TREE STRUCTURED NON-LINEAR SIGNAL MODELING AND PREDICTION

Olivier Michel   Alfred Hero   Anne Emmanuelle Badel

April 16, 1999

Abstract

In this paper we develop a regression tree approach to identification and prediction of signals which evolve according to an unknown non-linear state space model. In this approach a tree is recursively constructed which partitions the p-dimensional state space into a collection of piecewise homogeneous regions utilizing a 2^p-ary splitting rule with an entropy-based node impurity criterion. On this partition the joint density of the state is approximately piecewise constant leading to a non-linear predictor that nearly attains minimum mean square error. This process decomposition is closely related to a generalized version of the thresholded AR signal model (ART) which we call piecewise constant AR (PCAR). We illustrate the method for two cases where classical linear prediction is ineffective: a chaotic "double-scroll" signal measured at the output of a Chua-type electronic circuit, and a second order ART model. We show that the prediction errors are comparable to the nearest neighbor approach to non-linear prediction but with greatly reduced complexity.

Keywords: non-linear and non-parametric modeling and prediction, regression trees, recursive partitioning, chaotic signal analysis, piecewise constant AR models.

EDICS : SP 2.7, SP 3.9, SP 3.5

Corresponding Author :
Olivier Michel
Laboratoire de Physique
Ecole Normale Superieure de LYON
46 allee d'Italie
69394, LYON cedex 07, FRANCE
e-mail:omichel@ens-lyon.fr
tel: -33 (0) 472 72 83 78
fax: -33 (0) 472 72 80 80

1 Introduction

Non-linear signal prediction is an interesting and challenging problem especially in applications where the signal exhibits unstable or chaotic behavior [30, 61, 35]. A variety of approaches to modeling nonlinear dynamical systems and predicting non-linear signals from a sequence of N time samples have been proposed [12, 64] including: Hidden Markov Models (HMM) [24, 25], nearest neighbor prediction [20, 21], spline interpolation [39, 62], radial basis functions [10], and neural networks [32, 33]. This paper presents a stable low complexity tree-structured approach to non-linear modeling and prediction of signals arising from non-linear dynamical systems.

Tree-based regression models were first introduced as a non-parametric exploratory data analysis technique for non-additive statistical models by Sondquist and Morgan [58]. The regression-tree model represents the data in a hierarchical structure where the leaves of the tree induce a non-uniform partition of the data space over which a piecewise homogeneous statistical model can be defined. Each leaf can be labeled by a scalar or vector valued non-linear response variable. Once a cost-complexity metric is specified, known as a deviance criterion in the book by Breiman *etal* on classification and regression trees (CART) [9], the tree can be recursively grown to perform particular tasks such as non-linear regression, non-linear prediction, and clustering [9, 13, 55, 66]. The tree-based approach has several attractive features in the context of non-linear signal prediction. Unlike maximum likelihood approaches, no parametric model is required; however if one is available it can easily be incorporated into the tree structure as a constraint. Unlike approaches based on moments, since the tree-based model is based entirely on joint histograms all computed statistics are bounded and stable even in the case of heavy tailed densities. Unlike most methods, e.g. nearest neighbor, maximum likelihood, kernel density estimation, and spline interpolation, since the tree is constructed from rank order statistics its performance is invariant to monotonic non-linear transformations of the predictor variables. Furthermore, as different branches of the tree are grown independently, the tree can easily be updated as new data becomes available.

Our tree-based prediction algorithm has been implemented in Matlab[1] using a k-d tree growing procedure which is similar but not identical to that of the S-plus function `tree()` as described by Clarke and Pregibon [13]. Important features and contributions of this work are the following:

1. The Takens [19] time delay embedding method is used to construct a discrete time phase trajectory, i.e. a temporally evolving vector state, for the signal. This trajectory is then input to the tree growing procedure which attempts to partition the phase space into piecewise homogeneous regions.

2. The partitioning is accomplished by adding or deleting branches (nodes) of the tree according to a maximum entropy homogenization principle: we test that the joint probability density function (j.p.d.f.) is approximately uniform within any node (parent cell) by comparing the conditional entropy of the data points in the candidate partition of the node (children cells) to the maximum achievable conditional entropy in that partition. Cross-entropy criteria for node splitting have been used in the past, e.g. the Kullback-Liebler (KL) "node impurity" measure goes back to Breiman *etal* [9] and has been proposed as a splitting criterion for tree-structured vector quantization (TSVQ) in Perlmutter *etal* [47]. More recently, Zhang [66] proposed an entropy criterion for multivariate binomial classification trees which is closer in spirit to the method

[1] The Matlab code is available by request.

in this paper. Zhang found that the use of the entropy criterion produces regression trees that are more structurally stable, i.e. exhibit less variability as a function of N, than those produced using the standard squared prediction error criterion. For the non-linear prediction application, which is the subject of this paper, we have observed similar advantages using the Pearson Chi-square test of region homogeneity in place of the maximum entropy criterion of Zhang.

3. Similarly to Clarke and Pregibon [13], a median-based splitting rule is used to create splits of a parent cell along orthogonal hyperplanes, referred to as a median perpendicular splitting tree in the book by Devroye *etal* [17]. However, unlike previous methods which split only along the coordinate exhibiting the most spread, here the median splitting rule is applied simultaneously to each of the p coordinates of the phase space vector producing 2^p subcells. This has the advantage of producing denser partitions per node of the tree and, as the 2^p-ary split is balanced only in the case of uniform data, the cell probabilities in the split can be used directly for homogeneity testing of the parent cell.

4. In order to reduce the complexity of the tree a local singular value decomposition (SVD) orthogonalization of the phase space data is performed prior to splitting each node. This procedure, which can be viewed as applying a sequence of local coordinate transformations, produces a partition of the phase space into polygons whose edges are defined by non-orthogonal hyperplanes. This is similar to the principal component splitting method proposed for vector quantization of color images by Orchard and Bouman [46], and other non-orthogonal splitting rules for binary classification trees [17]. However, for phase space dimension $p > 2$ our method utilizes all components of the SVD as contrasted to the principal component alone.

5. When applied to non-linear signal prediction in phase space the SVD-based splitting rule yields a hierarchical signal model, which we call piecewise constant AR (PCAR), which is a generalization of the non-linear auto-regressive threshold (ART) model, called SETAR by Tong [61]. This thresholded AR model has been proposed for many physical signals exhibiting stochastic resonance or bistable/multi-stable trajectories such as ECG cardiac signals, EEG brain signals, turbulent flow, economic time series, and the output of chaotic dynamical systems (see references [61] and [31] for examples). A set of coefficients of the ART model can be extracted from a matrix obtained as the product of the local SVD coordinate transformation matrices. A causal and stable model can then be obtained by Cholesky factorization of this matrix.

6. We give a simple upper bound on the difference between the mean squared error of a fixed regression tree predictor and the minimum attainable mean squared prediction error. The bound establishes that a fixed regression tree predictor attains the optimal MSE when the j.p.d.f. is piecewise constant over the generated partition. This bound can be interpreted as an asymptotic bound on the actual MSE of our regression tree under the assumption that, as the training set increases, the generated partition converges a.s. to a non-random limiting partition. Many authors have obtained conditions for asymptotic convergence of tree-based classifiers and vector quantizers [17, 45, 43, 44]. However, as this theory requires strong conditions on the input data, e.g. independence, strong mixing, or stationarity, we do not pursue issues of asymptotic convergence in this paper.

7. It is shown by both simulations and experiments with real data that the non-linear prediction error performance of our regression tree is comparable to that of the popular

but more computationally intensive non-parametric nearest neighbor prediction method introduced by Farmer [20, 21]. A similar performance/computation advantage of our regression tree method has been established by Badel *etal* [10] relative to predictors based on radial basis functions (RBF).

The outline of the paper is as follows. In Section 2 some background on non-linear dynamical models and their phase space representation is given. Section 3 continues with background on regression tree prediction and the basic tree growing algorithm is described. In Section 4 the local SVD orthogonalization method is described and the equivalence of our SVD-based predictor to ART is established. Finally, in Section 5 experiments and simulations are presented.

2 Problem statement

It will be implicitly assumed that all random processes are ergodic so that ensemble averages associated with the signal can be consistently estimated from time averages over a single realization.

2.1 Non-linear modeling context

A very general class of non-linear signal models can be obtained by making non-linear modifications to the celebrated linear *ARMA(p,q)* model

$$x(n) = \sum_{i=1}^{p} a_i x(n-i) + \sum_{j=0}^{q} b_j e(n-j) \tag{1}$$

where $e(n)$ is a white Gaussian driving noise with variance σ^2, and the coefficients $\{a_i, i = 1, \ldots, p\}$ and $\{b_j, j = 1, \ldots, q\}$ are constants independent of $x(n)$ or e_n. Let

$$\mathbf{X}_n^{(k)} = [x(n-1)\ x(n-2)\ \ldots\ x(n-k)]^T$$

and

$$\mathbf{E}_n^{(k')} = [e(n-1)\ e(n-2)\ \ldots\ e(n-k')]^T$$

be vectors constructed from k and k' past values of x and e, respectively. Non-linear ARMA(p,q) models can be obtained by letting the coefficients (1) be functions of the ARMA state variables:

$$\{a_i, i = 1, \ldots, p\} = \mathcal{A}(\mathbf{X}_n^{(k)}, \mathbf{E}_n^{(k')})$$

$$\{b_j, i = j, \ldots, p\} = \mathcal{B}(\mathbf{X}_n^{(k)}, \mathbf{E}_n^{(k')})$$

where $\mathcal{A}$ and $\mathcal{B}$ are functions of $\mathrm{I\!R}^{k+k'}$ into $\mathrm{I\!R}^p$ and $\mathrm{I\!R}^q$, respectively.

This formulation has been used by Tong [61] and others to generate a wide class of non-linear stochastic models. For example, one easily obtains second order Volterra models or bilinear models by choosing $\mathbf{B} = [1, 0, \ldots, 0]^T$, $\mathcal{B}(\mathbf{X}_n^{(k)}, \mathbf{E}_n^{(k')}) = \mathbf{B}^T \cdot \mathbf{E}_n^{(k')}$, and $\mathcal{A}$ as the endomorphism

$$\mathcal{A}(\mathbf{X}_n^{(k)}, \mathbf{E}_n^{(k')}) \quad = \quad A \cdot [\mathbf{X}_n^{(k)^T}, \mathbf{E}_n^{(k')^T}]^T$$

where A is a constant matrix with p rows and $(k + k')$ columns.

Similarly, by exchanging the definitions for $\mathcal{A}$ and $\mathcal{B}$ in the preceding equations we obtain models for which the variance of the driving noise is a function of past values of x. These latter models are called heteroscedastic models, and are common in econometrics and other fields (see [18, 31]).

Moreover, one is not restricted to linear operators for $\mathcal{A}$ and $\mathcal{B}$. Piecewise constant state dependent values for the matrix $\mathcal{A}$ ($\mathcal{B}$ being kept constant) leads to a class of non linear model which are known as "piecewise ARMA" and referred to as generalized threshold autoregressive (TAR) or TARMA models [60]. As TARMA model coefficients depend on the previous states

$\mathbf{X}_n^{(k)}$ they belong to the general class of state dependant models developed by Priestley [51]. TARMA models arise in areas of time series analysis including: biology, oceanography, and hydrology. For more detailed discussion of non-linear models and their range of application the reader can consult the references [30, 52, 61].

In what follows the observed data will be represented by the sampled dynamical equation:

$$\begin{aligned} \mathbf{S}(n+1) &= F(\mathbf{S}(n)) + \varepsilon(n) \\ x(n+1) &= G(\mathbf{S}(n+1)) + \eta(n) \end{aligned} \tag{2}$$

where $\mathbf{S}(n)$ stands for the state vector at time n , F and G are (in general) unknown continuous functions from $\mathbb{R}^p$ into $\mathbb{R}^p$ and $\mathbb{R}^q$ respectively. $\varepsilon(n)$ is an i.i.d. state noise, and $\eta(n)$ is an i.i.d. observation noise. For $q > 1$ the observed quantity $x(n)$ is a multichannel measurement. We focus on the case $q = 1$ here. Note that the well known linear scalar AR process of order p may be represented within this framework by identifying $F(\mathbf{S}(n)) = A\mathbf{S}(n)$, A a $p \times p$ matrix in companion form, $G(\mathbf{S}(n+1)) = \mathbf{E}_1^T\mathbf{S}(n+1)$, $\mathbf{E}_1 = [1, 0, \ldots, 0]^T$, $\eta(n) = 0$, and $\mathbf{S}(n) = [x(n), \ldots, x(n-p+1)]^T$.

2.2 State space reconstruction method

Any process $x(n)$ obeying the pair of dynamical equations (2) is specified by its state vector $\mathbf{S}(n)$, called the *state trajectory*, evolving over $\mathbb{R}^p$, called the *state space*. The process of reconstruction of the state trajectory from real measurements is called state space embedding. For continuous time measurements $x(t)$ the reconstructed state trajectory is

$$\mathbf{X}(n) = [x(n)\ x(n-\tau_1)\ \ldots\ x(n-\tau_{\hat{p}-1})]^T \tag{3}$$

where by $x(n)$ we mean $x(nT_s)$ with $T_s > 0$ the sampling period, $\hat{p}$ is a positive integer, called the (estimated) embedding dimension, and τ_i are positive real numbers, called the embedding delays.

State space reconstruction was first proposed by Whitney [65], who stated conditions for identifiability of the continuous time state trajectory in the absence of observation noise. These conditions were formally proved and extended by Takens for the case of non-linear dynamical systems exhibiting chaotic behavior [19, 59], [11]. In practice only a finite number of (generally) equispaced samples are available and the embedding delay is set to $\tau_k = k\tau$, $\tau = mT_s$, where m is an integer value. In this finite case the value used for τ is very important: insufficiently large values lead to strong correlation or apparent linear dependences between the coordinates. On the other hand, overly large delays τ excessively decorrelate the components of $\mathbf{X}(n)$ so that the dynamical structure is lost [22, 23, 40], and [2, Ch. 3] or [35, Ch. 9].

