Foundational Methods for Foundation Models for Scientific Machine Learning

Michael W. Mahoney

ICSI, LBNL, & Dept of Statistics, UC Berkeley

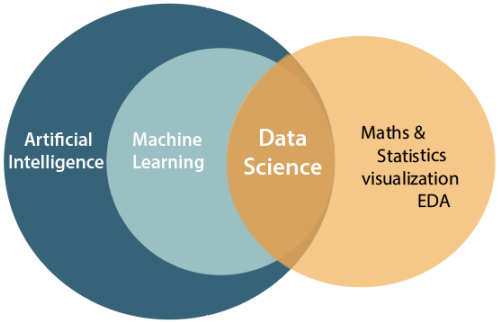
December 2024

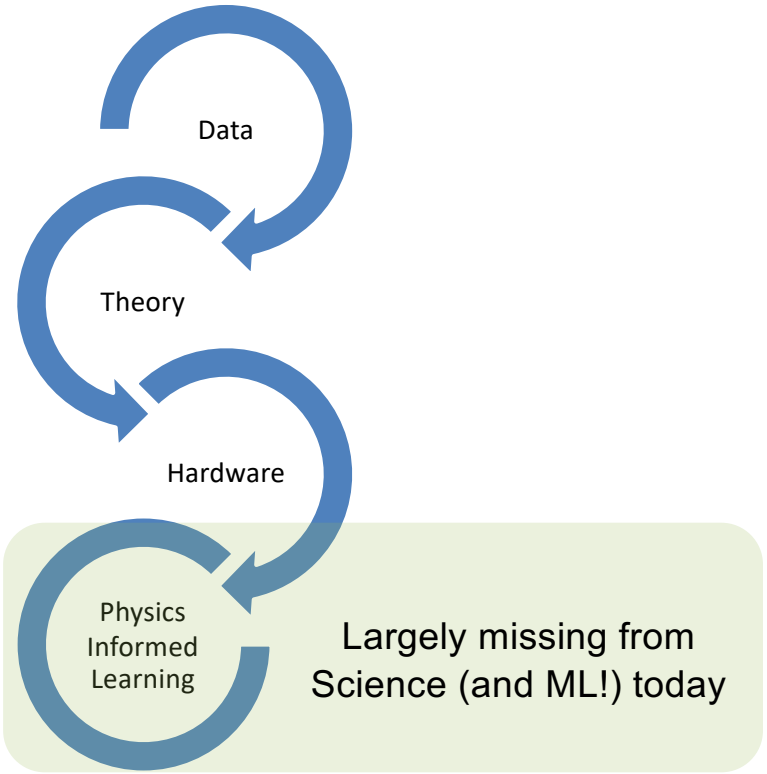
Physics Informed Learning?

Computational Science is an important tool that we can use to incorporate physical invariances into learning, but until recently it was missing from mainstream ML.

"**Computational Science** can **analyze past events** and **look into the future**. It can explore the effects of thousands of scenarios for or in lieu of actual experiment and be used to study events beyond the reach of expanding the boundaries of experimental science"
–Tinsley Oden, 2013

To make further progress in ML it is crucial that we incorporate computational science into learning.





J. Tinsley Oden's Commemorative Speech: "THE THIRD PILLAR: The Computational Revolution of Science and Engineering", Honda Prize, 2013.

Questions?

- Q0: What is a “Foundation Model”?
- Q1: Can we hope to train a “Foundation Model” for SciML?
- Q2: Would incorporating physical knowledge help? If so, how to do it?
- Q3: Foundations?
- Q4: Implementations?
- Q6: Applications?
- Q6: Looking forward?

Questions?

- **Q0: What is a "Foundation Model"?**
- Q1: Can we hope to train a "Foundation Model" for SciML?
- Q2: Would incorporating physical knowledge help? If so, how to do it?
- Q3: Foundations?
- Q4: Implementations?
- Q6: Applications?
- Q6: Looking forward?

What is a foundation model?

General-purpose technologies that can support a diverse range of use cases.

Built using well-established techniques from ML:

- NNs, self-supervised learning, transfer learning, etc.

New paradigm in ML:

- general-purpose models are “reusable infrastructure,” instead of bespoke/one-off solutions
- building foundation models is highly resource-intensive (100M - 1B USD, people, data, compute)
- adapting a foundation model for a specific use case or using it directly is much less expensive.

Term was created/popularized by Stanford Institute for Human-Centered Artificial Intelligence (HAI) Center for Research on Foundation Models (CRFM):

- Bommasani et al. ``On the Opportunities and Risks of Foundation Models'' arXiv:2108.07258.

What is a foundation model?

Other possible names:

- large language model - too narrow, given the focus is not only language
- self-supervised model - too specific, to the training objective
- pretrained model - suggests the important action happened after pretraining
- foundational model - suggests the model provides fundamental principles

Foundation model:

- emphasize the intended function (i.e., amenability to subsequent further development) rather than modality, architecture, or implementation.

Early examples were language models (LMs) like Google's BERT and OpenAI's GPT-n series.

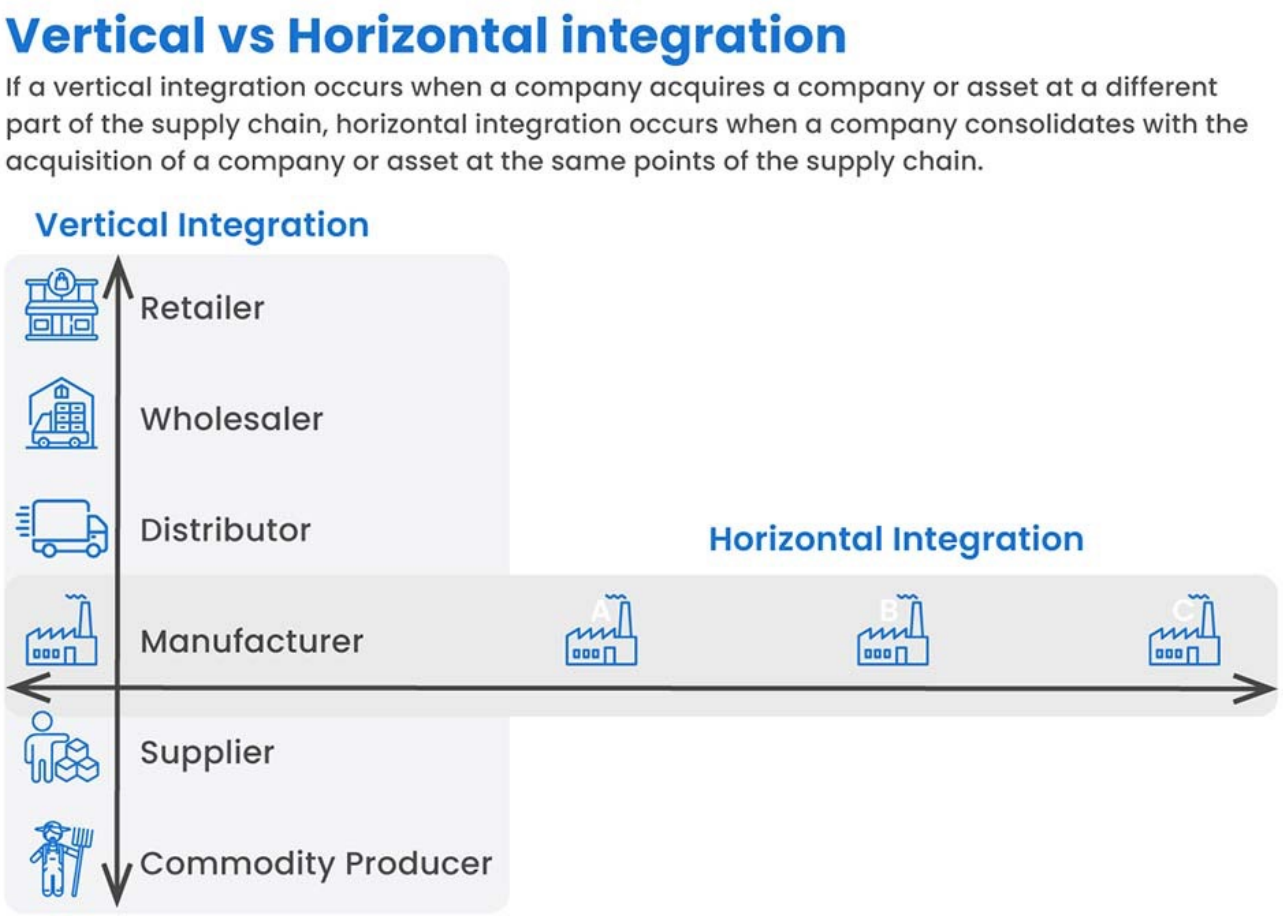
More recently, developed across a range of modalities:

- images; music; time series; robotic control; etc. (?)
- Lots of areas of science: astronomy, radiology, climate, genomics, coding, mathematics, etc. (?)

Questions?

- Q0: What is a “Foundation Model”?
- **Q1: Can we hope to train a “Foundation Model” for SciML?**
- Q2: Would incorporating physical knowledge help? If so, how to do it?
- Q3: Foundations?
- Q4: Implementations?
- Q6: Applications?
- Q6: Looking forward?

How to view Scientific Machine Learning



How to view Scientific Machine Learning

ML is a "horizontal":

- Provides a standard applicable across multiple cross-areas
- Like the iphone, or roads/railroads, or energy infrastructure, or HPC

Domain Sciences are "verticals":

- They own domain acquisition, insight, analysis, interpretation, etc.
- You need to be a domain expert to push state of the art

High-profile successes of SciML have taken place in industry:

- "Horizontal" companies that provide tech platforms and have lots of ML expertise
- Not "vertical" companies that know one science domain and use ML for that one goal

What do business leaders care about?

- No CEO cares about ML; they care about money
- Winners are those who invest heavily in this "means to an end" ML infrastructure

What might be possible with a meaningfully *Scientific* FM?

Train on data from:

- Atmosphere: Climate and Weather processes
- Land: Water and Ecosystem processes
- Subsurface: Heterogeneous flows and seismicity
- Language models: e.g., if you want to learn 1/f noise

Transfer learn on data from:

- Astronomy: to discover habitable exoplanets
- Materials Science: to learn physics across scales in an end-to-end way
- Chemistry: to learn interatomic potentials for MD simulation
- Fire/Floods/Etc.: to learn distributions of extreme events well enough to create insurance markets
- Nuclear Physics: to learn classified data from public data

How is this even possible? My data are special/unique?

- No; NOT so.
- You are NOT so unique/special: ML algorithms predict movies you watch better than you do

Just call ChatGPT? Or apply the M.O. of ML to Science?

