Time-series modeling and forecasting using Reservoir Computing models

Sebastián Basterrech

IT4Innovations

VŠB – Technical University of Ostrava

October 21, 2013

e-mail: Sebastian.Basterrech.Tiscordio@vsb.cz
web: http://tinyurl.com/bomdgzn
Brief research statement

Academic background:


Research topics of interest:

- Statistical Learning Methods.
- Neural Computing and Applications.
- Numerical Optimization Algorithms.
Background: Time-series forecasting

Motivation:

- Time-series modeling is an important research and application area.
- Many **real world tasks** are time dependent in nature, for example:
  - data prediction in areas such as financial, traffic, weather problems, etc;
  - pattern recognition in visual and speech processing;
  - modeling cognitive processing in neural systems;
  - in robotic and system control purposes;
  - medical classification problems;
Background: Time-series forecasting

In the Machine Learning literature there are numerous efforts and attempts to develop methods in this research field:

- **Linear models:**
  - Moving average.
  - Exponential smoothing.
  - Auto-regressive integrated moving average (ARIMA).

- **Non-linear models:**
  - Non-linear modeling using multivariate adaptive regression splines (MARS).
  - Fuzzy System models.
  - Non-linear ARX models.
  - Turing Machines.
  - Non-linear additive autoregressive models.
  - Recurrent Neural Networks (RNNs).
Recurrent Neural Networks

Functionality description:
- Is a Neural Network that operates in time.
- At each time step, it accepts an input vector, update its hidden state via non-linear activation functions, and uses it to make a prediction of its output.

Main characteristics:
- Connectionist models where there is at least one circuit.
- Circuits can be used to learn and memorize temporal information.
- They are fascinating tools for nonlinear time series processing applications.

Main drawbacks: the training process is hard.
Main drawbacks:

- The training methods which use the first differential information have often stability problems (Bengio et al., 1994).
- The volatile relation between the internal state of the dynamics and the model parameters affect the gradient error norm (Bengio et al., 1994). The problem of vanishing and exploding gradient:
  - The vanishing gradient occurs when the norm of the gradient get arbitrarily close and fast to 0.
  - Exploding gradient problem refers to the opposite, when the gradient norm large increases during the training process.
- As a consequence much longer training times are necessary.
Motivation:

- How to use the potential for *memorization* of recurrent networks without the difficulties in the training process?
- How to represent *time*?

The RC approach:

- The RNN is used to transform the spatio-temporal input information in a spatial representation.
- Additionally, this projection should enhance the linear separability of the input data.
- The RNN is deemed fixed during the learning process (it is only used to encode input data in another space).
Reservoir Computing models

Motivation:

- How to use the potential for memorization of recurrent networks without the difficulties in the training process?
- How to represent time?

The RC approach:

- The RNN is used to transform the spatio-temporal input information in a spatial representation.
- Additionally, this projection should enhance the linear separability of the input data.
- The RNN is deemed fixed during the learning process (it is only used to encode input data in another space).
Motivation:

- How to use the potential for *memorization* of recurrent networks without the difficulties in the training process?
- How to represent *time*?

The RC approach:

- The RNN is used to transform the spatio-temporal input information in a spatial representation.
- Additionally, this projection should enhance the linear separability of the input data.
- The RNN is deemed fixed during the learning process (it is only used to encode input data in another space).
Reservoir Computing models

RC approach:

- Input information
- A RNN as a dynamical system to project inputs in another space (reservoir)
- A supervised learning step, from the reservoir to the outputs

inputs: \((a(t), a(t - 1), \ldots)\),
targets: \(b(t), \forall t\)

Model parameters:
- \(w_{in}: \mathbb{R}^{N_a \times N_a}\)
- \(w_r: \mathbb{R}^{N_x \times N_x}\)
- \(w_{out}: \mathbb{R}^{N_b \times N_x}\)
Reservoir Computing models

RC approach:

- Input information
- A RNN as a dynamical system to project inputs in another space (reservoir)
- A supervised learning step, from the reservoir to the outputs

inputs: \((a(t), a(t-1), \ldots), \)
targets: \(b(t), \forall t\)
Reservoir Computing models

RC approach:

- Input information
- A RNN as a dynamical system to project inputs in another space (reservoir)
- A supervised learning step, from the reservoir to the outputs

Inputs: \((a(t), a(t - 1), \ldots),\)
Target: \(b(t), \forall t\)

\[a \in \mathbb{R}^{N_a}, \quad x \in \mathbb{R}^{N_x}, \quad b \in \mathbb{R}^{N_b}\]
Reservoir Computing models

RC approach:

- Input information
- A RNN as a dynamical system to project inputs in another space (reservoir)
- A supervised learning step, from the reservoir to the outputs

inputs: \((a(t), a(t - 1), \ldots)\),
targets: \(b(t), \forall t\)
RC models

Some questions?