Numerous authors have addressed the problem of finding the best embedding parameters τ and $\hat{p}$ (see e.g. [22, 23, 40] for detailed discussion). Selection of the dimension $\hat{p}$ requires investigation of the effective dimension of the space spanned by the estimated residuals. Overestimation of p creates state reconstructions with excessive variance while underestimation creates overly smooth (biased) reconstructions. A widely used method (see [2] or [35] for a discussion on this topic) which we will use for estimating p is the following: if $\hat{p} > p$, the true state dimension, then the estimated trajectories will lie on a lower dimensional manifold in $\mathbb{R}^{\hat{p}}$. This occurrence can be detected by testing a trajectory-dependent dimensionality criterion, e.g. the behavior of the algebraic dimension of the state trajectory vectors as $\hat{p}$ is increased [38]. We will adopt here the method of Fraser [22] for selection of τ: τ equals the time at which the first zero of the autocorrelation function occurs, i.e $\tau = \min\{\delta > 0 : \hat{C}(\delta) = 0\}$

where

$$\hat{C}(\delta) = \frac{1}{N-1}\sum_k x(k)x(k+\delta).$$

3 Growing the Tree

In this section we discuss the construction of the tree-structured predictor and give a bound on the mean squared prediction error of any fixed tree for the case that $\hat{p} = p$. As above let $\mathbf{X}(n) = [x(n-1), \ldots, x(n-p)]^T$ be a state vector of dimension p. A p-th order tree-structured predictor implements a regression function $\widehat{x}(n) = g(\mathbf{X}(n))$ which is piecewise constant as $\mathbf{X}(n)$ ranges over cells π_k in a partition $\{\pi_k\}$ of $\mathbb{R}^p$ [13]. The most common tree growing procedure [66, 9, 13] for regression and classification tries to find the partition of the phase space such that the predictive density $f(x(n)|x(n-1), \ldots, x(n-p))$ is approximately constant as the predictor variables $x(n-1), \ldots, x(n-p)$ vary over any of the partition cells. As is shown below, if the tree growing procedure does this sucessfully, the tree-based predictor $\widehat{x}_p(n)$ can attain mean squared error which is virtually identical to that of the optimal predictor $E[x(n)|x(n-1), \ldots, x(n-p)]$.

3.1 Regression Tree as a Quantized Predictor

Let $I_{\pi_k}(\mathbf{X}(n))$ be the indicator of the partition cell π_k and define the vector quantizer function

$$Q(\mathbf{X}(n)) = \sum_{k=1}^{L} \mathbf{q}_k I_{\pi_k}(\mathbf{X}(n)),$$

where $\mathbf{q}_k = [q_{k1}, \ldots, q_{kp}]^T$ is an arbitrary point in π_k. Typically, $\mathbf{q}_k$ is taken as the centroid of region π_k but this is immaterial in the following. Since the predictor function $\widehat{x}(n) = g(\mathbf{X}(n))$ is piecewise constant it is obvious that $g(\mathbf{X}(n)) = g(Q(\mathbf{X}(n)))$, i.e. the tree-structured predictor can be implemented using only the quantized predictor variables $Q(\mathbf{X}(n))$. Therefore, given the partition $\{\pi_k\}$, the optimal tree-based predictor can be constructed from the multidimensional histogram (the partition cell probabilities) as the conditional mean of $x(n)$ given the vector $Q(\mathbf{X}(n))$.

3.2 A Bound on MSE of Tree-structured Predictor

It follows from Theorem 1 in the Appendix that if the conditional density $f(x(n)|\mathbf{X}(n))$ is (Lipschitz) continuous of order α within all partition cells of the partition $\{\pi_k\}$ of $\mathbb{R}^p$, the mean squared error $E\left[(x(n) - E[x(n)|Q(\mathbf{X}(n))])^2\right]$ of the tree-structured predictor satisfies the following bound:

$$0 \leq E\left[(x(n) - E[x(n)|Q(\mathbf{X}(n))])^2\right] - E\left[(x(n) - E[x(n)|\mathbf{X}])^2\right] \leq 2 \max_i K_{\pi_i} m_x^2 E[\|\mathbf{X} - Q^o(\mathbf{X})\|^\alpha], \quad (4)$$

where $Q^o(\mathbf{X}(n))$ is the minimum mean squared error quantizer on $\{\pi_k\}$, m_x is an upper bound on the mean squared valued of $x(n)$ given $\mathbf{X}(n)$, and K_{π_k} is a Lipschitz constant characterizing the modulus of continuity within π_k. The upper bound in (4) is decreasing in the minimum α-th power quantization error $E[\|\mathbf{X}(n) - Q^o(\mathbf{X}(n))\|^\alpha]$ associated with optimal vector quantization of the predictor variables. Bounds and asymptotic expressions exist for this quantity [29, 42] which can be used to render the upper bound (21) more explicit, however this will not be explored here.

Note that the upper bound in (4) is decreasing in $\max_i K_{\pi_i}$ and equals zero when $f(x(n)|\mathbf{X}(n))$ is piecewise constant in $\mathbf{X}(n)$, i.e. $f(x(n)|\mathbf{X}(n) = \mathbf{x}) = \sum_i f(x(n)|\mathbf{q}_i) I_{\pi_i}(\mathbf{x})$ where $\mathbf{q}_i \in \pi_i$ are arbitrary. Thus in the case of a piecewise uniform conditional density, the optimal predictor of $x(n)$ given quantized data $Q(\mathbf{X}(n))$ is identical to the optimal non-linear predictor given unquantized data $\mathbf{X}(n)$, i.e. the tree-structured predictor attains the minimum possible prediction MSE. Note that for a general conditional density, both $E[\|\mathbf{X} - Q^o(\mathbf{X})\|^\alpha]$ and the total variations $\{K_{\pi_k}\}$ decrease as the sizes of the partition cells $\{\pi_k\}_i$ decrease. Hence the mean square prediction error can be seen from (4) to improve monotonically as the conditional density $f(x(n)|\mathbf{X}(n))$ becomes well approximated by the staircase function $f(x(n)|Q(\mathbf{X}(n))$ over $\mathbf{X}(n) \in \mathbb{R}^p$. This forms the basis for tree-based non-linear prediction as explained in more detail below.

3.3 Branch Splitting and Stopping Rules

Here we describe the generic recursive procedure used for growing the tree from training data. Let $\hat{p}$ be an estimate of the phase space dimension p of the signal. Assume that at iteration l of the tree growing procedure we have created a partition Π^l and consider the partition cells π_i^l, which we call the i-th parent nodes at depth l. We refine the partition Π^l by recursively splitting each partition cell π_i^l into $2^{\hat{p}}$ smaller cells which are called children-nodes of the i-th parent.

To control the number of nodes of the tree we test the residuals in each partition element of Π^l against a uniform distribution. If the test for uniformity fails in a particular cell, that cell is split and $2^{\hat{p}}$ parent nodes at depth $l+1$ are created. Otherwise the cell is not split and is declared a terminal node. The set of terminal nodes are called the leaves of the tree. See Fig. 1 for a graphical illustration of the generic tree growing procedure. The final tree specifies a set of leaves $\pi_1, \ldots, \pi_L$ partitioning the state-space together with the empirical histogram (cell occupancy rate): $\widehat{p}_j = P(\mathbf{X} \in \pi_j) = N_{\pi_j}/N$, where N_{π_j} is the number of samples $\{\mathbf{X}(k)\}_{k=1}^N$ which fall into leaf π_j.

3.3.1 Cell Uniformity Test

Here we discuss the selection of the goodness-of-split criterion that is used to test uniformity. As above let $\mathbf{X}(k)$ denote the $\hat{p}$-dimensional vector sampled at time kT_s, where the reconstruction dimension $\hat{p}$ is fixed. Many discriminants are available for testing uniformity including Kolmogorov-Smirnov tests [37], rank-order statistical tests [16], and scatter matrix tests [26].

Following Breiman *etal* and Zhang [9, 66] we adopt an entropy-like criterion. However, as contrasted to previous implementations [9, 66] this criterion is implemented using simple Chi-square goodness-of-fit test of significance over the distribution of child cell probabilities.

For a partition $\{\pi_1, \ldots, \pi_{2^{\hat{p}}}\}$ of a cell Π, let N_{π_i} be the number of vectors $\mathbf{X}(k)$, $k = 1, \ldots, N$, found in π_i. We assume that the vectors falling into the cells are approximately i.i.d. and that N_{π_i}, $i = 1, \ldots, 2^{\hat{p}}$, are approximately multinomial distributed random variables with class probabilities $p_i = P(\mathbf{X}(p) \in \pi_i | \mathbf{X}(p) \in \Pi) = E[N_{\pi_i}]/N$. These are reasonable assumptions when the volume of cell Π is small and $x(n)$ satisfies a long range decorrelation property (weak mixing), but we do not pursue a proof of this here. The test of uniformity is implemented by using the empirical cell probabilities $\hat{p}_i = \frac{N_{\pi_i}}{N}$ to test the uniform hypothesis $H_0 : p_i = 2^{-\hat{p}}$, $i = 1, \ldots, 2^{\hat{p}}$ against the composite alternative hypothesis $H_1 : p_i \neq 2^{-\hat{p}}$, $i = 1, \ldots, 2^{\hat{p}}$. Define the Kullback-Liebler (KL) distance between $\hat{p}_i$ and the uniform distribution $p_i = 2^{-\hat{p}}$

$$\begin{aligned} D(\hat{p}_i, p_i) &= \sum_{i=1}^{2^{\hat{p}}} p_i \log\left(\frac{\hat{p}_i}{p_i}\right) \\ &= \log 2^{\hat{p}} + \sum_{i=1}^{2^{\hat{p}}} p_i \log \hat{p}_i. \end{aligned} \tag{5}$$

It is easy to show that the generalized likelihood ratio test of H_0 vs. H_1 is to decide H_0 if $D(\hat{p}_i, p_i) < \eta$ where the threshold η selected to ensure that the probability of false rejection of H_0 is equal to a prescribed false alarm rate (see, e.g. [7, Ch. 8]).

However, since the distribution of $D(\hat{p}_i, p_i)$ is intractable under H_0 the decision threshold cannot easily be chosen to satisfy a prespecified false alarm level. We instead propose Pearson's Chi square goodness-of-fit test statistic χ^2 which has a central Chi square distribution under H_0. In particular, it can be shown [7, 6] that Pearson's Chi square statistic is a local approximation to the KL distance statistic (5) in the sense that

$$D(\hat{p}_i, p_i) = \frac{1}{2N_\Pi}\chi^2 + o(\max_i(\hat{p}_i - p_i)^2),$$

where $\chi^2 = N_\Pi \sum_{i=1}^{2^{\hat{p}}} \frac{(\hat{p}_i - p_i)^2}{p_i}$ is distributed as a central Chi square with $2^{\hat{p}} - 1$ degrees of freedom undr H_0.

3.3.2 Separable Splitting Rule

Another component of the procedure for growing a tree is the method of splitting parent cells into children cells. The standard cell splitting rule attempts to create a pair of rectangular subcells for which all marginal probabilities are identical regardless of the underlying distribution. The median-based binary splitting method for constructing k-d trees [5, 15, 13, 17] is commonly used for this purpose. As the median is a rank order statistic this gives the property that the predictor is invariant to monotone transformations of the predictor variables, a property not shared by most other non-linear predictors. Here we present a variant of the standard median splitting rule which generates $2^{\hat{p}}$ rectangular children cells which only have equal probabilities when the data is uniform over the parent cell. A version of this $2^{\hat{p}}$-ary splitting rule which generates non-rectangular cells is discussed in Section 4.

Let $\times_{i=1}^{\hat{p}}[\alpha_i, \beta_i]$ denote the hyper-rectangle constructed from the Cartesian product of intervals $[\alpha_i, \beta_i]$, $\alpha_i < \beta_i$, e.g. $\times_{i=1}^{2}[\alpha_i, \beta_i] = [\alpha_1, \beta_1] \times [\alpha_2, \beta_2]$ is a right parallelepiped in $\mathbb{R}^2$. We start with a partition element $\Pi = \times_{i=1}^{\hat{p}}[\alpha_i, \beta_i]$. Let this partition element contain N_Π of the reconstructed state vectors $\{\mathbf{X}(k)\}_{k=1}^{N}$. Define the N_Π-element vector $\mathbf{X}_\Pi^j = [\mathbf{e}_j^T \mathbf{X}(k) : \mathbf{X}(k) \in \Pi, k = 1, \ldots, N]$ as the projection of the inscribed reconstruction vectors onto the j-th coordinate axis. That is $\mathbf{X}_\Pi^j$ is the set of j-th coordinates of those $\mathbf{X}(k)$ falling into Π, $k = 1, \ldots, N$. Denote by $\widehat{T}_\Pi^j$ the sample median of the j-th coordinate axis projections

$$\widehat{T}_{\Pi}^{j} = \text{median}\{\mathbf{e}_j^T \mathbf{X}(k) : \mathbf{X}(k) \in \Pi, k = 1, \ldots, N\}.$$

where, for a scalar sequence $\{x_i\}_{i=1}^n$, the sample median is a threshold such that half fall to the left and half to the right:

$$\text{median}\{x_i\} = \begin{cases} x_{(n/2)}, & n \text{ even} \\ x_{([n+1]/2)}, & n \text{ odd} \end{cases}$$

and $x_{(1)} \leq \ldots \leq x_{(n)}$ denotes the rank ordered sequence. Note that when the points $\{\mathbf{X}(k)\}_k$ are truly uniform over parent cell the medians $\{\widehat{T}_{\Pi}^{j}\}_j$ will tend to be near the midpoints of the edges of the parent cell.

The standard median tree implements a binary split of parent cell Π about a hyperplane perpendicular to that coordinate axis j having the largest spread of points $\mathbf{X}_{\Pi}^{j}$ where the hyperplane intersects this coordinate axis at the median $\widehat{T}_{\Pi}^{j}$. This produces a pair of children cells which contain an identical number of points. In contrast, we split Π into $2^{\hat{p}}$ rectangular children cells whose edges are defined by all $\hat{p}$ perpendicular hyperplanes of the form: $\{\mathbf{X} : \mathbf{e}_j^T \mathbf{X} = \widehat{T}_{\Pi}^{j}\}$, $j = 1, \ldots, \hat{p}$. This produces a tree with a denser partition than the standard median tree having the same number of nodes. Unlike the standard median splitting rule tree, these $2^{\hat{p}}$ children cells will not have identical numbers of points unless the points are truly uniform over Π. This allows the cell occupancies in the $2^{\hat{p}}$-ary split to be used directly for uniformity testing as described in the previous section.

3.3.3 Stopping Rule

The last component of the tree growing procedure is a stopping rule to avoid over-fitting. As above, define $\mathbf{X}_{\Pi}^{j} = \{\mathbf{e}_j^T \mathbf{X}(k) : \mathbf{X}(k) \in \Pi, k = 1, \ldots, N\}$ as the j-th coordinates of the vectors $\mathbf{X}(k)$ falling into the hyper-rectangle $\Pi = \times_{i=1}^{\hat{p}}[\alpha_i, \beta_i]$. Thus each of the elements of $\mathbf{X}_{\Pi}^{j}$ lies in the interval $[\alpha_j, \beta_j]$. Under the assumption that these elements are i.i.d. with continuous marginal probability density function $f_{x^j|\Pi}$, the sample median $\widehat{T}_{\Pi}^{j}$ is an asymptotically unbiased and consistent estimator of the theoretical median T_{Π}^{j}, which is the half mass point of the marginal cumulative distribution function. Conditioned on N_{Π} the sample median has an asymptotic normal distribution [41]:

$$\widehat{T}_{\Pi}^{j} \sim \mathcal{N}\left(T_{\Pi}^{j}, \frac{1}{4N_{\Pi}[f_{x^j|\Pi}(T_{\Pi}^{j})]^2}\right). \tag{6}$$

The stopping rule is constructed under the assumption that $f_{x^j|\Pi}$ is a uniform density $f_{x^j|\Pi}(x) = 1/(\beta_j - \alpha_j)$ over $x \in [\alpha_i, \beta_i]$. Under this assumption $T_{\Pi}^{j} = (\beta_i + \alpha_i)/2$ is the midpoint and the sample medians $\hat{T}_{\Pi}^{j}$, $j = 1, \dots, \hat{p}$, are statistically independent. A natural stopping criterion is to require that the number N_{Π} of data points within Π be sufficiently large so that the Gaussian approximation to the density of $\hat{T}_{\Pi}^{j}$ has negligible mass outside of the interval $[\alpha_i, \beta_i]$. When this is the case it can be expected that $\hat{T}_{\Pi}$ will be a reliable estimate of the interval midpoint. More concretely, we will require that N_{Π} satisfy

$$P\left(|\hat{T}_{\Pi}^{j} - T_{\Pi}^{j}| \leq (\beta_j - \alpha_j)/2,\ j = 1, \dots, \hat{p}\right) \geq 1 - \epsilon. \tag{7}$$

where $\epsilon \in [0, 1]$ is a suitable (small) prespecified constant.

Since the $\hat{T}_{\Pi}^{j}$ are independent, the criterion (7) is equivalent to

$$1 - \prod_{j=1}^{\hat{p}} P(|\hat{T}_{\Pi}^{j} - T_{\Pi}^{j}| \leq (\beta_j - \alpha_j)/2) \leq \epsilon,$$

which, under the Gaussian approximation (6), gives

$$1 - P\left(|Z| \leq \sqrt{N_{\Pi}}\right)^{\hat{p}} \leq \epsilon$$

where Z is a standard normal random variable (zero mean and unit variance). Thus we obtain the following stopping criterion: continue subdividing the cell Π as long as

$$N_{\Pi} \geq 2\left[\text{erf}^{-1}\left([1-\epsilon]^{1/\hat{p}}\right)\right]^2 \tag{8}$$

where $\mathrm{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt$. Use of the asymptotic representation $1 - \mathrm{erf}(x) = \mathrm{erfc}(x) = e^{-x^2}/(x\sqrt{\pi}) + o(1/x)$ [3, 26.2.12] gives the log-linear small ϵ version of (8)

$$N_\Pi \geq 2 \ln \left(\frac{2\hat{p}}{\epsilon} \right). \tag{9}$$

The right hand side of the inequality (8) is plotted as a function of ϵ for several values of $\hat{p}$ in Fig. 2. Note that as predicted by the asymptotic (small ϵ) bound (9) the curves are very close to log linear in ϵ. As a concrete example, the criterion $\epsilon = 0.01$ (99 percent of Gaussian probability mass is inside π) gives for $\hat{p} = 2$: $N_\Pi = 8$, and for $\hat{p} = 4$: $N_\Pi = 10$, as the minimum number of data points N_Π for which a cell will be further subdivided. These numbers are of the same order as those obtained from the volume estimation criterion used by Badel *etal* [6].

3.4 Computational cost

The steps outlined in the preceding subsections may be summarized by the following tree growing algorithm:

```
1. input sampled time series and embedding parameters, $(\hat{p}, \tau)$, Pearson's $\chi^2$ test threshold
2. initialize $\Pi^0$ = set of all state vectors
3. while non-empty non-terminal leaves exist, at current depth $l$
4. for each cell $\Pi^l$
5.     if $\Pi^l$ contains $N_{\Pi^l} \geq \hat{p}^2 4^{\hat{p}-1}$ vectors
6.         compute the splitting thresholds $T^j_{\Pi^l}, j = 1, \ldots, \hat{p}$
7.         estimate empirical probabilities at depth $l+1$ for the children of $\Pi^l$
8.         Compute Pearson's $\chi^2$ statistic from the $2^{\hat{p}}$ probabilites
9.         if $\chi^2$ is less than threshold
10.            $\Pi^l$ is stored as a terminal leaf
11.          else
12.            $\{T^j_{\Pi^l}, j = 1, \ldots, \hat{p}\}$ are stored
13.            $\{\Pi^{l+1}_k, k = 1, \ldots, 2^{\hat{p}}\}$ are stored
14.          endif
15.      else $\Pi^l$ is stored as an 'empty' terminal leaf
16.      endif
17.  endfor
18.  l=l+1;
19.  goto line 3.
```

The computational cost associated with this tree estimation algorithm is signal dependent. For example, in the case of a state space containing N realizations of a p-dimensional white noise, the trivial partition $\Pi_0 = \mathbb{R}^p$ will generally pass the uniformity test and the algorithm will stop at the root node. In this case, only a very few computations are needed. In the following, we give an estimate of the worst case cost occurring when the terminal nodes all occur at the same depth.

At depth l of the tree, under the assumption that all obtained cells were stored as non-empty, non-terminal leaves, the tree has $2^{\hat{p}}l$ cells. The average number of $\hat{p}$-dimensional vectors in each leaf is

$$< N_{\Pi^l} >= \frac{N_{\Pi^0}}{2^{\hat{p}l}}.$$

The most computation consuming step in the algorithm is the splitting threshold determination procedure which requires rank ordering each of the coordinates of the inscribed state vectors. Using an optimized method (e.g. the heap-sort algorithm [50]) leads to a cost proportional to

$$C_l \simeq 2^{\hat{p}l} < N_{\Pi^l} > \log_2 < N_{\Pi^l} >= N_{\Pi^0}\hat{p} \log_2 \frac{N_{\Pi^0}}{2^{\hat{p}l}}.$$

By adding the computational costs obtained for each depth in the range $l = 0, \ldots, l_{max} - 1$, one obtains the following expression of the total cost

$$C_{Tot} \simeq N_{\Pi^0}\hat{p}\; l_{max} \log_2 \frac{N_{\Pi^0}}{2^{\frac{\hat{p}(l_{max}-1)}{2}}}.$$

Note that the expression of C_{Tot} corresponds to the worst case where no cells pass the χ^2 uniformity test until the minimal cell residency stopping criterion is reached. This computational cost is well below that of the well-known nearest neighbor one-step prediction method: $C_{NN} \simeq \hat{p}\frac{N_{\Pi_0}^2}{2} + N_{\Pi_0} \log N_{\Pi_0}$.

4 Complexity Reduction via SVD Orthogonalization

The number of leaves in the final tree, i.e. the number of cells in the partition of state space, is a reasonable measure of model complexity. However, without additional preprocessing of the data the separable splitting rule described in the previous section can produce trees of greatly varying complexity for state space trajectories which are identical up to a rotation in $\mathbb{R}^p$. This is an undesirable feature since a simple transformation of coordinates in the state space, such as translation, scale, and rotation, does not change the intrinsic complexity of the process, e.g. as measured by process entropy or Lyapunov exponent.

As a particularly simple example, consider the case where the state trajectory evolves about line segment in 2 dimensions $x(n) = ax(n-1) + \epsilon(n)$, where $\epsilon(n)$ is a white noise, with variance σ^2. Under the separable 2^p-ary partitioning rule when $a = +1$ or $a = -1$ a very complex tree will result. This is because the Chi-square splitting criteria will lead to a tree with cell sizes on the order of magnitude of σ. This is troublesome, as a simple rotation of the axis coordinates by an angle of $\pi/4$ will lead the partitioning algorithm to stop at the root node. Here we perform a local recursive orthogonalization of the state vector prior to node splitting in order to produce trees with fewer leaves. This produces a new orthogonalized sequence of node variables $\mathbf{Z}_{\pi_j^l}$ which are used in place of $\mathbf{X}_{\pi_j^l}$ to perform separable splitting and goodness-of-split tests discussed in the previous section. The local recursive orthogonalization described below differs from a similar principal component orthogonalization for binary partitioning, first described by Orchard and Bouman [46], in that all the components of the SVD are utilized for the $2^{\hat{p}}$-ary partition used in this paper.

4.1 Local Recursive Orthogonalization

We recursively define a set of orthogonalized node variables as follows. Let $\mathbf{X}$ be a $\hat{p} \times N$ matrix of samples $\mathbf{X}(k)$, $k = 1, \ldots, N$, of the $\hat{p}$-dimensional state trajectory. Let the covariance of $\mathbf{X}(k)$ be denoted $\Lambda_{\mathbf{X}}$ and let it have the SVD (eigendecomposition) $\Lambda_{\mathbf{X}} = M_{\mathbf{X}}^T \mathrm{diag}(\lambda_{\mathbf{X}(k)}) M_{\mathbf{X}}$. Define the root node $\pi^0 = \mathbb{R}^{\hat{p}}$. Next define the orthogonalized set of vectors $\mathbf{Z}_{\pi^0}$

$$\mathbf{Z}_{\pi^0} = M_{\mathbf{X}}(\mathbf{X} - E[\mathbf{X}]).$$

The matrix $\mathbf{Z}_{\pi^0}$ is now used in place of $\mathbf{X}$ to determine the split of the root node into children $\pi_1^1, \ldots, \pi_{2^{\hat{p}}}^1$ according to the same separable splitting and stopping criteria as before. In practice the empirical mean $\hat{\mathbf{X}} = \frac{1}{N}\mathbf{X}\mathbf{1}$ and empirical covariance $(\mathbf{X} - \hat{\mathbf{X}})(\mathbf{X} - \hat{\mathbf{X}})^T/(N-1)$ are used in place of $E[\mathbf{X}]$ and $\Lambda_{\mathbf{X}}$.

Now assume a split occurs at the root node and define $\mathbf{Z}^0_{\pi_j^1}$ as the matrix of columns of $\mathbf{Z}_{\pi^0}$ which lie inside π_j^1. The (empirical) mean and covariance matrix $\Lambda_{\mathbf{Z}^0_{\pi_j^1}}$ of $\mathbf{Z}^0_{\pi_j^1}$ are computed. Next the unitary matrix $M_{\mathbf{Z}^0_{\pi_j^1}}$ of eigenvectors of $\Lambda_{\mathbf{Z}^0_{\pi_j^1}}$ is extracted via SVD. This unitary matrix is applied to $\mathbf{Z}^0_{\pi^1_j}$ to produce an equivalent but uncorrelated set of vectors $\mathbf{Z}_{\pi^1_j}$:

$$\mathbf{Z}_{\pi^1_j} = M_{\mathbf{Z}^0_{\pi^1_j}}(\mathbf{Z}^0_{\pi^1_j} - E[\mathbf{Z}^0_{\pi^1_j}]) + E[\mathbf{Z}^0_{\pi^1_j}]\mathbf{1}^T_{\pi^1_j}$$

where $\mathbf{1}^T_{\pi^1_j}$ stands for the transpose of the vector containing $N_{\pi^1_j}$ ones. Application of this local orthogonalization procedure over all $2^{\hat{p}}$ hyper-rectangles $\pi^1_1, \ldots, \pi^1_{2^{\hat{p}}}$ produces a set of local coordinate rotations which results in changing the shape of the hyper-rectangles into hyper-parallelepipeds. When this process is repeated these hyper-parallelepipeds are further subdivided producing, at termination of the algorithm, a partition of the state space into general polytopes π^1_j.

The general recursion from depth l to depth $l+1$ can be written as

$$\mathbf{Z}_{\pi^{l+1}_j} \quad = \quad M_{\mathbf{Z}^l_{\pi^{l+1}_j}}\mathbf{Z}^l_{\pi^{l+1}_j} + \mathbf{C}^l\mathbf{1}^T_{\pi^{l+1}_j} \tag{10}$$

where $\mathbf{C}^l = E[\mathbf{Z}^l_{\pi^{l+1}_j}] - M_{\mathbf{Z}^l_{\pi^{l+1}_j}}E[\mathbf{Z}^l_{\pi^{l+1}_j}]$.

4.2 Relation to Piecewise Constant AR (PCAR) models

Once the tree growing procedure terminates the partitions π^l_j can be mapped back to the original state space by a sequence of backward recursions which back-projects the π^{l+1}_j node variables $\mathbf{Z}_{\pi^{l+1}_j}$ into the parent cell π^l via the relation

$$\mathbf{Z}^l_{\pi^{l+1}_j} \quad = \quad M^T_{\mathbf{Z}^l_{\pi^{l+1}_j}}(\mathbf{Z}_{\pi^{l+1}_j} - \mathbf{C}_{\pi^l}), \tag{11}$$

Iteration of (11) over l yields an equation for back-projection of $\mathbf{Z}_{\pi^{l+1}_j}$ to the root node. By induction on l the forward recursion (10) gives the relation

$$\mathbf{Z}_{\pi^l} \quad = \quad \mathcal{M}_{\pi^l}\mathbf{X}_{\pi^l} - \mathcal{C}_{\pi^l}\mathbf{1}^T_{\pi^l} \tag{12}$$

where $\mathbf{X}_{\pi^l}$ denotes the subset of columns of $\mathbf{X}$ that are mapped to terminal node π^l at depth l via the sequence of bijective maps (10), $\mathcal{M}_{\pi^l}$ and $\mathcal{C}_{\pi^l}$ are matrices formed from the telescoping series

$$\mathcal{M}_{\pi^l} = \prod_{i=0}^{l} M_{\mathbf{Z}^i_{\pi^l}} \tag{13}$$

$$\mathcal{C}_{\pi^l} = \sum_{i=0}^{l} \left[\prod_{j=i}^{l} M_{\mathbf{Z}^j_{\pi^l}} \right] \mathbf{C}_{\pi^i} \tag{14}$$

where $M_{\mathbf{Z}^0_{\pi^l}}$ is defined as the p-dimensional identity matrix.

For any parent node π^l the covariance matrix of the rotated data $\mathbf{Z}_{\pi^l}$ is diagonal, which means that the components of $\mathbf{Z}^{\pi^l}$ are separable (in the mean squared sense) but not necessarily uniform. On this rotated data the Chi-square test for uniformity can easily be implemented on a coordinate-by-coordinate basis. When the tree growing procedure terminates we will have found a set of partition cells $\pi_1^{l_1}, \ldots, \pi_L^{l_L}$ such that each $\pi^l = \pi^l_j$ contains points $\mathbf{Z}_{\pi^l}$ which are (approximately) uniformly distributed over π^l. Thus, relation (12) gives an autoregressive AR(p-1) model whose coefficients are piecewise constant over regions of state space $\mathbf{X}_{\pi^l}$.

