
Rishi Relan
Dieter Verbeke
Koen Tiels

Department ELEC

12.05.2017
Idea of this workshop: Nonlinear System

“All the world is a nonlinear system
He linearised to the right
He linearised to the left
Till nothing was right
And nothing was left”

Stephen A Billings, The University of Sheffield, U.K
Idea of this workshop: Different methods

“Based on a careful analysis of the data, the only logical conclusion we can draw is that our method gets the thumbs-up.”
Overview

- Wiener-Hammerstein benchmark
- Linear adaptive filters
- Nonlinear adaptive filters
- Kernel methods
- Online Kernel Methods
- Results
- Take home messages
Benchmark

Nonlinear system identification challenges

- the process noise that is present in the system,
- the static nonlinearity which is not directly accessible from neither the measured input nor output
- the output dynamics are difficult to invert due to the presence of a transmission zero
Physical Electronic Circuit
Linear Adaptive Filters

\[ w(i) = w(i - 1) + \Delta w(i) \]

\[ J(i) = \mathbb{E}[e^2(i)] \quad i = 1, 2, \ldots \]

- Widrow and Hoff (1960): LMS filter
- Kalman (1960): Kalman filter
Two Simple Adaptive Filters

**Least Mean Squares (LMS)**

\[ J(i) = \frac{1}{2} e(i)^2 \]

\[ e(i) = d(i) - w(i - 1)^T u(i) \]

**Recursive Least Squares (RLS)**

\[ J(i) = \sum_{j=1}^{i} [d(j) - w^T u(j)]^2 \]

**Regularized Recursive Least Squares (RRLS)**

\[ J(i) = \sum_{j=1}^{i} [d(j) - w^T u(j)]^2 + \lambda \|w\|^2 \]
Nonlinear Adaptive Filter
Kernel Methods
Putting it formally

\[ \varphi : \mathcal{U} \rightarrow \mathcal{H} \]
\[ \mathbf{u} \mapsto \mathcal{K}(\mathbf{u}, \cdot) \]

**RKHS**

\[ f(\cdot) = \sum_{i=1}^{n} \alpha_i \mathcal{K}(\mathbf{u}_i, \cdot) \]

\( x_1, x_2, \ldots, x_n \) is a set of basis vectors, and \( \alpha_1, \alpha_2, \ldots, \alpha_n \) are the coefficients.

**Representer Theorem**

\[ \| f(\cdot) - \sum_{i=1}^{n} \alpha_i \mathcal{K}(\mathbf{u}_i, \cdot) \| \leq \epsilon \]

In the Hilbert space \( \mathcal{H} \) associated with \( \mathcal{K} \), we can define a corresponding vector \( \tilde{h} \in \mathcal{H} \) as

\[ \tilde{h} = \sum_{i=1}^{m} \alpha_i \varphi(\mathbf{u}_i) \]

\[ \| f(\cdot) - \tilde{h}^T \varphi(\mathbf{u}_i) \| \leq \epsilon \]
Kernel Regression

Kernel Ridge Regression

$$\min_f J(f) = \sum_{i=1}^{N} [d(i) - \varphi(u(i))]^2 + \lambda \|f\|_H^2$$

Kernel RLS

$$\min_w J(i) = \sum_{j=1}^{i} [d(j) - w^T \varphi(j)]^2 + \lambda \|w\|_H^2$$
The Standard Online Algorithm
Some relevant questions?

- How to select the basis functions in the dictionary?
- How many basis functions or how to remove basis?
- What is the criterion for basis removal?
- How to handle nonstationary data?
The Bayesian Viewpoint

\[ D_t \equiv \{u_i, y_i\}_{i=1}^t \]

where \( u_i \in \mathcal{R}^D \) and \( y_i \in \mathcal{R} \)

**Model**

\[ y_i = f(u_i) + \epsilon_i \]

**Prior**

\[ f(u) \approx \mathcal{GP}(m(u), \mathcal{K}(u, u')) \]

**Likelihood**

\[ p(y_i | f(u_i)) = \mathcal{N}(y_i | f(u_i), \sigma_n^2) \]

**Posterior**

\[ p(f_t | D_t) = \frac{\text{Prior} \times \text{likelihood}}{\text{evidence}} = \mathcal{N}(f_t | \mu_t, \Sigma_t) \]

**Posterior at new data**

\[ p(f_{t+1} | D_{t+1}) = \mathcal{N}(f_{t+1} | \mu_{t+1}, \Sigma_{t+1}) \]
Further Extensions

**Fixed Budget KRLS**

Adaptive KRLS

\[ M = \text{Size of basis expansion} \]

\[ \lambda = \text{Forgetting factor} \]


Observations

$M = 100, \ K = 10$

$RMSE : -41.55 \ dB$
Observations

$M=1, \ K=10$

$RMSE : -40.81 \ dB$
$M=1, \ K=1$

$RMSE: -36.50$ dB
Conclusions

- Kernel adaptive learning can be useful in certain sequential prediction or tracking tasks.
- It can deal with nonstationary data.
- It requires further testing on challenging nonlinear system identification benchmarks.
Thank you for your attention!