Expressivity of Neural Networks with Random Weights and Learned Biases

Ezekiel Williams[1], Alexandre Payeur[1], Avery Hee-Woon Ryoo[1], Thomas Jiralerspong[1], Matthew Perich[1], Luca Mazzucato[2], Guillaume Lajoie[1] ✉ezekiel.williams@mila.quebec

Université de Montréal 1. UNIVERSITY OF OREGON 2.

Mila CIFAR NSERC CRSNG Fonds de recherche Nature et technologies Québec

Summary

When all weights are randomly initialized and frozen and only biases are optimized, which we term *bias learning*, we provide math proofs and experiments showing that for wide enough hidden layers:

1. Single-hidden layer Multi-Layer Perceptrons (MLPs) can arbitrarily approximate continuous functions with high probability.
2. Recurrent Neural Networks (RNNs) can arbitrarily approximate finite trajectories from continuous dynamical systems with high probability.

Motivation

Neuro: many biological network parameters are plastic beyond just synapses. In a rate model these are often modelled by bias terms (e.g. tonic input). Could the brain span a wide range of dynamics by adapting only such non-synaptic parameters?

AI: tuning weights and biases in a RNN (MLP) yields arbitrary approximation of many dynamical systems (functions) as hidden layer width increases. We don't know if such a result holds for bias learning, despite its relevance to multi-task and in-context learning.

Theoretical Results for RNNs

Consider the dynamical system, with compact invariant set, given by:

$$z[t+1] = F\big(z[t], x[t]\big)$$
$$y[t+1] = Cz[t+1],$$

where F is continuous, and the RNN:

$$r[t+1] = r[t] + \phi\big(Wr[t] + Bx[t] + b\big)$$
$$\hat{y}[t+1] = Ar[t+1],$$

where $r, b \in \mathbb{R}^m$.

Let $\theta \sim \text{unif}[-a, a], a > 0$ for θ any element of A, B or W.

For finite T, $\varepsilon > 0$, $\delta \in (0,1)$, we can find m, b, and $r_0(z_0)$ such that

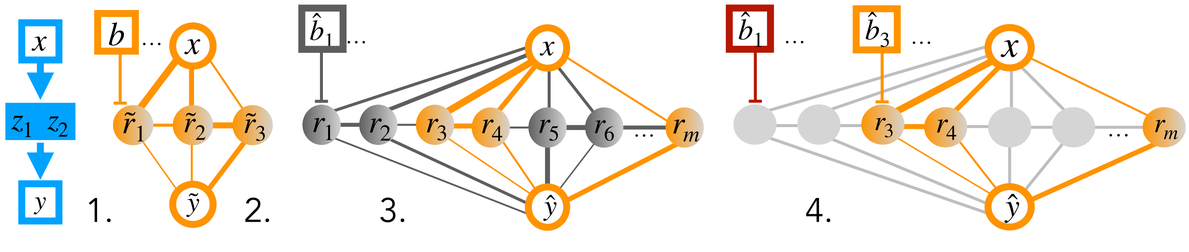
$$\sup_{z_0, \mathbf{x}} \sum_{t=1}^{T} || y_t(z_0, \mathbf{x}) - \hat{y}_t(r_0, \mathbf{x}) || < \varepsilon,$$

with probability greater than $1 - \delta$ for ϕ ReLU.