This can be made more transparent by writing the i-th component of relation (12) as

$$x(n) = -\sum_{j=1}^{p-1} a_{\pi^l}(i,j)x(n-j) + w_{\pi^l}(n), \quad \mathbf{X}(n) \in \pi^l \tag{15}$$

where $a_{\pi^l}(i,j) = m_{\pi^l}(i,j+1)/m_{\pi^l}(i,1)$, $m_{\pi^l}(i,j)$ denotes the i,j element of $\mathcal{M}_{\pi^l}$, and $w_{\pi^l}(n) = (\mathbf{Z}_{\pi^l}(n) + \mathcal{C}_{\pi^l}\mathbf{1}^T_{\pi^l})\mathbf{e}^i_1$ is a white noise.

Note that the coefficients for the PCAR representation (15) may not be stable. There is an alternative approach to orthogonalizing the node variables which uses Gramm-Schmidt recursions and guarantees that all PCAR coefficients are stable. This method is equivalent to constructing the Schur complement by adding one coordinate to each vector in the node; amounting to recursively synthesizing a local stable AR(p-1) model over $p = 1, 2, 3, \ldots$. This is tantamount to performing Cholesky (LDU) factorization of the local covariance matrices

$\Lambda_{\mathbf{Z}^{l}_{\pi_j^{1+1}}}$ [56] as contrasted with the SVD factorization described above. In the sequel the former method will lead to what will be called a Schur-Tree while the latter will lead to a tree called the SVD-Tree.

The PCAR model (12) is a generalization of the AR-threshold (ART) model, called SETAR in Tong [61]. Similarly to the PCAR model (12), SETAR is an AR model whose coefficients are piecewise constant over regions of state space; but unlike the PCAR model these regions are restricted to half planes. In particular a 2-level single coordinate p-th order SETAR model is

$$x(n) = \begin{cases} a_{10} + a_{11}x(n-1) + \ldots + a_{1d}x(n-p) + \sigma_1\epsilon(n) & \text{if } x(n-d) \leq T_0 \\ a_{20} + a_{21}x(n-1) + \ldots + a_{2d}x(n-p) + \sigma_2\epsilon(n) & \text{if } x(n-d) > T_0 \end{cases} \tag{16}$$

where $d \in \{1, \ldots, p\}$. As far as we know, filtering, prediction, and identification of SETAR models have only been studied for the case where the switching of the AR coefficients depends on a single coordinate $x(n-d)$ and where the switching threshold T_0 is known. The PCAR generalization of SETAR models allows transition thresholds to be applied to linear combinations of past values. As will be illustrated below, the orthogonalized version of the tree based partitioning algorithm is well adapted to filtering, prediction and identification over these models.

5 Examples and Applications

In this section the tree-structured predictors are applied to various real and simulated data examples.

5.1 Illustrative Examples

To illustrate the parsimony of the local recursive orthogonalization method, we first consider a rather artificial random process which follows a piecewise linear trajectory through state space (see Figure (3)). A trajectory made of three linear segments in a 2 dimensional state space was simulated. The segments have slopes 1.25, -.25 and -2.5 respectively.

Each segment contains 128 realization of the 2 dimensional state vector. White Gaussian i.i.d. noise of variance $\sigma^2 = 5$ was added to the trajectory. We first applied the recursive tree (RT) method in $p = 2$ state dimensions without SVD orthogonalization. Both the rectangular partition of the state space (a) and the tree partitioning algorithm (b) exhibit high complexity. The number of terminal leaves of the resulting quad-tree is driven exclusively by the variance of the additive noise. We next grew a quad-tree using the local recursive SVD orthogonalization procedure, which will be called SVD-Tree here, described in Section 4. The orthogonalization procedure re-expresses the state vectors in their local eigenbases at each splitting iteration and, as seen from Figure 4, produces a tree partitioning of lower complexity with many fewer leaves. As explained in Section 4, applying the recursive SVD orthogonalization on a cell π_j^l synthesizes the local AR(1) model (recall (15)

$$x(n) = -a_{\pi_j^l} x(n-1) + w_{\pi_j^l}(n), \quad [x(n), x(n-1)] \in \pi_j^l.$$

We denote by $\mathbf{A}_j^l = [1, a_{\pi_j^l}]/\sqrt{1 + a_{\pi_j^l}^2}$ the unit-length vector of the AR model synthesized in the cell π_j^l. Figure 4-c plots the set of unit-length vectors for all cells π_j^l resulting from the SVD-Tree partition.

The length of each segment is plotted proportionally to the number of points falling into the corresponding cell. Note that this graphical representation clearly reveals the existence of 3 distinct linear segments governing the state trajectories.

5.2 Chua Circuit Experiments

We ran experiments on a physical chaotic voltage waveform, measured at the output of a "double-scroll' Chua electronic circuit (see [40], [63], [49]).

The non-linear differential equations governing the Chua circuit are :

$$\left\{\begin{array}{l} \frac{dx}{dt} = \alpha(y - \gamma) \\ \frac{dy}{dt} = x - y + z \\ \frac{dz}{dt} = -\beta y \end{array}\right. \tag{17}$$

We built the circuit from "off the shelf" components chosen so as to get the following set of parameter values $\alpha = 9$, $\beta = \frac{100}{7}$, $m_0 = -\frac{1}{7}$ and $m_1 = \frac{2}{7}$. The voltage signal at the output of the electronic circuit was digitized. The sampling frequency was 14.4 kHz. We chose an embedding dimension $\hat{p} = 4$ to generate the state trajectory $\mathbf{X}(k) = [x(k), x(k-\tau), x(x-2\tau), x(k-3\tau)]^T$. We used a stopping threshold of $N_\pi > 16$ data points which corresponds to $\epsilon \simeq 3.10^{-3}$ (for $p = 4$) via relation (8). The reconstruction delay τ was chosen in such a way as to minimize the mutual information between the coordinates (see [23] and [19]): in this case, $\tau = 4$ sampling periods. A training set of $N_{\Pi_0} = 8192$ points was used to grow the Schur-Tree and obtain the empirical histogram $\{N_i/N\}$ on the leaves $\{\pi_i\}$ of the tree. A non-linear predictor of $x(n)$ given $x(n-1), \ldots, x(n-\hat{p}+1)$ was implemented by approximating the conditional mean $\hat{x}(n) = E[x(n)|x(n-1), \ldots, x(n-\hat{p}+1)]$ using the tree-induced vector quantizer function $Q()$ and the empirical histogram. Specifically, with $Q(x(n), \ldots, x(n-\hat{p}+1)) = \sum_i \mathbf{q}_i I_{\pi_i}(x(n), \ldots, x(n-\hat{p}+1))$

$$\widehat{x}(n) \quad = \quad \frac{\sum_{j=1}^{\hat{p}} \xi_{j1} \widehat{P}(\xi_{j1}, q_{n-1}, \ldots, q_{n-\hat{p}+1})}{\sum_{j=1}^{\hat{p}} \widehat{P}(\xi_{j1}, q_{n-1}, \ldots, q_{n-\hat{p}+1})}, \tag{18}$$

where $\{\xi_i\}$ are centroids of the partition cells $\{\pi_i\}$ at the leaves of the tree, ξ_{j1} denotes the 1-st element of the vector ξ_j, $q_{n-1}, \ldots, q_{n-\hat{p}+1}$ are the 2-nd through $\hat{p}$-th elements of the vector $Q(x(n), \ldots, x(n-\hat{p}+1))$, and $\hat{P}(\mathbf{q}) = \frac{1}{N}\sum_i N_i I_{\pi_i}(\mathbf{q})$ is the empirical histogram indexed by $\mathbf{q}$.

Figures 5.a and 5.b show time segments of actual measured and predicted output Chua circuit voltages using the Schur-Tree predictor and the popular but costlier nearest neighbor

(NN) prediction method, respectively. The NN prediction method is briefly summarized below.

The NN prediction method consists of finding in a learning sequence $L = \mathbf{X}(n), n = 1, \ldots, N$ the point $\mathbf{X}(j), 1 \leq j \leq N$ in the state space which is the closest (in some metric) to the current observation $\mathbf{X}(t)$ and defining the predictor as $\mathbf{X}(t \hat{+} 1) = \mathbf{X}(j + 1)$. As is shown in Devroye *etal* [17], under certain technical conditions the mean squared prediction error of the NN predictor decreases to zero in N. The NN predictor was implemented in a manner identical to the one proposed by Farmer [21]. While more sophisticated implementations of NN predictors are available, see e.g. [1, 20, 54], they require higher implementation complexity than Farmer's implementation, for only a small improvement in prediction error performance. We performed benchmarks in Matlab 4.2c on a Sun Ultra-1 workstation for 512 one-step predictions of the SETAR model described above. The CPU run times were $33.2s$ for SVD-Tree versus $115.3s$ for the NN prediction algorithm, respectively, with comparable prediction error performance.

5.3 SETAR Time Series Simulations

Figure 6 presents results for the simulated SETAR model;

$$x(k) = \begin{cases} 1.71x_{k-1} - .81x_{k-2} + .356 + \varepsilon_k, & x_{k-1} > 0 \\ -.562x_{k-2} - 3.91 + \varepsilon_k, & x_{k-1} \leq 0 \end{cases} .$$

The time series $\{x(k)\}$ was embedded in a 3 dimensional reconstructed state space ($\hat{p} = 3$), with unit delay τ. The 8-ary Schur-Tree was grown according to the methods described in Sec. 4. Figure 6.a shows time segments of the actual and predicted SETAR time series and the associated prediction error. Figure 6.b gives a graphical depiction of the 8-ary tree. Figure 6.c shows the the estimates of the AR vectors governing the SETAR model in each cell obtained from the recursive local orthogonalization. Note that these estimated AR vectors cluster in two directions which closely correspond to the two AR(2) regimes of the actual SETAR model (5.3).

5.4 Rössler Simulations

The discrete time Rössler system generates a chaotic measurement $x(t)$ generated by the non-linear set of differential equations :

$$\begin{cases} \frac{dx}{dt} = -y - z \\ \frac{dy}{dt} = -x + ay \\ \frac{dz}{dt} = b + xz - cz \end{cases} \tag{19}$$

where x, y, z are components of the three dimensional state vector of the Rössler system.

We simulated (19) using the following set of parameter values: $a = 0,15$, $b = 0,2$ et $c = 10$. The set of non linear coupled ordinary differential equations were numerically integrated, using an order 3 Runge-Kutta approach. The recorded time series correspond to the first coordinate $(x(t))$ of the system, sampled at a period $h = .4s$ [2]

The reconstruction dimension was varied from 2 to 5, but the reconstruction delay is maintained to a constant value $\tau = 4h$. The prediction error variance is estimated from N predicted values by

$$V = \frac{\sum_{i=1}^{N}(e_i - \overline{e}_i)^2}{\sum_{i=1}^{N}(x_i - \overline{x}_i)^2}.$$

The Schur tree was grown from phase space time series of duration N=500 and the training set consisted of 8192 phase space state vectors. Figure 7 shows the one-step forward prediction and errors for NN and Schur-Tree methods applied to the Rössler time series. Note that both NN and Schur-Tree predictors have similar trajectories although the more complex NN implementation achieves somewhat smaller prediction error. The spikes observed in the Schur-Tree prediction residuals are due to transitions between the local models in phase space.

[2] For simulating this chaotic system by numerically integrating this set of ordinary differential equations, h must be set to a much smaller value than the sampling step of the recorded time series, in order to avoid numerical instabilities. The integration was performed with a time increment $h' = h/64$

5.5 Algorithm Comparisons

A comparison of the performance of the four different one-step forward prediction methods discussed in this paper is illustrated in Figure 8 for the Chua circuit measurements and the Rössler simulation. The four methods studied are: the tree-structured predictor of Sec. 3.3 (RT), the SVD-Tree and the Schur-Tree discussed in Sec. 4, and the nearest neighbor (NN) algorithm. Note the relative performance advantage of the recursive Schur-Tree as compared to the SVD-Tree. We believe that this is due to the instability of the AR model obtained from SVD-Tree; the Schur-Tree is guaranteed to give a stable model. In all the cases the NN algorithm slightly outperforms the tree-based methods, but the improvement is obtained at a significant increase in computational burden.

6 Conclusions

We have presented a low complexity algorithm based on recursive state space partitioning for performing near-optimal non-linear prediction and identification of non-linear signals. We have also derived local SVD and Schur decomposition versions which are naturally suited to piecewise constant AR models (SETAR). These algorithms were numerically illustrated for simulated SETAR measurements, simulated chaotic measurements, and voltage measurements obtained from a Chua electronic circuit.

The tree based prediction approach presented here is related to the classification and regression tree (CART) technique [9] and adaptive tree structured vector quantization (TSVQ) [14]. The main difference is our use of a locally defined recursive SVD orthogonalization and its intrinsic applicability to piecewise linear generalizations of thresholded AR (SETAR) models [61]. Our tree-structure with SVD orthogonalization is also related to (unitary) transform coding [27], the difference being that the orthogonalization is applied locally and recursively to each splitting node. Future work will include detection of the local linearized dynamics and regularization for smoothing out model discontinuity between partition cells. A related issue for future study is how to deal with larger values of the imbedding dimension $\hat{p}$. The $2^{\hat{p}}$-ary splitting rule proposed here produces subcells of equal volume but gives a model with complexity, i.e. the number of free parameters, exponential in $\hat{p}$. Therefore, to avoid the need for unreasonably large amounts of training data $\hat{p}$ must be held as small as possible without sacrificing quantization error performance. A reasonable alternative would be to use the standard binary splitting rule for growing the model; restricting the $2^{\hat{p}}$ splitting rule to implementation of the subcell uniformity tests.

Appendix

Let $\mathbf{X}$, and Y be real vector ($\mathbb{R}^m$) and scalar valued random variables, respectively. Let the joint distribution of $\mathbf{X}, Y$ have the Lebesgue density $f_{X,Y}(\mathbf{x}, y)$. Define the marginals $f_X(\mathbf{x})$ and $f_Y(y)$ and, for any $\mathbf{x}$ satisfying $f(\mathbf{x}) > 0$, the conditional density $f_{Y|X}(y|\mathbf{x})$. Given a set $\mathcal{A}$, the conditional density function $f_{Y|X}(y|\mathbf{x})$ is said to be Lipschitz continuous of order $\alpha > 0$ almost everywhere in $\mathbf{x} \in \mathcal{A}$ (in the Hellinger metric) if there exists a finite constant $K_{\mathcal{A}}$, called a Lipschitz constant, such that for any $\mathbf{x}_1, \mathbf{x}_2 \in \mathcal{A}$ for which $f_X(\mathbf{x}_1), f_X(\mathbf{x}_2) > 0$

$$\int \left| f^{\frac{1}{2}}_{Y|X}(\mathbf{y}|\mathbf{x}_1) - f^{\frac{1}{2}}_{Y|X}(\mathbf{y}|\mathbf{x}_2) \right|^2 dy \leq K_{\mathcal{A}} \|\mathbf{x}_1 - \mathbf{x}_2\|^{\alpha}. \tag{20}$$

Lipschitz continuity of the above form is a common explicit smoothness condition assumed for probability measures and densities [34, 36]. Lipschitz continuity implies pointwise continuity of $f_{Y|X}(y|\mathbf{x})$ for almost all y [34].