Option 1:

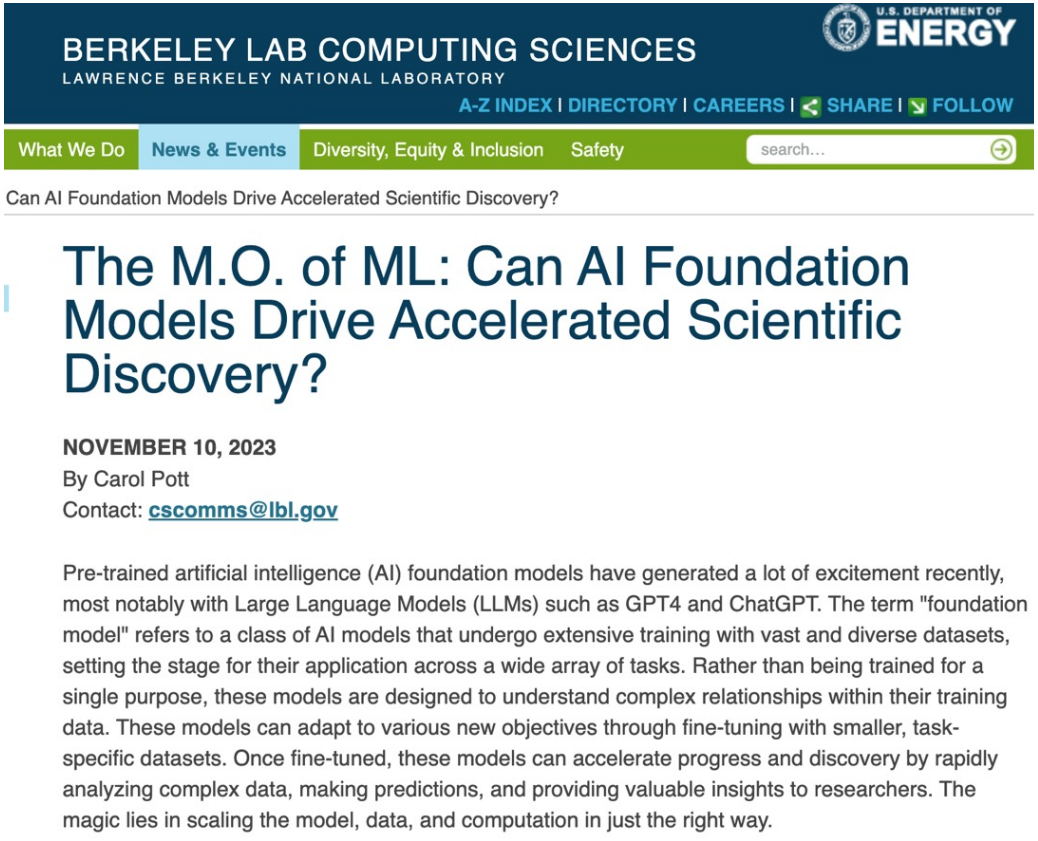
- Ask ChatGPT (or whatever LLM), post fine-tuning, to hypothesize new drugs, or what comes after the Top quark, or …

Option 2:

- Use ChatGPT embeddings in a model for some other scientific objective.

Option 3:

- Understand the *methodology of ML**
- Apply that methodology to Scientific data
- Multi-modal Scientific data could be text
- It could be simulation, experiment, etc.
- Incorporate spatio-temporal inductive biases into architecture and compute
- Develop foundations for SciML



BERKELEY LAB COMPUTING SCIENCES
LAWRENCE BERKELEY NATIONAL LABORATORY
U.S. DEPARTMENT OF ENERGY
A-Z INDEX | DIRECTORY | CAREERS | SHARE | FOLLOW
What We Do | News & Events | Diversity, Equity & Inclusion | Safety | search…
Can AI Foundation Models Drive Accelerated Scientific Discovery?

The M.O. of ML: Can AI Foundation Models Drive Accelerated Scientific Discovery?

NOVEMBER 10, 2023
By Carol Pott
Contact: cscomms@lbl.gov

Pre-trained artificial intelligence (AI) foundation models have generated a lot of excitement recently, most notably with Large Language Models (LLMs) such as GPT4 and ChatGPT. The term "foundation model" refers to a class of AI models that undergo extensive training with vast and diverse datasets, setting the stage for their application across a wide array of tasks. Rather than being trained for a single purpose, these models are designed to understand complex relationships within their training data. These models can adapt to various new objectives through fine-tuning with smaller, task-specific datasets. Once fine-tuned, these models can accelerate progress and discovery by rapidly analyzing complex data, making predictions, and providing valuable insights to researchers. The magic lies in scaling the model, data, and computation in just the right way.

**Scale data size, model size, and compute so none of them saturate, then transfer learn.*

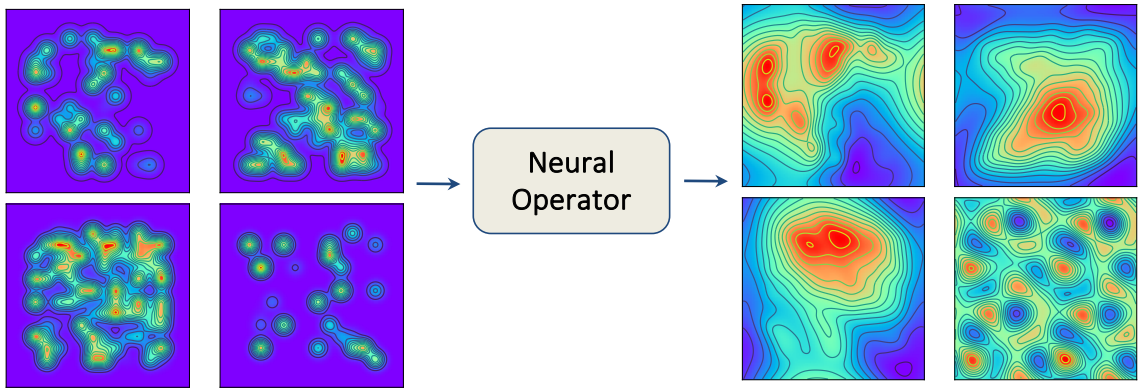
The M.O. of ML: Foundation models for SciML?

Scale data size, model size, and compute so none of them saturate, then transfer learn.

Create and pre-train on diverse PDE systems

Vary/Sample all inputs (PDE coefficients, source functions, …)
Include multiple differential operators, predict PDE solution

$$\nabla \cdot \mathrm{K} \nabla u + v \cdot \nabla u + \ \ldots = f$$

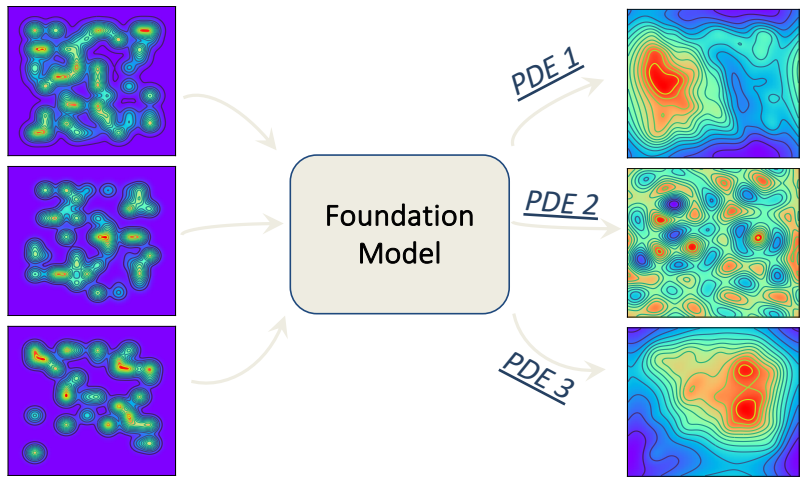


Inputs
f, K, v, …

Outputs
u

Foundation Models for SciML

Solve multiple systems using the same pre-trained model, outperforming training from scratch

