NONLINEAR FILTERING BY KRIGING, WITH APPLICATION TO SYSTEM INVERSION

J.-P. Costa, L. Pronzato and E. Thierry

Laboratoire I3S, CNRS-UNSA, Les Algorithmes/Bât. Euclide,
2000 route des Lucioles, Sophia-Antipolis
06410 Biot, FRANCE

ABSTRACT

Prediction by kriging does not rely on any specific model structure, and is thus much more flexible than approaches based on parametric behavioural models. Since accurate predictions are obtained for extremely short training sequences, it generally performs better than prediction methods using parametric models. Application to nonlinear system inversion is considered

1. INTRODUCTION

We consider the situation where the input and output sequences $\{x_k\}$ and $\{y_k\}$ of a SISO nonlinear system S are observed on a given horizon n, and the response of the system to future inputs must be predicted. Such a situation is rather common, and finds applications in many signal processing problems.

Usually, one builds an input/output model for S and uses this model to predict its response for new values of the input. When no prior knowledge on S is available, a behavioural model is used.

The Volterra and Wiener functional series [6], or the NARMAX (Nonlinear AutoRegressive Moving Average model with eXogenous inputs) model proposed in [4], are traditional parametric representations for nonlinear systems of unknown structure. The inclusion of information from both lagged inputs and outputs in the NARMAX model gives a lot of flexibility, but the choice of the characteristics of the structure (polynomial degree D, memory lengths m_x and m_y of the input and output signals) is rather difficult. Moreover, the number p of parameters of the model, determined by D, m_x and m_y, is generally very large, which makes their estimation difficult (long data sets are required).

The parametric approaches mentioned above rely on *a priori* choices, which strongly influence the quality of the model obtained. Kriging, which originated in geostatistics, see e.g., [3, 2], is a statistical tool for modeling spatial observations, with or without observation errors, and does not rely explicitly on any specific structure. To the best of our knowledge, it has never been used in a signal processing context to solve problems such as system inversion.

The kriging approach is described in Section 2. It can be called semi–parametric since the model contains a linear regression part (parametric), and a non–parametric part considered as the realization of a random process. The covariance matrix of this process is parameterized and, assuming that the process is Gaussian, the parameters of the covariance are estimated by maximum likelihood. The memory length m_x of the input is the only important prior choice concerning the structure, and a prior over–estimation of m_x only results in heavier computations [1]. This makes the approach especially attractive when the structure of nonlinear system involved is totally unknown.

2. SEMI–PARAMETRIC MODELING BY KRIGING

Let S be a system with output y depending in an unknown (possibly nonlinear) manner on a vector $\mathbf{x}$ of inputs. Once inputs $\{\mathbf{x}_k\}$ and associated outputs $\{y_k\}$ are observed, $k = 1, \ldots, n$, we predict the value of y at new unsampled values of $\mathbf{x}$, that is $\mathbf{x}_{n+1}, \mathbf{x}_{n+2}, \ldots$, by interpolating previous data by the best linear unbiased predictor. Note that although we are predicting *future* responses $y_{n+1}, y_{n+2}, \ldots$, we are not approximating a function of the time index k from observed values at $k = 1, \ldots, n$. This is crucial, since the approach has rather bad *extrapolating* properties, see, e.g., [8]. On the other hand, prediction by kriging has nice *interpolating* properties: we only need the training sample $\{\mathbf{x}_k, y_k\}$, $k = 1, \ldots, n$, to be representative of the data to be predicted. Kriging then yields accurate predictions even for extremely short training sequences, which does not seem to be the case for neural networks.

Consider first the case of a deterministic system (no observation errors), where

$$y_k = y(\mathbf{x}_k) = F(\mathbf{x}_k)\,, \tag{1}$$

with $F(\cdot)$ an unknown nonlinear function, and $\mathbf{x}_k$ the vector formed by lagged scalar inputs, that is:

$$\mathbf{x}_k = (x_k, x_{k-1}, \ldots, x_{k-m_x+1})^T\,.$$