For an arbitrary vector $\mathbf{x} \in \mathbb{R}^m$ and a discrete set of vectors $\mathcal{Q} = \{\mathbf{q}_1, \mathbf{q}_2, \ldots\}$ in $\mathbb{R}^m$, let Q, $Q : \mathbb{R}^m \to \mathcal{Q}$, be a vector function (a vector quantizer) operating on $\mathbf{x}$. The set of quantization cells $\{\pi_1, \pi_2, \ldots\}$ are defined as the inverse images $\{Q^{-1}(\mathbf{q}_1), Q^{-1}(\mathbf{q}_2), \ldots\}$ of elements of $\mathcal{Q}$. The following theorem provides a bound on the increase in the minimum mean square prediction error due to quantization of the predictor variables $\mathbf{x}$.

Theorem 1 *Let $\{\pi_i\}_i$ be a partition of $\mathbb{R}^m$. Assume that for each i the density $f_{Y|X}(y|\mathbf{x})$ is Lipschitz continuous of order $\alpha > 0$ almost everywhere in $\mathbf{x} \in \pi_i$ and let K_{π_i} be the associated Lipschitz constant. Assume also that $E[Y^2|\mathbf{X}] \leq m_Y^2 < \infty$ (a.s.). Then,*

$$0 \leq E\left[(Y - E[Y|Q(\mathbf{X})])^2\right] - E\left[(Y - E[Y|\mathbf{X}])^2\right] \leq 2 \max_i K_{\pi_i} m_Y^2 E[\|\mathbf{X} - Q^o(\mathbf{X})\|^{\alpha}]. \tag{21}$$

where $Q^o(\mathbf{x}) = \sum_i \xi_i I_{\pi_i}(\mathbf{x})$ and $\xi_i \in \mathbb{R}^m$ are the quantization vectors defined in Lemma 2.

The upper bound in (21) is decreasing in $\max_i K_{\pi_i}$ and equals zero when $f(y|\mathbf{x})$ is piecewise constant in $\mathbf{x}$, i.e. $f(y|\mathbf{x}) = \sum_i f(y|\xi_i) I_{\pi_i}(\mathbf{x})$ where $\xi_i \in \pi_i$ are arbitrary. Thus in this case use of quantized predictor variables do not degrade optimal prediction MSE. Also note that the upper bound in (21) is decreasing in the mean square quantization error associated with quantizing the predictor variables $E[\|\mathbf{X} - Q(\mathbf{X})\|^2]$. Bounds and asymptotic expressions exist for this quantity [29, 42] which can be used to make the bound (21) more explicit.

The following lemmas will be useful in the proof of Theorem 1.

Lemma 1 *Define the optimal predictor $\hat{\mu}_{Y|X}(\mathbf{X}) = E[Y|\mathbf{X}]$ based on the predictor variables $\mathbf{X}$. Assume that for some subset $\mathcal{A}$ of $\mathrm{I\!R}^m$ the density $f_{Y|X}(y|\mathbf{x})$ is Lipschitz continuous of order α almost everywhere in $\mathbf{x} \in \mathcal{A}$ and that $E[Y^2|\mathbf{X}] \leq m_Y^2 < \infty$ (a.s.). Then $\hat{\mu}_{Y|X}(\mathbf{x})$ is pointwise continuous almost everywhere over $\mathbf{x} \in \mathcal{A}$.*

Proof of Lemma 1

First observe that for any two functions f_1 and f_2 we have by the triangle inequality:

$$\begin{aligned} |f_1 - f_2| &= \left| f_1^{\frac{1}{2}}(f_1^{\frac{1}{2}} - f_2^{\frac{1}{2}}) + f_2^{\frac{1}{2}}(f_1^{\frac{1}{2}} - f_2^{\frac{1}{2}}) \right| \\ &\leq \left| f_1^{\frac{1}{2}} \right| \left| f_1^{\frac{1}{2}} - f_2^{\frac{1}{2}} \right| + \left| f_2^{\frac{1}{2}} \right| \left| f_1^{\frac{1}{2}} - f_2^{\frac{1}{2}} \right|. \end{aligned} \tag{22}$$

Therefore, by definition of the conditional mean, for arbitrary $\mathbf{x}_1, \mathbf{x}_2 \in \mathcal{A}$

$$\begin{aligned} |\hat{\mu}_{Y|X}(\mathbf{x}_1) - \hat{\mu}_{Y|X}(\mathbf{x}_2)| &\leq \int dy\ |y| |f_{Y|X}(y|\mathbf{x}_1) - f_{Y|X}(y|\mathbf{x}_2)| \\ &\leq \int dy\ |y| f_{Y|X}^{\frac{1}{2}}(y|\mathbf{x}_1) \left| f_{Y|X}^{\frac{1}{2}}(y|\mathbf{x}_1) - f_{Y|X}^{\frac{1}{2}}(y|\mathbf{x}_2) \right| \\ &\quad + \int dy\ |y| f_{Y|X}^{\frac{1}{2}}(y|\mathbf{x}_2) \left| f_{Y|X}^{\frac{1}{2}}(y|\mathbf{x}_1) - f_{Y|X}^{\frac{1}{2}}(y|\mathbf{x}_2) \right|. \end{aligned} \tag{23}$$

Applying the Cauchy-Schwartz inequality to the two integrals in the expression at bottom of (23)

$$\left(\int dy\ |y| f_{Y|X}^{\frac{1}{2}}(y|\mathbf{x}_1) \left| f_{Y|X}^{\frac{1}{2}}(y|\mathbf{x}_1) - f_{Y|X}^{\frac{1}{2}}(y|\mathbf{x}_2) \right| \right)^2 \leq \int dy\ |y|^2 f_{Y|X}(y|\mathbf{x}_1) \int dy \left| f_{Y|X}^{\frac{1}{2}}(y|\mathbf{x}_1) - f_{Y|X}^{\frac{1}{2}}(y|\mathbf{x}_2) \right|^2,$$

and similarly for the second integral. Hence, Lipschitz continuity of $f_{Y|X}(y|\mathbf{x})$ over $\mathbf{x} \in \mathcal{A}$ gives the bound

$$
\begin{aligned}
|\hat{\mu}_{Y|X}(\mathbf{x}_1) - \hat{\mu}_{Y|X}(\mathbf{x}_2)|^2 &\leq 2 \max_{\mathbf{x}} E[y^2|\mathbf{x}] \int dy \left| f^{\frac{1}{2}}_{Y|X}(y|\mathbf{x}_1) - f^{\frac{1}{2}}_{Y|X}(y|\mathbf{x}_2) \right|^2 \\
&\leq 2m_Y^2 K_{\mathcal{A}} \|\mathbf{x}_1 - \mathbf{x}_2\|^{\alpha}, \qquad (24)
\end{aligned}
$$

where $K_{\mathcal{A}} < \infty$ is the associated Lipschitz constant. This establishes the lemma. □

Lemma 2 *Define the optimal predictor $\hat{\mu}_{Y|Q}(\mathbf{x}) = E[Y|Q(\mathbf{X})]$ based on quantized predictor variables $Q(\mathbf{X})$. Assume that $f_{Y|X}(y|\mathbf{x})$ is Lipschitz continuous of order α almost everywhere in $\mathbf{x} \in \pi_i$ and that $E[Y^2|\mathbf{X}] \leq m_Y^2 < \infty$ (a.s.). Then for any quantization cell $\pi_i \subset \mathrm{I\!R}^m$ there exists a point $\xi_i \in \pi_i$ such that*

$$\hat{\mu}_{Y|Q}(\mathbf{x}) = \hat{\mu}_{Y|X}(\xi_i), \quad \forall \mathbf{x} \in \pi_i.$$

Furthermore, the point ξ_i satisfies the equation

$$\int_{\pi_i} d\mathbf{x}\, \hat{\mu}_{Y|X}(\mathbf{x}) f(\mathbf{x}) = \hat{\mu}_{Y|X}(\xi_i) P_X(\pi_i),$$

where $P_X(\pi_i) = \int_{\pi_i} f(\mathbf{x}) d\mathbf{x}$.

Proof of Lemma 2

By definition of conditional expectation: $\hat{\mu}_{Y|Q}(\mathbf{x}) = \int dy\ y f_{Y|Q}(y|Q(\mathbf{x}))$ where

$$f_{Y|Q}(y|Q(\mathbf{x})) = \int_{\pi_i} d\mathbf{x}\ f_{Y|X}(y|\mathbf{x}) f(\mathbf{x}) / P_X(\pi_i), \quad \mathbf{x} \in \pi_i$$

is the conditional density of Y given $Q(\mathbf{X}) = Q(\mathbf{x})$. Invoking Fubini's theorem [53] to permute the order of integration, we obtain the Lebesgque-Steiltjes integral representation

$$
\begin{aligned}
\hat{\mu}_{Y|Q}(\mathbf{x}) &= \frac{1}{P_X(\pi_i)} \int_{\pi_i} d\mathbf{x}\ f(\mathbf{x}) \int dy\ y f_{Y|X}(y|\mathbf{x}) \\
&= \frac{1}{P_X(\pi_i)} \int_{\pi_i} dP(\mathbf{x})\ \hat{\mu}_{Y|X}(\mathbf{x}), \quad \mathbf{x} \in \pi_i
\end{aligned}
$$

where $dP(x) = f(x)dx$. By Lemma 1 $\hat{\mu}_{Y|X}(\mathbf{x})$ is continuous and therefore, by the mean value theorem for Lebesgue-Steiltjes integrals [53], there exists a point $\xi_i \in \pi_i$ such that

$$\frac{1}{P_X(\pi_i)} \int_{\pi_i} dP(\mathbf{x}) \hat{\mu}_{Y|X}(\mathbf{x}) = \hat{\mu}_{Y|X}(\xi_i), \quad \mathbf{x} \in \pi_i.$$

This establishes the Lemma. □

Proof of Theorem 1

Define

$$\Delta^2 \overset{\text{def}}{=} E\left[(Y - E[Y|Q(\mathbf{X})])^2\right] - E\left[(Y - E[Y|\mathbf{X}])^2\right].$$

That $\Delta^2 > 0$ follows directly from the fact that the conditional mean estimator $E[Y|\mathbf{X}]$ minimizes mean square prediction error. Next we deal with the right hand side of the inequality (21). It is easily verified by iterated expectation that $E\left[(Y - E[Y|Q(\mathbf{X})])E[Y|Q(\mathbf{X})]\right] = 0$ and $E\left[(Y - E[Y|\mathbf{X}])E[Y|\mathbf{X}]\right] = 0$ (orthogonality principle of non-linear estimation). Therefore

$$\begin{aligned}
\Delta^2 &= E\left[(Y - E[Y|Q(\mathbf{X})])Y\right] - E\left[(Y - E[Y|\mathbf{X}])Y\right] \\
&= E\left[(E[Y|\mathbf{X}] - E[Y|Q(\mathbf{X})])\, Y\right] \\
&= E\left[(E[Y|\mathbf{X}] - E[Y|Q(\mathbf{X})])\, E[Y|\mathbf{X}]\right].
\end{aligned}$$

Thus, by Fubini [8], we have the integral representation

$$\begin{aligned}
\Delta^2 &= \int d\mathbf{x} \, \left[\hat{\mu}_{Y|X}(\mathbf{x}) - \hat{\mu}_{Y|Q}(\mathbf{x})\right] \hat{\mu}_{Y|X}(\mathbf{x}) f(\mathbf{x}) \\
&= \sum_i \int_{\pi_i} d\mathbf{x} \, \left[\hat{\mu}_{Y|X}(\mathbf{x}) - \hat{\mu}_{Y|Q}(\mathbf{x})\right] \hat{\mu}_{Y|X}(\mathbf{x}) f(\mathbf{x}), \qquad (25)
\end{aligned}$$

where the $\hat{\mu}$ quantities are defined as in Lemmas 1 and 2. Invoking the latter lemma, there exists a point $\xi_i \in \pi_i$ such that $\hat{\mu}_{Y|Q}(\mathbf{x}) = \hat{\mu}_{Y|X}(\xi_i)$, $\mathbf{x} \in \pi_i$ and $\int_{\pi_i} d\mathbf{x} \, [\hat{\mu}_{Y|X}(\mathbf{x}) - \hat{\mu}_{Y|X}(\xi_i)] f(\mathbf{x}) = 0$. Therefore, from (25)

$$\Delta^2 = \sum_i \int_{\pi_i} d\mathbf{x} \, \left|\hat{\mu}_{Y|X}(\mathbf{x}) - \hat{\mu}_{Y|X}(\xi_i)\right|^2 f(\mathbf{x}).$$

Application of the bound (24) on $|\hat{\mu}_{Y|X}(\mathbf{x}_1) - \hat{\mu}_{Y|X}(\mathbf{x}_2)|^2$ obtained in the course of proving Lemma 1 yields

$$\begin{aligned}\Delta^2 &\leq 2m_Y^2 \max_i K_{\pi_i} \sum_i \int_{\pi_i} d\mathbf{x}\, \|\mathbf{x} - \xi_i\|^\alpha f(\mathbf{x}) \\ &= 2m_Y^2 \max_i K_{\pi_i}\, E\left[\|\mathbf{X} - Q^o(\mathbf{X})\|^\alpha\right]\end{aligned}$$

where $Q^o(\mathbf{x}) = \sum_i \xi_i I_{\pi_i}(\mathbf{x})$. □

References

[1] H.Abarbanel, R.Brown, J.Sidorowitch, L.Tsimring, " The Analysis of Observed Chaotic Data in Physical Systems",*Rev. Mod. Phys.*, vol.65(4), pp.1331-1391, 1990.

[2] H. Abarbanel, "Analysis of Observed Chaotic Data," Institute for Nonlinear Science, Springer-Verlag, NY 1996.

[3] M. Abromowitz and I. Stegun, "Handbook of Mathematical Functions," Dover, New York, NY, 1977.

[4] M. Basseville, "Distances Measures for Signal Processing and Pattern Recognition", *Signal Processing*, vol.18, pp 349-369.

