SVM-based Time Series Prediction with Nonlinear Dynamics Methods

Francesco Camastra[1] and Maurizio Filippone[2]

[1] Dipartimento di Scienze Applicate, Universitá di Napoli Parthenope Via A. De Gasperi 5, 80133 Napoli, Italy e-mail: francesco.camastra@uniparthenope.it
[2] DISI, University of Genova and CNISM, Via Dodecaneso 35, 16146 Genova, Italy, e-mail: filippone@disi.unige.it

Abstract. A key problem in time series prediction using autoregressive models is to fix the *model order*, namely the number of past samples required to model the time series adequately. The estimation of the model order using cross-validation is a long process. In this paper we explore faster alternative to cross-validation, based on nonlinear dynamics methods, namely Grassberger-Procaccia, Kégl and False Nearest Neighbors algorithms. Once the model order is obtained, it is used to carry out the prediction, performed by a SVM. Experiments on three real data time series show that nonlinear dynamics methods have performances very close to the cross-validation ones.

1 Introduction

Time series prediction is the problem of determining the future values of a given time series. This problem has great importance in several fields, ranging from finance (e.g. predicting the future behavior of stock markets) to engineering (e.g. estimating future electrical consumption). A key problem in time series prediction is to fix the *model order*, namely the number of past samples required to model the time series adequately. In principle *cross-validation* [4] [17] is the simplest solution, just picking the model order which gives the lowest prediction error. However the computational cost of cross-validation can be very high and an estimate of the model order is helpful, either to be used directly or to narrow down the range for cross-validation. In this paper we use nonlinear dynamics methods, namely Kégl, Grassberger-Procaccia and False Nearest Neighbors algorithms, to estimate the model order. Then the model order is used to carry out the prediction, performed by a *Support Vector Machine* (*SVM*) [16] [19] [21]. We investigate the effectiveness of nonlinear dynamics methods comparing their performances with the cross-validation ones. The paper structure is as following: in Section 2 a description of the nonlinear dynamics methods investigated is provided; in Section 3 some experimental results are reported; in Section 4 conclusions are drawn.

2 Nonlinear Dynamics Methods

We consider a time series $x(t)$, with $(t = 1, 2, \ldots, N)$. An autoregressive model describes a time series as :
$x(t) = f(x(t-1), \ldots, x(t-(d-2)), x(t-(d-1))) + \epsilon_t$. The function $f(\cdot)$ is called the *skeleton* of the time series [20] [9], the term ϵ_t represents the noise. The key problem in the autoregressive models is to fix the model order *(d-1)*. Nonlinear Dynamics methods can be used for the *model reconstruction* of the time series. This is performed by the method of delays [5] [15]. The time series can be represented as a series of a set of points $\{X(t) : X(t) = [x(t), x(t-1), \ldots, x(t-d+1)]\}$ in a d-dimensional space. If d is adequately large, between the manifold M obtained by the points $X(t)$ and the attractor U of the dynamic system that generated the time series, there is a diffeomorphism[3]. The *Takens-Mañé embedding theorem* [12][18] states that to obtain a faithful reconstruction of the system dynamics, it must be

$$2S + 1 \leq d \tag{1}$$

where S is the dimension of the system attractor U and d is called the *embedding dimension* of the system. Hence it is adequate to measure S to infer the embedding dimension d and the model order $d-1$. A unique definition of the dimension has not been given yet. Popular definitions of set dimensions are the *Box-Counting* Dimension [14] and the *Correlation dimension* [6]. In the next sections we shall discuss two methods to estimate attractor dimension (Grassberger-Procaccia and Kégl) methods and a method to estimate the embedding dimension, without using Takens-Mañé embedding theorem (False Nearest Neighbors methods).

