. Time series prediction by using a connectionist network with internal delay lines. In A. S. Weigend and N. A. Gershenfeld, editors, *Time series prediction: Forecasting the future and understanding the past*, pages 195–217. Addison Wesley, Reading, MA, 1993.
- [30] A. S. Weigend and N. A. Gershenfeld. *Time Series Prediction: Forecasting the Future and Understanding the Past*. Addison-Wesley, 1993. Proceedings of the NATO Advanced Research Workshop on Comparative Time Series Analysis held in Santa Fe, New Mexico, May 14–17, 1992.
- [31] H. White. *Artificial neural networks. Approximation and learning theory*. Blackwell Publ., Oxford, 1992.

Description

Nearest-neighbors time series prediction with different local models

Syntax

`y = nnpredic(insample, outofsample, k, T, opt1, opt2, nhidden)`

Input

insample: learning set represented by a d -dimensional vector of inputs (generally the output of the `rebuild` command)
outofsample: testing set
k: number of nearest neighbors (this parameter should be chosen by the `choosedk` command)
T: prediction step
opt1: local model type: 'constant', 'linear', 'quadratic', 'neural'
opt2: type of prediction: 'direct', 'iterative' (when $T = 1$ direct and iterative predictions give the same result)
nhidden: number of unit in the hidden layer

Output

y: predicted points in the reconstructed phase space

Example

```
load 'lor.txt' -ascii
x=rebuild(lor(:,1),3,5);
z=x(602:902,:);
insample=x(1:500,:);
outofsample=x(600:900,:);
y = nnpredic(insample,outofsample,30,2,'linear','direct');
plot3(y(:,1),y(:,2),y(:,3),'.',z(:,1),z(:,2),z(:,3))
```

choosedk

Description

Algorithm for the optimal choice of the d and k parameters based on the minimum (cross-validation) nearest-neighbors prediction error. A portion of the reconstructed series (the learning set) is used for estimating the local regression model. On a second portion (the cross-validation set) the algorithm evaluates the prediction error of the model.

Syntax

```
[krange, drange, errors] = choosedk(x, noofpreds, T, tau, drange, krange, opt1, opt2, opt3, nhidden)
```

Input

x: vector or matrix data (state variables must be placed in columns)

noofpreds: number of points in the testing set

T: prediction step

tau: time lag

drange: range of values for the embedding dimension

krange: range of values for the number of nearest neighbors

opt1: local model type: 'constant', 'linear', 'quadratic', 'neural' (default: 'linear')

opt2: type of prediction: 'direct', 'iterative' (when $T = 1$ direct and iterative predictions give the same result) (default: 'direct')

opt3: type of graphic representation: '2dplot', '3dplot', 'no-plot' (default = '2dplot')

Output

errors: matrix of prediction errors for different values of d and k

Example

```
load lor.txt -ascii
[krange, drange, erroridk] =
choosedk(lor(1:500,2), 100, 1, 5, [2:5], [10:10:100], 'lineare', 'diretta', '2dplot')
```

See also

nmse, nnpredic, plotdk, rebuild

rebuild

Description

Method of delay for attractor reconstruction

Syntax

`y = rebuild(x, m, tau)`

Input

x: vector or matrix data (state variables must be placed in columns)
m: embedding dimension
tau: time lag

Output

y: matrix of reconstructed points (each row represents an m -dimensional point)

Example

```
load lor.txt -ascii
x=lor(.1);
y = rebuild(x,5,8)
```

See also

rc

Description

The routine filters the input series according to the method based on SSA and developed by [28]

Syntax

`y = rc(x, m, p, tau)`

Input

x: vector or matrix input data
m: embedding dimension
p: number of principal components
tau: lag time

Output

y: filtered signal

Example

```
load lornoise.txt -ascii
x=lornoise(.1);
n=length(x);
y=rc(x,25,3,1);
plot(1:n,x,1:n,y)
```

See also

`nnpredict`

ssaplot

Description

Using different graphical representations, this command plots the singular values and singular vectors of the (uni or multivariate) input series

Syntax

`h=ssaplot(x, m, tau, eigvectno, opt)`

Input

- x:** vector or matrix input data
- m:** vector of embedding dimensions
- tau:** lag time
- eigvectno:** vector containing the number of singular vectors to be plotted (es. `eigvectno=[1,2,3,4]` plots the first 4 eigenvectors)
- opt:** type of plot required:
- 'spec' plot of the spectrum of singular vectors
 - 'logspec' plot of the logarithmic spectrum of the singular vectors
 - 'diff' plot of the difference between successive singular vectors: $\frac{\lambda_{i+1}-\lambda_i}{\lambda_i}$ $i = 1, 2, \dots, m-1$
 - 'cum' plot of the cumulative percentage of total variance explained by the first i principal vectors
 - 'all' the four above graphics plotted together
 - 'eigvects' plot the four singular vectors specified by `eigvectno`