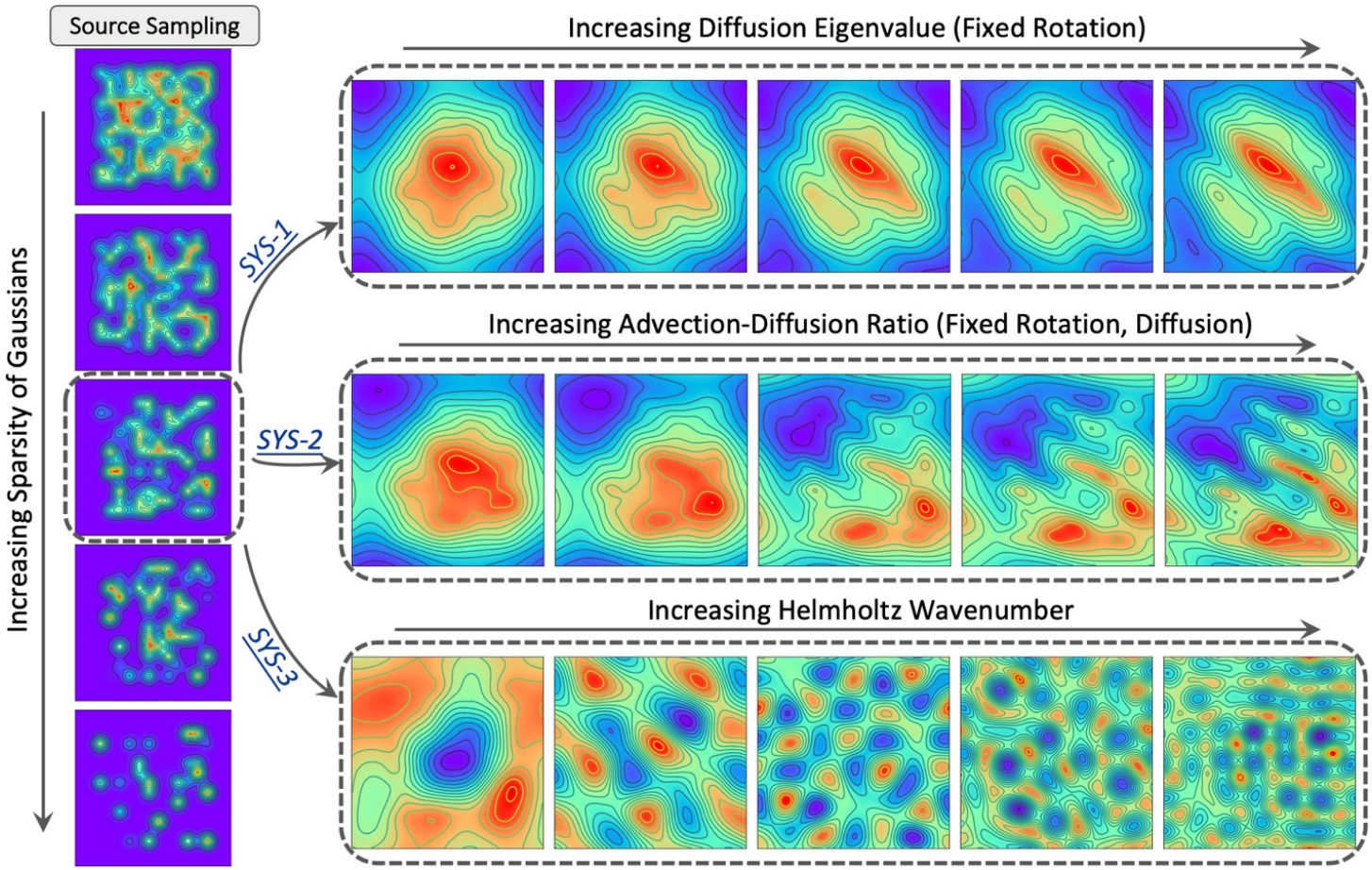


Inputs
f, K, v, …

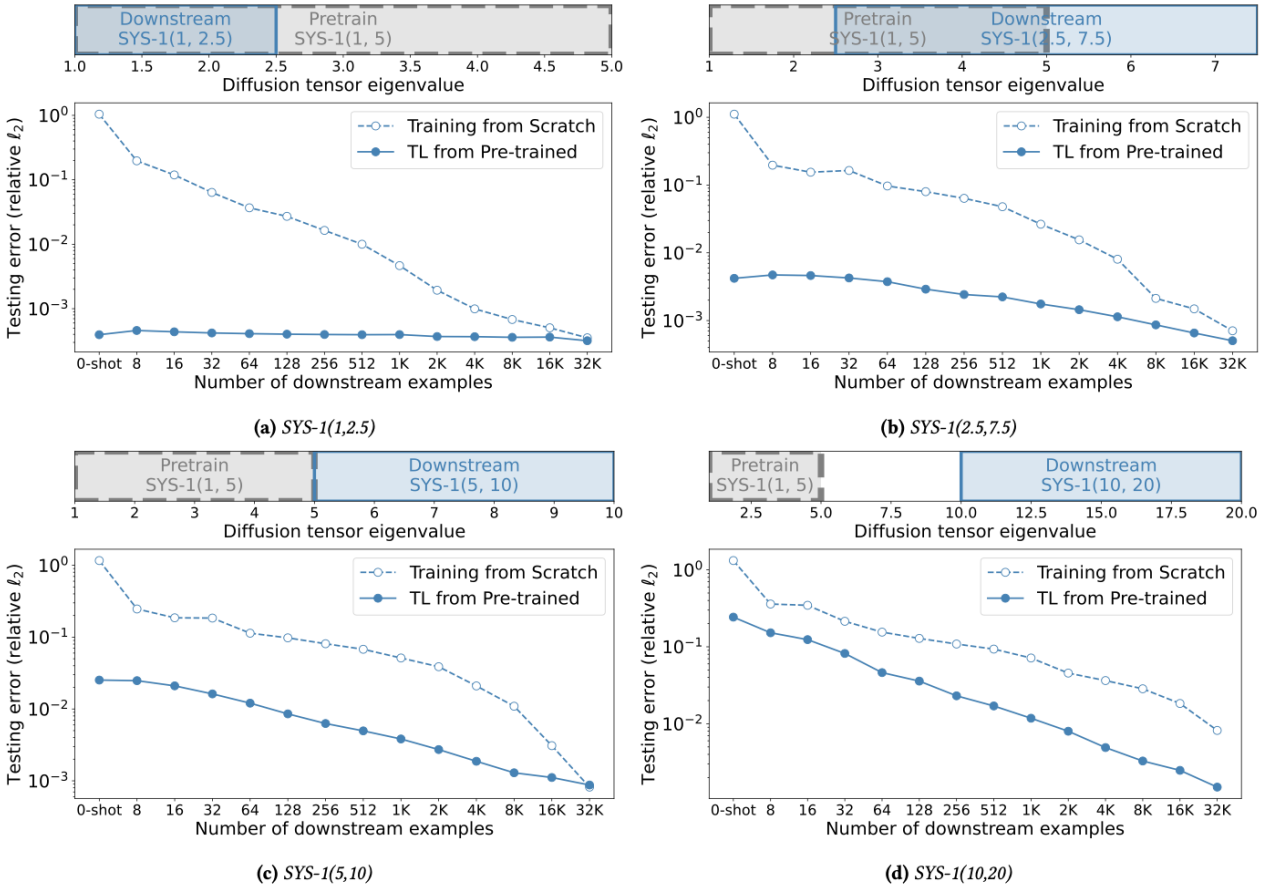
Outputs
u

"Towards Foundation Models for Scientific Machine Learning: Characterizing Scaling and Transfer Behavior," Subramanian, Harrington, Keutzer, Bhimji, Morozov, Mahoney, and Gholami, arXiv:2306.00258, NeurIPS23.

The M.O. of ML: Physics control knobs for changing solutions

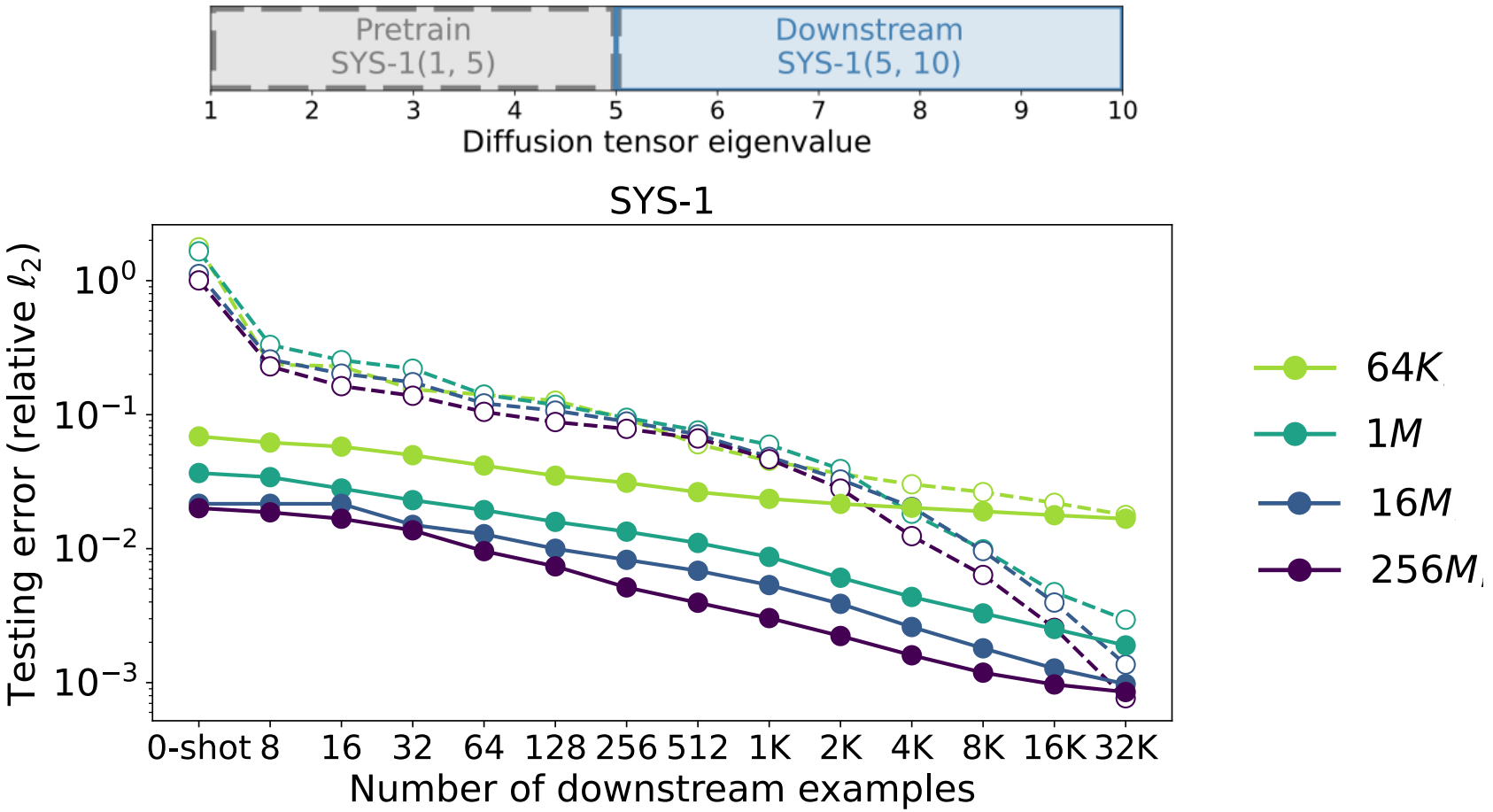


The M.O. of ML: OOD transfer behavior



"Towards Foundation Models for Scientific Machine Learning: Characterizing Scaling and Transfer Behavior," Subramanian, Harrington, Keutzer, Bhimji, Morozov, Mahoney, and Gholami, arXiv:2306.00258, NeurIPS23.

Transfer behavior with model size



Many open problems/limitations

- **Going beyond simulation data to with "real" experimental/observational data**
  - Our data comes from numerical simulations of dynamical systems with known coefficients.
  - Need to test data from observations or (simulations + observations)

- **NN architecture**
  - We did not change the FNO model architecture.
  - We know it is not the right model for all kinds of SciML problems.

- **More complex PDEs**
  - 3D, time, space-time, high-resolution, multi-scale

- **Self-supervision in pre-training**
  - Physics losses, spatiotemporal masking (from CV)
  - Inductive biases to be continuous, well-posed w.r.t. constraints/discontinuities

Unsupervised pretraining and in-context learning?

Computer Science > Machine Learning

[Submitted on 24 Feb 2024]

Data-Efficient Operator Learning via Unsupervised Pretraining and In-Context Learning

Wuyang Chen, Jialin Song, Pu Ren, Shashank Subramanian, Dmitriy Morozov, Michael W. Mahoney

Recent years have witnessed the promise of coupling machine learning methods and physical domain-specific insight for solving scientific problems based on partial differential equations (PDEs). However, being data-intensive, these methods still require a large amount of PDE data. This reintroduces the need for expensive numerical PDE solutions, partially undermining the original goal of avoiding these expensive simulations. In this work, seeking data efficiency, we design unsupervised pretraining and in-context learning methods for PDE operator learning. To reduce the need for training data with simulated solutions, we pretrain neural operators on unlabeled PDE data using reconstruction-based proxy tasks. To improve out-of-distribution performance, we further assist neural operators in flexibly leveraging in-context learning methods, without incurring extra training costs or designs. Extensive empirical evaluations on a diverse set of PDEs demonstrate that our method is highly data-efficient, more generalizable, and even outperforms conventional vision-pretrained models.

Subjects: **Machine Learning (cs.LG)**; Machine Learning (stat.ML)
Cite as: arXiv:2402.15734 **[cs.LG]**
(or arXiv:2402.15734v1 **[cs.LG]** for this version)
https://doi.org/10.48550/arXiv.2402.15734

"Data-Efficient Operator Learning via Unsupervised Pretraining and In-Context Learning," Chen, Song, Ren, Subramanian, Morozov, and Mahoney, arXiv:2402.15734

Questions?

- Q0: What is a “Foundation Model”?
- Q1: Can we hope to train a “Foundation Model” for SciML?
- **Q2: Would incorporating physical knowledge help? If so, how to do it?**
- Q3: Foundations?
- Q4: Implementations?
- Q6: Applications?
- Q6: Looking forward?

Combining domain-driven and data-driven models?

Characterizing possible failure modes in physics-informed neural networks

Aditi S. Krishnapriyan[*,1,2], **Amir Gholami**[*,2],
Shandian Zhe[3], **Robert M. Kirby**[3], **Michael W. Mahoney**[2,4]
[1]Lawrence Berkeley National Laboratory, [2]University of California, Berkeley,
[3]University of Utah, [4]International Computer Science Institute
{aditik1, amirgh, mahoneymw}@berkeley.edu, {zhe, kirby}@cs.utah.edu

Abstract

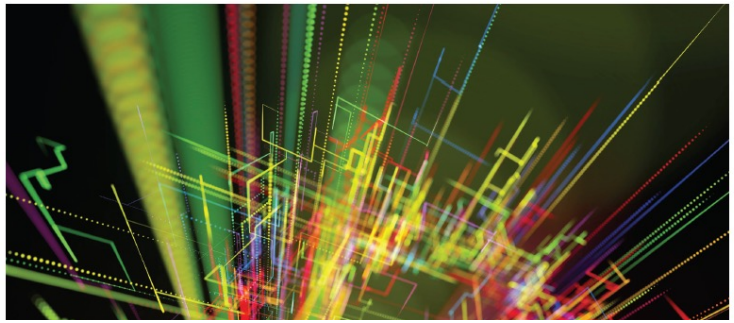
Recent work in scientific machine learning has developed so-called physics-

Science | DOI:10.1145/3524015 Chris Edwards

Neural Networks Learn to Speed Up Simulations

Physics-informed machine learning is gaining attention, but suffers from training issues.