2.1 Kégl algorithm

Let $\Omega = \{x_1, x_2, \ldots, x_N\}$ be a set of points in $\mathbb{R}^n$ of cardinality N. The Box-Counting dimension (or *Kolmogorov capacity*) D_B of the set Ω is defined as follows [14]: if $\nu(r)$ is the number of the boxes of size r needed to cover Ω, then D_B is

$$D_B = \lim_{r \to 0} \frac{\ln(\nu(r))}{\ln(\frac{1}{r})} \tag{2}$$

Recently Kégl [11], has proposed a fast algorithm (*Kégl algorithm*) to estimate the Box-Counting dimension. The algorithm has originally been proposed for intrinsic data dimensionality estimation. In this paper we propose a novel application of Kégl's algorithm, consisting in the dimension estimation of an attractor. Kégl algorithm is based on the observation that $\nu(r)$ is equivalent to the cardinality of maximum independent vertex set $MI(G_r)$ of the graph $G_r(V, E)$ with vertex set $V = \Omega$ and edge set $E = \{(x_i, x_j) | d(x_i, x_j) < r\}$. Kégl has proposed

[3] M is *diffeomorphic* to U iff there is a differentiable map $m : M \mapsto U$ whose inverse m^{-1} exists and is also differentiable.

to estimate MI(G) using the following greedy approximation. Given a data set Ω we start with an empty set $\mathcal{C}$ and in an iteration over Ω we add to $\mathcal{C}$ data points that are at distance of at least r from all elements of $\mathcal{C}$. The cardinality $\mathcal{C}$, after every point in Ω has been visited, is the estimate of $\nu(r)$. The Box-Counting dimension estimate is given by:

$$D_B = -\frac{\ln \nu(r_2) - \ln \nu(r_1)}{\ln r_2 - \ln r_1} \tag{3}$$

where r_2 and r_1 are values that can be set up heuristically.

2.2 Grassberger-Procaccia algorithm

The *Correlation dimension* [6] of a set Ω is defined as follows. If the *correlation integral* $C_m(r)$ is defined as:

$$C_m(r) = \lim_{N\to\infty} \frac{2}{N(N-1)} \sum_{i=1}^{N} \sum_{j=i+1}^{N} I(\|x_j - x_i\| \leq r) \tag{4}$$

where I is an *indicator function*[4], then the Correlation dimension D of Ω is:

$$D = \lim_{r\to 0} \frac{\ln(C_m(r))}{\ln(r)} \tag{5}$$

It can be proved that Correlation Dimension is a lower bound of the Box-Counting Dimension. The most popular method to estimate Correlation dimension is the *Grassberger-Procaccia algorithm* [6]. This method consists in plotting $\ln(C_m(r))$ versus $\ln(r)$. The Correlation dimension is the slope of the linear part of the curve (see Figure 1a). For increasing increasing values of d one can notice a saturation effect. The limit value is the correlation dimension.

2.3 Method of False Nearest Neighbors

Kégl and Grassberger-Procaccia algorithms estimate the attractor dimension and permit to estimate the model order of the time series by means of the Takens-Mañé embedding theorem. An alternative approach is proposed by the *False Nearest Neighbors method* [2] [10]. This method estimates directly the embedding dimension without using the Takens-Mañé theorem. False Nearest Neighbors method is based on a simple geometric concept. If the dimension d used to reconstruct the attractor is too small, many points that appear *near* will become widely separated when $d + 1$ dimensions are used in the attractor reconstruction. Nearest neighbor points that show this wide separation when comparing their distance in dimension d and $d + 1$ are called *False Nearest Neighbors* in dimension d. Conversely, true nearest neighbors will remain near each other in attractor reconstructions of both d and $d + 1$ dimensions. More

[4] $I(\lambda)$ is 1 iff condition λ holds, 0 otherwise.

formally a pair of points are considered False Nearest Neighbors in dimension d if $\frac{R_{d+1}^2(j)}{R_d^2(j)} > \alpha$ where $R_d(j)$ $R_{d+1}(j)$ are respectively the Euclidean distance between the jth point and its nearest neighbors in d and $d+1$ dimensions and α is an heuristic threshold. Typical values for α are suggested in [2]. The adequacy of dimension d for reconstructing an attractor can be evaluated by calculating for each data point of the attractor the nearest neighbors in dimension d and then evaluating the percentage of False Nearest Neighbors. Then the percentage of False Nearest Neighbors is plotted versus the dimension d. The lowest dimension corresponding to this minimum value of the percentage of False Nearest Neighbors is the embedding dimension.