[5] J. Bentley, "Multidimensional binary search trees used for associative searching," *Communic. Assoc. Comput. Mach.*, vol. 18, pp. 509–517, 1975.

[6] A.E. Badel, O. Michel, A.O. Hero, " Arbres de Régression : Modélisation Non Paramétrique et Analyse des Séries Temporelles" *Revue de Traitement du Signal*, Vol. 14, No. 2, pp. 117-133, June 1997.

[7] P. J. Bickel and K. A. Doksum, "Mathematical Statistics: Basic Ideas and Selected Topics", Holden-Day, San Francisco (CA), 1977.

[8] P. Billingsley, *Probability and Measure*, Wiley, New York, 1979.

[9] L. Breiman, J.H.Friedman, R.A. Olshen, C.J. Stone, "Classification and Regression Trees," Wadsworth Advanced Books and Software, 1984.

[10] A.E. Badel, L. Mercier, D. Guegan, O. Michel, “Comparison of several methods to predict chaotic time series.”, Proceedings of IEEE-ICASSP’97, Munich, Vol.V, pp.3793-3796, April 1997.

[11] M. Casdagli, T. Sauer, J.A Yorke, “Embedology”, *J. Stat. Phys*, vol.**65**, 1991, pp.579-616

[12] “Nonlinear Modeling and Forecasting”, M. Casdagli and S. Eubank Ed., Proceedings Volume in the Santa Fe Institute Studies for the Sciences of Complexity, Vol. **12**, 1991.

[13] L. Clarke and D. Pregibon, “Tree-based models,” in *Statistical Models in S*, J. Chambers and T. Hastie, editors, pp. pp. 377–419, Wadsworth, 1992.

[14] P.A. Chou, T. Lookabaugh, R.M. Gray, “Optimal Pruning with Applications to Tree-Structures Source Coding and Modeling,” *IEEE Trans on Inf. theory*, vol.35, No.2, pp 299-315, 1989.

[15] W. Cleveland, E. Grosse, and W. Shyu, “Local regression models,” in *Statistical Models in S*, J. Chambers and T. Hastie, editors, pp. pp. 309–376, Wadsworth, 1992.

[16] H. A. David, “Order Statistics”, Wiley, New York, 1981.

[17] L.Devroye, Lásló Györfi, Gábor Lugosi, “ A Probabilistic Theory of Pattern Recognition”, Application of Math. Series, vol.31, Springer Verlag, 1996.

[18] R.F. Engle, “Autoregressive conditional heteroscedasticity with estimates of the variance of United Kingdom inflation”, *Econometrica*, vol.50, pp.987-1007, 1982.

[19] J.P. Eckmann and D. Ruelle, “Ergodic Theory of Chaos and Strange Attractors,” *Rev. Mod. Phys.*, Vol. 57, No. 3, pp. 617–656, 1985.

[20] J.D.Farmer, J.Sidorowitch, " Predicting Chaotic Time Series", *Phys. Rev. Lett.*, 59:845, 1987

[21] J.D.Farmer, J.Sidorowitch, "Exploiting Chaos to Predict the Future and Reduce Noise", in Evolution, Learning and Cognition, Y.C.Lee Ed., World Scientific, Singapore 1988, 277-330.

[22] A.M. Fraser, "Information and Entropy in Strange Attractors," *IEEE Trans on Inf. theory*, vol.35, No.2, pp 245-262, 1989.

[23] A.M. Fraser, H.L. Swinney, "Independent Coordinates for Strange Attractors from Mutual Information," *Phys. Rev. A*, Vol.33, No.2, pp.1134-1140, 1981.

[24] A.M. Fraser, "Modeling NonLinear TimeSeries," ICASSP'92, vol.**5** San Francisco, 1992, pp.V313-V317.

[25] A.M. Fraser, A. Dimitriadis, "Forecasting Probability Densisty by Using Hidden Markov Models with mixed States," Time Series Prediction, A.S. Weigend, N.A.Gershenfeld ed., Proceeding volume of the Santa Fe institute, vol.XV, pp.265-282.

[26] K. Fukunaga, "Statistical Pattern Recognition (2nd Ed), Academic Press, San Diego (CA), 1990.

[27] A. Gersho, R.M. Gray, "Vector Quantization and Signal Compression", Kluwer Academic Press, 1992.

[28] P.Grassberger, I.Procaccia, "Measuring the Strangeness of Strange Attractors," *Physica D*, Vol.9, pp.189-208, 1983.

[29] R. M. Gray, *Source Coding Theory*, Kluwer Academic, Norwell MA, 1990.

[30] D.Guegan, “On the identification and prediction of nonlinear models”, Proceedings of the Workshop on New Directions in Time Series Analysis, Springer Verlag, 1992.

[31] D.Guegan, Séries Chronologiques Non Linéaires à temps discret,” Statistique Mathématique et Probabilité, Economica, Paris, 1994.

[32] S.Haykin, J.Principe “Using neural networks to dynamically model chaotic events such as sea clutter; Making sense of a complex world”, *IEEE Signal Processing Magazine*, 66:81, Mai 1998.

[33] D.R.Hush, B.G.Horne “ Progress in Supervised Neural Networks”, *Signal Processing Magazine*, IEEE, 8:38, janvier 1993.

[34] I. A. Ibragimov and R. Z. Has'minskii, *Statistical estimation: Asymptotic theory*, Springer-Verlag, New York, 1981.

[35] H. Kantz and T.Schreiber, “Nonlinear Time Series Analysis”, Cambridge Nonlinear Science Series 7, Cambridge University Press 1997.

[36] L. LeCam, *Asymptotic Methods in Statistical Decision Theory*, Springer-Verlag, New York, 1986.

[37] E.L. Lehmann, “Testing Statistical Hypotheses”, Wiley, New York, 1959.

[38] W. Liebert, K. Pawelzik, H.G. Schuster, “ Optimal embedding of Chaotic Attractors from topological Considerations”, *Europhysics Letters*, Vol.14, No.6, pp.521-526, 1991.

[39] A. Mead, R.D. Jones et al., “Prediction of Chaotic Time Series Using CNLS-Net-Example: The mackey-Glass Equation,”, in “Nonlinear Modeling and Forecasting”,

M.Casdagli and S.Eubank Ed., Proc. of the Santa Fe Institute vol.**12**, Addison Wesley, 1992, pp.39-72. .

[40] O. Michel, P. Flandrin, "Higher Order Statistics For Chaotic Signal Processing," Control and Dynamic Systems, Vol.75, pp.105-153, Academic Press, 1996.

[41] A.M. Mood, F.A. Graybill, D.C. Boes, "Introduction to the Theory of Statistics," Mc Graw Hill International Editions, Statistics Series, 3rd ed. 1974.

[42] D. N. Neuhoff, "On the asymptotic distribution of the errors in vector quantization," *IEEE Trans. on Inform. Theory*, vol. IT-42, pp. 461–468, March 1996.

[43] A. Nobel, "Vanishing distortion and shrinking cells," *IEEE Trans. on Inform. Theory*, vol. IT-42, no. 4, pp. 1303–1305, 1996.

[44] A. Nobel, "Recursive partitioning to reduce distortion," *IEEE Trans. on Inform. Theory*, vol. IT-43, no. 4, pp. 1122–1133, 1997.

[45] A. Nobel and R. Olshen, "Termination and continuity of greedy growing for tree-structured vector quantizers," *IEEE Trans. on Inform. Theory*, vol. IT-42, no. 1, pp. 191–205, 1996.

[46] M. Orchard and C. Bouman, "Color quantization of images," *IEEE Trans. on Signal Processing*, vol. SP-39, no. 12, pp. 2677–2690, 1991.

[47] K. Perlmutter, S. Perlmutter, R. Gray, R. Olshen, and K. Oehler, "Bayes risk weighted vector quantization with posterior estimation for image compression and classification," *IEEE Trans. on Image Processing*, vol. IP-5, no. 2, pp. 347–360, 1996.

[48] A. Papoulis, "Probability, Random Variables, and Stochastic Processes", McGraw-Hill Int. Editions, 2nd edition, 1984.

[49] T. S. Parker, L. O. Chua, "Practical Numerical Algorithms for Chaotic Systems", Springer Verlag, 1989.

[50] W.H. Press, B.P. Flannery, S.A. Teukolsky, W.T. Vetterling, "Numerical recipes in C", Cambridge University Press, 1989.

[51] M.B. Priestley, "State Dependant models: a general approach to nonlinear time series analysis", *Journal of Time Series Analysis*, vol.1, pp.47-71, 1980.

[52] M.B. Priestley, "Non Linear and Non Stationary Time Series Analysis," Academic Press, San Diego, 1988.

[53] F. Riesz and B. Sz.-Nagy, *Functional analysis*, Ungar, New York, 1955.

[54] T. Sauer, "A noise reduction method for signal from nonlinear systems.", *Physica D*, vol.**58**, pp.193-201, 1992.

[55] M.R. Segal, "Tree structured methods for longitudinal data," Journ. Amer. Stat. Assoc. (JASA), vol. 87, pp. 407-418, May 1992.

[56] L. L. Scharf, "Statistical Signal Processing, Detection, Estimation and Time Series Analysis", Addison-Wesley Pub. Co., 1990.

[57] R. Shaw, "Strange Attractors, Chaotic Behavior and Information Flow", in *Zeitschrift in NaturForschung*, vol.36A, No.1, pp.80-112, 1981.

[58] J.N. Sonquist and J.N. Morgan, "The detection of interaction effects," Monograph 35, Survey Research Center, Institute for Social Research, University of Michigan, 1964.

[59] F. Takens, “Detecting Strange Attractors in Turbulence,” *Lecture Notes in Mathematics*, Vol.898, pp.366-381, 1981.

[60] H. Tong “Threhold models in nonlinear time series analysis”, *Lecture Notes in Statistics*, vol.21, Springer Verlag, 1983.

[61] H. Tong, “Non Linear Time Series : a Dynamical system Approach,” Oxford Science Publication, Oxford University Press, NY 1990.

[62] G.Wahba, “Multivariate Function and Operator Estimation, Based on Smoothing Splines and Reproducing Kernels,” in “Nonlinear Modeling and Forecasting”, M.Casdagli and S.Eubank Ed., Proc. of the Santa Fe Institute vol.**12**, Addison Wesley, 1992, pp.95-112.

[63] T.P. Weldon, “An Inductorless Double-Scroll Chaotic Circuit”, *American Journal of Physics*, vol. 58, No.10, pp.936-941, 1990.

[64] “Time Series Prediction : Forecasting the Future and Understanding the Past”, A.S.Weigend and N.A.Gershenfeld Ed., Proceedings Volume in the Santa Fe Institute Studies for the Sciences of Complexity, Vol. **15**, 1992.

[65] H. Whitney, “Differentiable Manifolds”, *Annals of Mathematics*, Vol.37 No.3, pp.645-680, 1936.

[66] H. Zhang, “ Classificationtrees for multiple binary responses,” Journ. Amer. Stat. Assoc. (JASA), vol. 93, No. 441, pp 180-193, Mar. 1998.

FIGURE CAPTIONS

Figure 1: Graphical depiction of the tree growing algorithm using separable 2^p-ary splitting rule. For a $\hat{p} = 2$ dimensional state space embedding the tree is a quad-tree. The root-node is split into 4 subcells and the sample distribution of points is found be non-uniform. Among the derived subsets, only the one depicted by the lower left corner square is found to be non-uniform and is split further.

Figure 2: Family of curves describing cell subdivision stopping rule in terms of minimum number of points N_Π falling into a rectangular cell π and the probability criterion $\epsilon \in [0,1]$. Vertical axis is the minimum number of points that will be assigned to a subdivided cell and horizontal axis is the log of ϵ.

Figure 3: Tree-structured predictor for separable splitting rule applied to a piecewise linear phase space trajectory in 2 dimensions. (a) Simulated state space trajectory in 2 dimensions, with superimposed rectangular partition produced by recursive tree (RT) growing algorithm. (b) Representation of the quad-tree associated to the state space trajectory depicted in (a).

Figure 4 : (a) Same simulated state space trajectory as in Fig. 3 but with recursive SVD-Tree partitioning. (b) Representation of the SVD-tree associated to the state space trajectory depicted in (a). (c) Pairs of estimated (normalized) AR coefficients governing the dynamics in each cell are plotted with lengths proportional to the occupancy rate (number of points) of the cell.

Figure 5 : One step forward predictor for the sampled output of the Chua electronic circuit: (a) the SVD-Tree algorithm; (b) the nearest neighbor algorithm.

Figure 6: SETAR time series from equation (5.3): (a) one step forward predictor trajectory and prediction errors obtained from Schur-Tree algorithm; (b) 8-ary tree constructed from a 3 dimensional state phase space using Schur-Tree algorithm; (c) unit-length AR direction vectors.

Figure 8: Normalized prediction error variance as a function of the reconstruction dimension for: (a) the voltage output of a Chua electronic circuit; and (b) the simulated Rössler time series.

Figure 7: simulated time series, one step forward predicted values and prediction errors for the first coordinate of the Rössler system ($\hat{p} = 3$) using: (a) NN algorithm; and (b) Schur-Tree algorithm.

