Robust, Deep

Recurrent Gaussian processes

Andreas Damianou
with César Lincoln Mattos, Zhenwen Dai, Neil Lawrence,
Jeremy Forth, Guilherme Barreto

Royal Society, 06 June 2016

RECURRENT GAUSSIAN PROCESSES

César Lincoln C. Mattos[1], **Zhenwen Dai**[2], **Andreas Damianou**[3], **Jeremy Forth**[4], **Guilherme A. Barreto**[5] **& Neil D. Lawrence**[6]
[1,5]Federal University of Ceará, Fortaleza, Ceará, Brazil
[2,3,6]University of Sheffield, Sheffield, UK
[1]cesarlincoln@terra.com.br
[2,3]{z.dai,andreas.damianou}@sheffield.ac.uk
[4]jforth@iweng.org
[5]gbarreto@ufc.br
[6]N.Lawrence@dcs.sheffield.ac.uk

Preprint, 11th IFAC Symposium on Dynamics and Control of Process Systems, including Biosystems
June 6-8, 2016. NTNU, Trondheim, Norway

Latent Autoregressive Gaussian Processes Models for Robust System Identification

César Lincoln C. Mattos * **Andreas Damianou** **
Guilherme A. Barreto * **Neil D. Lawrence** **

* *Federal University of Ceará, Dept. of Teleinformatics Engineering, Center of Technology, Campus of Pici, Fortaleza, Ceará, Brazil (e-mail: cesarlincoln@terra.com.br; gbarreto@ufc.br).*
** *Dept. of Computer Science & SITraN, The University of Sheffield, Sheffield, UK (e-mail: andreas.damianou@sheffield.ac.uk; N.Lawrence@dcs.sheffield.ac.uk)*

Variational Gaussian Process State-Space Models

Roger Frigola, Yutian Chen and Carl E. Rasmussen
Department of Engineering
University of Cambridge
{rf342,yc373,cer54}@cam.ac.uk

Variational inference for Student-*t* models: Robust Bayesian interpolation and generalised component analysis

Michael E. Tipping[a], , Neil D. Lawrence[b],
[a] Microsoft Research, 7 J.J. Thomson Avenue, Cambridge CB3 0FB, UK
[b] Department of Computer Science, The University of Sheffield, Regent Court, 211 Portobello St., Sheffield S1 4DP, UK

Semi-described and semi-supervised learning with Gaussian processes

Andreas Damianou
Dept. of Computer Science & SITraN
The University of Sheffield
Sheffield, UK

Neil D. Lawrence
Dept. of Computer Science & SITraN
The University of Sheffield
Sheffield, UK

The Unreasonable Effectiveness of Recurrent Neural Networks

May 21, 2015

There's something magical about Recurrent Neural Networks (RNNs). I still remember when I trained my first recurrent network for Image Captioning. Within a few dozen minutes of training my first baby model (with rather arbitrarily-chosen hyperparameters) started to generate very nice looking descriptions of images that were on the edge of making sense. Sometimes the ratio of how simple your model is to the quality of the results you get out of it blows past your expectations, and this was one of those times. What made this result so shocking at the time was that the common wisdom was that RNNs were supposed to be difficult to train (with more experience I've in fact reached the opposite conclusion). Fast forward about a year: I'm training RNNs all the time and I've witnessed their power and robustness many times, and yet their magical outputs still find ways of amusing me. This post is about sharing some of that magic with you.

Challenge: Learn patterns from sequences

- Recurrent Gaussian Processes (RGP): a family of recurrent Bayesian nonparametric models
  (data efficient, uncertainty handling).
- Latent deep RGP: a deep RGP with latent states
  (simultaneous representation + dynamical learning).
- Recurrent Variational Bayes (REVARB) framework
  (efficient inference + coherent propagation of uncertainty)
- Extension: RNN-based sequential recognition models
  (Regularization + parameter reduction).
- Extension: Robustness to outliers.
- Comparison with LSTMs, parametric and non-latent models.