Output

- h:** the output of this command is the handle of the output figure

Example

```
load lor.txt -ascii
ssaplot(lor(1:500,1),25,5,[1,2,3,4],'all')
ssaplot(lor(1:500,1),25,5,[1,2,3,4],'eigvects')
```

See also

`eigspec`, `rebuild`

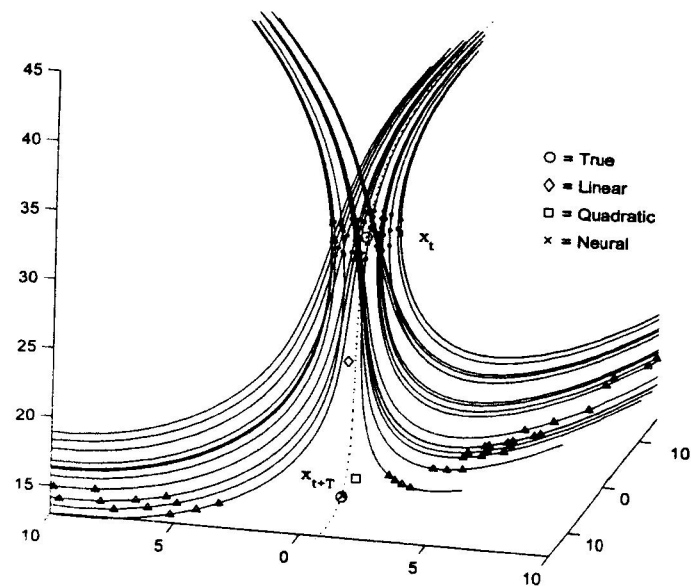


Figure 1: Example of nearest-neighbor prediction with three types of local models: linear, quadratic and neural. Balls and triangles represent the nearest neighbors of x_t and x_{t+T} , respectively.

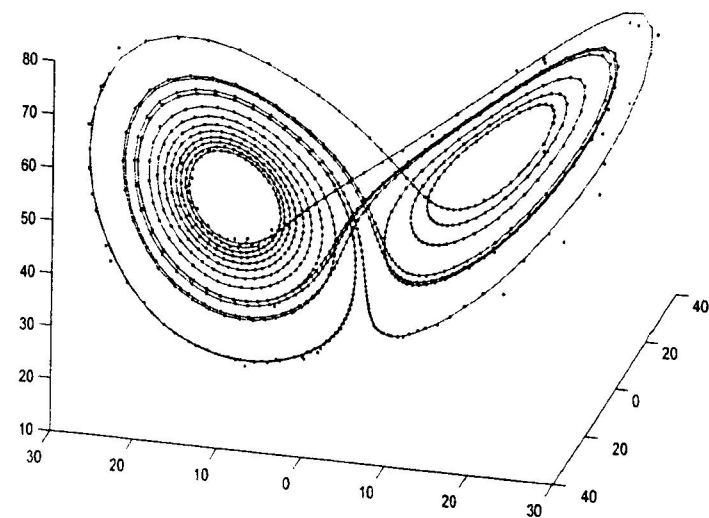


Figure 2: Prediction of a 3-dimensional noise-free Lorenz series with a local neural model. Balls are the predicted values. The solid line is the true trajectory.