If the RNN is fixed during the learning process, thus:

- what is the best network topology?
- what activation functions do we use?
- can we guarantee stability?
- how to initialize it?
- and so on ...
RC models

Echo State Networks (Jaegger, 2001):

- The inputs of the model: \( a(t), t = 0, 1, \ldots \)
- The reservoir state at any time \( t \):

\[
x(t) = \tanh(w^{\text{in}} a(t) + w^{\text{r}} x(t - 1)).
\]

- The output of the ESN model:

\[
y(t) = w^{\text{out}} x(t).
\]

- Adjustable parameters: only \( w^{\text{out}} \).
RC models
RC model examples

- **Liquid State Machine** (LSM) introduced in 2002 (Maass *et al.*, 2002): the model comes from the interest in making a conceptual representation of the cortical microstructures in the brain. It is built using a special kind of Spiking Neural Neurons (SNNs) derived from the Hodgkin–Huxley neuron model: *Leaky Integrate and Fire* (LIF) neurons.

- Backpropagation-decorrelation Recurrent Learning introduced in 2004 (Steil, 2004), Leaky Integrator Echo State Networks studied in 2007 (Jaeger *et al.*, 2007), and so on.
RC models

Liquid State Machine (LSM) introduced in 2002 (Maass et al., 2002): the model comes from the interest in making a conceptual representation of the cortical microstructures in the brain. It is built using a special kind of Spiking Neural Neurons (SNNs) derived from the Hodgkin–Huxley neuron model: Leaky Integrate and Fire (LIF) neurons.

Backpropagation-decorrelation Recurrent Learning introduced in 2004 (Steil, 2004), Leaky Integrator Echo State Networks studied in 2007 (Jaeger et al., 2007), and so on.
Echo State Queueing Networks (ESQN) comes from the Theory of Queueing (Basterrech et al., 2013):

- Three-layered recurrent topology.
- For all input units $u = 1, \ldots, N_a$,
  $$\varrho_u(t) = \frac{a_u(t)}{r_u}, \quad (a_u(t) < r_u).$$
- For all reservoir units $u = N_a + 1, \ldots, N_a + N_x$ (using ML notation):
  $$\varrho_u(t) = \frac{\sum_{v=1}^{N_a} \frac{a_v(t)}{r_v} w_{u,v}^+ + \sum_{v=N_a+1}^{N_a+N_x} \varrho_v(t-1) w_{u,v}^+}{r_u + \sum_{v=1}^{N_a} \frac{a_v(t)}{r_v} w_{u,v}^- + \sum_{v=N_a+1}^{N_a+N_x} \varrho_v(t-1) w_{u,v}^-}. \quad (1)$$
Experimental results

Simulated data: Fixed 10th order NARMA series data

ESQN estimation with 80 reservoir units.
Experimental results

The Internet traffic prediction (hour scale, UKERNA data set)

(a) Test data.

(b) ESQN estimation with 150 reservoir units.
Experimental results

The Internet traffic prediction (5 minutes scale, ISP data set), example of 20 instances of test predictions:
Experimental results

Time series data:

- Simulated data: 10th order NARMA data, ...
- Real data: the Internet Traffic data from two Internet service providers (ISP and UKERNA).

Example: Performance comparison between ESN and ESQN

<table>
<thead>
<tr>
<th>Series</th>
<th>Model</th>
<th>NMSE</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>NARMA</td>
<td>ESN</td>
<td>0.1401</td>
<td>±0.0504</td>
</tr>
<tr>
<td></td>
<td>ESQN</td>
<td>0.1004</td>
<td>±0.0025</td>
</tr>
<tr>
<td>ISP</td>
<td>ESN</td>
<td>0.0062</td>
<td>±9.8885 \times 10^{-7}</td>
</tr>
<tr>
<td></td>
<td>ESQN</td>
<td>0.0100</td>
<td>±1.2436 \times 10^{-4}</td>
</tr>
<tr>
<td>UKERNA</td>
<td>ESN</td>
<td>0.3781</td>
<td>±0.0066</td>
</tr>
<tr>
<td></td>
<td>ESQN</td>
<td>0.2030</td>
<td>±0.0335</td>
</tr>
</tbody>
</table>
Parameter Sensitivity Analysis

Reservoir size and parameter initialization:

(c) Fixed 10th order NARMA times series. $N_a = 10$. 390 validation samples.

(d) Hénon Map data. $N_a = 3$. 800 validation samples.
Parameter Sensitivity Analysis

Spectral radius and sparsity

10th Fixed NARMA: NMSE computed with 150 reservoir units

10th NARMA validation data set. Influence of spectral radius and density of weights.
Conclusions and future work

Summary

Conclusions:

- Main properties: accuracy, simple learning rules, usefulness in many applications (traffic data prediction, times series data problems, etc).
- Important parameters: reservoir size (important impact), spectral radius (the impact is not clear in some RC models), sparsity (enables fast reservoir updates).
- Important aspects: the dynamics stability.
Future work:

- ESQN: more experimentation is needed in order to analyze the performance impact in the model memory capabilities.
- Using unsupervised method to initialize the reservoir is a potential area for future research in Reservoir Computing field.
- The analysis of the stability of the reservoir dynamics (i.e.: using Lyapunov exponents) of the ESQN remains to be done (in order to state conditions to avoid chaotic systems).
- The connections between the model and dynamic properties of the mammalian brain are also a possible research direction to follow.
Conclusions and future work

Future work

Specifically, at this moment we are investigated:

- RC models to estimate weather forecasting.
- Ensemble systems using RC models.
- RC models for outliers and drift detection.
- The relationship between the dynamics stability and pseudo-spectra of the reservoir.
Some of our results (since June/2013) were accepted in:


Thank you for your attention!
References:


