Adaptive Memory-Based Regression Methods

Hugues Bersini, Mauro Birattari, and Gianluca Bontempi*

Abstract

Memory-based methods obtain accurate predictions from empirical data without explicitly modeling the underlying process. For each query, a local model is first tailored on the query itself, then used to perform the prediction, and finally discarded. In this paper, we consider local models which are linear in the parameters. This allows us to adopt, on a local scale, powerful and well-understood methods from classical linear statistics. In particular, we discuss here a fast cross-validation procedure that can be effectively used, on a query-by-query basis, to select the features, the neighbors and the polynomial degree of the local approximator. Experimental results in the time-series prediction domain are presented.

1 Introduction

In learning theory, the problem of function estimation [1] is formulated as the minimization of a global cost function J, the *risk functional* [2], which measures the discrepancy over the whole input space between the target function and the approximator $f(\mathbf{x},\alpha)$. The cost function J has the following form:

$$J(\alpha) = \int L\big(y, f(\mathbf{x},\alpha)\big)\, p(\mathbf{x},y)\,\mathrm{d}\mathbf{x}\,\mathrm{d}y, \tag{1}$$

where $L\big(y, f(\mathbf{x},\alpha)\big)$ is the loss function in a point $\mathbf{x}$ of the input space. It is well known that if the loss function is the squared error

$$L\big(y, f(\mathbf{x},\alpha)\big) = \big(y - f(\mathbf{x},\alpha)\big)^2, \tag{2}$$

*The authors are with IRIDIA, Université Libre de Bruxelles, Brussels, Belgium. The work of M. Birattari was supported by the FIRST program of the Région Wallonne, Belgium. The work of G. Bontempi was supported by the European Union TMR Grant FMBICT960692.

the function which minimizes the risk functional (1) is the *regression function*:

$$y(\mathbf{x}) = \int y\, p(y|\mathbf{x})\, \mathrm{d}y = E[y|\mathbf{x}]. \tag{3}$$

Since only a finite amount of observation is available, the risk functional cannot be analytically evaluated and has to be replaced by an approximation, the *empirical risk functional*, calculated on the basis of the training set.

In the neural network approach, the problem of learning an input-output mapping is seen as a problem of function estimation, and is thus reduced to the problem of choosing from a given set of parametric functions $f(\mathbf{x}, \alpha)$ with $\alpha \in \Lambda$, the one which best approximates the unknown data distribution. The problem of predicting the value that the unknown function will assume in a point $\mathbf{q}$, is solved in two steps: first an approximating function is estimated using the data, and then the prediction is computed evaluating the estimated function for $\mathbf{x} = \mathbf{q}$. In this scheme, the relatively simple problem of estimating the value of the function in one point, is solved by first solving a much more difficult intermediate problem of function estimation.

Memory-based methods [3, 4, 5, 6] reformulate the learning problem in order to avoid the minimization of the global risk functional (1). When a memory-based approach is adopted, the problem of learning an unknown mapping is reduced to a collection of simpler problems: the estimation of the value assumed in specific query points by the target function.

In this paper we distinguish between the classical *non-adaptive* memory-based approach, and the adaptive memory-based approach. In the classical *non-adaptive* memory-based approach the structure of the local approximator is manually tuned by the data analyst according to some heuristic, and is then kept unchanged for all the queries. We propose here an adaptive memory-based method (AMB) that extends the classical approach with an automatic tuning, performed on a query-by-query basis, of the relevant neighbors, of the features [7], and of the degree of the local polynomial approximator. The method we introduce belongs to the class of *lazy learning* methods [8] which groups all the methods which defer the learning procedure until a specific query needs to be answered.

In this paper we propose an application of AMB to the problem of time-series prediction which has been already the focus of numerous studies in the classic memory-based literature [9, 10, 11]. In particular, we present some experimental results obtained in the prediction of the Mackey-Glass chaotic time-series.

2 The adaptive memory-based paradigm

In a function estimation approach, the dominant criterion is the global performance of the resulting approximator over the whole input space: what is required to the model is a good performance on average. This might have some drawbacks. Let us consider for example an input-output mapping where the distribution of the input is not uniform. The definition of the learning problem as a risk minimization assumes that those areas of the input space where the density $p(\mathbf{x})$ is higher, deserve more attention. The risk functional, in fact, weights each prediction error $L\big(y, f(\mathbf{x}, \alpha)\big)$ according to the density value $p(\mathbf{x})$. As a consequence, the minimization procedure is biased towards approximators which perform better on the regions where $p(\mathbf{x})$ is higher.

