Nonlinear state space model identification using a regularized basis function expansion

Andreas Svensson*, Thomas B. Schön*, Arno Solin**, Simo Särkkä***

*Department of Information Technology, Uppsala University. **Department of Neuroscience and Biomedical Engineering. ***Department of Electrical Engineering and Automation, Aalto University

The short story

To learn nonlinear models for dynamical systems without making strong assumptions about the functional form of f and g, we use a basis function expansion f(x) ≈ ∑ωμ(Ψ(x)), where the weights ωμ are to be learned. Such a basis can, e.g., be the Fourier basis. This gives a model able to describe a broad class of functions f and g:

\[
x_{k+1} = \begin{bmatrix} 1 & \cdots & 1 \end{bmatrix} A \begin{bmatrix} \mu(\phi(x_1)) \\ \vdots \\ \mu(\phi(x_n)) \end{bmatrix} + \nu_{t}, \\
y_{k} = \begin{bmatrix} \omega(x) \end{bmatrix} \begin{bmatrix} \mu(\phi(x_1)) \\ \vdots \\ \mu(\phi(x_n)) \end{bmatrix} + \epsilon_{t}. \tag{1}
\]

This model is nonlinear in its states, requiring particle methods (or similar), but linear in its parameters, allowing for efficient parameter estimation (A, C, Q, R, learning).

Basis function expansions in a state space model

A general structure for a state space model is

\[
x_{k+1} = f(x_k) + \nu_{t}, \\
y_{k} = g(x_k) + \epsilon_{t},
\]

where with x_k ∈ ℝ^n and y_k being stochastic noise. We observe y_{1:T} (not x_{1:T}), and an exogenous input can also be included. We assume Gaussian noise with E[ν_t] = 0 and E[ε_t] = 0.

The goal is to learn f, g, Q and R from data, without making strong assumptions about the functional form of f and g.

As a flexible model for f and g, we use a basis function expansion

\[
f(x) = \sum_{\mu} \omega(\mu) \phi(\mu(x)),
\]

where the weights ω(μ) are to be learned. Such a basis can, e.g., be the Fourier basis. This gives a model able to describe a broad class of functions f and g:

\[
x_{k+1} = \begin{bmatrix} 1 & \cdots & 1 \end{bmatrix} A \begin{bmatrix} \mu(\phi(x_1)) \\ \vdots \\ \mu(\phi(x_n)) \end{bmatrix} + \nu_{t}, \\
y_{k} = \begin{bmatrix} \omega(x) \end{bmatrix} \begin{bmatrix} \mu(\phi(x_1)) \\ \vdots \\ \mu(\phi(x_n)) \end{bmatrix} + \epsilon_{t}. \tag{1}
\]

This model is nonlinear in its states, requiring particle methods (or similar), but linear in its parameters, allowing for efficient parameter estimation (A, C, Q, R, learning).

Output (in pixels)

Simulated position

The model (1) is indeed very flexible. Thus, there is a need to encode prior assumptions to "help" the model in generalizing the training data and to avoid over-fitted models and lack of unique solutions.

We therefore suggest to use a L² regularization scheme on ω(μ):

\[
\|ω(μ)\|_{L²} = \sum_{μ} \omega(μ)^2,
\]

or equivalently, put Gaussian priors on ω(μ).

With a particular choice of basis functions and such priors, the assumptions on f can actually be interpreted as an (approximate) Gaussian process prior (Svensson et al 2015).

We can also put an inverse Wishart prior on Q.

Regularization – Tuning the complexity of nonlinear models

The proposed method is also possible to tailor to more specific model structures. The paper contains an example on how to learn a Hammerstein–Wiener model as a special case of (1).

Learning a MIMO model

The proposed framework is applicable also to MIMO data. Here, we consider learning of the input-output behavior of real data from a motorized camera with 6 inputs and 2 outputs. We model the data using a two-dimensional state space, and a simple regularization with a prior precision proportional to the order of the basis function.

In this example, we tailor the model by assuming a known linear measurement function g, and let f catch both the dynamics and the nonlinear phenomenon.