Deep Convolutional Networks are Useful in System Identification

Antônio H. Ribeiro[1,2,*], Carl Andersson[1,*], Koen Tiels[1], Niklas Wahlström[1] and Thomas B. Schön[1]

[1]Uppsala University, [2]UFMG, [*] Equal contribution

Deep Neural Networks

Yoshua Bengio, Geoffrey Hinton and Yann LeCun *"for conceptual and engineering breakthroughs that have made **deep neural networks** a critical component of computing."*

– Turing award (2018)

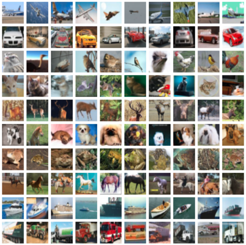
Classifying ECG abnormalities

Antônio H. Ribeiro et. al. (2018)
Automatic Diagnosis of Short-Duration 12-Lead ECG using a Deep Convolutional Network
Machine Learning for Health (ML4H) Workshop at NeurIPS (2018).
arXiv:1811.12194.

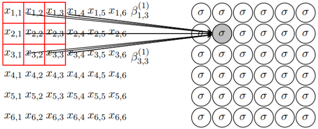
Antônio H. Ribeiro et. al. (2019)
Automatic Diagnosis of the Short-Duration12-Lead ECG using a Deep Neural Network: the CODE Study
arXiv:1904.01949.

Convolutional neural networks

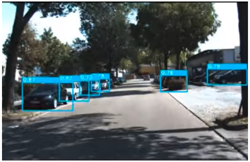
(a) MNIST dataset



(c) CIFAR-10



(b) Conv. layer (2D)

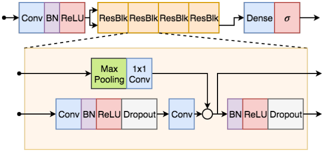


(d) Object detection

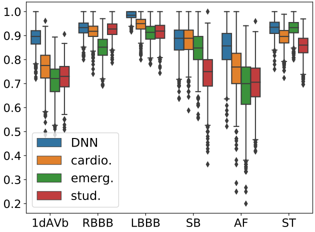
UPPSALA UNIVERSITET

UFMG

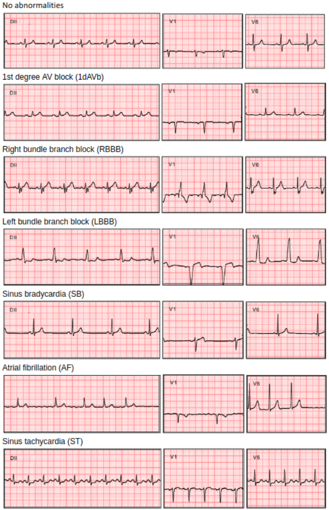
Classifying ECG abnormalities



(a) Convolutional Neural Network



(b) F1 score



(c) Abnormalities classified

Convolutional neural networks for sequence models

Shaojie Bai, J. Zico Kolter, Vladlen Koltun (2018)
An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling
arXiv:1803.01271.

A. van den Oord et. al. (2016)
WaveNet: A Generative Model for Raw Audio
arXiv:1609.03499.

N. Kalchbrenner et. al. (2016)
Neural Machine Translation in Linear Time
arXiv:1610.10099.

The basic neural network

The basic neural network:

$$
\begin{aligned}
\hat{y} &= g^{(L)}(z^{(L-1)}),\\
z^{(l)} &= g^{(l)}(z^{(l-1)}), \quad l = 1, \dots, L-1,\\
z^{(0)} &= x,
\end{aligned}
$$

where $g^{(l)}(z) = \sigma(W^{(l)} z + b^{(l)})$.

UPPSALA UNIVERSITET

UFMG

The causal convolution

Causal Convolution

The causal convolution can be interpreted as a NARX model:

$$\hat{y}[k+1] = g(x[k], x[k-1], \ldots x[k-(n-1)]),$$

with $x[k] = (u[k],\ y[k])$.