NARX model

A **standard NARX model** considers an input vector $\mathbf{x}_i \in \mathbb{R}^D$ comprised of L_y past observed outputs $y_i \in \mathbb{R}$ and L_u past exogenous inputs $u_i \in \mathbb{R}$:

$$\mathbf{x}_i = [y_{i-1}, \cdots, y_{i-L_y}, u_{i-1}, \cdots, u_{i-L_u}]^\top,$$
$$y_i = f(\mathbf{x}_i) + \epsilon_i^{(y)}, \qquad \epsilon_i^{(y)} \sim \mathcal{N}(\epsilon_i^{(y)} | 0, \sigma_y^2),$$

State-space model:

$$x_i = f(x_{i-1}, \cdots, x_{i-L_x} u_{i-1}, \cdots, u_{i-L_u}) + \epsilon_i^{(x)}, \quad \textit{(transition)}$$
$$y_i = x_i + \epsilon_i^{(y)} \quad \textit{(emission)}$$

Non-linear emission:

$$y_i = g(x_i) + \epsilon_i^{(y)}$$

NARX model

A **standard NARX model** considers an input vector $\mathbf{x}_i \in \mathbb{R}^D$ comprised of L_y past observed outputs $y_i \in \mathbb{R}$ and L_u past exogenous inputs $u_i \in \mathbb{R}$:

$$\mathbf{x}_i = [y_{i-1}, \cdots, y_{i-L_y}, u_{i-1}, \cdots, u_{i-L_u}]^\top,$$
$$y_i = f(\mathbf{x}_i) + \epsilon_i^{(y)}, \qquad \epsilon_i^{(y)} \sim \mathcal{N}(\epsilon_i^{(y)} | 0, \sigma_y^2),$$

State-space model:

$$x_i = f(x_{i-1}, \cdots, x_{i-L_x} u_{i-1}, \cdots, u_{i-L_u}) + \epsilon_i^{(x)}, \quad \textit{(transition)}$$
$$y_i = x_i + \epsilon_i^{(y)} \quad \textit{(emission)}$$

Non-linear emission:

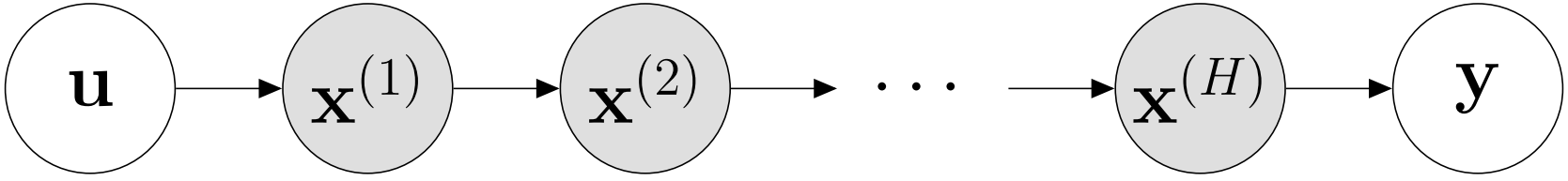
$$y_i = g(x_i) + \epsilon_i^{(y)}$$

NARX vs State-space

- Latent inputs allow for simultaneous representation learning and dynamical learning.
- Latent inputs means that noisy predictions are not fed back to the model.

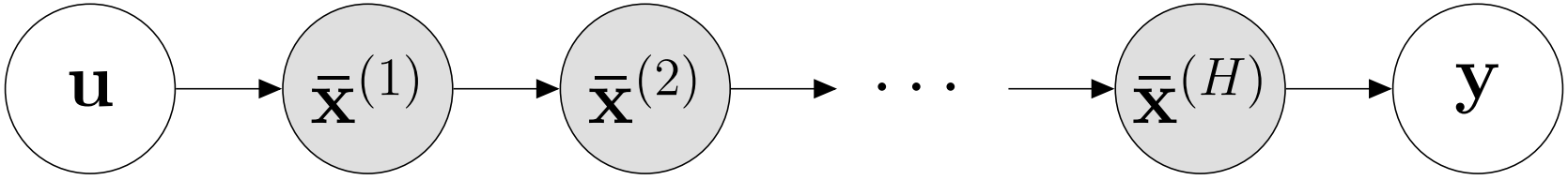
(Deep) RGP

Start from a deep GP:



(Deep) RGP

Latent states formed from lagged window of length L:

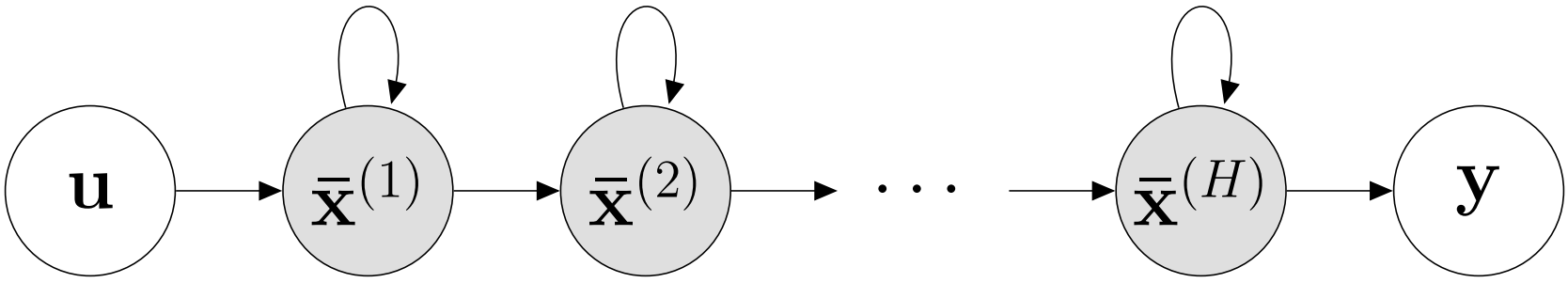


For one layer:

$$\bar{\mathbf{x}}_i = [x_i, \cdots, x_{i-L+1}]^\top, \quad x_j \in \mathbb{R}$$

(Deep) RGP

Add recursion in the latent states:



For one layer:

$$\bar{\mathbf{x}}_i = \left[x_i, \cdots, x_{i-L+1}\right]^\top, \quad x_j \in \mathbb{R}$$

So that:

$$x_i = f(\bar{\mathbf{x}}_{i-1}, \bar{\mathbf{u}}_{i-1}) + \epsilon_i^x$$

$$\mathbf{y}_i = g(\bar{\mathbf{x}}_i) + \epsilon_i^y$$

REVARB: REcurrent VARiational Bayes

Extend joint probability with *inducing points*:

$$p(\text{joint}) = p\left(\mathbf{y}, \mathbf{f}^{(H+1)}, \mathbf{z}^{(H+1)}, \left\{\mathbf{x}^{(h)}, \mathbf{f}^{(h)}, \mathbf{z}^{(h)}\right\} \middle|_{h=1}^{H}\right)$$

Lower bound: $\log p(\mathbf{y}) \leq \int_{\mathbf{f},\mathbf{x},\mathbf{z}} \mathcal{Q} \frac{p(\text{joint})}{\mathcal{Q}=q(\mathbf{x}^{(h)},\mathbf{f}^{(h)},\mathbf{z}^{(h)}),\forall h}$

Posterior marginal: $q\left(\mathbf{x}^{(h)}\right) = \prod_{i=1}^{N} \mathcal{N}\left(x_i^{(h)} \middle| \mu_i^{(h)}, \lambda_i^{(h)}\right)$

Mean-field for $q(x)$ allows analytical solution without having to resort to sampling. Additional layers to compensate for the uncorrelated posterior.

REVARB: REcurrent VARiational Bayes

Extend joint probability with *inducing points*:

$$p(\text{joint}) = p\left(\mathbf{y}, \mathbf{f}^{(H+1)}, \mathbf{z}^{(H+1)}, \left\{\mathbf{x}^{(h)}, \mathbf{f}^{(h)}, \mathbf{z}^{(h)}\right\} \Big|_{h=1}^{H}\right)$$

Lower bound: $\log p(\mathbf{y}) \leq \int_{\mathbf{f},\mathbf{x},\mathbf{z}} \mathcal{Q} \frac{p(\text{joint})}{\mathcal{Q}=q(\mathbf{x}^{(h)},\mathbf{f}^{(h)},\mathbf{z}^{(h)}),\forall h}$

Posterior marginal: $q\left(\mathbf{x}^{(h)}\right) = \prod_{i=1}^{N} \mathcal{N}\left(x_i^{(h)} \middle| \mu_i^{(h)}, \lambda_i^{(h)}\right)$

Mean-field for $q(x)$ allows analytical solution without having to resort to sampling. Additional layers to compensate for the uncorrelated posterior.

REVARB: REcurrent VARiational Bayes

Extend joint probability with *inducing points*:

$$p(\text{joint}) = p\left(\mathbf{y}, \mathbf{f}^{(H+1)}, \mathbf{z}^{(H+1)}, \left\{\mathbf{x}^{(h)}, \mathbf{f}^{(h)}, \mathbf{z}^{(h)}\right\} \Big|_{h=1}^{H}\right)$$

Lower bound: $\log p(\mathbf{y}) \leq \int_{\mathbf{f},\mathbf{x},\mathbf{z}} \mathcal{Q} \frac{p(\text{joint})}{\mathcal{Q}=q(\mathbf{x}^{(h)},\mathbf{f}^{(h)},\mathbf{z}^{(h)}),\forall h}$

Posterior marginal: $q\left(\mathbf{x}^{(h)}\right) = \prod_{i=1}^{N} \mathcal{N}\left(x_i^{(h)} \middle| \mu_i^{(h)}, \lambda_i^{(h)}\right)$

Mean-field for $q(x)$ allows analytical solution without having to resort to sampling. Additional layers to compensate for the uncorrelated posterior.

REVARB: REcurrent VARiational Bayes

Extend joint probability with *inducing points*:

$$p(\text{joint}) = p\left(\mathbf{y}, \mathbf{f}^{(H+1)}, \mathbf{z}^{(H+1)}, \left\{\mathbf{x}^{(h)}, \mathbf{f}^{(h)}, \mathbf{z}^{(h)}\right\} \Big|_{h=1}^{H}\right)$$

Lower bound: $\log p(\mathbf{y}) \leq \int_{\mathbf{f},\mathbf{x},\mathbf{z}} \mathcal{Q} \frac{p(\text{joint})}{\mathcal{Q}=q(\mathbf{x}^{(h)},\mathbf{f}^{(h)},\mathbf{z}^{(h)}),\forall h}$

Posterior marginal: $q\left(\mathbf{x}^{(h)}\right) = \prod_{i=1}^{N} \mathcal{N}\left(x_i^{(h)} \middle| \mu_i^{(h)}, \lambda_i^{(h)}\right)$

Mean-field for $q(x)$ allows analytical solution without having to resort to sampling. Additional layers to compensate for the uncorrelated posterior.

RNN-based recognition model

Reduce variational parameters by reparameterizing the variational means $\mu_i^{(h)}$ using RNNs:

$$\mu_i^{(h)} = g^{(h)}\left(\hat{\mathbf{x}}_{i-1}^{(h)}\right),$$

where $g(\mathbf{x}) = \mathbf{V}_{L_N}^\top \boldsymbol{\phi}_{L_N}(\mathbf{W}_{L_N-1}\boldsymbol{\phi}_{L_N-1}(\cdots \mathbf{W}_2\boldsymbol{\phi}_1(\mathbf{U}_1\mathbf{x})))$,

Amortized inference also regularizes the optimization procedure.

RNN-based recognition model

Reduce variational parameters by reparameterizing the variational means $\mu_i^{(h)}$ using RNNs:

$$\mu_i^{(h)} = g^{(h)}\left(\hat{\mathbf{x}}_{i-1}^{(h)}\right),$$

where $g(\mathbf{x}) = \mathbf{V}_{L_N}^\top \boldsymbol{\phi}_{L_N}(\mathbf{W}_{L_N-1}\boldsymbol{\phi}_{L_N-1}(\cdots \mathbf{W}_2\boldsymbol{\phi}_1(\mathbf{U}_1\mathbf{x})))$,

Amortized inference also regularizes the optimization procedure.

Robustness to outliers

Recall the RGP variant with parametric emission:

$$
\begin{aligned}
x_i &= f(x_{i-1}, \cdots, x_{i-L_x} u_{i-1}, \cdots, u_{i-L_u}) + \epsilon_i^{(x)}, \\
y_i &= x_i + \epsilon_i^{(y)}, \\
\epsilon_i^{(x)} &\sim \mathcal{N}(\epsilon_i^{(x)} | 0, \sigma_x^2), \\
\epsilon_i^{(y)} &\sim \mathcal{N}(\epsilon_i^{(y)} | 0, \tau_i^{-1}), \quad \tau_i \sim \Gamma(\tau_i | \alpha, \beta),
\end{aligned}
$$

- "Switching-off" outliers by including the above Student-t likelihood.
- **Modified REVARB** allows for analytic solution.

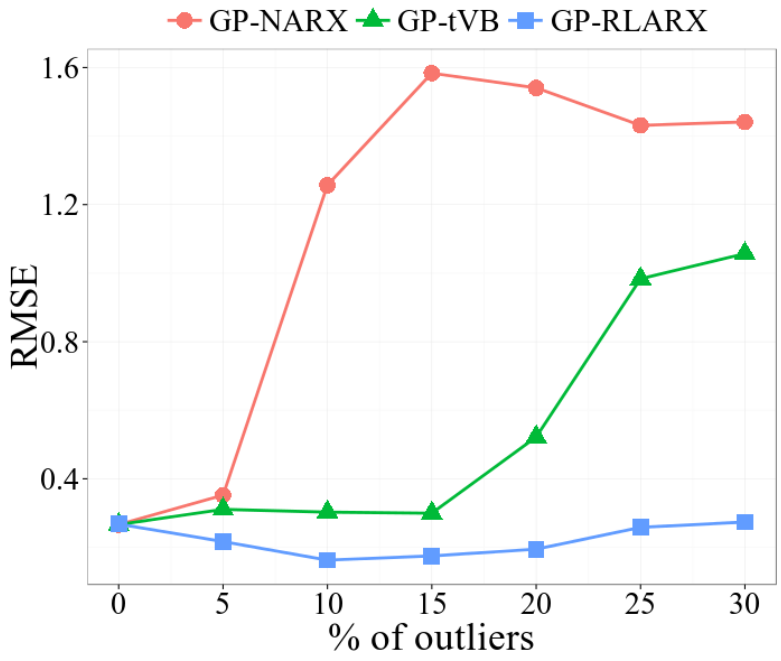
Robustness to outliers

Recall the RGP variant with parametric emission:

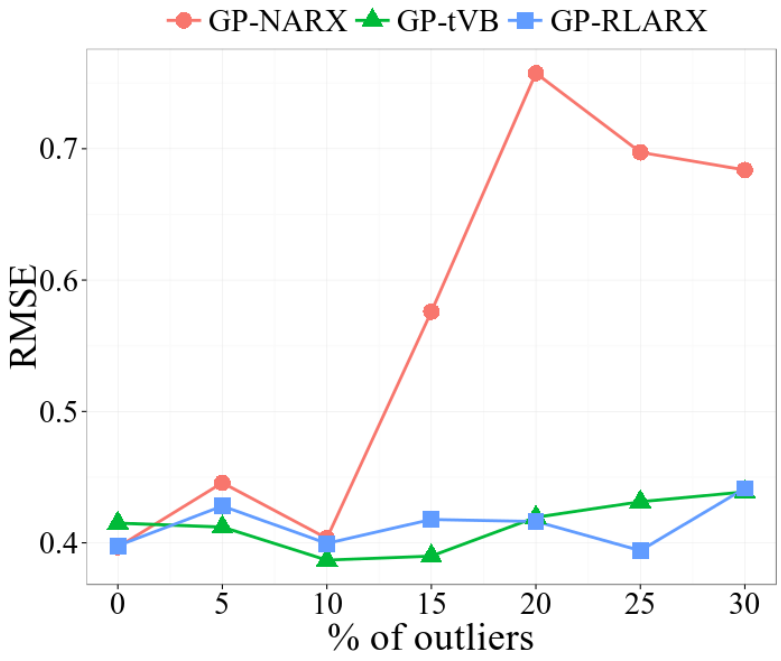
$$
\begin{aligned}
x_i &= f(x_{i-1}, \cdots, x_{i-L_x} u_{i-1}, \cdots, u_{i-L_u}) + \epsilon_i^{(x)}, \\
y_i &= x_i + \epsilon_i^{(y)}, \\
\epsilon_i^{(x)} &\sim \mathcal{N}(\epsilon_i^{(x)} | 0, \sigma_x^2), \\
\epsilon_i^{(y)} &\sim \mathcal{N}(\epsilon_i^{(y)} | 0, \tau_i^{-1}), \quad \tau_i \sim \Gamma(\tau_i | \alpha, \beta),
\end{aligned}
$$

- "Switching-off" outliers by including the above Student-t likelihood.
- **Modified REVARB** allows for analytic solution.

Robust GP autoregressive model: demonstration

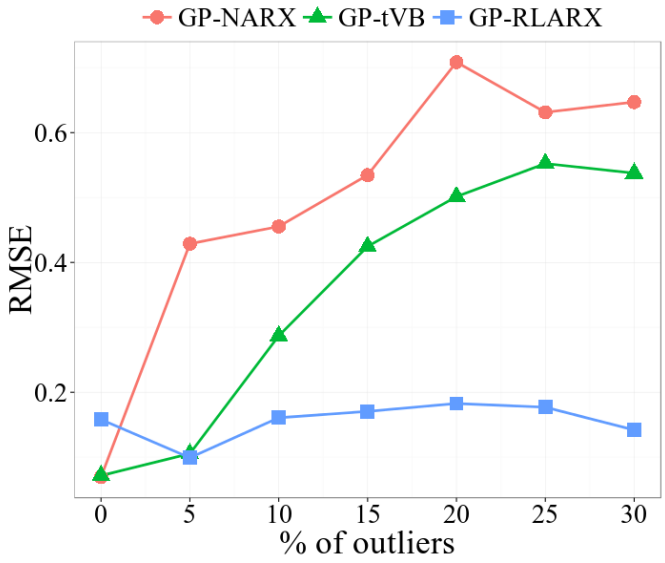


(a) Artificial 1.