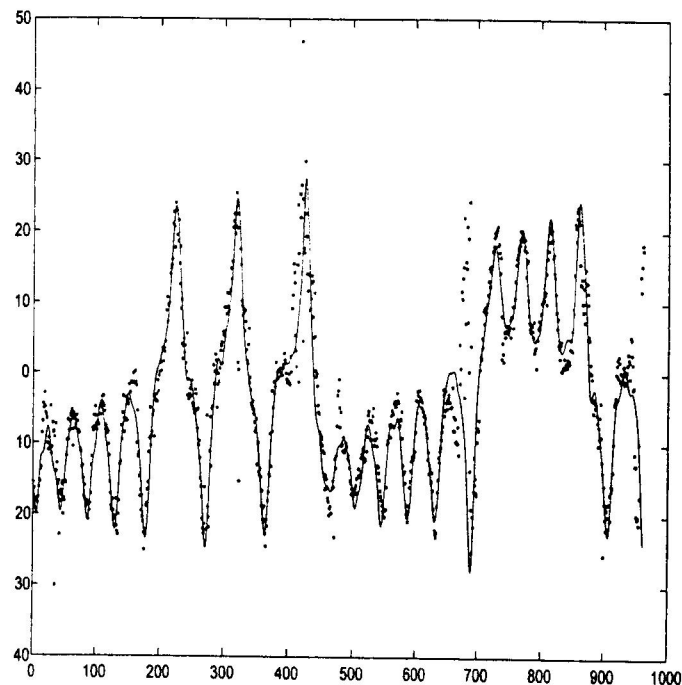


Figure 5: Prediction of a noisy Lorenz series with a local neural model (solid line: true noise-less trajectory, points: predicted values)

DIPARTIMENTO DI SCIENZE ECONOMICHE — WORKING PAPER

- 1 Man and work according to the social doctrine of the Church, particularly in the thought of John Paul II
G. TONDINI (marzo 1998)
- 2 Rational addiction and alcohol consumption: evidence from the nordic countries
J. BENTZEN – T. ERIKSSON – V. SMITH (aprile 1998)
- 3 Considerazioni sulla natura e ruolo della Banca Centrale Europea
P. DONGILI (aprile 1998)
- 4 Scale di equivalenza e misurazione della povertà in Italia
V. ATELLA – A. CAIUMI – F. PERALI (aprile 1998)
- 5 Equity versus efficiency: an application to Colombia between 1985 and 1990
F. PERALI (aprile 1998)
- 6 L'impatto sulla economia locale delle manifestazioni fieristiche veronesi
G. GABURRO – M. POMINI (luglio 1998)
- 7 Imprenditori o lavoratori? L'uso dei contratti di soccida nella filiera delle carni
A.M. ZAGO (settembre 1998)
- 8 The long-run effects of the italian pension reforms
N. SARTOR (settembre 1998)
- 9 Debates and perspectives in social market economy: the german experience
G. GABURRO – V. PAIUSCO (novembre 1998)
- 10 The Barten-Gorman model in the Aids framework: an application to the demand for leisure differentiated by gender in Northcentral U.S.
M.L. FERREIRA – C.F. PERALI (novembre 1998)
- 12 The early endogenous neoclassical growth models: a case study in macroeconomic thought
M. POMINI (dicembre 1998)
- 13 Le strategie della politica monetaria della BCE
P. DONGILI (maggio 1999)
- 14 Introduzione alla teoria della crescita in una prospettiva storica (I parte)
M. POMINI (giugno 1999)
- 15 Introduzione alla teoria della crescita in una prospettiva storica (II parte)
M. POMINI (giugno 1999)
- 16 L'euro nel processo di regionalizzazione dell'emisfero europeo
P. DONGILI (giugno 2000)