The causal convolution

Causal Convolution

The causal convolution can be interpreted as a NARX model:

$$\hat{y}[k+1] = g(x[k], x[k-1], \ldots x[k-(n-1)]),$$

with $x[k] = (u[k],\ y[k])$.

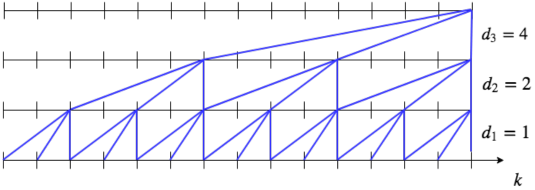
Causal Convolution *with dilations*

Dilations can be interpreted as subsampling the signals:

$$\hat{y}[k+1] = g(x[k], x[k-d_l], \ldots x[k-(n-1)d_l]).$$

antonio.ribeiro@it.uu.se

Temporal convolutional networks



A full TCN:

$$\hat{y}[k+1] = g^{(L)}(Z^{(L-1)}[k]),$$
$$z^{(l)}[k] = g^{(l)}(Z^{(l-1)}[k]), \quad l = 1, \dots, L-1,$$
$$z^{(0)}[k] = x[k],$$

where:

$$Z^{(l-1)}[k] = \left(z^{(l-1)}[k], z^{(l-1)}[k-d_l], \dots, z^{(l-1)}[k-(n-1)d_l] \right).$$

ResNet: residual network

Other Layers

- Nonlinear activation: ReLU
- Dropout
- Batch Normalization:

$$\tilde{z}^{(l)}[k] = \gamma \frac{z^{(l)}[k] - \hat{\mu}_z}{\hat{\sigma}_z} + \beta.$$

- Skip Conections:

$$z^{(l+p)} = \mathcal{F}(z^{(l)}) + z^{(l)}.$$

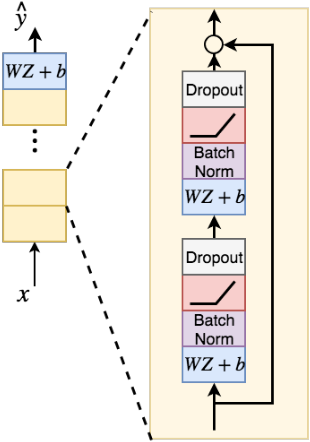


Figure: ResNet

ResNet: residual network

Other Layers

- Nonlinear activation: ReLU
- Dropout
- Batch Normalization:

$$\tilde{z}^{(l)}[k] = \gamma \frac{z^{(l)}[k] - \hat{\mu}_z}{\hat{\sigma}_z} + \beta.$$

- Skip Conections:

$$z^{(l+p)} = \mathcal{F}(z^{(l)}) + z^{(l)}.$$

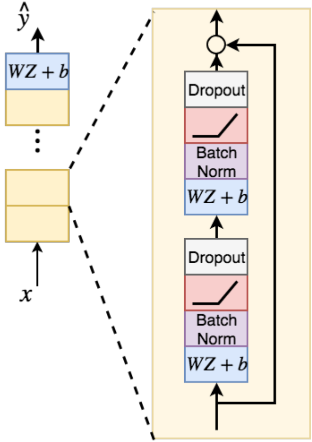


Figure: ResNet

ResNet: residual network

Other Layers

- Nonlinear activation: ReLU
- Dropout
- Batch Normalization:

$$\tilde{z}^{(l)}[k] = \gamma \frac{z^{(l)}[k] - \hat{\mu}_z}{\hat{\sigma}_z} + \beta.$$