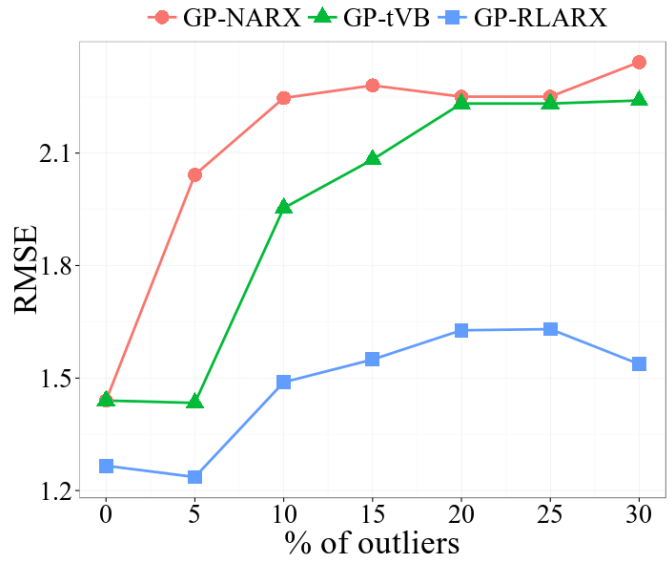


(b) Artificial 2.

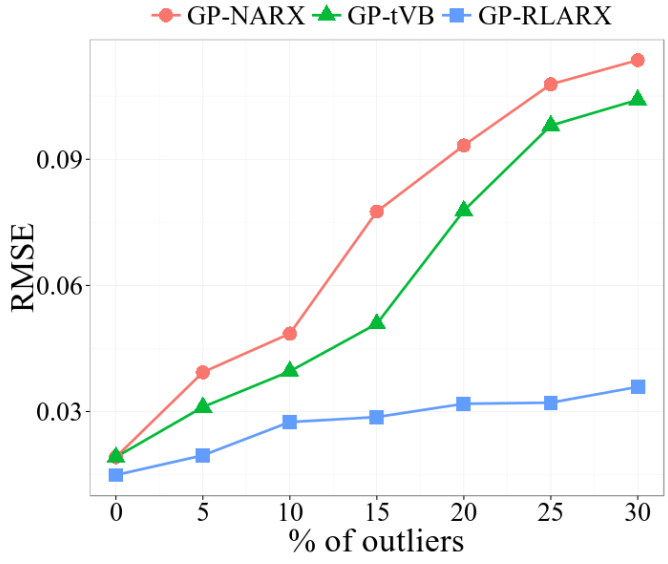
Figure: RMSE values for free simulation on test data with different levels of contamination by outliers.



(c) Artificial 3.



(d) Artificial 4.



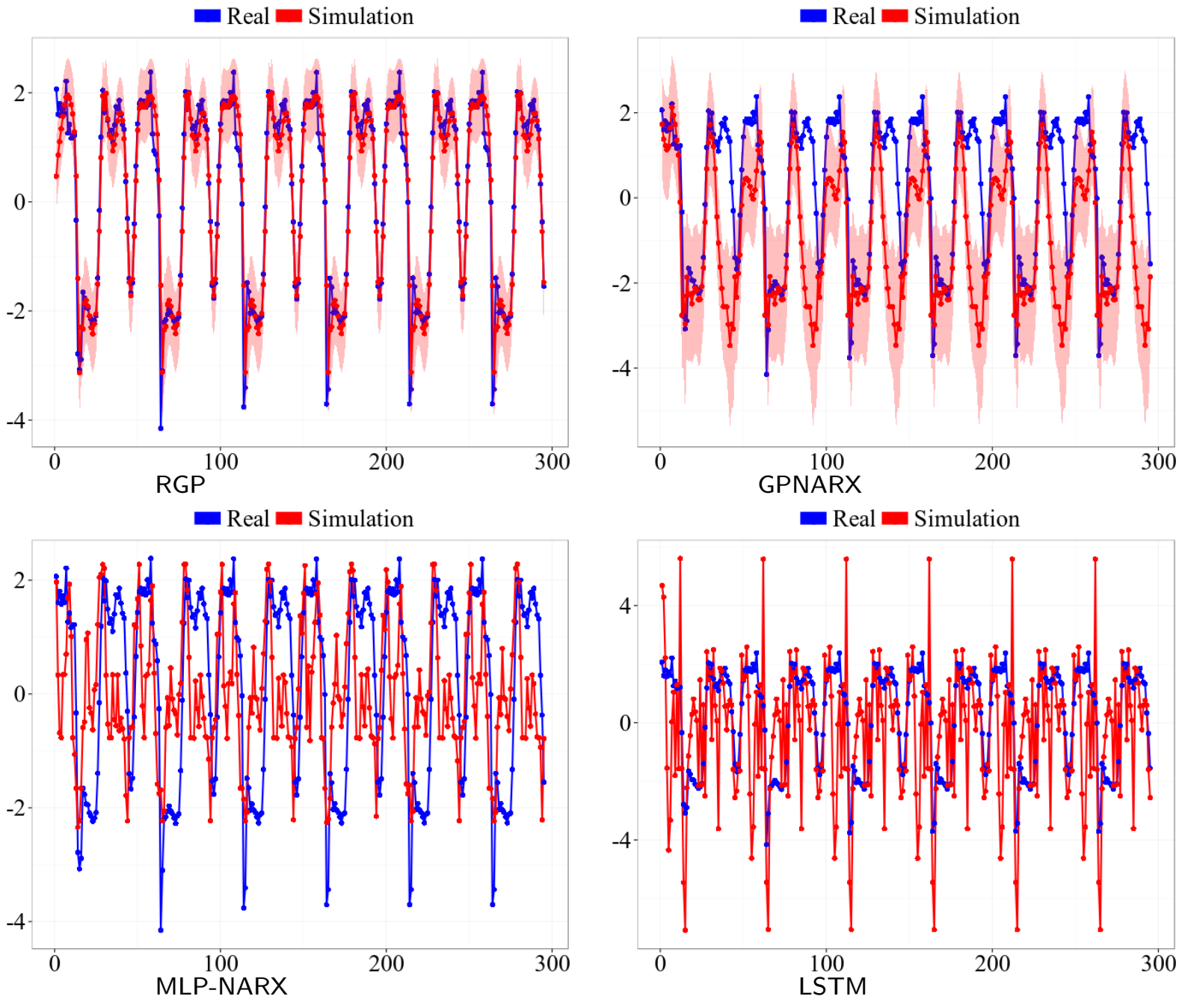
(e) Artificial 5.