On the contrary, in the memory-based approach, the estimation of the value of the unknown function is obtained giving the whole attention to the region surrounding the point where the estimation is required. The classical *non-adaptive* memory-based procedure consists essentially of these steps:

- For each query point $\mathbf{q}$, a set of neighbors is selected, each weighted according to some relevance criterion which is typically a function decreasing with the distance from the query point.

- Through a locally weighted regression, a local approximation f of the regression function is chosen from a restricted family of parametric functions.

- The prediction $f(\mathbf{q})$ is obtained by evaluating the local approximator in the query point.

In the classical memory-based framework, the data analyst who adopts a local regression approach, has to tune manually a set of structural parameters of the local models: the number of neighbors, the weight function, the parametric family, and the fitting criterion to estimate the parameters.

We extend the classical approach with a method that automatically selects the correct configuration for each query. The key element of our adaptive memory-based method, is the PRESS statistic [12] which is a simple and economical way to perform leave-one-out cross-validation [13] and therefore to assess the generalization performance of local linear models. For each query point, different model configurations are considered and to each of them the PRESS statistic assigns a quantitative index of performance. According to this index the best model is selected and used to perform the prediction. This same selection strategy is indeed exploited to select a sub set of neighbors to be used as local training set, as well as various structural aspects like the feature sub-set, and the degree of the polynomial used as a local approximator.

In this paper we present a version of the AMB algorithm for time-series prediction. A time-series is a sequence of measurements φ^t of an observable φ at equal time intervals. The Takens theorem [14] implies that for a wide class of deterministic systems, there exists a *diffeomorphism* (one-to-one differential mapping) between a finite window of the time-series $\{\varphi^{t-1}, \varphi^{t-2}, \ldots, \varphi^{t-m}\}$ (*lag vector*), and the state of the underlying dynamic system. This means that in theory it exists a multi-input single-output mapping $F : R^m \rightarrow R$ so that:

$$\varphi^{t+1} = F(\varphi^t, \varphi^{t-1}, \ldots, \varphi^{t-m+1}) \tag{4}$$

where m is the *order* of the time-series. As a consequence, a future value can be predicted by solving a regression problem where the regressors are time delayed observations.

3 The AMB algorithm

The general ideas of the proposed adaptive memory-based approach can be summarized in the following way:

1. The task of learning an input-output mapping is decomposed into a series of linear estimation problems;
2. Each single prediction involves a search in the space of alternative model configurations;
3. The estimation ability of each alternative model is measured by the cross-validation performance computed using the PRESS statistic.

In the version of the algorithm adapted for time-series problems, for each time step to be predicted, the structural parameter of the local model that are automatically tuned are the degree of the local approximator, the number of neighbors, and the order, as defined in eq. 4. In this paper, the space of the structural parameters is searched in an exhaustive way through three nested for-loops. In the outer one the algorithm loops over a range of different lag vector length, in the middle one over a range of nearest neighbors, and in the inner one over a range of polynomial degrees.

For each of the explored triples $< m, k, d >$, where m is the order, k the number of neighbors, and d the polynomial degree, a local model is obtained as the solution of a weighted least squares problem:

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}'\mathbf{W}'\mathbf{W}\mathbf{X})^{-1}\mathbf{X}'\mathbf{W}'\mathbf{W}\mathbf{y} = (\mathbf{Z}'\mathbf{Z})^{-1}\mathbf{Z}'\mathbf{v} = \mathbf{P}\mathbf{Z}'\mathbf{v}, \tag{5}$$