- Skip Conections:

$$z^{(l+p)} = \mathcal{F}(z^{(l)}) + z^{(l)}.$$

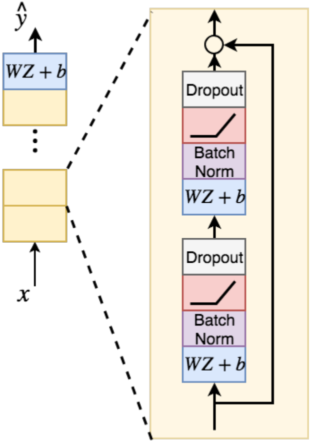


Figure: ResNet

ResNet: residual network

Other Layers

- Nonlinear activation: ReLU
- Dropout
- Batch Normalization:

$$\tilde{z}^{(l)}[k] = \gamma \frac{z^{(l)}[k] - \hat{\mu}_z}{\hat{\sigma}_z} + \beta.$$

- Skip Conections:

$$z^{(l+p)} = \mathcal{F}(z^{(l)}) + z^{(l)}.$$

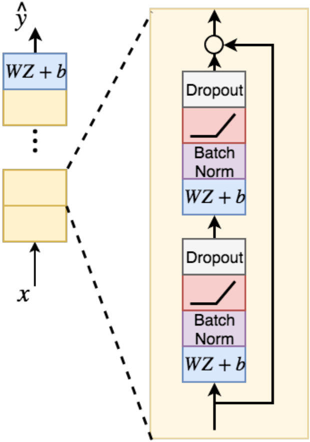


Figure: ResNet

Example 1: Nonlinear toy problem

The nonlinear system:

$$\begin{aligned} y^*[k] &= (0.8 - 0.5e^{-y^*[k-1]^2})y^*[k-1] - \\ &\quad (0.3 + 0.9e^{-y^*[k-1]^2})y^*[k-2] + u[k-1] + \\ &\quad 0.2u[k-2] + 0.1u[k-1]u[k-2] + v[k], \\ y[k] &= y^*[k] + w[k], \end{aligned}$$

S. Chen, S. A. Billings, and P. M. Grant (1990)
Non-linear system identification using neural networks
International Journal of Control, vol. 51, no. 6, pp. 1191-1214,

Example 1: Nonlinear toy problem

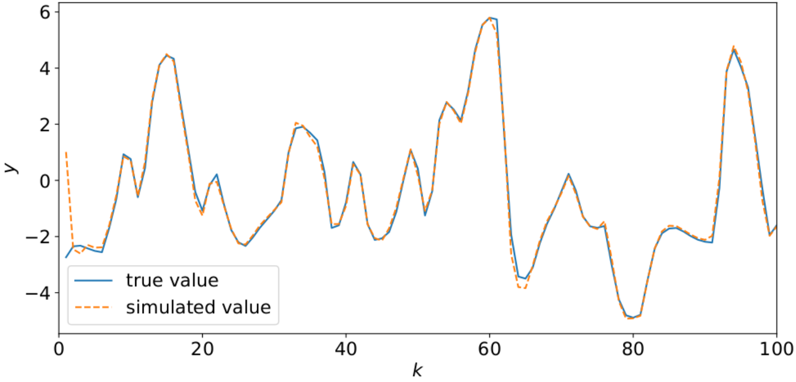


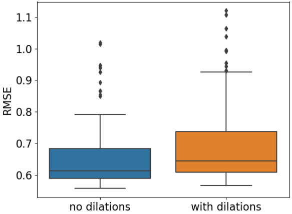
Figure: Displays 100 samples of the free-run simulation TCN model vs the simulation of the true system.

Example 1: Nonlinear toy problem

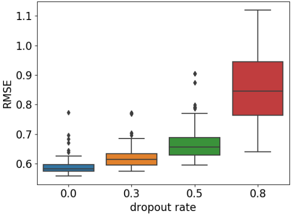
Table: One-step-ahead RMSE on the validation set for the models trained on datasets generated with: different noise levels (σ) and lengths (N)

	N=500			N=2 000			N=8 000		
σ	LSTM	MLP	TCN	LSTM	MLP	TCN	LSTM	MLP	TCN
0.0	0.362	0.270	**0.254**	0.245	0.204	**0.196**	0.165	**0.154**	0.159
0.3	0.712	0.645	**0.607**	0.602	0.586	**0.558**	**0.549**	0.561	0.551
0.6	1.183	1.160	**1.094**	1.105	1.070	**1.066**	**1.038**	1.052	1.043