PHYSICAL SCIENTISTS AND engineering research and development (R&D) teams are embracing neural networks in attempts to accelerate their simulations. From quantum mechanics to the prediction of blood flow in the body, numerous teams have reported on speedups in simulation by swapping conventional finite-element solvers for models trained on various combinations of experimental and synthetic data.



Methods for Incorporating Physics into Learning

- **Method 1:** Enforce physical laws as hard constraints either in:
  - NN Architecture: still an open problem
  - Optimization: very difficult to train the NN with such constraints

- **Method 2:** Train on lots of data and let NN learn physics based operators
  - Neural Operator like methods

- **Method 3:** Use penalty methods and add the PDE residual to the loss.
  - PINN like methods: *very* easy to implement with any NN architecture

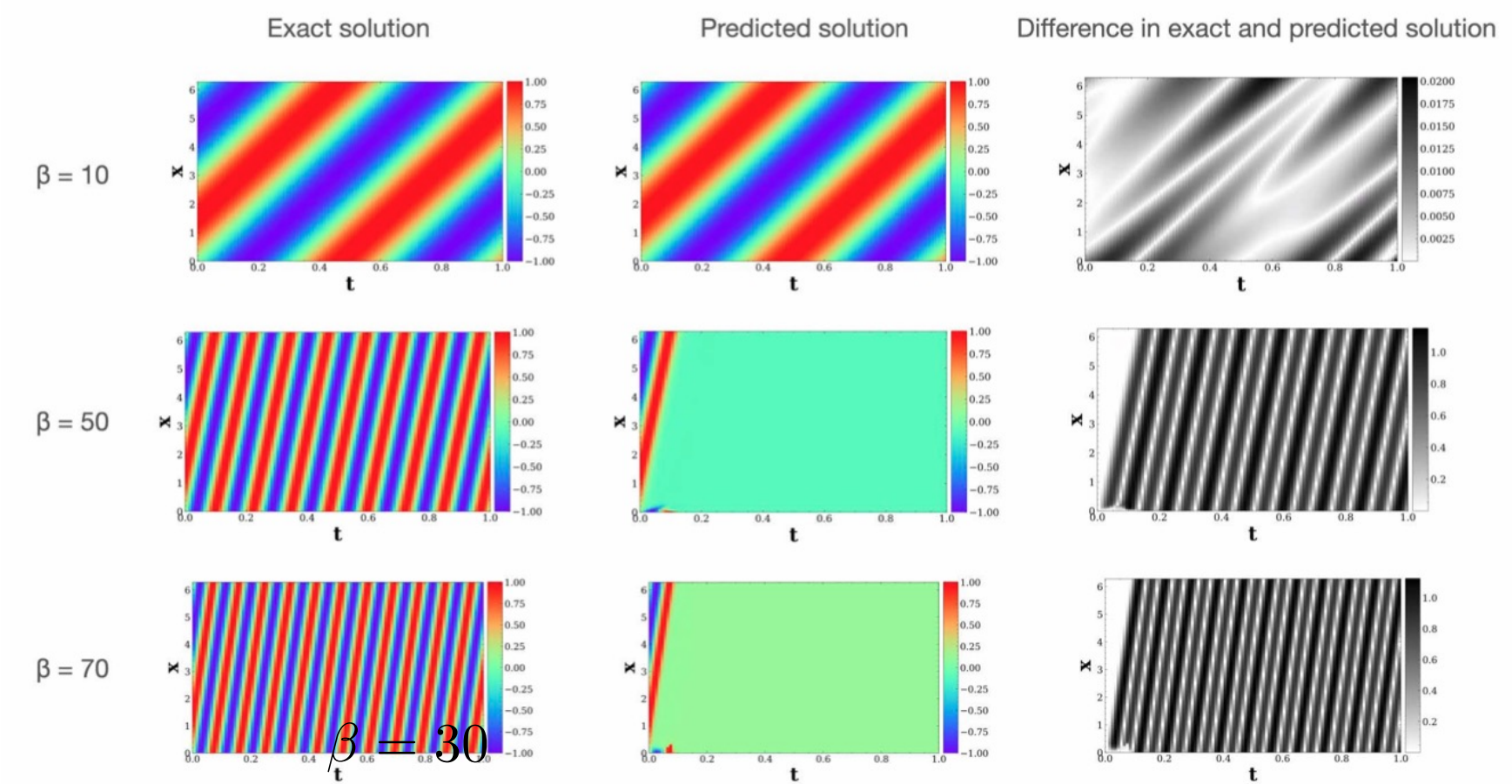
- **Method 4:** Use a combination of Neural Operator and PINNs
  - Uses a combination of observation data points as well as physical constraints added as soft penalty to the loss

Xu K, Darve E. Physics constrained learning for data-driven inverse modeling from sparse observations. arXiv preprint arXiv:2002.10521. 2020 Feb 24.
Raissi M, Perdikaris P, Karniadakis GE. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. Journal of Computational Physics. 2019 Feb 1;378:686-707.
*[Weinan et al. 2017; **Raissi et al. 2019**; Rackauckas et al. 2020; Hennigh et al. 2021; Lu et al. 2021]*
Li et al. 2021]
Etc.

Sounds good … but this is not the entire story …

- There are a lot of subtleties in adding a soft-constraint:
  - Methods actually do not work so well, even for simple problems
- To study this, we chose three families of PDEs:
  - Advection (aka wave equation)
  - Reaction
  - Reaction-Diffusion
- Soft-constrained PINN-like models fail to learn relevant physics in all these cases
  - Since there are many moving parts to ML training
  - The relevant ML methodologies don't play well with scientific methodologies
  - Reasons for the failure modes are interesting and informative

PINN can fail to learn Advection*



*Nothing special about advection: same holds true for other PDEs..

*Krishnapriyan AS, **Gholami*A**, Zhe S, Kirby RM, Mahoney MW. Characterizing possible failure modes in physics-informed neural networks. NeurIPS, 2021.*

Training: Optimization Challenges with PINNs

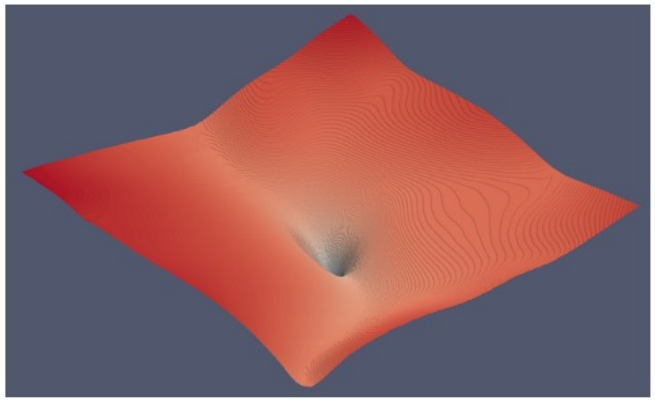
Data Loss Function:

$$\mathcal{L}_u = \|\hat{u} - u\|_2^2$$

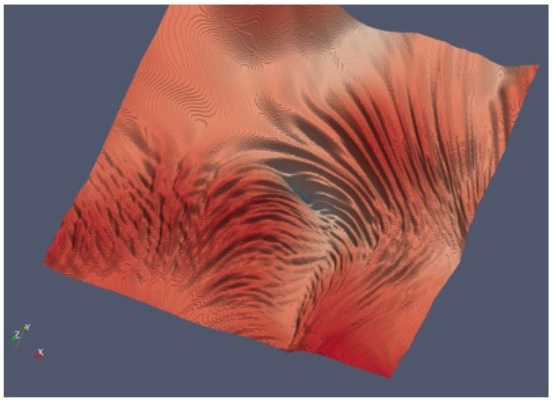
Physics Loss Function:

$$\mathcal{L}_{\mathcal{F}} = \|\hat{u}_t + \hat{u}\hat{u}_x - \hat{u}_{xx}\|_2^2$$

$$\min_{\theta} \mathcal{L} = \mathcal{L}_u + \lambda_{\mathcal{F}} \mathcal{L}_{\mathcal{F}}$$



Without Physics Loss



With Physics Loss

Illustration credit: Roman Amici, Mike Kirby

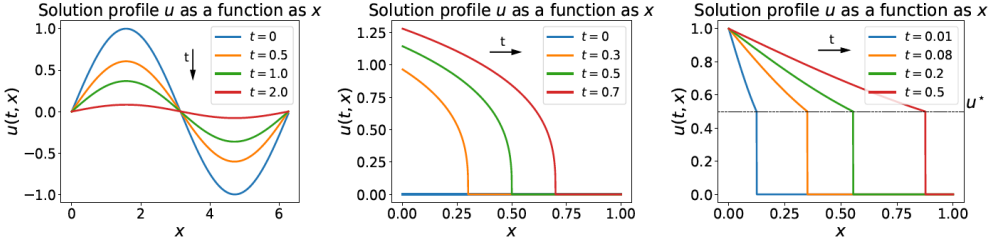
Krishnapriyan AS, **Gholami* A**, Zhe S, Kirby RM, Mahoney MW. Characterizing possible failure modes in physics-informed neural networks. NeurIPS, 2021.*

Questions?

- Q0: What is a “Foundation Model”?
- Q1: Can we hope to train a “Foundation Model” for SciML?
- Q2: Would incorporating physical knowledge help? If so, how to do it?
- **Q3: Foundations?**
- Q4: Implementations?
- Q6: Applications?
- Q6: Looking forward?

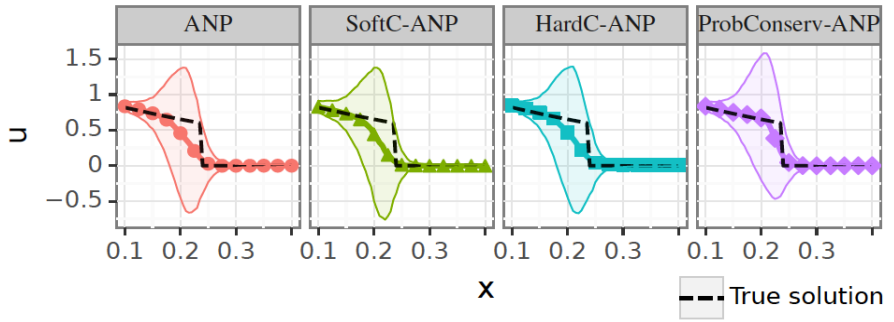
One way to address failure modes: ProbConserve

1. Compute mean and variance estimates
2. Update model (with oblique projection, depending on heteroscedasticity structure)
3. Good for sharp discontinuities

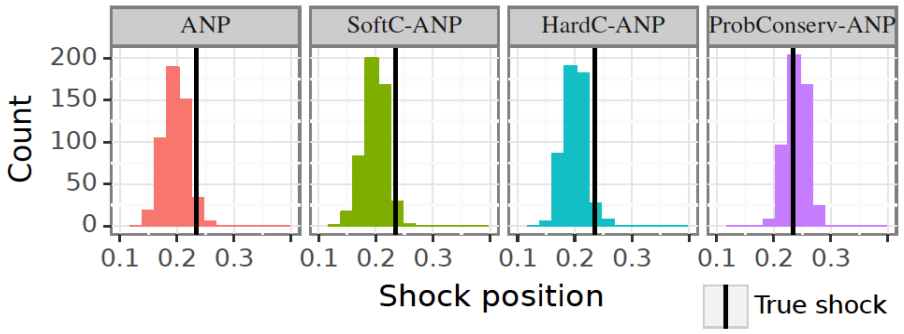


(a) easy: Diffusion equation ($k = 1$) (b) medium: PME ($k(u) = u^3$) (c) hard: Stefan (discont. $k(u)$)

Figure 1: Illustration of the "easy-to-hard" paradigm for PDEs, for the GPME family of conservation equations: (a) "easy" parabolic smooth (diffusion equation) solutions, with constant parameter $k(u) = k \equiv 1$; (b) "medium" degenerate parabolic PME solutions, with nonlinear monomial coefficient $k(u) = u^m$, with parameter $m = 3$ here; and (c) "hard" hyperbolic-like (degenerate parabolic) sharp solutions (Stefan equation) with nonlinear step-function coefficient $k(u) = \mathbf{1}_{u \geq u^\star}$, where $\mathbf{1}_{\mathcal{E}}$ is an indicator function for event $\mathcal{E}$.