where $\mathbf{W}$ is a diagonal matrix that weights the observation according to their distance from the query points giving a non-zero weights only to the k nearest neighbors, $\mathbf{y}$ is the vector of observed values, and $\mathbf{X}$ is the input matrix. Moreover, $\mathbf{Z} = \mathbf{WX}$, $\mathbf{v} = \mathbf{Wy}$, and the matrix $\mathbf{X'W'WX} = \mathbf{Z'Z}$ is assumed to be non-singular so that its inverse $\mathbf{P} = (\mathbf{Z'Z})^{-1}$ is defined. According to the degree of the local approximator, the matrix $\mathbf{X}$ assumes different forms. If $d = 0$, $\mathbf{X}$ is a vector whose elements are all equal to the constant 1. If $d = 1$, the i^{th} row of the matrix $\mathbf{X}$ is the i^{th} observed lag vector to which a constant 1 has been appended in order to include a constant term in the regression. For polynomials of generic degree d, the i^{th} row contains the constant 1, the components of the i^{th} lag vector together with their powers up to the d^{th}, and the appropriate cross-terms.

The local approximation $\hat{\boldsymbol{\beta}}$ is then assessed in a leave-one-out cross-validation procedure. This step is carried out through the PRESS statistic whose formulation for the case at hand is the following:

$$e_j^{cv} = y_j - \mathbf{x}_j'\hat{\boldsymbol{\beta}}_{-j} = \frac{y_j - \mathbf{x}_j'\mathbf{PZ'v}}{1 - \mathbf{z}_j'\mathbf{Pz}_j} = \frac{y_j - \mathbf{x}_j'\hat{\boldsymbol{\beta}}}{1 - h_{jj}}. \quad (6)$$

The scalar $\mathbf{z}_j'$ is the j^{th} row of $\mathbf{Z}$ and therefore $\mathbf{z}_j = w_{jj}\mathbf{x}_j$, where w_{jj} is the j^{th} diagonal element of the weight matrix $\mathbf{W}$. The term h_{jj} is the j^{th} diagonal element of the *Hat matrix* $\mathbf{H} = \mathbf{ZPZ'} = \mathbf{Z}(\mathbf{Z'Z})^{-1}\mathbf{Z'}$.

Equation 6 shows how the leave-one-out errors e_j^{cv} are efficiently calculated without the need of an explicit re-training of the local model: the matrix $(\mathbf{Z'Z})^{-1}$ does not need to be re-calculated for each example j and, moreover, it is obtained as a by-product of the local model identification (5). On the basis of these errors e_j^{cv}, different statistics can be evaluated to estimate the generalization properties of the local model. In particular, we use here the *mean squared error* to compare the models obtained with different values of the parameters $< m, k, d >$ and to select the one which will be used to forecast the future value of the time-series. A pseudo-code fragment describing the AMB algorithm is proposed in fig. 1.

The analysis of efficient implementations of memory-based methods is outside the scope of this paper. Anyway, it is worth noting here that, as far as the proposed implementation is concerned, the optimization of a structural parameter is obtained at the cost of a wrapping for-loop as it is shown in fig. 1. Thanks to the adoption of fast linear identification and validation procedure in the core of the loops, the computational cost of the local structural optimization is not dramatically higher than the cost of retrieving the neighbors of each query point which is the major computational bottleneck shared by all the nearest-neighbor-*like* methods. More detailed descriptions of efficient

```
best_Press := Inf;
for $m$ := min_$m$ to max_$m$
    for $k$ := min_$k$ to max_$k$
        for $d$ := 0 to max_$d$
            if Press($m$,$k$,$d$) < best_Press
                best_Press := Press($m$,$k$,$d$);
                best_$m$ := $m$;
                best_$k$ := $k$;
                best_$d$ := $d$;
            end
        end
    end
end
Prediction := ValueEstimation($\mathbf{q}$,best_$m$,best_$k$,best_$d$);
```

Figure 1: The AMB algorithm. In this fragment, m is the length of the *lag vector*, k is the number of neighbors, and d is the degree of the local polynomial approximator.

implementations of memory-based methods can be found in Moore *et al.* [15] and in Birattari *et al.* [16]. As far as the problem of retrieving relevant data is concerned, further references can be found in the comprehensive tutorial on local learning by Atkeson *et al.* [4].

4 Experiments

The adaptive memory-based approach has been tested on the prediction of the chaotic Mackey-Glass time-series, a well-known benchmark in time-series prediction (fig. 2). We used a training set of 500 points and a test set with an equal number of samples according to the benchmark definition[1].