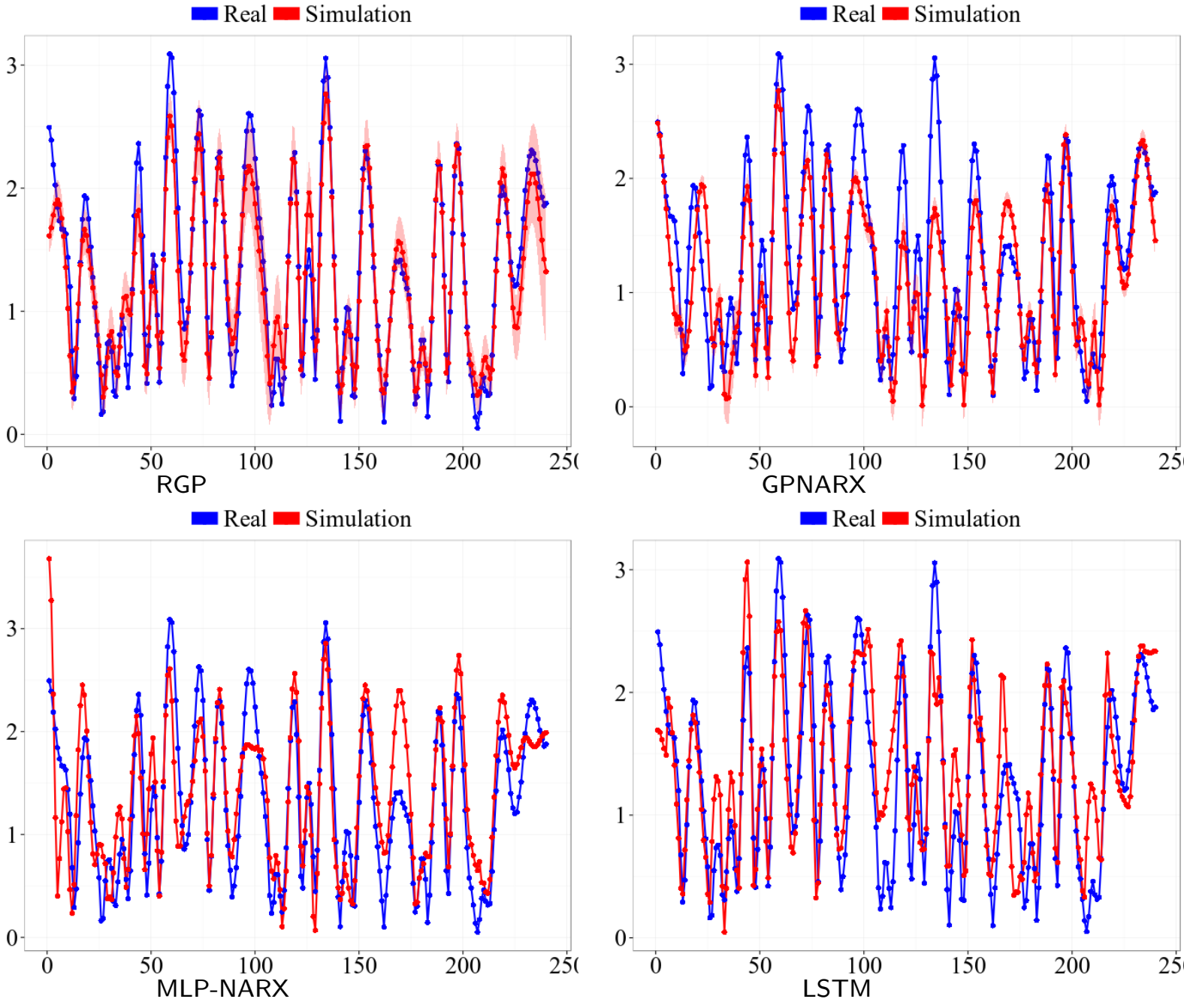
Results

Results in nonlinear systems identification:

1. artificial dataset

2. "drive" dataset: by a system with two electric motors that drive a pulley using a flexible belt.
   - input: the sum of voltages applied to the motors
   - output: speed of the belt.


Real
Simulation
2
0
-2
-4
0
100
200
300
RGP
Real
Simulation
2
0
-2
-4
0
100
200
300
GPNARX
Real
Simulation
2
0
-2
-4
0
100
200
300
MLP-NARX
Real
Simulation
4
0