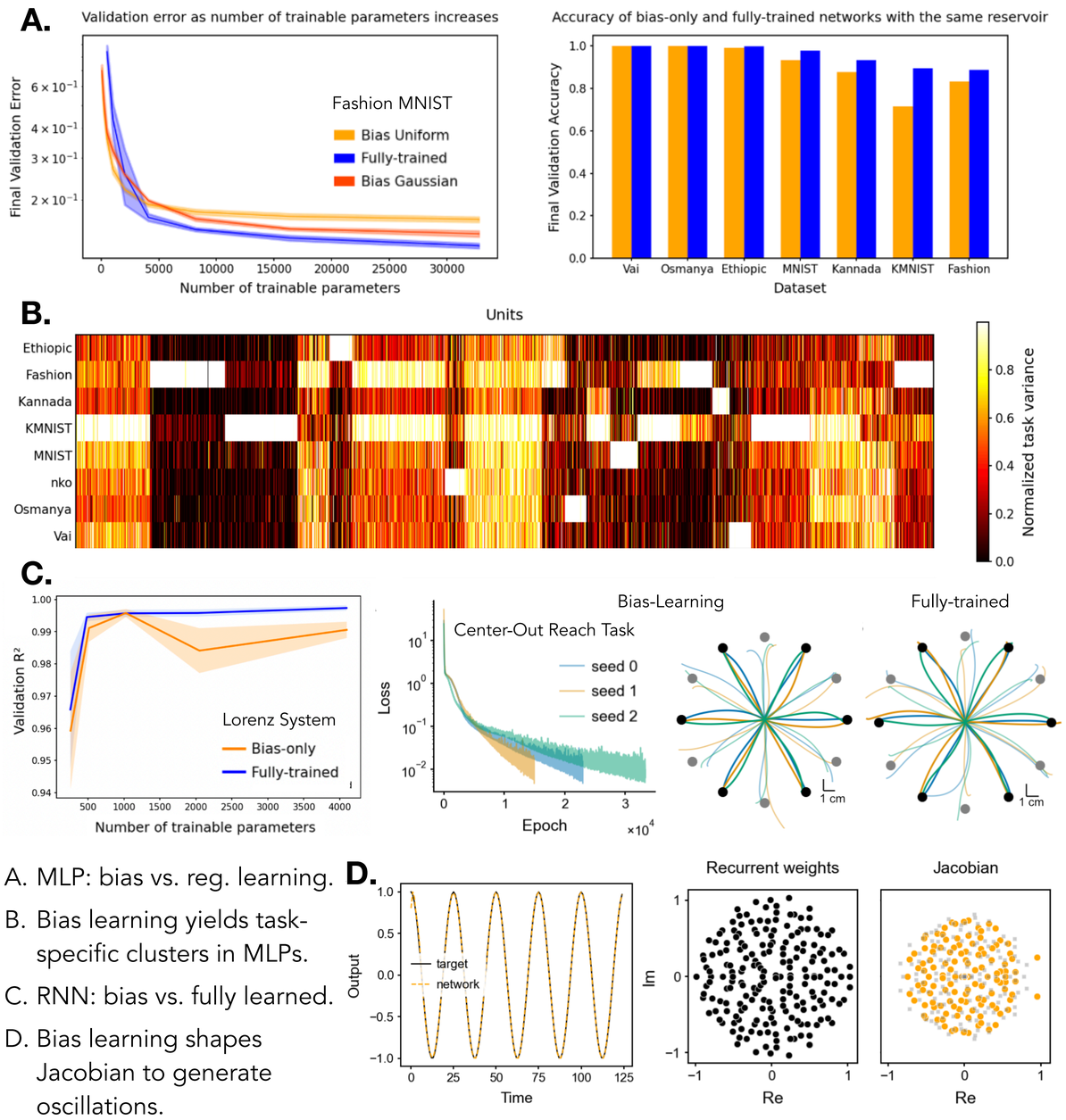
Result for MLPs in preprint

Theory Intuition

1. Consider a continuous, partially observed Dynamical System (**DS**).
2. Let $0 < t \leq T$. Approximate **DS** using **RNN** with trained weights and biases.
3. Randomly initialize a larger RNN. Check for sub-network with weights matching **RNN**.
4. Adjust biases outside (inside) sub-network to be very negative (match those of **RNN**).



Empirical Results



A. MLP: bias vs. reg. learning.
B. Bias learning yields task-specific clusters in MLPs.
C. RNN: bias vs. fully learned.
D. Bias learning shapes Jacobian to generate oscillations.

Conclusions and Future Directions

Synaptic plasticity is not necessarily required for learning task-relevant dynamics in sufficiently large neural networks.

Characterization of network size needed for given degree of error?

What are better distributions for initializing synaptic weights?

Might tonic inputs be adjusted for rapid learning?

Could bias learning yield simpler temporal credit assignment?

Link to Paper