4.1 Adaptive memory-based method without adaptation of the number of regressors

In the first experiment we consider only the adaptive selection of the number of neighbors in the range $[3, 80]$, and the degree of the local model in the range $[0, 3]$. As required by the Mackey-Glass benchmark, we predicted the

[1] `http://www.boltz.cs.cmu.edu/`

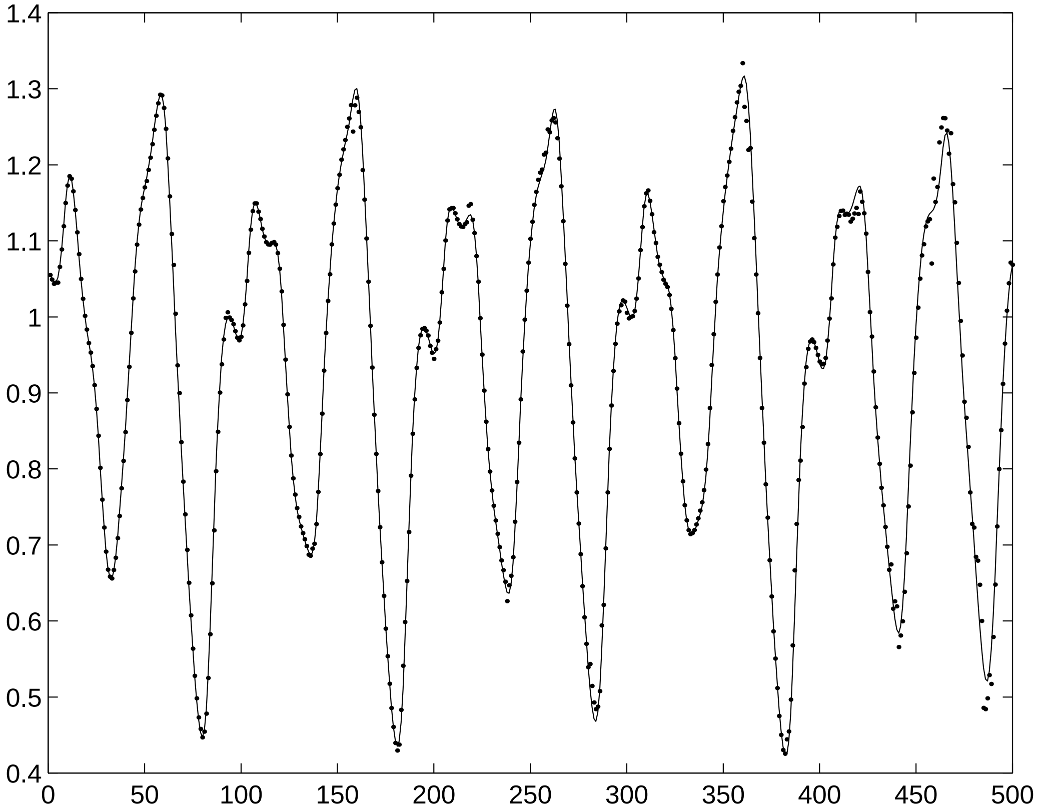

Figure 2: Mackey-Glass time-series and AMB prediction (dotted line).

value of the series at time $t + 85$ from inputs at time t, $t - 6$, $t - 12$ and $t - 18$. We achieved a *normalized mean squared error* (NRMSE) equal to 0.059. One referential result obtained with the RAN approach is NRMSE = 0.075 [17]. In fig. 3 we present the prediction on a time window of 100 samples (diagram a), and the relative prediction squared error (diagram b). Moreover, for each of the predicted time step we report in diagram (c) the optimal polynomial degree and in diagram (d) the number of neighbors taken in consideration in our iterative selection procedure. It is worth noting that the output of the method is not simply a good prediction but a more complete information about the local behavior of the dynamical system underlying the time-series. As an example, in fig. 4 we propose a visual representation of the relation existing between the squared error estimated in cross-validation by the PRESS statistic and the real squared error for the prediction of one time-step. Each point in the figure represents a different model analyzed in the model search procedure. The points roughly distribute along the diagonal,

