Exploring Randomly Wired Neural Networks for Climate Model Emulation

William Yik[1,2] Sam Silva[2,3] Andrew Geiss[4] Duncan Watson-Parris[5]

[1]Harvey Mudd College, Claremont, California

[2]Department of Earth Sciences, University of Southern California
Los Angeles, California

[3]Department of Civil and Environmental Engineering, University of Southern California
Los Angeles, California

[4]Pacific Northwest National Laboratory, Richland, Washington

[5]Atmospheric, Oceanic and Planetary Physics, Department of Physics, University of Oxford
Oxford, UK

Table of Contents

1. Introduction and Background
2. Randomly Wired Neural Networks
3. Experiments and Results

Table of Contents

1. Introduction and Background
2. Randomly Wired Neural Networks
3. Experiments and Results

Motivation

- Modeling many emissions scenarios is important, but expensive.

Motivation

- Modeling many emissions scenarios is important, but expensive.
- Machine learning emulators can provide cheap, fast solutions. However, comparing emulators is difficult without standardized testing frameworks.

Motivation

- Modeling many emissions scenarios is important, but expensive.
- Machine learning emulators can provide cheap, fast solutions. However, comparing emulators is difficult without standardized testing frameworks.
- Watson-Paris et al. (2022) introduced Climatebench, a standardized dataset and testing framework

ClimateBench

- Benchmark Earth system model: Norwegian Earth System Model v2 (NorESM2)

ClimateBench

- Benchmark Earth system model: Norwegian Earth System Model v2 (NorESM2)
- Inputs: four primary drivers of NorESM2 experiments
  - Long-lived: CO_2, CH_4
  - Short-lived: SO_2, BC

ClimateBench

- Benchmark Earth system model: Norwegian Earth System Model v2 (NorESM2)
- Inputs: four primary drivers of NorESM2 experiments
  - Long-lived: CO_2, CH_4
  - Short-lived: SO_2, BC
- Outputs: four predicted output variables (annual means)
  - Surface air temperature (TAS)
  - Diurnal temperature range (DTR)
  - Precipitation (PR)
  - 90th percentile of precipitation (PR90)

Table of Contents

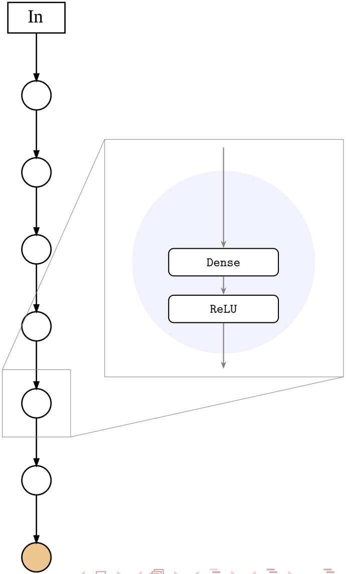
1 Introduction and Background

2 Randomly Wired Neural Networks

3 Experiments and Results

Multilayer Perceptron (MLP)

- Most basic type of neural network
- Information flows in one direction from one layer to the next
- White circle: hidden dense layer and activation function



