Parameter Estimation with Dense and Convolutional Neural Networks Applied to the FitzHugh–Nagumo ODE

Johann Rudi

Mathematics and Computer Science Division, Argonne National Laboratory

collaboration with
Julie Bessac (Argonne National Laboratory)
Amanda Lenzi (Argonne National Laboratory)

Outline

Introduce the forward and inverse problem

Propose approach: Reconstruction / inverse maps with dense and convolutional neural networks

Demonstrate parameter estimation capabilities

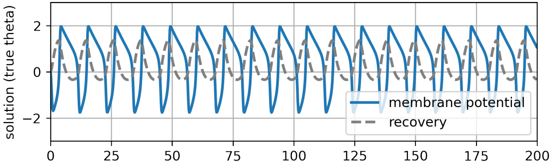
Inspect sensitivity of neural network estimates

Extend inference to ODE parameters and noise model parameters

Forward problem: FitzHugh–Nagumo ODE modeling neuron spikes

The FitzHugh–Nagumo[1] model is a nonlinear system of two ODEs

$$\frac{\mathrm{d}u}{\mathrm{d}t} = \gamma\left(u - \frac{u^3}{3} + v + \zeta\right),$$
$$\frac{\mathrm{d}v}{\mathrm{d}t} = -\frac{1}{\gamma}\left(u - \theta_0 + \theta_1 v\right)$$



Solution of FitzHugh–Nagumo system *(blue and gray lines)*.

- Unknown: membrane potential u, recovery variable v
- Known: stimulus $\zeta \equiv$ const., damping parameter $\gamma \equiv$ const.
- Parameters considered for inference: θ_0 and θ_1

The FitzHugh–Nagumo model is simple from a physiological perspective, however it exhibits similar challenges as complex neuron models, if considered in an inference setting.

[1] FitzHugh 1961; Nagumo, Arimoto, and Yoshizawa 1962.

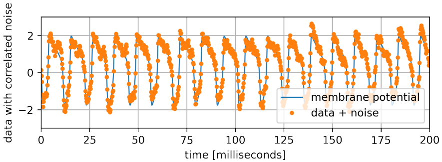
Inverse problem is problematic for gradient-based methods

Consider for inference: MAP point estimate

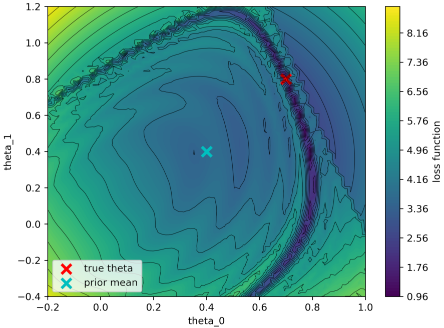
$$\min_{\boldsymbol{\theta}} \frac{1}{2} \left\|(d(t) - u_{\boldsymbol{\theta}}(t))/\sigma_{\text{noise}}\right\|_{L_2}^2 + \frac{1}{2} \left|\boldsymbol{\theta} - \bar{\boldsymbol{\theta}}_{\text{pr}}\right|_{\Sigma_{\text{pr}}^{-2}}^2 ,$$

Data: $d(t) = u_{\boldsymbol{\theta}_{\text{true}}}(t) + \eta(t)$

Noise: $\eta(t_i) = \rho\, \eta(t_{i-1}) + \epsilon(t_i), \quad \eta(t) \sim \mathcal{N}\left(0, \sigma^2/\Delta_t^2\right)$



Data of inverse problem *(orange dots)* is membrane potential $u_{\boldsymbol{\theta}}$ with added correlated noise η (AR process).



Highly nonlinear loss function of inverse problem with weak priors.

Challenges:

- Highly nonlinear and nonconvex loss
- Sharp gradients, strong nonlinear dependencies between parameters, multiple local minima
- Weak assumptions on regularization / prior, because little is known about the parameter values in practice

Outline

Introduce the forward and inverse problem

Propose approach: Reconstruction / inverse maps with dense and convolutional neural networks

Demonstrate parameter estimation capabilities

Inspect sensitivity of neural network estimates

Extend inference to ODE parameters and noise model parameters

Idea: Reconstruction maps based on deep artificial neural networks

Propose: Replace optimization of inverse problem by computationally learning reconstruction / inverse maps[2] using deep neural networks (NNs)

$$\underbrace{\hat{\boldsymbol{\theta}} := y_L}_{\text{NN output}}, \quad y_\ell = \mathcal{F}_\ell(y_{\ell-1}) \text{ for } 1 \le \ell \le L, \quad \underbrace{y_0 := \boldsymbol{d}}_{\text{NN input}}$$

- Observational data $\boldsymbol{d}$ (membrane potential + noise) is input to the NN
- Parameters of ODE $\hat{\boldsymbol{\theta}}$ are output of the NN
- NN is learning to directly represent a "pseudoinverse" of the forward operator
- Network layers $\mathcal{F}_\ell$ are dense, convolutional, or average pooling layers; swish activation

Limitations:

- No convergence analysis
- NN architecture has to be selected and optimized heuristically
- Generation of training data increases computational cost before inference can begin

[2] Adler and Öktem 2017; Khoo and Ying 2019; Fan, Bohorquez, and Ying 2019.