- 1 Une analyse interpretative du modele ciclique de Fanno
C. NARDI SPILLER (luglio 1992)
- 2 Evolutionary selection of "Chivalrous" conventions in coordination games without common expectations
P. SACCO – M. SANDRI (luglio 1992)
- 3 Latin America between neoliberal policies and integration: wich way out of the crisis?
M. D. PARRILLI (luglio 1992)
- 4 Le regole e il contratto
A. VILLANI (agosto 1992)
- 5 Aspetti macro e microeconomici del sistema vitivinicolo mantovano negli ultimi dieci anni
R. RUGGERI (settembre 1992)
- 6 Cambiamento tecnologico e crescita economica: nuove acquisizioni e quesiti aperti nella riflessione teorica
G. ANTONELLI (ottobre 1992)
- 7 Le fluttuazioni economiche: punti convergenti e divergenti tra l'apporto di Fanno e il contributo di altri studiosi
C. NARDI SPILLER (dicembre 1992)
- 8 Il trattato di Maastricht
M. SIRONI MARIOTTI – ALFREDO SOLUSTRI (gennaio 1993)
- 9 Il capitale produttivo in Italia
S. MONTRESOR (febbraio 1993)
- 10 Metodi di valutazione quantitativa e predizione della quantità di vini a lungo invecchiamento
G. GABURRO – P. FRONTERO – O. LOVADINA (marzo 1993)
- 11 Alcune considerazioni sul ruolo della riserva obbligatoria
P. DONGILI (maggio 1993)
- 12 Input-Output dynamics as an iterated function system
P. L. SACCO – M. SANDRI (dicembre 1993)
- 13 Alcune note metodologiche sulla matrice di contabilità sociale
G. MARANGONI – S. MONTRESOR (dicembre 1993)
- 14 Ethics and economics Catholic Thinkers of the Twentieth Century
G. GABURRO (febbraio 1994)
- 15 Il messaggio etico di Marco Fanno
C. NARDI SPILLER – MARIA LUISA BERGAMINI (luglio 1994)
- 16 Riflessioni sul ruolo macroeconomico e sui problemi microeconomici del credito al consumo
P. DONGILI (agosto 1994)
- 17 Recenti contributi teorici sull'inflazione. Parte 1ª
V. PEDERZOLI (ottobre 1994)
- 18 Una modellistica in tema di inflazione. Parte 2ª
V. PEDERZOLI (dicembre 1994)
- 19 Inflazione, tassi d'interesse e saggi cambio nell'esperienza della comunità europea
C. NARDI SPILLER (marzo 1995)
- 20 I problemi del coordinamento monetario internazionale
P. DONGILI (aprile 1995)
- 21 Spunti di analisi economica sulla centenaria normativa in tema di lavori pubblici
C. NARDI SPILLER (maggio 1995)
- 22 Aspetti teorici relativi all'ottimalità dell'offerta di moneta
V. PEDERZOLI – M. COSTALUNGA (giugno 1995)
- 23 La politica monetaria in alcune posizioni teoriche
V. PEDERZOLI (luglio 1995)
- 24 Il finanziamento del terzo settore: aspetti e problemi
P. DONGILI (agosto 1995)
- 25 Consumptionin and wages dynamics: an input-output approach
G. MARANGONI (settembre 1995)
- 26 Le tavole input-output regionali in Italia
G. MARANGONI (ottobre 1995)
- 27 Classici, neoclassici, keynesiani e il modello di Leontief
G. MARANGONI (novembre 1995)
- 28 Coefficienti tecnici in termini fisici e in valore in un modello input-output: alcune osservazioni
G. MARANGONI (dicembre 1995)
- 29 Sistemi finanziari e trasformazione delle economie a pianificazione centrale: teoria e pratica
C. FRATESCHI (gennaio 1996)
- 30 Mezzogiorno italiano e problematicità occupazionale
P. BONIFANTI – C. NARDI SPILLER (febbraio 1996)
- 31 Le problematiche inflazionistiche nella U.E.: un riferimento alla passata esperienza
C. NARDI SPILLER (marzo 1996)
- 32 Assorbimento occupazionale e "terzo settore"
C. NARDI SPILLER (aprile 1996)
- 33 Connessioni tra regimi di tassi di cambio e variabili macroeconomiche
V. PEDERZOLI (giugno 1996)
- 34 The structure of demand for alcoholic drinks in the United Kingdom
J. D. MCAVINCHIEY (luglio 1996)
- 35 Debito Pubblico, tassi d'interesse e dinamica inflattiva nell'Unione Europea nel dopo Maastricht
C. NARDI SPILLER (ottobre 1996)
- 36 L'apport novateur controversé de la nouvelle économie keynésienne appliquée au marché des capitaux
C. NARDI SPILLER (novembre 1996)

- 37 I costi dell'Europa
P. DONGILI (dicembre 1996)
- 38 Analisi, interdipendenza e simulazione di un modello di equilibrio generale
F. PERALI (gennaio 1997)
- 39 La nuova legislazione cubana sugli investimenti stranieri
E. RODRIGUEZ (febbraio 1997)
- 40 Income, Schooling, and Ability: Evidence from a new Sample of Twins
O. ASHENFELTER (marzo 1997)
- 41 Justice Principles and Individual Rights in social Choice: how to Use Justice to Solve Gibbard's Paradox
G. PALOMBA (aprile 1997)
- 42 Economia di mercato e valori dell'impresa alla luce dell'enciclica "Centesimus Annus"
G. GABURRO (maggio 1997)
- 43 Competition in international telecommunication tariffs
G. DE FRAJA - P. VALBONESI
- 44 "L'economia è invece più di una collezione di tecniche, tese a investigare il funzionamento del sistema economico. È un modo di guardare il mondo, essendo un caso speciale di una più generale logica dell'azione razionale... Così è, credo, nel caso dell'arte"
C. NARDI SPILLER
- 45 È la fine dell'inflazione nella U.E.?
C. NARDI SPILLER