(a) Solution profile.



(b) Posterior of the shock position.

"Learning Physical Models that Can Respect Conservation Laws," Hansen, Maddix, Alizadeh, Gupta, and Mahoney, arXiv:2302.11002, ICML23, Physica D (2024)
"Using Uncertainty Quantification to Characterize and Improve Out-of-Domain Learning for PDEs," Mouli, Maddix, Alizadeh, Gupta, Stuart, Mahoney, Wang, arXiv:2403.10642

Foundations more generally

Illustrative recent proof-of-principle directions:

- ContinuousNet: “numerical” convergence tests
- Traditional vs Modern ML UQ: Over- vs under-parameterized models
- Weight diagnostics: WeightWatcher Analysis and HTSR
- Time Series: LEM, ConvLEM, Chronos

Foundations more generally: ContinuousNet

1. Convergence test based on numerical analysis theory
2. Verifies whether a model has learned an underlying continuous dynamics
3. Good for super-resolution, iterative dynamics, etc.
4. Applies to NNs, SINDy, etc.

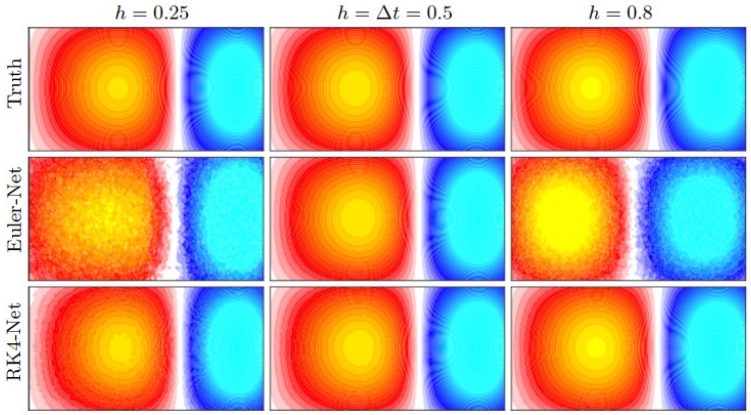
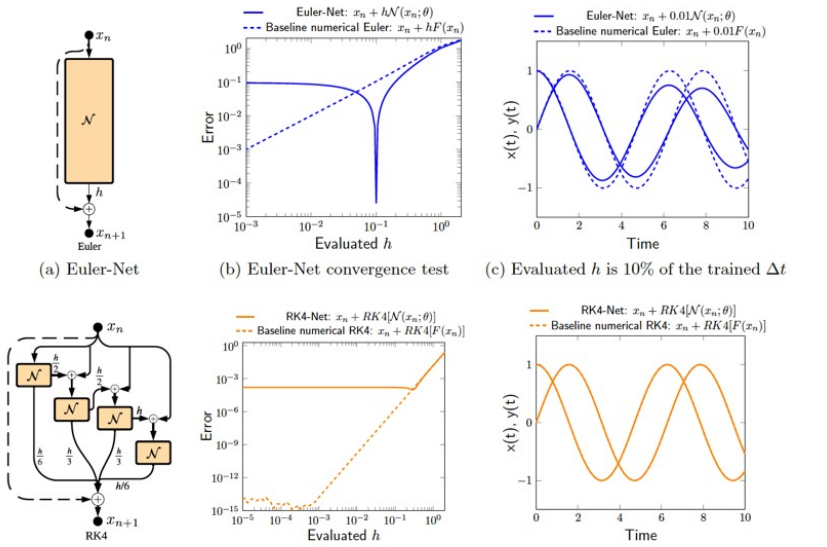


Figure 4: ***Double gyre fluid flow: Reconstructing fine-scale flow fields from coarse training***



(a) Euler-Net (b) Euler-Net convergence test (c) Evaluated h is 10% of the trained Δt

(d) RK4-Net (e) RK4-Net convergence test (f) Evaluated h is 10% of the trained Δt

Figure 2: ***Illustration of our convergence test with different ODE-Nets.*** *(a) Schematic of an ODE-Net*

"Learning continuous models for continuous physics," Krishnapriyan, Queiruga, Erichson, and Mahoney, arXiv:2202.08494, Comm Phys (2023)
"Continuous-in-Depth Neural Networks," Queiruga, Erichson, Taylor, and Mahoney, arXiv:2008.02389

Foundations more generally: traditional vs modern ML UQ

Traditional UQ versus Modern UQ in overparameterized vs underparameterized models

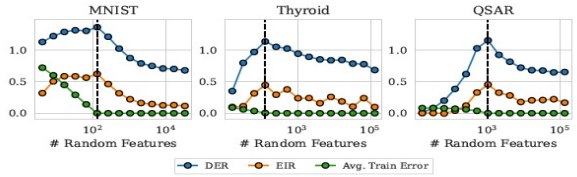


Figure 3: **Bagged random feature classifiers.** Blacked dashed line represents the interpolation threshold. Across all tasks, DER and EIR are maximized at this point, and then decrease thereafter.

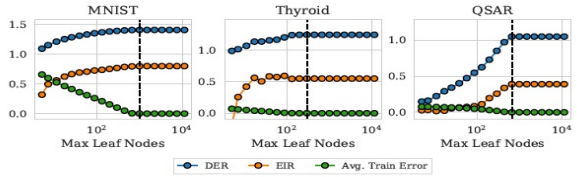
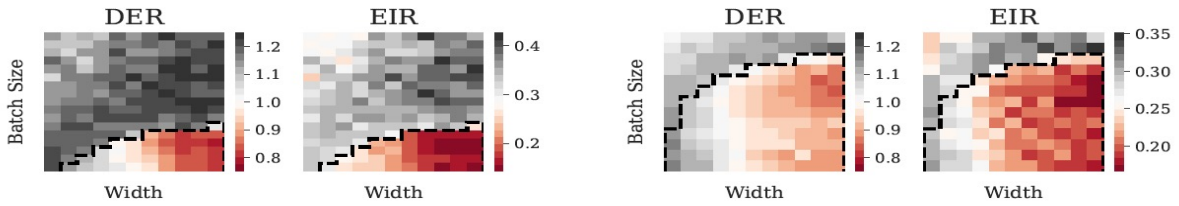


Figure 4: **Random forest classifiers.** Blacked dashed line represents the interpolation threshold. Across all tasks, DER and EIR are maximized at this point, and then remain constant thereafter.



(a) **Without LR decay.** (b) **With LR decay.**

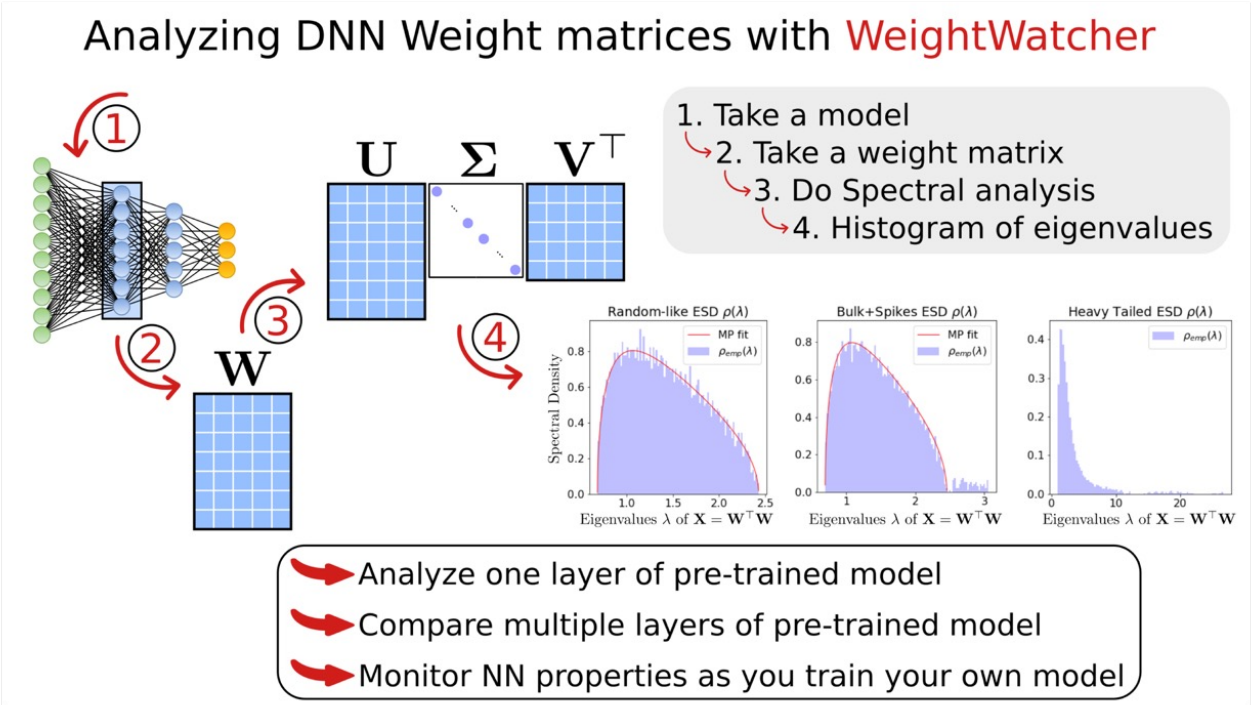
Figure 5: **Large scale studies of deep ensembles on ResNet18/CIFAR-10.** We plot the DER and EIR across a range of hyper-parameters, for two training settings: one with learning rate decay, and one without. The black dashed line indicates the *interpolation threshold*, i.e., the curve below which individual models achieve exactly zero training error. Observe that interpolating ensembles attain distinctly lower EIR than non-interpolating ensembles, and correspondingly have low DER (< 1), compared to non-interpolating ensembles with high DER (> 1).