RandDense Networks

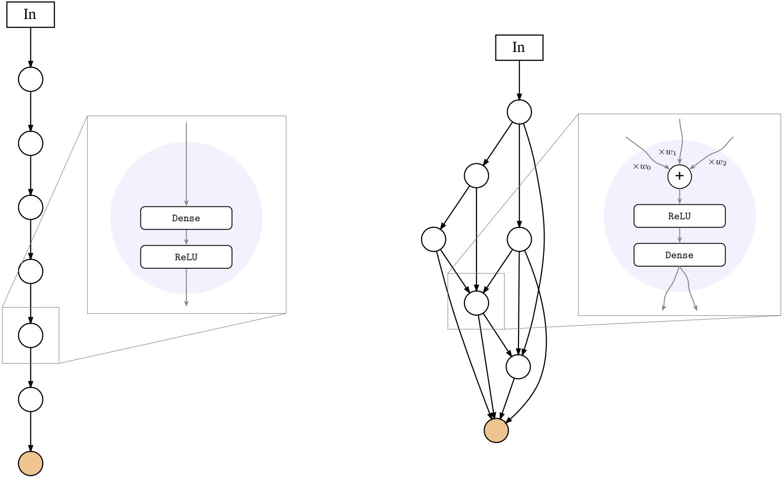


Table of Contents

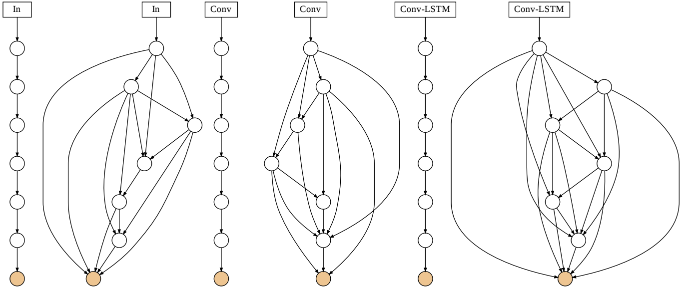
1 Introduction and Background

2 Randomly Wired Neural Networks

3 Experiments and Results

Experimental Setup

- Three baseline architectures: MLP, convolutional neural network (CNN), convolutional long short-term memory network (CNN-LSTM)
- 2-10 hidden layers, 1M and 10M parameters
- Generate 50 standard networks and 50 randomly wired networks for comparison

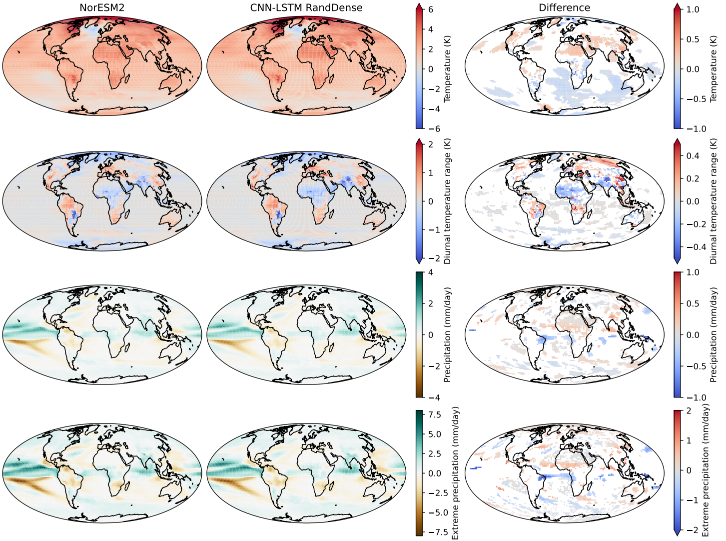


Results

		TAS	DTR	PR	PR90
MLP	Standard	1.928	15.62	4.663	5.651
	RandDense	**1.612**	**14.67**	**4.472**	**5.206**
CNN	Standard	**3.350**	23.15	9.235	10.30
	RandDense	3.353	**22.92**	**8.681**	**9.964**
CNN-LSTM	Standard	**0.262**	11.85	2.861	3.880
	RandDense	0.263	**11.66**	**2.775**	**3.810**
ClimateBench		0.327	16.78	3.175	4.339

Table 1: Best total RMSE performance for each model class and predicted variable across all generated models, along with the original CNN-LSTM model from Watson-Parris et al. (2022). Lower is better, and the better RMSE between the standard and RandDense models is bolded.

NorESM2 vs. CNN-LSTM RandDense



Summary and Conclusion

Key takeaways

- Randomization appears to provide performance benefits in multiple models!
- Same prediction speed as standard models
- Suggests replacing dense layers with randomly wired ones for the task of climate model emulation

Summary and Conclusion

Key takeaways

- Randomization appears to provide performance benefits in multiple models!
- Same prediction speed as standard models
- Suggests replacing dense layers with randomly wired ones for the task of climate model emulation

Thank you! Correspondence to wyik@hmc.edu.