Idea: Use prior distribution to generate training data

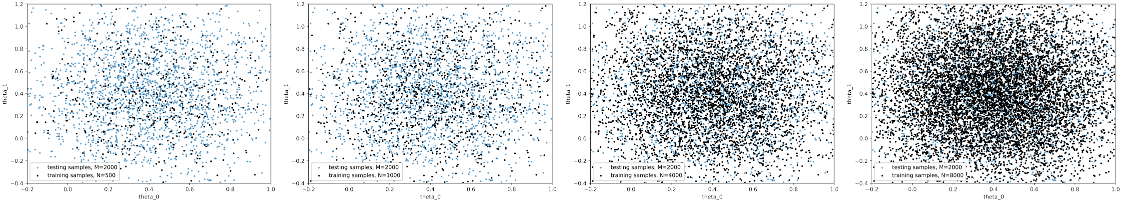
Propose: Sample parameters from prior distribution and simulate ODE

Training data: Gaussian prior with wide variance

$$\theta_0 \sim \mathcal{N}\left(0.4, 0.3^2\right), \quad \theta_1 \sim \mathcal{N}\left(0.4, 0.4^2\right)$$

and lower and upper bounds

$$-0.2 \leq \theta_0 \leq 1.0, \quad -0.4 \leq \theta_1 \leq 1.2$$



Testing data *(blue dots)* fixed ($M = 2000$); training data *(black dots)* increases ($N = 500, 1000, 4000, 8000$ *from left to right*).

Outline

Introduce the forward and inverse problem

Propose approach: Reconstruction / inverse maps with dense and convolutional neural networks

Demonstrate parameter estimation capabilities

Inspect sensitivity of neural network estimates

Extend inference to ODE parameters and noise model parameters

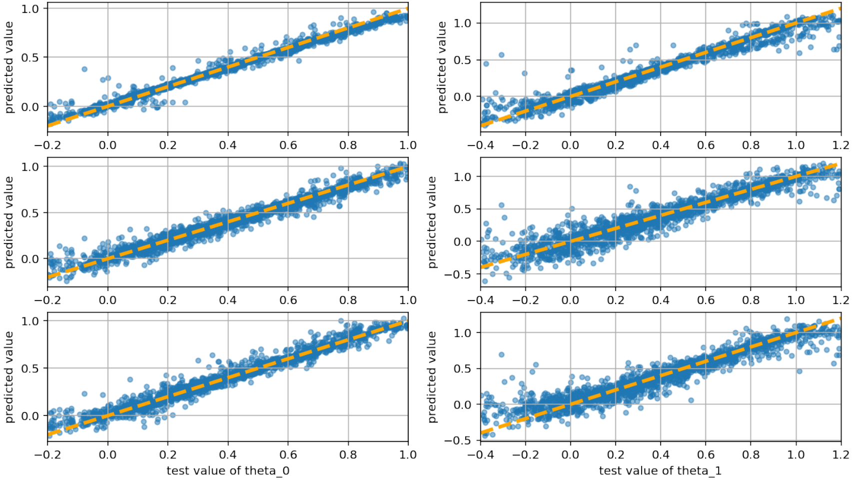
Result: Parameter estimation with dense neural network

Dense NN: 4 dense layers with 32 units (after optimization of network architectures)