"The Interpolating Information Criterion for Overparameterized Models," Hodgkinson, van der Heide, Salomone, Roosta, and Mahoney, arXiv:2307.07785
"When are ensembles really effective?," Theisen, Kim, Yang, Hodgkinson, and Mahoney, arXiv:2305.12313, NeurIPS23
"Monotonicity and Double Descent in Uncertainty Estimation with Gaussian Processes," Hodgkinson, van der Heide, Roosta, and Mahoney, arXiv:2210.07612, ICML23

Foundations more generally: weight diagnostics

Use methods from disordered systems theory, random matrix theory and statistical physics to diagnose practical problems in state-of-the art neural networks

- "Predicting trends in the quality of state-of-the-art neural networks without access to training or testing data," Martin, Peng, and Mahoney, arXiv:2002.06716 (2020)
- "Statistical Mechanics Methods for Discovering Knowledge from Modern Production Quality Neural Networks, Martin and Mahoney," KDD (2019)
- "Traditional and Heavy-Tailed Self Regularization in Neural Network Models, Martin and Mahoney," ICML (2019)
- "Heavy-Tailed Universality Predicts Trends in Test Accuracies for Very Large Pre-Trained Deep Neural Networks," Martin and Mahoney, SDM (2019)
- "Implicit Self-Regularization in Deep Neural Networks: Evidence from Random Matrix Theory and Implications for Learning," Martin and Mahoney, arXiv:1810.01075 (2018)
- (https://github.com/CalculatedContent/ww-trends-2020)



Foundations more generally: Time Series

Published as a conference paper at ICLR 2022

LONG EXPRESSIVE MEMORY FOR SEQUENCE MODELING

T. Konstantin Rusch
ETH Zürich
trusch@ethz.ch

Siddhartha Mishra
ETH Zürich
smishra@ethz.ch

N. Benjamin Erichson
University of Pittsburgh
erichson@pitt.edu

Michael W. Mahoney
ICSI and UC Berkeley
mmahoney@stat.berkeley.edu

ABSTRACT

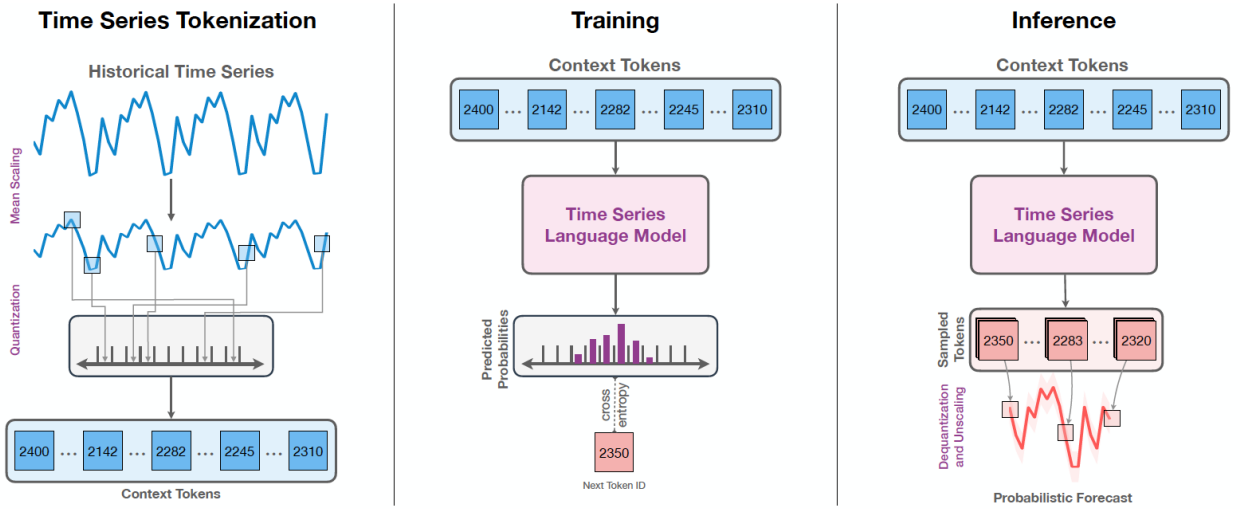


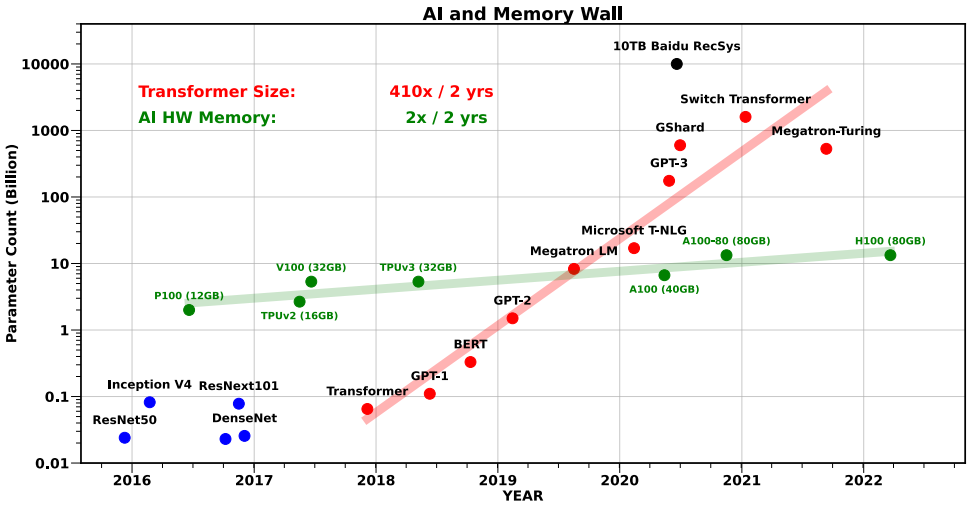
Figure 1: High-level depiction of CHRONOS. (**Left**) The input time series is scaled and quantized to obtain a sequence

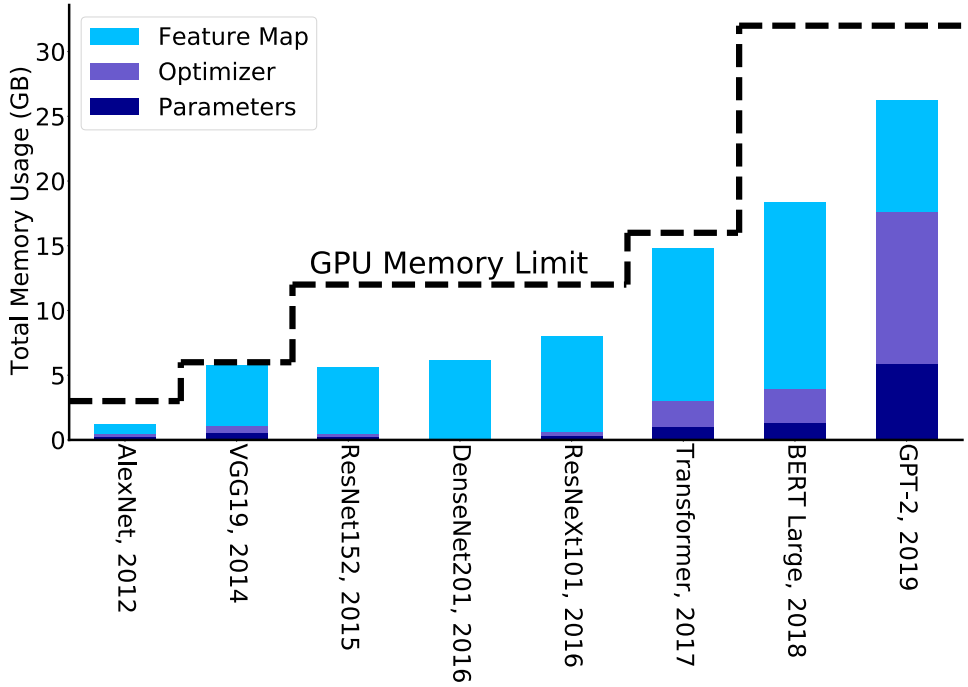
"Chronos: Learning the Language of Time Series," Ansari et al., arXiv:2403.07815
"Long Expressive Memory for Sequence Modeling," Rusch, Mishra, Erichson, and Mahoney, arXiv:2110.04744, ICLR22

Questions?

- Q0: What is a “Foundation Model”?
- Q1: Can we hope to train a “Foundation Model” for SciML?
- Q2: Would incorporating physical knowledge help? If so, how to do it?
- Q3: Foundations?
- **Q4: Implementations?**
- Q6: Applications?
- Q6: Looking forward?

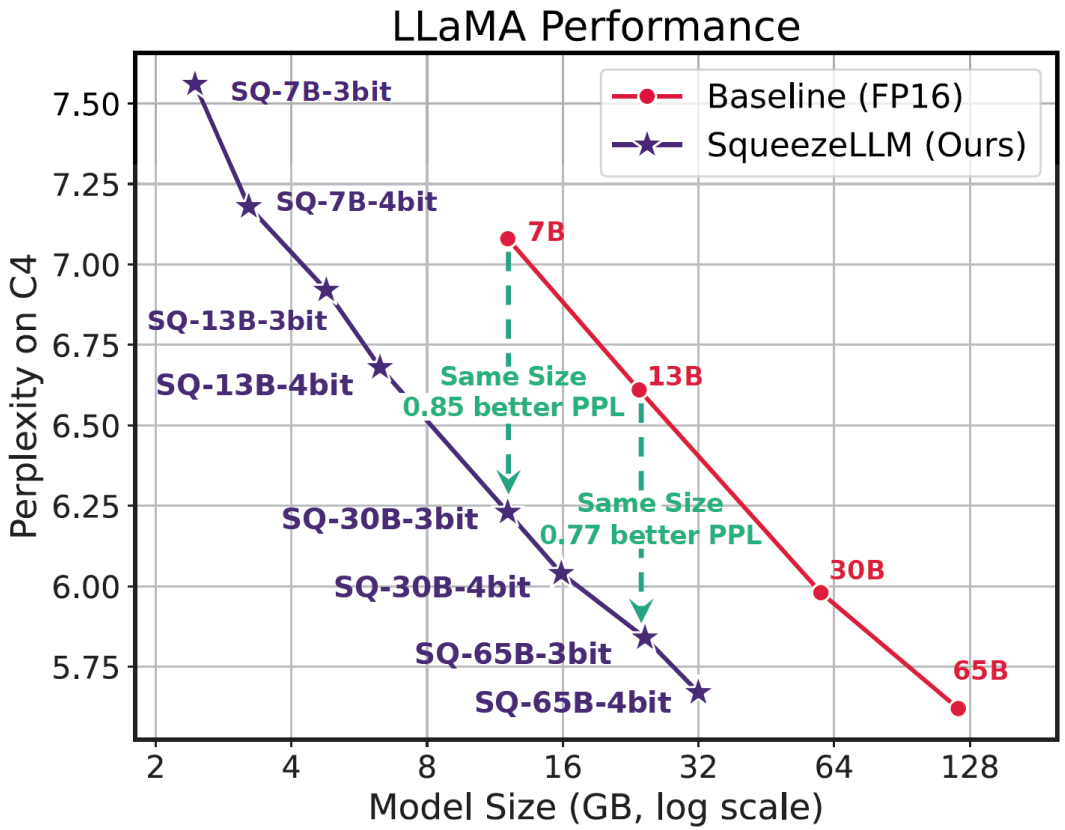
Model Size Increased Exponentially in 2018-22





Amir Gholami, Zhewei Yao, Sehoon Kim, Michael W. Mahoney, Kurt Keutzer, AI and Memory Wall, IEEE Micro, 2024.