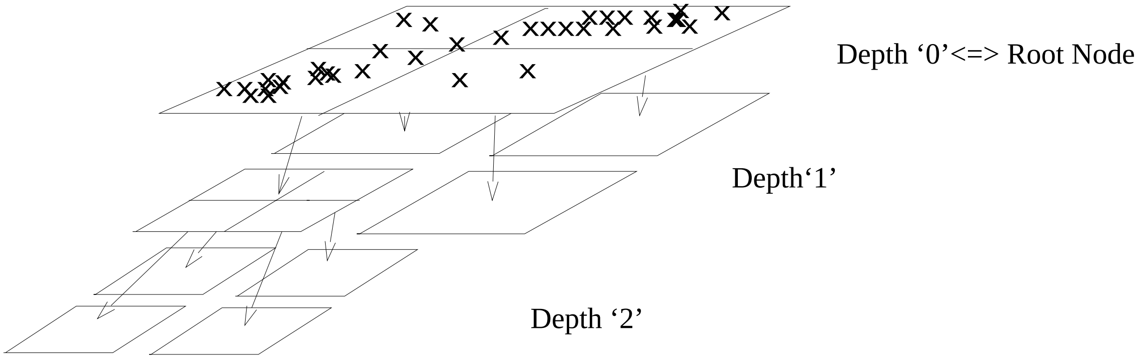


Figure 1:

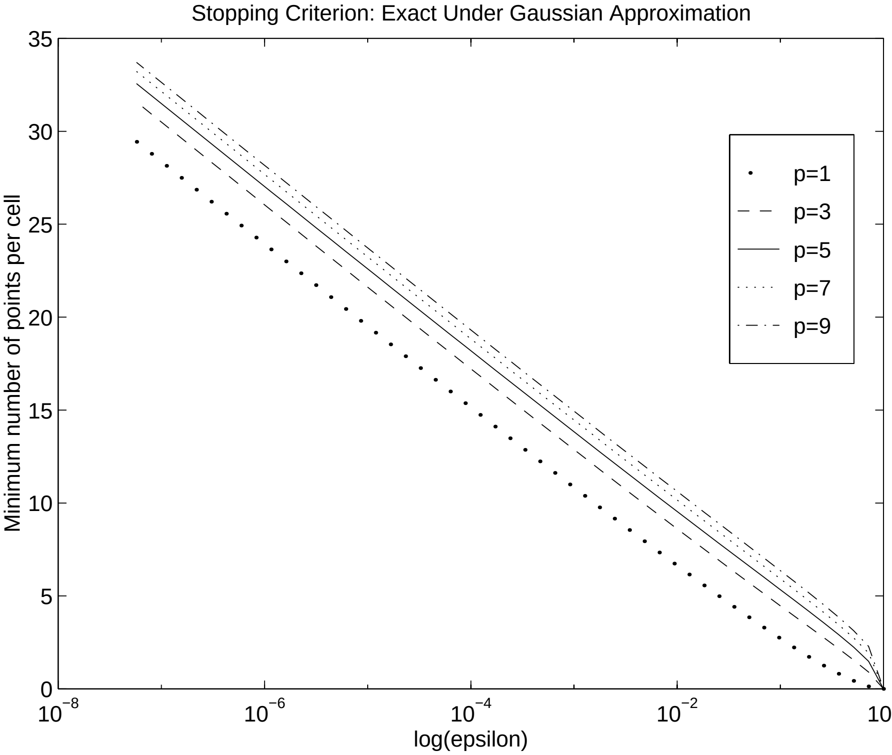
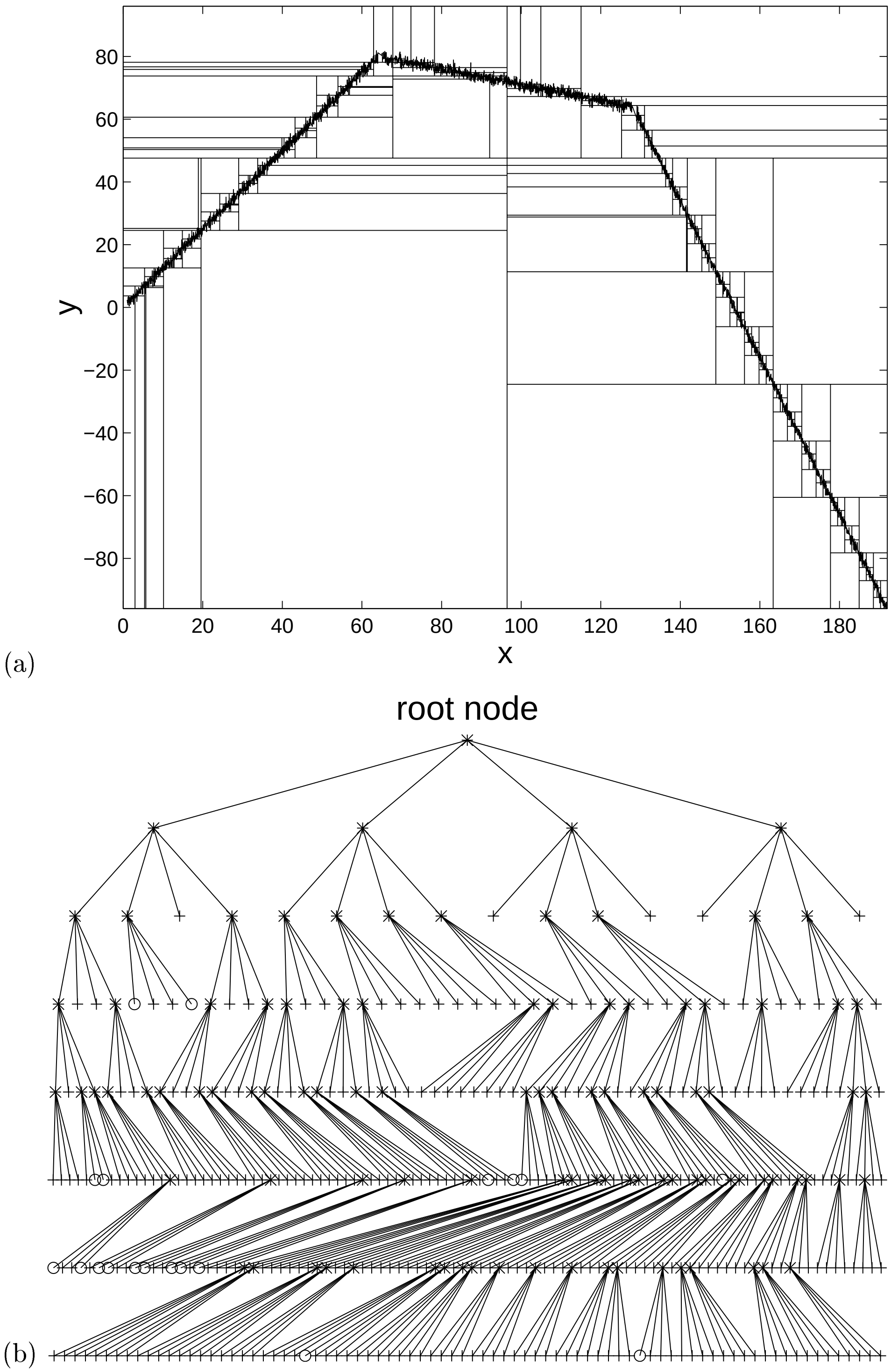


Figure 2:



(a)
y
80
60
40
20
0
−20
−40
−60
−80
0
20
40
60
80
100
120
140
160
180
x
(b)
root node

Figure 3:

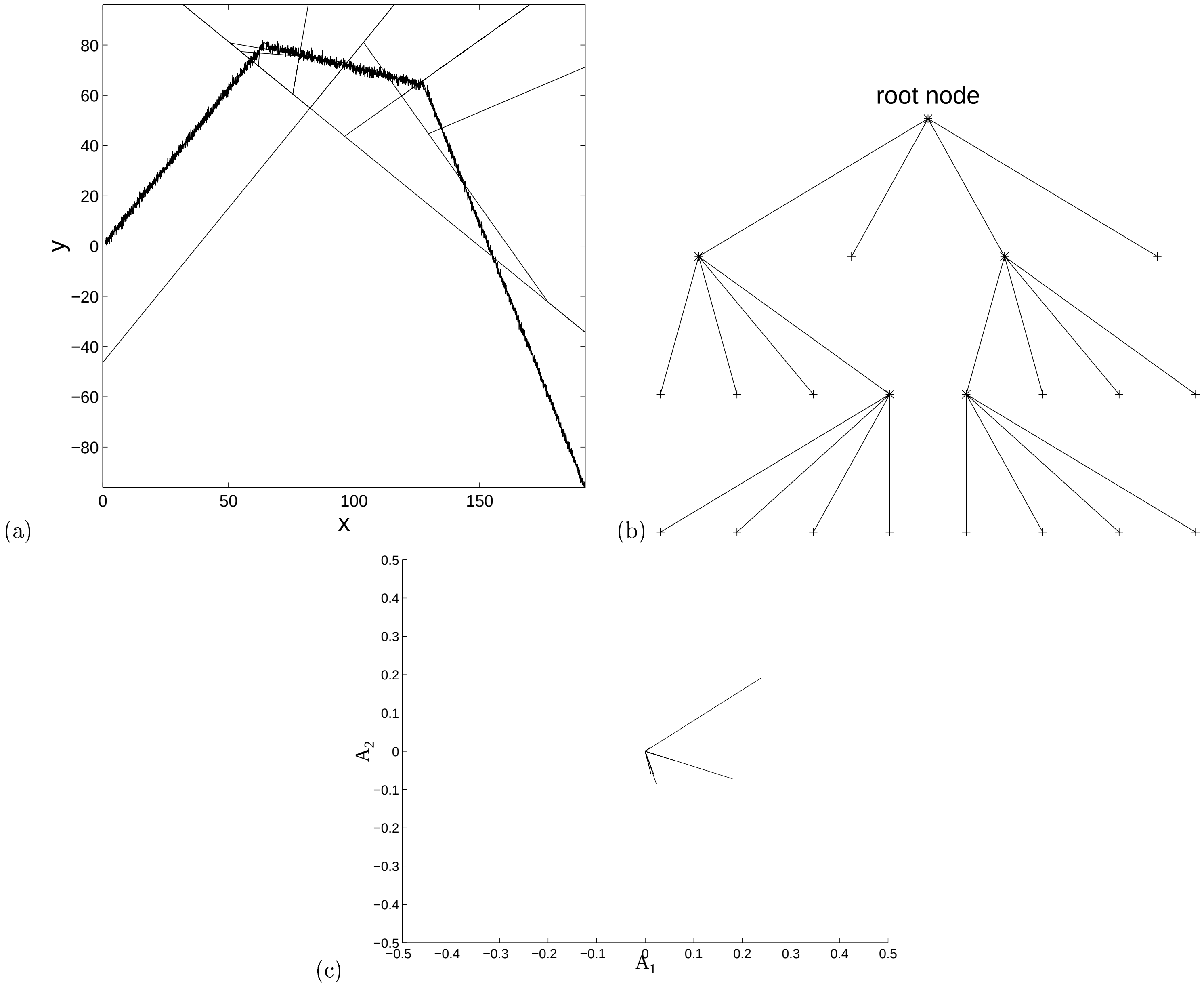
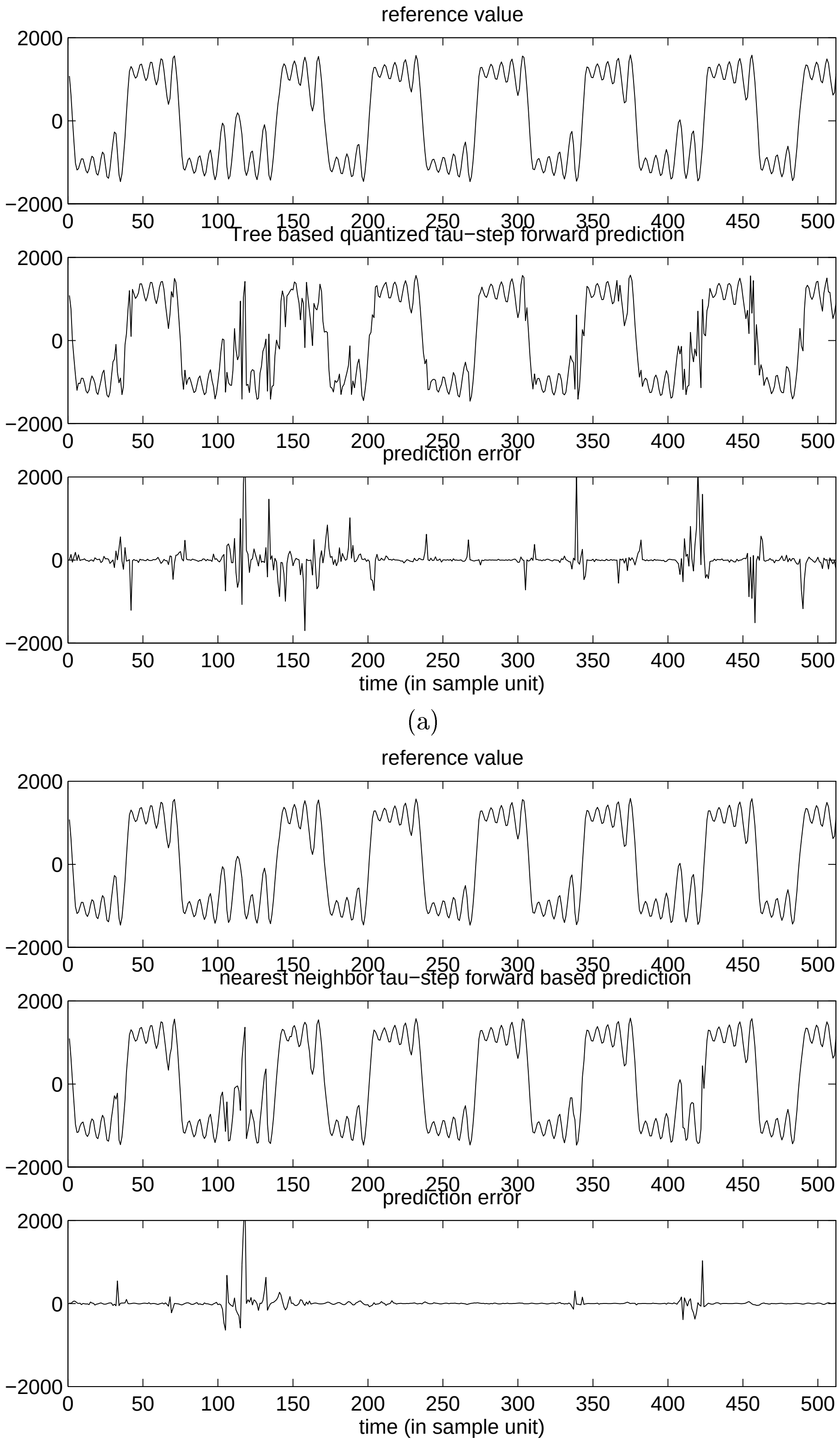

(a)
y
80
60
40
20
0
−20
−40
−60
−80
0
50
100
150
x
(b)
root node
(c)
A2
0.5
0.4
0.3
0.2
0.1
0
−0.1
−0.2
−0.3
−0.4
−0.5
−0.5
−0.4
−0.3
−0.2
−0.1
0
0.1
0.2
0.3
0.4
0.5
A1

Figure 4:


reference value
2000
0
−2000
0
50
100
150
200
250
300
350
400
450
500
Tree based quantized tau−step forward prediction
2000
0
−2000
prediction error
2000
0
−2000
time (in sample unit)
reference value
nearest neighbor tau−step forward based prediction
prediction error
time (in sample unit)

(a)

(b)

Figure 5:

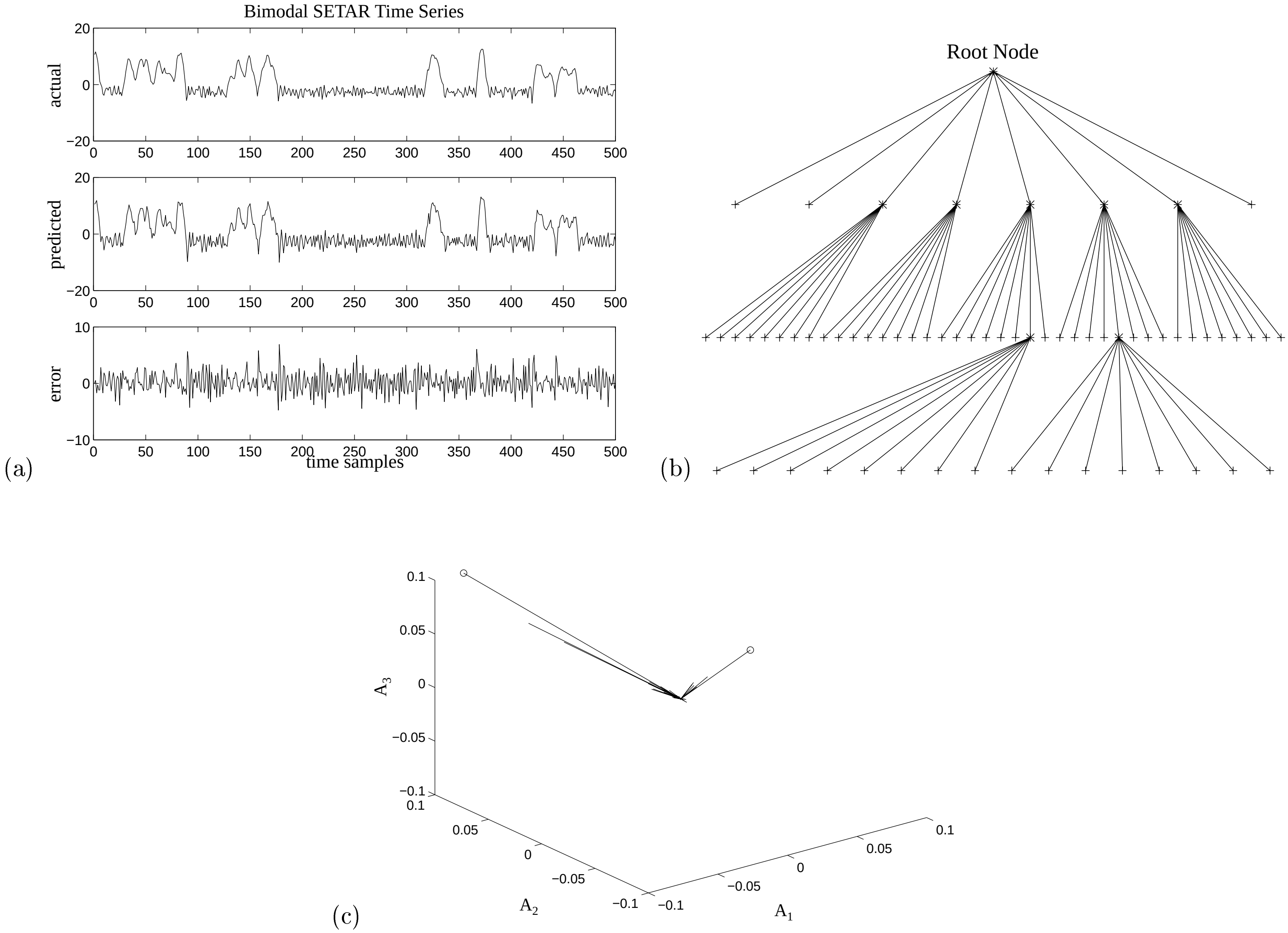
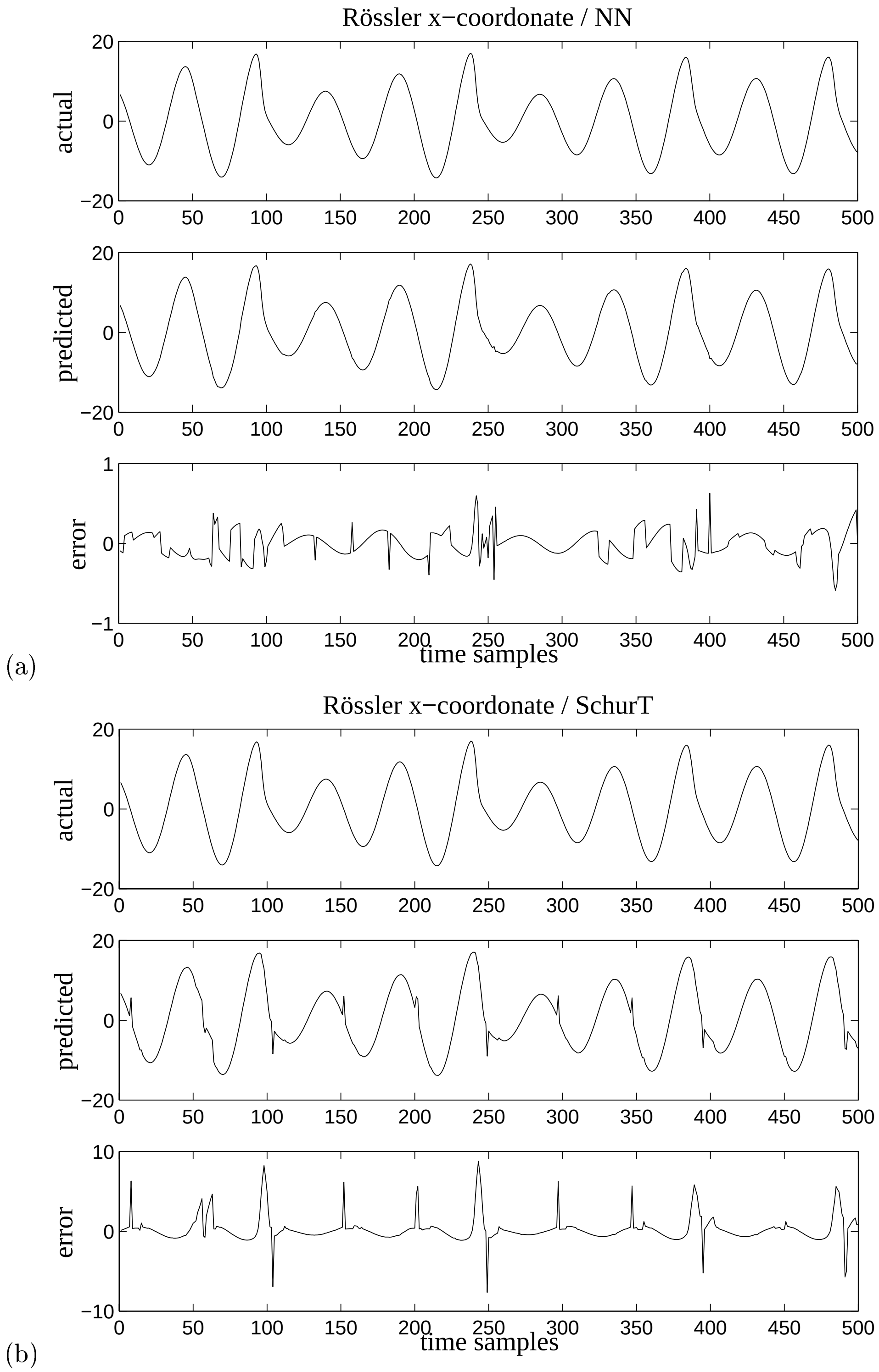

Bimodal SETAR Time Series
actual
predicted
error
time samples
20
0
−20
10
−10
0
50
100
150
200
250
300
350
400
450
500
(a)
Root Node
(b)
A_1
A_2
A_3
0.1
0.05
0
−0.05
−0.1
(c)

Figure 6:



Rössler x−coordonate / NN

Rössler x−coordonate / SchurT

Figure 7:

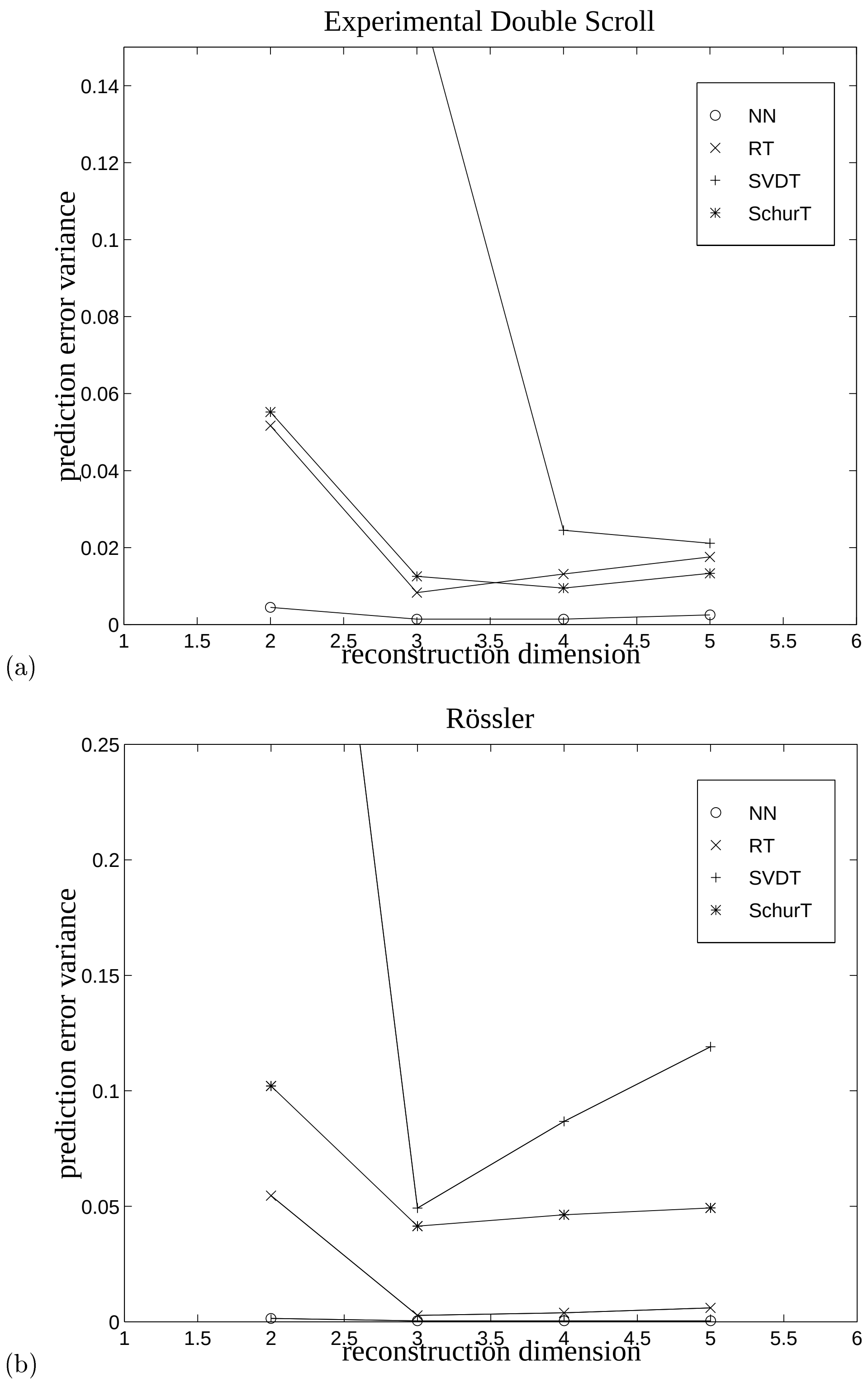

Experimental Double Scroll
prediction error variance
0.14
0.12
0.1
0.08
0.06
0.04
0.02
0
1
1.5
2
2.5
3
3.5
4
4.5
5
5.5
6
reconstruction dimension
NN
RT
SVDT
SchurT
(a)
Rössler
prediction error variance
0.25
0.2
0.15
0.1
0.05
0
1
1.5
2
2.5
3
3.5
4
4.5
5
5.5
6
reconstruction dimension
NN
RT
SVDT
SchurT
(b)

Figure 8:

Biographies

ALFRED O. HERO, III, (S '79, M '84, SM '96, F '97) was born in Boston, MA, USA in 1955. He received the B.S. (summa cum laude) from Boston University (1980) and the Ph.D. from Princeton University (1984), both in electrical engineering. He held the G.V.N. Lothrop Fellowship in Engineering at Princeton University. Since 1984 Alfred Hero has been with the Dept. of Electrical Engineering and Computer Science, University of Michigan - Ann Arbor, where he is currently Professor and Director of the Communications and Signal Processing Laboratory. He has held positions of Visiting Scientist at MIT Lincoln Laboratory, Lexington, MA (1987-89); Visiting Professor at l'Ecole Nationale de Techniques Avancees (ENSTA), Paris, France (1991); and William Clay Ford Fellow at the Ford Motor Company, Dearborn, MI (1993). He has served as consultant for US government agencies and private industry. His present research interests are in the areas of detection and estimation theory, statistical signal and image processing, statistical pattern recognition, signal processing for communications, channel equalization and interference mitigation, spatio-temporal sonar and radar processing, and biomedical signal and image analysis.

Alfred Hero is a Fellow of the IEEE, a member of Tau Beta Pi, the American Statistical Association, the New York Academy of Science, and Commission C of the International Union of Radio Science (URSI). In 1995 he received a Research Excellence Award from the College of Engineering at the University of Michigan. In 1999 he received a Best Paper Award from the IEEE Signal Processing Society. He was associate editor for the IEEE Transactions on Information Theory (1994-97); Chair of the IEEE SPS Statistical Signal and Array Processing Technical Committee (1996-98); and Treasurer of the IEEE SPS Conference Board (1997-2000). He was co-chair for the 1999 IEEE Information Theory Workshop and the 1999 IEEE Workshop on Higher Order Statistics. He served as publicity chair for the 1986 IEEE International Symposium on Information Theory and was general chair of the 1995 IEEE International Conference on Acoustics, Speech, and Signal Processing. He received the 1999 Meritorious Service Award from the IEEE Signal Processing Society.

OLIVIER J.J.MICHEL (S'84, M'85) was born in Mont Saint Martin, France, in 1963. He completed his studies at Ecole Normale Supérieure de Cachan, in the department of Applied Physics, where he received the "Agrégation de Physique" in 86. He received a Ph.D degree from University Paris-XI Orsay in 91, in signal processing. In 91, he joined the physics department at Ecole Normale Supérieure de Lyon, France, as an assistant professor. His research interest include non stationnary spectral analysis, array processing, non linear time series problems, information theory and dynamical systems studies, in close relationship with physical experiments in the field of chaos and hydrodynamical turbulence.

ANNE EMMANUELLE BADEL (S'92, M'93) was born in Lyon, France, in 1971. She completed her studies at Ecole Normale Supérieure de Lyon, France, in the physics department. She completed the Agrǵation de Physique in 1994, and completed a PhD in Physics at ENS-Lyon in 1998. Her research interests are in non linear time series analysis. in Physics in 1998, at ENS-Lyon.