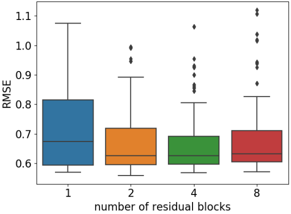
Example 1: Nonlinear toy problem



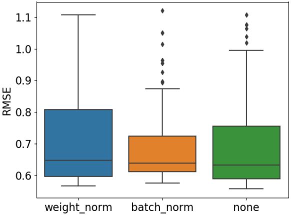
(a) Dilations



(b) Dropout



(c) Depth



(d) Normalization

Example 2: Silverbox

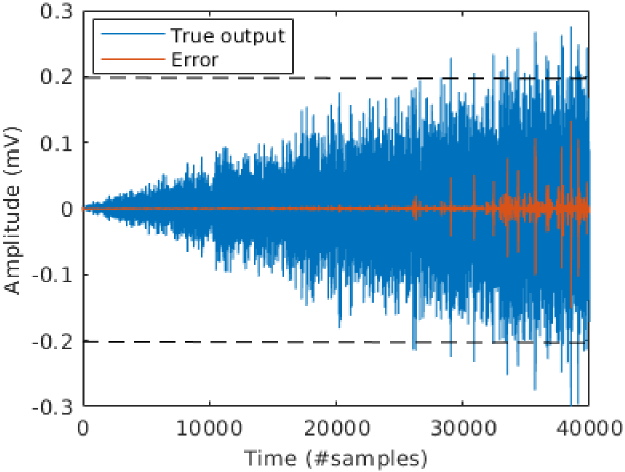


Figure: The true output and the prediction error of the TCN model in free-run simulation for the Silverbox data.

Example 2: Silverbox

Table: **(Example 2)** Free-run simulation results for the Silverbox example on part of the test data (avoiding extrapolation).

RMSE (mV)	Which samples	Approach	Reference
0.7	first 25 000	Local Linear S. Space	V. Verdult (2004)
0.24	first 30 000	NLSS with sigmoids	A. Marconato et. al. (2012)
1.9	400 to 30 000	Wiener-Schetzen	K. Tiels (2014)
0.31	first 25 000	LSTM	this paper
0.58	first 30 000	LSTM	this paper
0.75	first 25 000	MLP	this paper
0.95	first 30 000	MLP	this paper
0.75	first 25 000	TCN	this paper
1.16	first 30 000	TCN	this paper

UPPSALA UNIVERSITET

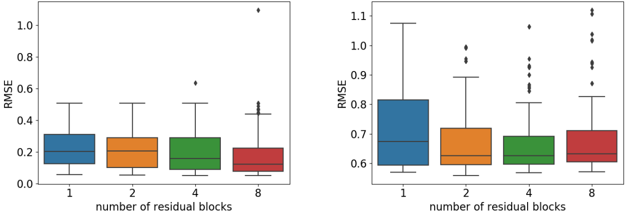
UFMG

Example 2: Silverbox

Table: Free-run simulation results for the Silverbox example on the full test data. (*Computed from FIT=92.2886%).

RMSE (mV)	Approach	Reference
0.96	Physical block-oriented	H. Hjalmarsson et. al. (2004)
0.38	Physical block-oriented	J. Paduart et. al. (2004)
0.30	Nonlinear ARX	L. Ljung (2004)
0.32	LSSVM with NARX	M. Espinoza (2004)
1.3	Local Linear State Space	V. Verdult (2004)
0.26	PNLSS	J. Paduart (2008)
13.7	Best Linear Approximation	J. Paduart (2008)
0.35	Poly-LFR	A. Van Mulders et. al.(2013)
0.34	NLSS with sigmoids	A. Marconato et. al. (2012)
0.27	PWL-LSSVM with PWL-NARX	M. Espinoza et. al. (2005)
7.8	MLP-ANN	L. Sragner et. al. (2004)
4.08*	Piece-wise affine LFR	E. Pepona et. al. (2011)
9.1	Extended fuzzy logic	F. Sabahi et. al. (2016)
9.2	Wiener-Schetzen	K. Tiels et. al. (2015)
3.98	LSTM	this paper
4.08	MLP	this paper
4.88	TCN	this paper