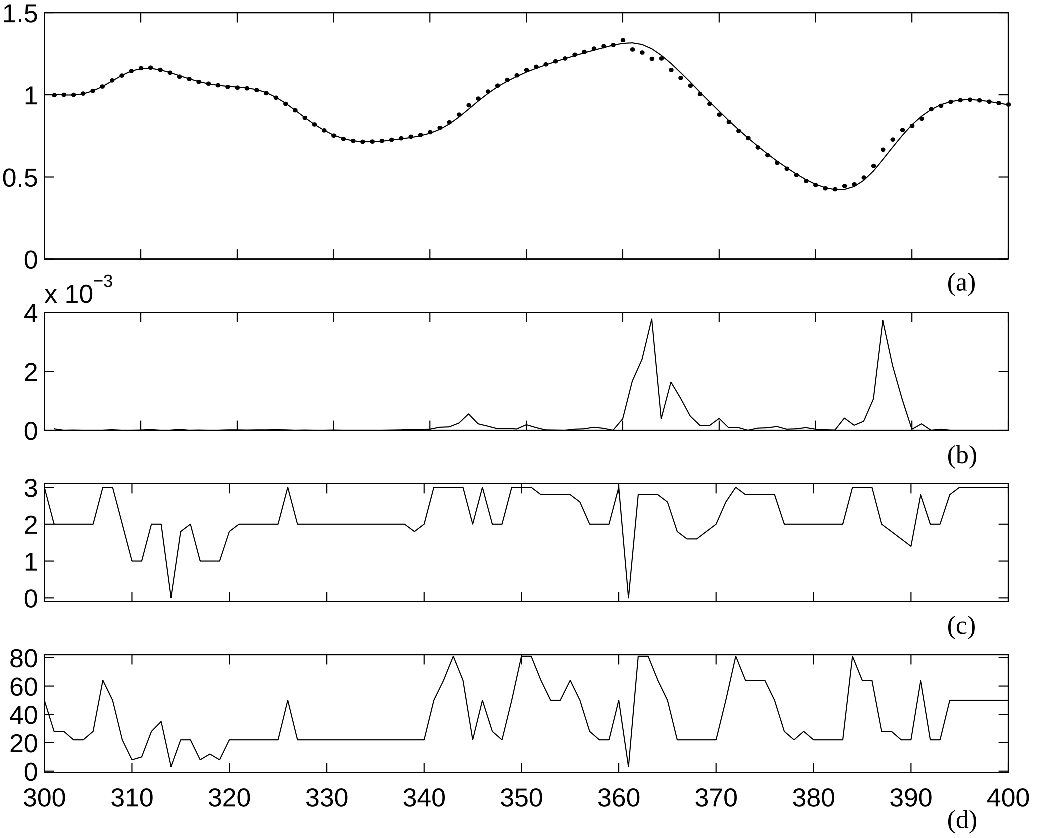

Figure 3: Mackey-Glass time-series and AMB prediction (dotted line) on a 100 samples time window (a); squared error (b); polynomial degree (c) and number of neighbors (d).

showing that the PRESS error is a satisfactory predictor of the actual error. We denote with a cross, approximately in $(0, 0)$, the model chosen by AMB and with a circle, approximately in $(0.02, 0)$, the optimal model. In the proposed example, the difference between the optimal model *a posteriori* and the selected model, in terms of real error, is not even perceivable from fig. 4.

4.2 Adaptive memory-based method with adaptation of the number of regressors

Our second experiment concerns the predictions of the same chaotic time-series using a AMB models which at each time step automatically selects the number of regressors that yields the most accurate prediction. In fig. 5 we report a comparison between the squared error obtained with a fixed number of regressors and the error obtained with the adaptive time step selection. In this case we limited the choice between 3 and 4 regressors which means, for each query, to select between the lag vector $[t, t-6, t-12]$ and $[t, t-6, t-12, t-18]$. We improved the previous result by achieving a NRMSE equal to 0.054. In fig. 6 we report the prediction on the same time window of fig. 3, and the squared error (diagram b). In diagram (c) we plot the number of regressors taken into consideration for each single prediction.

5 Conclusions

The adaptive memory-based approach is a powerful framework that allows the local adoption of well-understood linear methods in a globally nonlinear setting.

Pushing the idea of locality to the extreme, the adaptive memory-based method reduces the problem of learning an input-output mapping to a collection of simpler local estimation problems. Each of these problems is solved independently through a local linear regression. In particular, the proposed experiments show that the accuracy of the one-step-ahead prediction of a time-series can be improved by an automatic tuning, performed on a query-by-query basis, of some structural parameters of the local approximator such as its polynomial degree, the number of neighbors, and the features sub-set. This same methodology have indeed been successfully applied also to control problems [18], and to problems of multiple-step-ahead time-series forecasting through iterated one-step-ahead predictions [19].

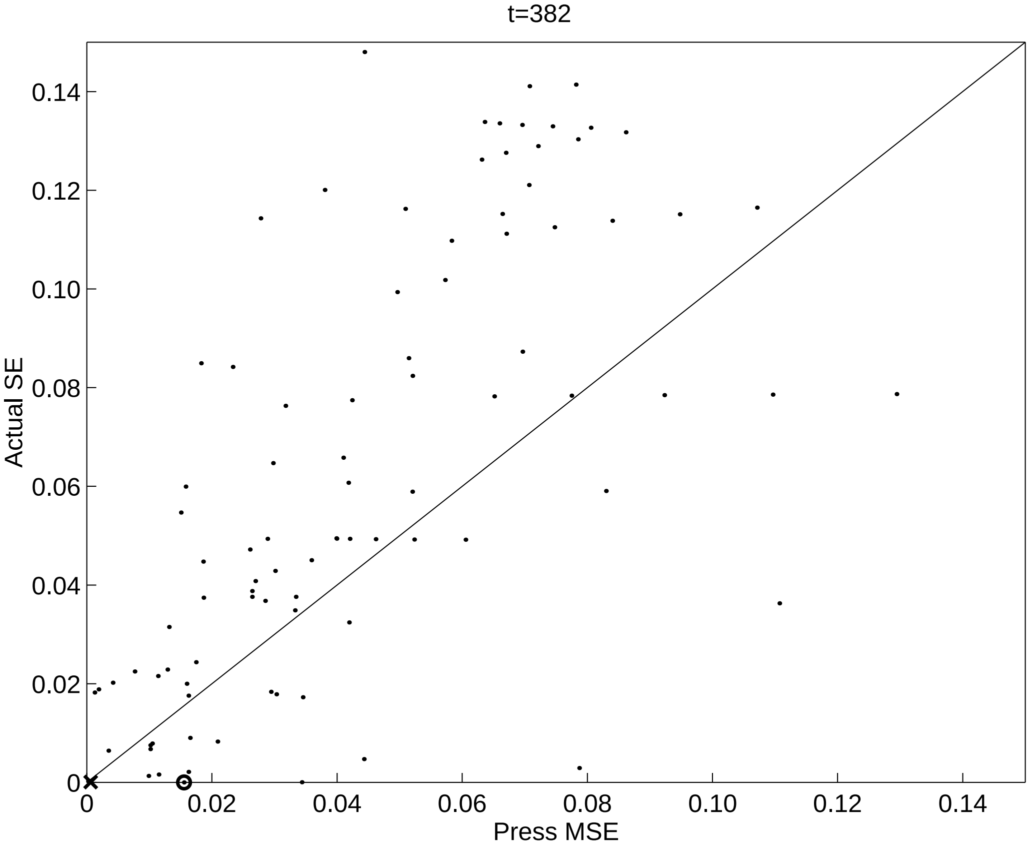

Figure 4: As an example, we propose a visual representation of the relation existing between the PRESS estimation of the squared error and the *a posteriori* squared error of the models considered for the prediction of one time-step, the 382^{nd}.