SqueezeLLM Overview
Breaking Memory Wall with Dense-and-Sparse Quantization

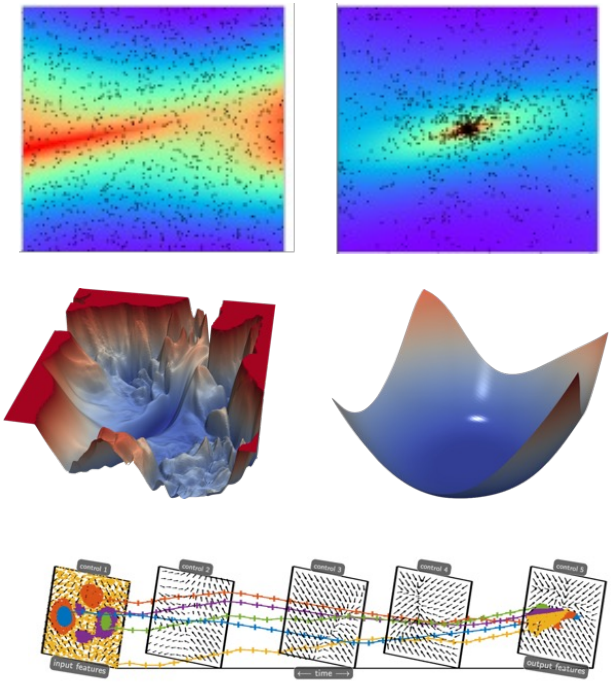


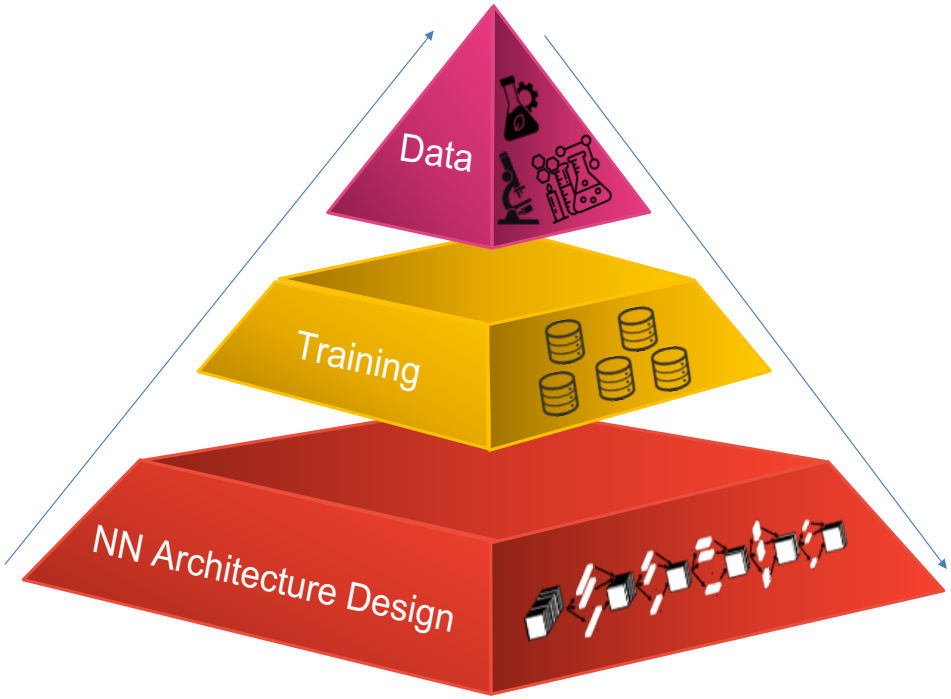
Kim*, S., Hooper*, C., Gholami*, A., Dong, Z., Li, X., Shen, S., Mahoney, M.W. and Keutzer, K. SqueezeLLM: Dense-and-Sparse Quantization. *arXiv:2306.07629*.

Implementations: “Full stack” design

Rethink the *design*, *training, inference,* and *role of data* for successful application of NNs in SciML

- Different than computational design for ML/LLMs in industry
- Different than computational design in HPC and scientific simulation





Questions?

- Q0: What is a “Foundation Model”?
- Q1: Can we hope to train a “Foundation Model” for SciML?
- Q2: Would incorporating physical knowledge help? If so, how to do it?
- Q3: Foundations?
- Q4: Implementations?
- **Q6: Applications?**
- Q6: Looking forward?

Example scientific challenges

Popular Past Challenges:

- Learn solutions to PDEs
- Learn operators new laws of physics
- Learn dynamical systems

Lesson 1: Don't solve a past problem that some well-established domain solves.*
Lesson 2: Don't solve domain problems that are only well-define to domain expert.**

Important Future Challenges:

- **Extreme value forecasting/estimation**
- **Multi-scale modeling/analysis**
- **High-frequency inverse scattering**

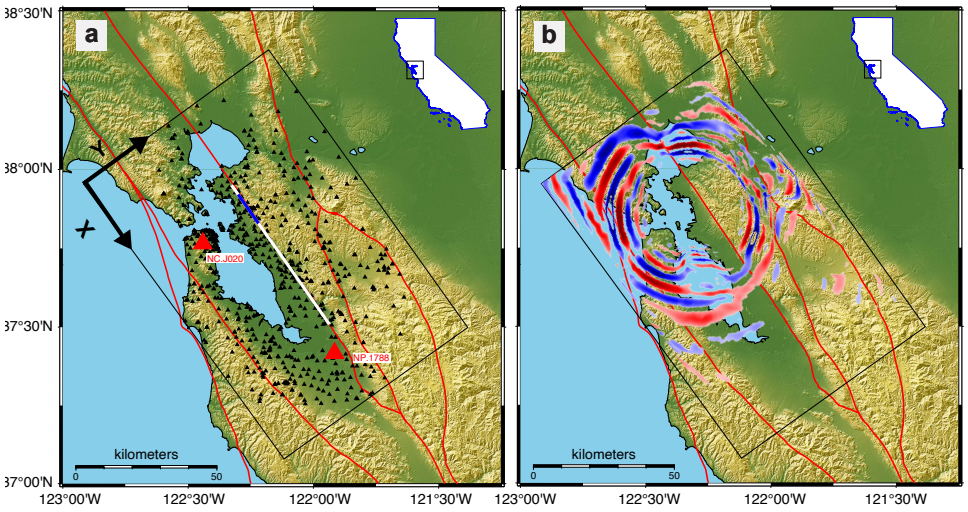
Goal: Focus on future challenges that are real scientific problems that cut across domains and that play well with ML methodologies.

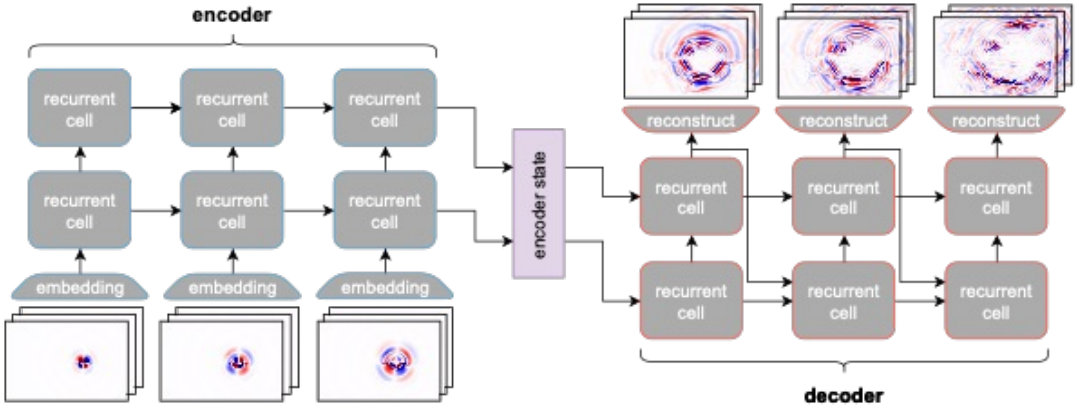
*They will beat you up, even if you do better than them.
**How ignorant can I be about your domain and still solve a problem you care about?

Foundational methods: can be useful in your vertical …

… for science:

- Earthquake early warning: to turn off critical infrastructure
- Scientific GenAI: to uncover physically-meaningful data ground motions
- Etc.





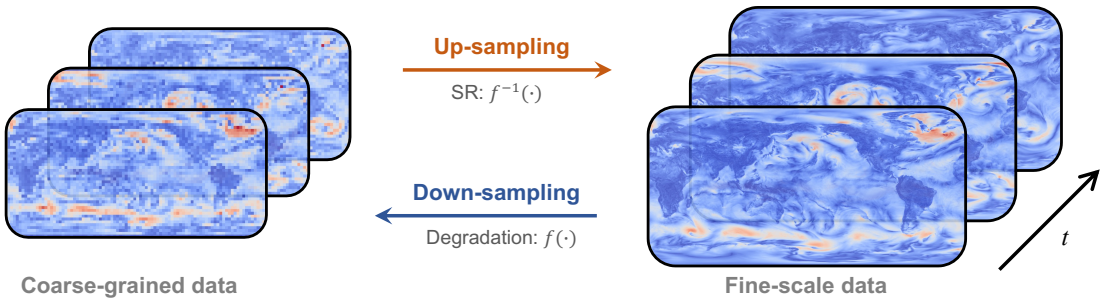
"Learning Physics for Unveiling Hidden Earthquake Ground Motions via Conditional Generative Modeling," Ren, Nakata, Lacour, Naiman, Nakata, Song, Bi, Malik, Morozov, Azencot, Erichson, and Mahoney, arXiv:2407.15089

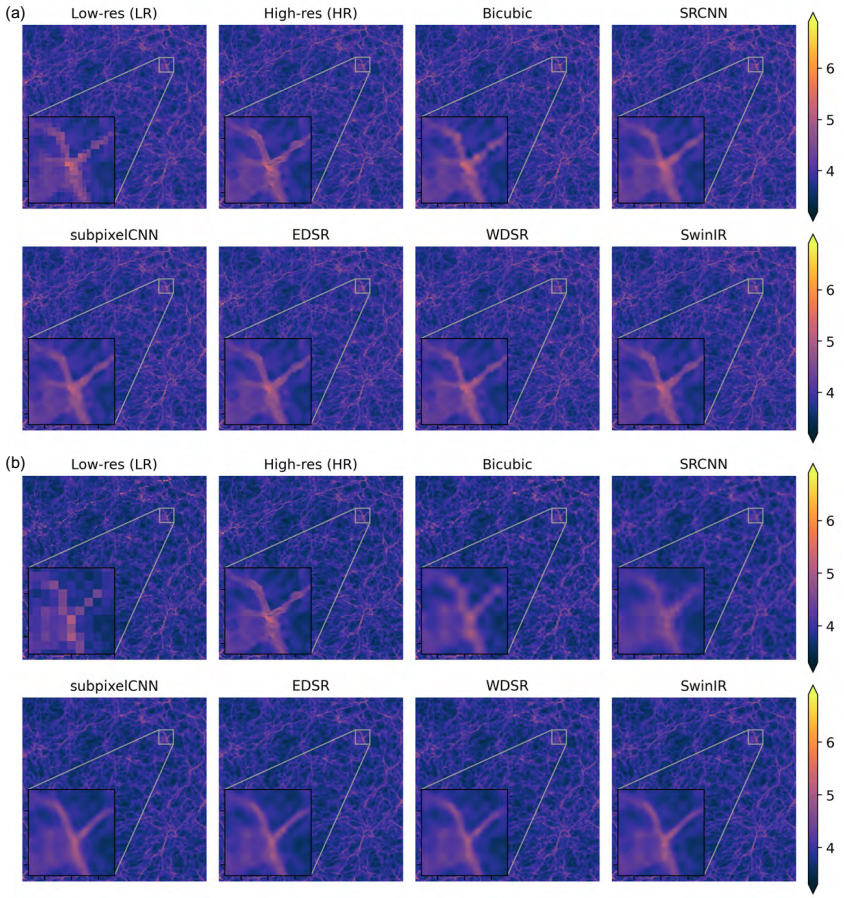
"WaveCastNet: An AI-enabled Wavefield Forecasting Framework for Earthquake Early Warning," Lyu, Nakata, Ren, Mahoney, Pitarka, Nakata, and Erichson, arXiv:2405.20516