train w/o noise
test w/o noise

train w/o noise
test w/ noise

train w/ noise
test w/ noise



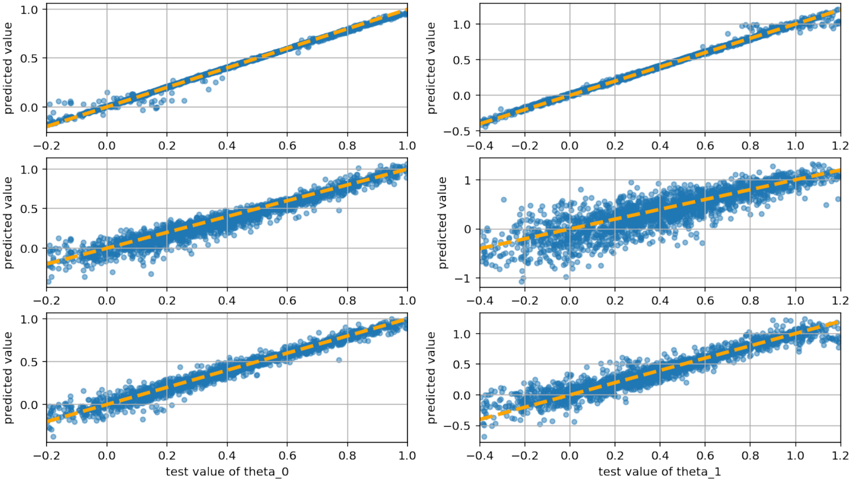
Result: Parameter estimation with convolutional neural network (CNN)

CNN: 3 conv. layers ([8,16,32] filters) and 2 dense layers (32 units) (after optimization of network architectures)

train w/o noise
test w/o noise

train w/o noise
test w/ noise

train w/ noise
test w/ noise

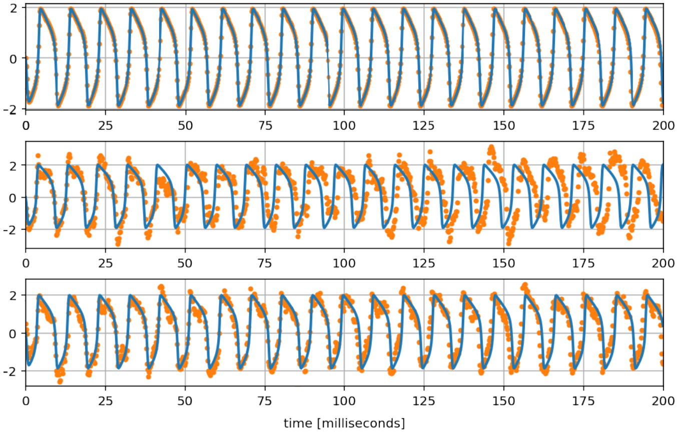


Simulated ODE output with parameters from CNN predictions

train w/o noise
test w/o noise

train w/o noise
test w/ noise

train w/ noise
test w/ noise



Time series error:
Sq. bias $= 2.8 \times 10^{-6}$
C-MSE $= 0.061$

Time series error:
Sq. bias $= 1.8 \times 10^{-3}$
C-MSE $= 2.001$

Time series error:
Sq. bias $= 1.2 \times 10^{-3}$
C-MSE $= 0.536$

Each graph shows the median percentile of MSE between testing and simulated time series.

Outline

Introduce the forward and inverse problem

Propose approach: Reconstruction / inverse maps with dense and convolutional neural networks

Demonstrate parameter estimation capabilities

Inspect sensitivity of neural network estimates

Extend inference to ODE parameters and noise model parameters

Sensitivity of NN predictions to partially observed time series

Want to know: Is a NN merely "remembering" a time series or can it learn underlying properties or dynamics?

Approach:

- For training: Split time series in half, 200 ms $\rightarrow$ 100 ms (doubles the amount of training samples)
- For testing: Choose random intervals of length 100 ms within 200 ms time frame
- New: Use time series, its Fourier transform, or combination of both as NN input

Predictions with partially observed time series, using a **dense NN** are relatively poor

Data type	Sq. bias	C-MSE	Med.-APE	R^2
Time	3.51×10^{-3}	0.0534	0.2632	0.475
Fourier	2.48×10^{-4}	0.0312	0.1478	0.620
Time & Fourier	4.27×10^{-3}	0.0468	0.2503	0.528

Predictions with partially observed time series, using a **CNN** are relatively accurate

Data type	Sq. bias	C-MSE	Med.-APE	R^2
Time	8.38×10^{-5}	0.003,34	0.0235	0.970
Fourier	1.70×10^{-4}	0.024,56	0.1235	0.685
Time & Fourier	8.59×10^{-6}	0.002,22	0.0289	0.980

Shown are "Sq. bias" (Squared bias), "C-MSE" (Centered Mean Squared Error), "Med.-APE" (Median of the Absolute Percentage Error), "R^2."

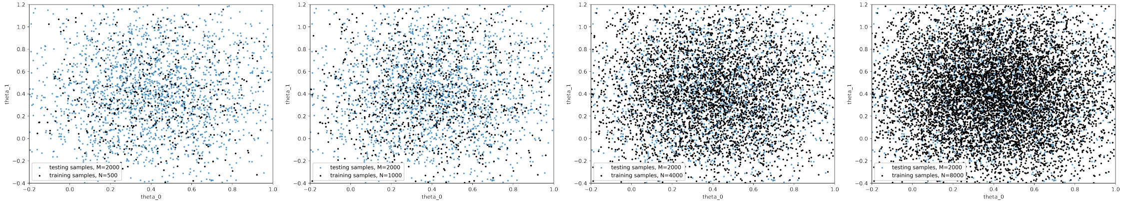
Sensitivity of NN predictions to training data sizes

Median-APE (R^2) of model parameter predictions, using a **dense NN**

N	train noise-free test noise-free	train noise-free test with noise	train with noise test with noise
500	0.043 (0.960)	0.098 (0.914)	0.105 (0.879)
1000	0.021 (0.978)	0.103 (0.918)	0.082 (0.921)
4000	0.014 (0.993)	0.089 (0.927)	0.061 (0.961)
8000	0.021 (0.992)	0.098 (0.921)	0.062 (0.968)

Median-APE (R^2) of model parameter predictions, using a **CNN**

N	train noise-free test noise-free	train noise-free test with noise	train with noise test with noise
500	0.023 (0.990)	0.169 (0.788)	0.098 (0.921)
1000	0.014 (0.995)	0.174 (0.763)	0.096 (0.938)
4000	0.014 (0.997)	0.204 (0.710)	0.060 (0.970)
8000	0.014 (0.998)	0.251 (0.617)	0.053 (0.976)



Testing data *(blue dots)* fixed ($M = 2000$); training data *(black dots)* increases ($N = 500, 1000, 4000, 8000$ *from left to right*).

Outline

Introduce the forward and inverse problem

Propose approach: Reconstruction / inverse maps with dense and convolutional neural networks

Demonstrate parameter estimation capabilities

Inspect sensitivity of neural network estimates

Extend inference to ODE parameters and noise model parameters

Extend inference to ODE parameters and noise model parameters

- Joint inference of parameters of (deterministic) physical models and of statistical models is rarely attempted with traditional methods; but estimation of noise can be relevant in cases where noise is unknown a-priori
- Challenging because of extremely different time scales in physical vs. statistical processes
- ODE: $\frac{\mathrm{d}u}{\mathrm{d}t} = \gamma\left(u - \frac{u^3}{3} + v + \zeta\right), \quad \frac{\mathrm{d}v}{\mathrm{d}t} = -\frac{1}{\gamma}\left(u - \theta_0 + \theta_1 v\right)$ Noise: $\eta(t_i) := \rho\,\eta(t_{i-1}) + \epsilon(t_i), \quad \eta(t) \sim \mathcal{N}(0, \sigma^2/\Delta_t^2)$

CNN: Median Absolute Percentage Error, R^2 in brackets ($R^2 = 1$ is ideal), N = #training samples

N	**Data type**	**FitzHugh–Nagumo parameter** θ_0	θ_1	**Noise parameter** σ	ρ
1000	Time	0.115 (0.914)	0.213 (0.812)	0.113 (−0.62)	0.063 (−0.80)
	Fourier	0.243 (0.460)	0.315 (0.577)	0.064 (0.524)	0.028 (0.603)
	Time & Fourier	0.103 (0.935)	0.192 (0.856)	0.058 (0.589)	0.028 (0.645)
8000	Time	0.070 (0.962)	0.138 (0.933)	0.058 (0.627)	0.030 (0.557)
	Fourier	0.162 (0.580)	0.215 (0.797)	0.051 (0.669)	0.023 (0.721)
	Time & Fourier	0.066 (0.968)	0.110 (0.942)	0.050 (0.684)	0.024 (0.722)

Thank you

References I

Adler, Jonas and Ozan Öktem (2017). "Solving ill-posed inverse problems using iterative deep neural networks." In: *Inverse Problems* 33.12, p. 124007. DOI: 10.1088/1361-6420/aa9581.

Fan, Yuwei, Cindy Orozco Bohorquez, and Lexing Ying (2019). "BCR-Net: A neural network based on the nonstandard wavelet form." In: *Journal of Computational Physics* 384, pp. 1–15. DOI: 10.1016/j.jcp.2019.02.002.

FitzHugh, Richard (1961). "Impulses and physiological states in theoretical models of nerve membrane." In: *Biophysical Journal* 1.6, pp. 445–466.

Khoo, Yuehaw and Lexing Ying (2019). "SwitchNet: A neural network model for forward and inverse scattering problems." In: *SIAM Journal on Scientific Computing* 41.5, A3182–A3201. DOI: 10.1137/18M1222399.

Nagumo, Jinichi, Suguru Arimoto, and Shuji Yoshizawa (1962). "An active pulse transmission line simulating nerve axon." In: *Proceedings of the IRE* 50.10, pp. 2061–2070.