3 Experimental Results

False Nearest Neighbors, Grassberger-Procaccia and Kégl algorithms have been tried on three benchmarks of real data, e.g. the *data set A* [7] of the Santa Fe time series competition; the *Paris-14E Parc Montsouris*[5][22] time series and the *DSVC1* [6] [1] [22] time series.

3.1 Data Set A

Data Set A is a real data time series, formed by 1000 samples, generated by a Lorenz-like chaotic system, implemented by NH_3-FIR lasers. Firstly the model order of the time series has been estimated by means of False Nearest Neighbors, Grassberger-Procaccia and Kégl algorithms. The estimates of the attractor dimension using Grassberger-Procaccia and Kégl algorithms are respectively 2.00 and 2.02. Since the attractor dimension of data set A is 2.06, the estimates of both algorithms can be considered quite satisfactory. Applying the equation (1) of the Takens-Mañé theorem we see that the embedding dimension estimate, provided by Grassberger-Procaccia and Kégl algorithms, is ~ 5. Hence the model order is 4. Then we have estimated the model order using False Nearest Neighbors method. As shown in Figure 1b, the percentage of False Nearest Neighbors is negligible for an embedding dimension value of 3. Hence the model order estimated by False Nearest Neighbors is 2. Then the model order, estimated by three different algorithms, has been used to carry out *one-step ahead prediction*, i.e the prediction of the next value of the time series. The former half of time series has been used for the training set, while the latter one has been used for the validation and test set, respectively formed by 200 and 300 samples. The prediction stage has been performed using *SVM-Light* [8], an implementation of SVM for Regression [13] [16]. In our experiments we have used the gaussian kernel and the kernel variance has been set up using cross-validation. Finally, as a comparison we have set up the model order by means of the cross-validation.

[5] The time series can be downloaded from www.knmi.nl/samenw/eca

[6] The time series can be downloaded from www.cpdee.ufmg.br/~MACSIN/services/data/data.htm

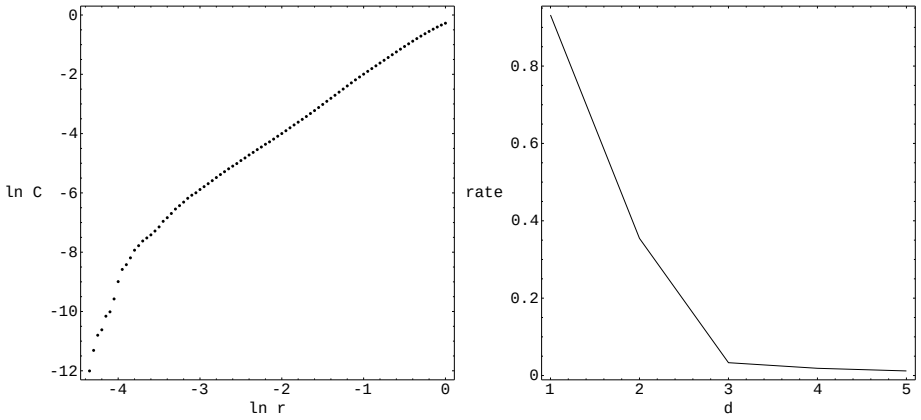

Fig. 1. Grassberger-Procaccia (a) and False Nearest Neighbors (b) algorithms on Data Set A.