Foundational methods: SuperBench

SuperBench
A Super-Resolution Benchmark Dataset for SciML

1. A super-resolution benchmark for SciML
2. High-resolution fluid flow, cosmology, and weather datasets with dimensions up to 2048 × 2048
3. Pixel-level difference, human-level perception domain-motivated error metrics
4. Extensible framework





"SuperBench: A Super-Resolution Benchmark Dataset for Scientific Machine Learning," Ren, Erichson, Subramanian, San, Lukic, and Mahoney, arXiv:2306.14070

Data Sets

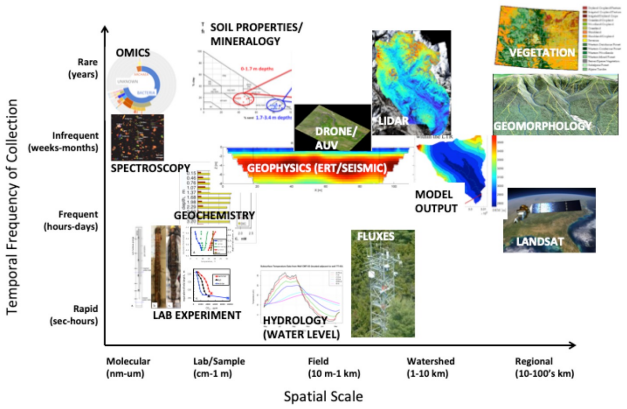


Figure 2: Example of the diverse Earth science data collected at a range of spatial and temporal scales (Figure from [131]). See also Table 1.

Datasets or Collections	Type	Spatial Extent (Resolution)	Temporal Resolution	Modality	Usage
ERA-5 [22]	Climate/weather data product	Global (25 km)	hourly	Gridded timeseries	Training
Daymet, PRISM, NARR [23]	Climate/weather data product	US (1km, 4 km, 32 km)	daily, daily, 3-hours	Gridded timeseries	Training
GHCN [23]	Climate observations	Global (point)	daily	Univariate sensor timeseries	Training
GRDC[24]	River flow observations	Global (point)	daily	Univariate Sensor Time-series	Training
FLUXNET*[25]	Land-Atmosphere energy, water flux observations	Global (point)	daily	Univariate sensor Time-series	Training
MODIS [26]	Remote sensing of land surface	Global (250 m)	daily	Images	Training
HLS [27]	Remote sensing of land surface	Global (30 m)	2-3 days	Images	Training
CMIP6/ESGF*[28]	Long-term climate simulations	Global (O(100)km)	daily, monthly	Gridded Time-series	Training (1-2 models only)
SEG Open data [29]	Seismic experiments, simulation, observation	Local (O(10)m)	milliseconds	Semi-gridded Time-series	Training
EarthScope DMC [30]	Earthquakes	Global (point)		Time-Series	Training
	Seismic experiment observations	Global (O(10)m)	milliseconds	Time-series	Training
SCEC BBPlatform [31]	Earthquakes simulations	Regional (O(1km))	milliseconds	Time-series	Training
ESS-DIVE* [32]	Observations, experiments, simulations	Pore-Global	Heterogeneous	Heterogeneous	Training and Validation
Energy Data eXchange* [33]	Geophysical observations and simulation	O(10)-O(1)km	milliseconds-daily	Time series, Images	Training and Validation
Geothermal Data Repository* [34]	Geophysical observations	O(10)m-O(1)km	milliseconds-daily	Time-Series, Images	Training and Validation
ARM Best estimate*[35]	Climate data product	Global (point)	hourly	Univariate sensor Time-series	Validation
ILAMB*[36]	Climate, ecosystem, water observations	Global (point)	Variable-dependent	Univariate sensor Time-series	Validation

* Data generated or served by DOE

Table 1: Data available for model training and validation including experiments, simulations, and observations across a range of spatial and temporal scales.

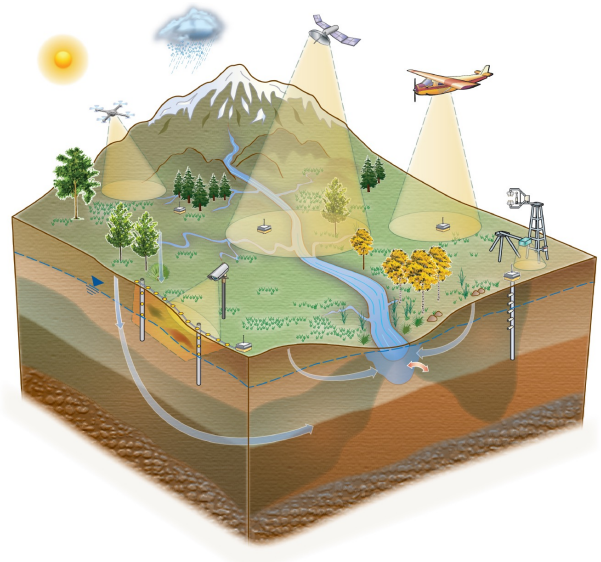


Figure 3: Example of diverse Earth science data from Atmosphere, Land and Subsurface.

SciGPT: Scalable Foundation Model for Scientific Machine Learning

Motivation: In spite of recent effort, there is no scientific foundation model (SFM) that:

(1) has been trained on a broad range of data
(2) across different domains, and space and time scales,
(3) to gain an understanding of multiple physical processes and their interactions in a complex scientific system.

Goal: To develop a broad-based SFM ``blueprint'' that:

(1) is applicable via transfer learning to multiple scientific domains and
(2) provides a clear blueprint to develop a general scientific foundation model.

Longer Term: Will provide a clear path forward for more general investment:

(1) for a general scientific foundation model and
(2) for multiple domain-specific scientific ML models.

SciGPT: Scalable Foundation Model for Scientific Machine Learning, cont.

Three main challenges: that currently block the development of a SFM:

(1) lack of "neural scaling" w.r.t. model/data/compute as well as spatio-temporal scaling;

(2) lack of control on out-of-distribution generalization; and

(3) lack of broad-based multi-modal data for training.

Approach: Adopt the main methodology that ML researchers do:

(1) used to develop CV and NLP FMs,

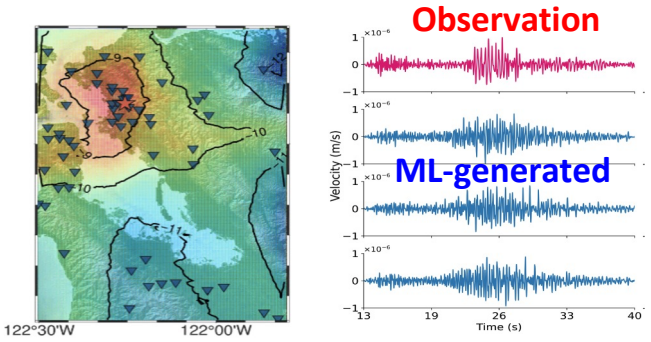
(2) adapting those methods as needed to the properties of scientific data.

"Scale model and data and compute so none of them saturate, then transfer learn"

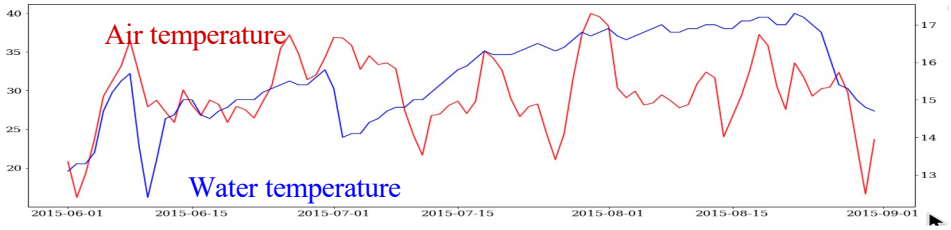
Possible SciGPT applications in X={Earth Sciences}?

Prediction of extreme events & impacts

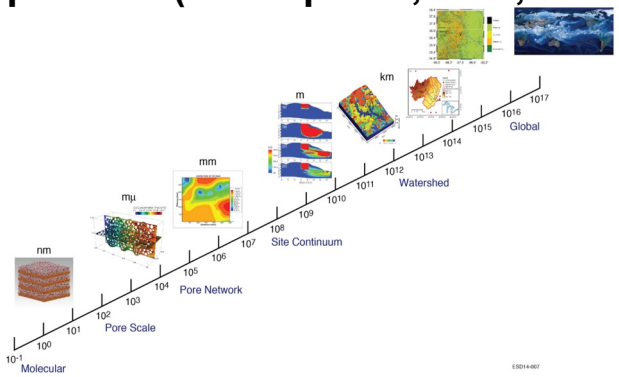
Earthquakes



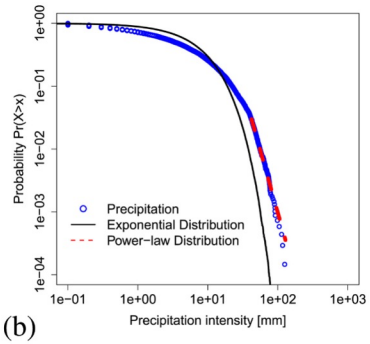
Climate impacts on watersheds, tipping points



Learning physics across scales & Earth system components (atmosphere, land, subsurface



E.g. power law dynamics common in natural systems



(b)

Possible Different Scientific Tasks

Prediction (in space and time).

- Temporal forecasting: weather, traffic, network intrusion, energy infrastructure, etc.
- Time series prediction across disciplines: extreme events; short, medium, long term
- E.g., predict the weather or air quality at a particular location using information from nearby observations, model forecasts, and remote sensing images.

Inversion and imaging of physical parameters.

- Indirect experiments (e.g., seismic waves) are often used to invert for physical parameters (e.g., seismic velocities) that can be used for prediction/discovery
- High-frequency regime: of particular interest, to identify fine-scale structures, but it is particularly challenging, doe to computationally expensive Fourier inversions

Sim-to-real.

- Transfer learn from simulations, using a small amount of real data.
- Transfer learn from low-quality simulations, using a small amount of higher-quality
- Transfer learn from ``in the lab,'' using a small amount of ``in the field'' data

Questions?

- Q0: What is a “Foundation Model”?
- Q1: Can we hope to train a “Foundation Model” for SciML?
- Q2: Would incorporating physical knowledge help? If so, how to do it?
- Q3: Foundations?
- Q4: Implementations?
- Q6: Applications?
- **Q6: Looking forward?**

Looking forward …

Foundation Models are infrastructure:

- A foundation upon which to do stuff
- Just like the computer, or iphone, or bridges, or electrical grid
- All these are impressive … until they are not

Look at history: computer science (industry) vs computational science (science)

- Very similar forcing functions
- Expect similar outcomes
- Do we compute on the metal or with multiple layers of abstraction?
- Do we fit SciML into the form factor provided by industrial LMs?

Question: How can we deliver on the promise of Scientific ML?

- Give it a strong, robust, principled foundations
- Rooted in both scientific principles and ML principles