-4
0
100
200
300
LSTM


Real
Simulation
RGP
GPNARX
MLP-NARX
LSTM

Avatar control

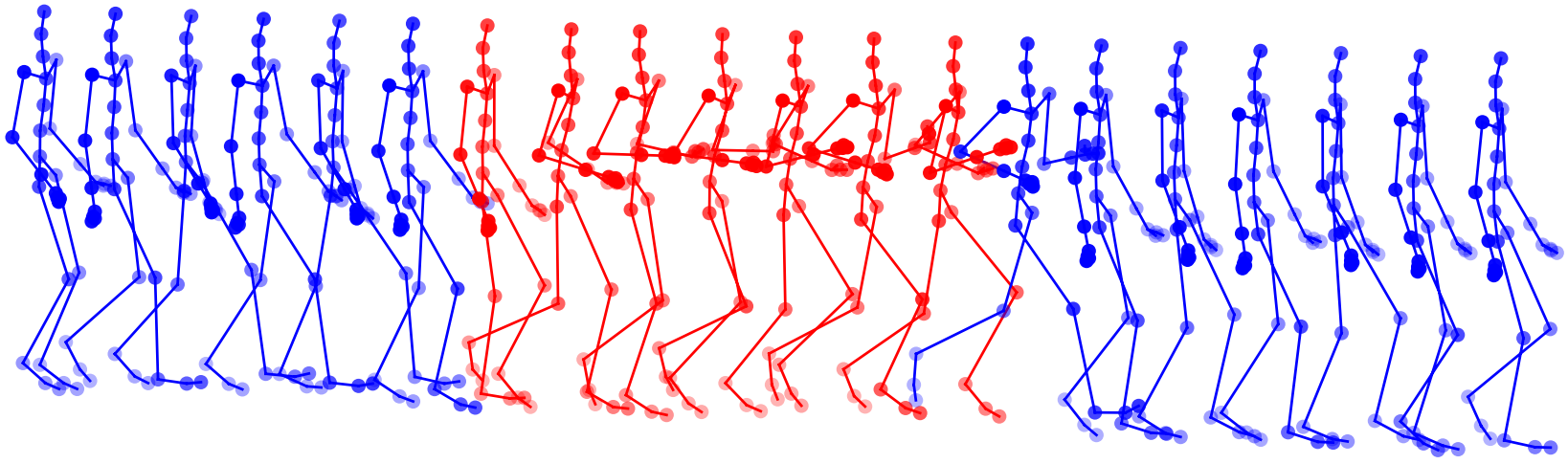


Figure: The generated motion with a step function signal, starting with walking (blue), switching to running (red) and switching back to walking (blue).

Videos:

- https://youtu.be/FR-oeGxV6yY *Switching between learned speeds*
- https://youtu.be/AT0HMtoPgjc *Interpolating (un)seen speed*
- https://youtu.be/FuF-uZ83VMw Constant unseen speed