Algorithm	Attractor Dimension	Embedding Dimension	Model Order	Quadratic Loss
False Nearest Neighbors		3	2	0.75
Kégl	2.02	∼5	4	0.65
Grassberger-Procaccia	2.00	∼5	4	0.65
cross-validation			4	0.65

Table 1. False Nearest Neighbors, Kégl, Grassberger-Procaccia and cross-validation method on Data Set A. Quadratic Loss has been measured on normalized data.

The results, expressed in terms of quadratic loss [16], are reported in the table 1.

3.2 Paris-14E Parc Montsouris

Paris-14E Parc Montsouris is a real data time series formed by the daily average temperatures, expressed in tenths of Celsius degrees, in Paris. The time series covers the whole period from January 1 1958 to December 31 2001 and has 15706 samples. The former half of time series (7853 samples) has been used for the training set, while the latter one has been used for the validation and test set, respectively formed by 2190 and 5663 samples. We have estimated the model order using False Nearest Neighbors, Grassberger-Procaccia and Kégl algorithms and we have performed the prediction stage using SVM-Light. Even in this case, we have used the gaussian kernel, setting the variance using cross-validation. As a comparison we have also estimated the model order by means of the cross-validation. The results on the test set, expressed in terms of quadratic loss, are reported in the table 2.

Algorithm	Attractor Dimension	Embedding Dimension	Model Order	Quadratic Loss
False Nearest Neighbors		5	4	14.55
Kégl	4.03	∼9	8	14.43
Grassberger-Procaccia	4.91	∼11	10	14.25
cross-validation			10	14.25

Table 2. False Nearest Neighbors, Kégl, Grassberger-Procaccia and cross-validation method on the Data Set Paris-14E Parc Montsouris. Quadratic Loss has been measured on normalized data.

3.3 DSVC1

DSVC1 is a real data time series, formed by 5000 samples, measured from a hardware realization of Chua's circuit [3]. The former half of time series (2500 samples) has been used for the training set, while the latter one has been used for the validation and test set, respectively of 750 and 1750 samples. The model order was estimated using the three methods (e.g. False Nearest Neighbors, Grassberger-Procaccia and Kégl) and the prediction stage was performed using SVM-Light. Even in this case, we have used the gaussian kernel, setting the variance using cross-validation. The estimates of the attractor dimension using Grassberger-Procaccia and Kégl algorithms are respectively 2.20 and 2.14. Since the attractor dimension of data set A is ~ 2.26, the estimates of both algorithms can be considered satisfactory. As a comparison the model order was also estimated by means cross-validation. The results expressed on the test set, in terms of quadratic loss, are reported in the table 3.

Algorithm	Attractor Dimension	Embedding Dimension	Model Order	Quadratic Loss
False Nearest Neighbors		6	5	2.71
Kégl	2.14	$5 \div 6$	$4 \div 5$	$3.72 \div 2.71$
Grassberger-Procaccia	2.20	$5 \div 6$	$4 \div 5$	$3.72 \div 2.71$
cross-validation			5	2.71

Table 3. False Nearest Neighbors, Kégl, Grassberger-Procaccia and cross-validation method on Chua Time Series. Quadratic Loss has been measured on normalized data. The model order estimated by Grassberger-Procaccia and Kegl is between 4 and 5, hence the quadratic loss is between 3.72 (model order = 4) and 2.71 (model order=5).

4 Conclusion

A key problem in time series prediction is to fix the model order, namely the number of past samples required to model the time series adequately. In this paper

we have investigated three nonlinear dynamics methods, False Nearest Neighbors, Grassberger-Procaccia and Kégl algorithms to estimate the model order of a time series. Then the model order has been used to carry out the prediction, performed by a SVM. The experiments on three real data time series have shown that the model order estimated by nonlinear dynamics methods is quite close to the one estimated using cross-validation. Even if the cross-validation remains the simplest way to set up the model order of a time series, nonlinear dynamics methods can be useful. They can be effectively used to narrow down the range for cross-validation, speeding up the crossvalidating process.

References

1. L.A. Aguirre, G.G. Rodrigues, E.M. Mendes, "Nonlinear identification and cluster analysis of chaotic attractors from a real implementation of Chua's circuit", *International Journal of Bifurcation and Chaos* 6(7), 1411-1423, 1997.
2. H.D.I. Arbabanel, *Analysis of Observed Chaotic Data*, Springer-Verlag, 1996.
3. L.O. Chua, M. Komuro, T. Matsumoto, "The double scroll", *IEEE Transactions on Circuits and Systems*, 32(8), 797-818, 1985.
4. R.O. Duda, P.E. Hart and D.G. Stork, *Pattern Classification*, Wiley Classification, John Wiley and Sons, New York, 2000.
5. J.P. Eckmann and D. Ruelle, "Ergodic Theory of Chaos and Strange Attractors", *Review of Modern Physics*, 57, 617-659, 1985.
6. P. Grassberger and I. Procaccia, "Measuring the Strangeness of Strange Attractors", *Physica*, D9, 189-208, 1983.
7. U. Hübner, C.O. Weiss, N.B. Abraham and D. Tang, "Lorenz-Like Chaos in NH_3-FIR Lasers", *Time Series Prediction. Forecasting the Future and Understanding the Past*, 73-104, Addison Wesley, 1994.
8. T. Joachim, "Making large-Scale SVM Learning Practical", Advances in Kernel Methods - Support Vector Learning, MIT Press, 1999.
9. H. Kantz and T. Schreiber, "Nonlinear time series analysis", Cambridge University Press, 1997.
10. M.B. Kennel, R. Brown and H.D.I. Arbabanel, "Determining Embedding Dimension for Phase-Space Reconstruction using a Geometrical Construction", Physical Review A, 45(6), 3403-3411, 1992.
11. B. Kégl, "Intrinsic Dimension Estimation Using Packing Numbers", *Advances in Neural Information Processing 15*, MIT Press, 2003.
12. R. Mañé, "On the dimension of compact invariant sets of certain nonlinear maps", *Dynamical Systems and Turbolence, Warwick 1980*, Lecture Notes in Mathematics no. 898, 230-242, Springer-Verlag, 1981.
13. K-R. Müller, G. Rätsch, J. Kohlmorgen, A. Smola, B. Schölkopf, V. Vapnik, "Time Series Prediction using support vector regression and neural network", in T. Higuchi and Y. Takizawa (eds.) *Proceedings of Second International Symposium on Frontiers of Time Series Modelling: Nonparametric Approach to Knowledge Discovery*, Institute of mathematical statistic publication, 2000.
14. E. Ott, *Chaos in Dynamical Systems*, Cambridge University Press, 1993.
15. N. Packard, J. Crutchfield, J. Farmer and R. Shaw, "Geometry from a time series", *Physical Review Letters*, 45(1), 712-716, 1980.
16. B. Schölkopf and A. Smola, *Learning with Kernels*, MIT Press, 2002.

17. M. Stone, "Cross-validatory choice and assessment of statistical prediction", Journal of the Royal Statistical Society, 36(1), 111-147, 1974.
18. F. Takens, "Detecting strange attractor in turbolence", *Dynamical Systems and Turbolence, Warwick 1980*, Lecture Notes in Mathematics no. 898, 366-381, Springer-Verlag, 1981.
19. J. Shawe-Taylor, N. Cristianini, *Kernel Methods for Pattern Analysis*, Cambridge University Press, 2004.
20. H. Tong, *Nonlinear Time Series*, Oxford University Press, 1990.
21. V.N. Vapnik. *Statistical Learning Theory*, John Wiley and Sons, New York, 1998.
22. J.B. Wijngaard , A.M.G. Klein Tank and G.P. Konnen, "Homogeneity of 20th century European daily temperature and precipitation series", International Journal of Climatology, 23, 679-692, 2003.