Example 3: F16 ground vibration test



(a) F16 ground vibration test (b) Chen et. al. (1990)

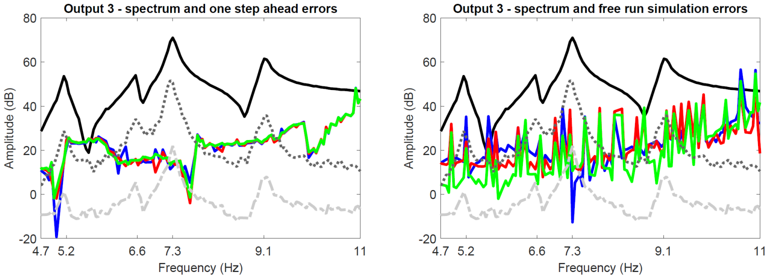
Figure: Box plot showing how different depths of the neural network affects the performance of the TCN.

Example 3: F16 ground vibration test

Table: RMSE for free-run simulation and one-step-ahead prediction for the F16 example averaged over the 3 outputs.

Mode	LSTM	MLP	TCN
Free-run simulation	0.74	0.48	0.63
One-step-ahead prediction	0.023	0.045	0.034

Example 3: F16 ground vibration test



(a) one-step-ahead (b) free-run-simulation

Figure: The error around the main resonance at 7.3 Hz.True output spectrum in black, noise distortion in grey dash-dotted line, total distortion (= noise + nonlinear distortions) in grey dotted line, error LSTM in green, error MLP in blue, and error TCN in red

UPPSALA UNIVERSITET

UFMG

Conclusion

- Potential to provide good results in sys. id. (even if this requires us to rethink these models).
- Traditional deep learning tricks did not always improve the performance.
  - Dilation (exponential decay of dynamical systems)
  - Dropout
  - Depth
- Causal convolutions $\sim$ NARX $\Rightarrow$ biased for non-white noise.
- Both LSTMs and the dilated TCNs are designed for long memory dependencies. Try to apply these models to system identification problems where those are needed, e.g. switched system.

Conclusion

- Potential to provide good results in sys. id. (even if this requires us to rethink these models).
- Traditional deep learning tricks did not always improve the performance.
  - Dilation (exponential decay of dynamical systems)
  - Dropout
  - Depth
- Causal convolutions $\sim$ NARX $\Rightarrow$ biased for non-white noise.
- Both LSTMs and the dilated TCNs are designed for long memory dependencies. Try to apply these models to system identification problems where those are needed, e.g. switched system.

UPPSALA UNIVERSITET

UFMG

Conclusion

- Potential to provide good results in sys. id. (even if this requires us to rethink these models).
- Traditional deep learning tricks did not always improve the performance.
  - Dilation (exponential decay of dynamical systems)
  - Dropout
  - Depth
- Causal convolutions $\sim$ NARX $\Rightarrow$ biased for non-white noise.
- Both LSTMs and the dilated TCNs are designed for long memory dependencies. Try to apply these models to system identification problems where those are needed, e.g. switched system.

UPPSALA UNIVERSITET

UFMG

Conclusion

- Potential to provide good results in sys. id. (even if this requires us to rethink these models).
- Traditional deep learning tricks did not always improve the performance.
  - Dilation (exponential decay of dynamical systems)
  - Dropout
  - Depth
- Causal convolutions $\sim$ NARX $\Rightarrow$ biased for non-white noise.
- Both LSTMs and the dilated TCNs are designed for long memory dependencies. Try to apply these models to system identification problems where those are needed, e.g. switched system.

UPPSALA UNIVERSITET

UFMG

Thank you!