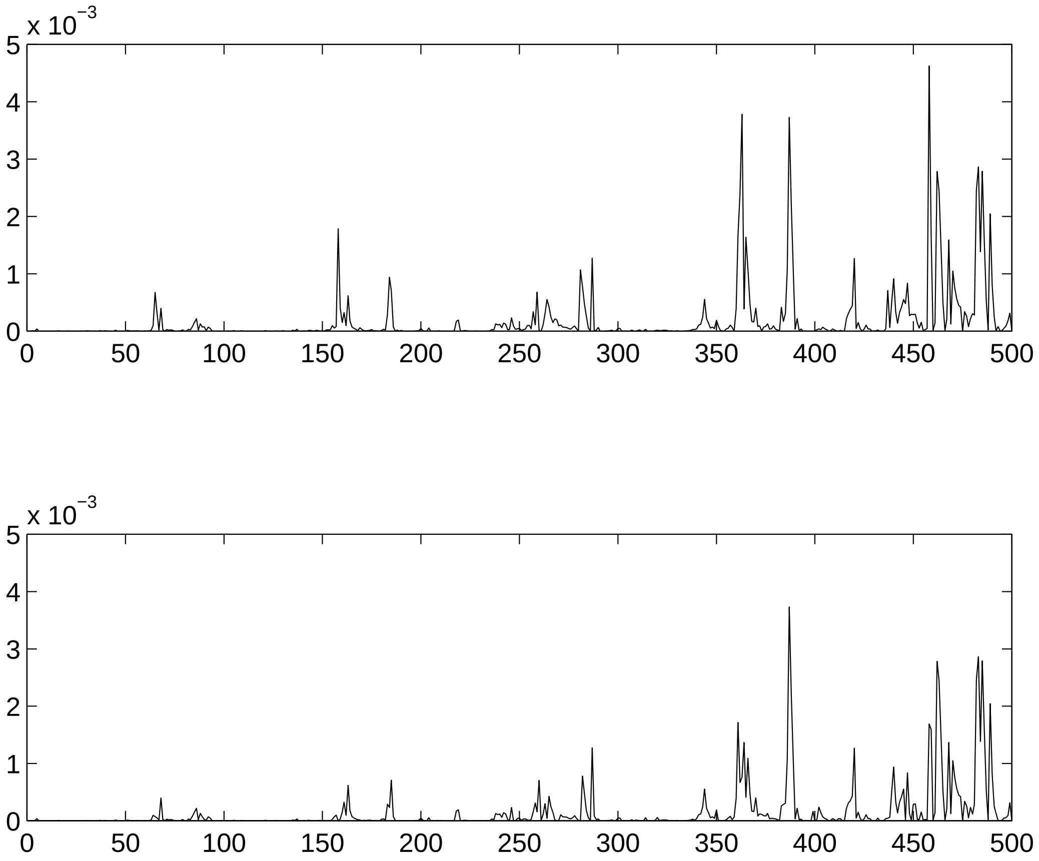

Figure 5: Squared error with a fixed number of regressors (above) and with the automatic selection procedure (below).

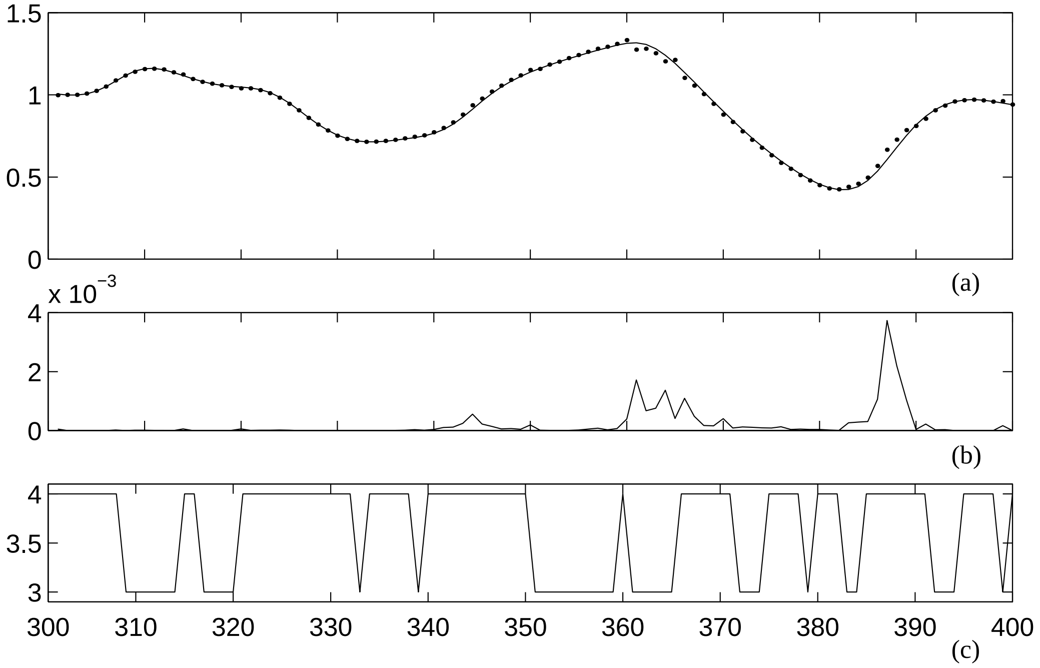

Figure 6: Mackey-Glass time-series and AMB prediction with regressors selection (dotted line) on a 100 samples time window (a); squared error (b); number of regressors (c).

References

[1] C.M. Bishop, *Neural Networks for Statistical Pattern Recognition*, Oxford University Press, Oxford, UK, 1994.

[2] V.N. Vapnik, "Principles of risk minimization for learning theory", in *Advances in Neural Information Processing Systems 4*, pp. 831–838, Morgan Kaufmann Publishers, San Mateo, CA, 1992.

[3] C.G. Atkeson, "Memory-based approaches to approximating continuous functions", in *Nonlinear Modeling and Forecasting*, M. Casdagli and S. Eubank, Eds., pp. 503–521. Addison Wesley, Harlow, UK, 1992.

[4] C.G. Atkeson, A.W. Moore, and S. Schaal, "Locally weighted learning", *Artificial Intelligence Review*, vol. 11, no. 1–5, pp. 11–73, 1997.

[5] W.S. Cleveland, "Robust locally weighted regression and smoothing scatterplots", *Journal of the American Statistical Association*, vol. 74, pp. 829–836, 1979.

[6] T. Hastie and R. Tibshirani, "Discriminant adaptive nearest neighbor classification and regression", in *Advances in Neural Information Processing Systems 8*, pp. 409–415, MIT Press, Cambridge, MA, 1996.

[7] R. Kohavi and G.H. John, "Wrappers for feature subset selection", *Artificial Intelligence journal*, Special issue on relevance, vol. 97, no 1-2, pp. 273–324, 1997

[8] D.W. Aha, "Editorial", *Artificial Intelligence Review*, vol. 11, no. 1–5, pp. 1–6, 1997.

[9] J.D. Farmer and J.J. Sidorowich, "Predicting chaotic time series", *Physical Review Letters*, vol. 8, no. 59, pp. 845–848, 1987.

[10] S. Yakowitz, "Nearest-neighbour methods for time series analysis", *Journal of Time Series Analysis*, vol. 8, no. 2, pp. 235–247, 1987.

[11] M. Casdagli, "Chaos and deterministic versus stochastic non-linear modelling", *Journal of the Royal Statistical Society*, vol. 55, no. 2, pp. 303–328, 1991.

[12] R.H. Myers, *Classical and Modern Regression with Applications*, PWS-KENT, Boston, MA, 1990.

[13] B. Efron and R.J. Tibshirani, *An Introduction to the Bootstrap*, Chapman and Hall, New York, NY, 1993.

[14] N.H. Packard, J.P. Crutchfeld, J.D. Farmer, and R.S. Shaw, "Geometry from a time series", *Physical Review Letters*, vol. 45, no. 9, pp. 712–716, 1980.

[15] A.W. Moore, J. Schneider, and K. Deng "Efficient locally weighted polynomial regression predictions", in *Machine Learning: Proceedings of the Fourteenth International Conference*, pp. 236–244, Morgan Kaufmann Publishers, San Francisco, CA, 1997.

[16] M. Birattari, G. Bontempi, and H. Bersini. "Lazy learning meets the recursive least-squares algorithm", in *Advances in Neural Information Processing Systems 11*, to be published, MIT Press, Cambridge, MA, 1999.

[17] J. Platt, "Resource-allocating network for function interpolation", *Neural Computation*, vol. 3, no. 2, pp. 213–225, 1991.

[18] G. Bontempi, M. Birattari, and H. Bersini, "Lazy learning for local modelling and control design", *International Journal of Control*, vol. 72, no. 7/8, pp. 643–658, 1999

[19] G. Bontempi, M. Birattari, and H. Bersini, "Local learning for iterated time series prediction", in *Machine Learning: Proceedings of the Sixteenth International Conference*, to be published, Morgan Kaufmann Publishers, San Francisco, CA, 1999.