Differentiable modeling to unify machine learning and physical models and advance Geosciences

Chaopeng Shen[1*], Alison P. Appling[2], Pierre Gentine[3], Toshiyuki Bandai[4], Hoshin Gupta[5], Alexandre Tartakovsky[6], Marco Baity-Jesi[7], Fabrizio Fenicia[7], Daniel Kifer[8], Li Li[1], Xiaofeng Liu[1], Wei Ren[9], Yi Zheng[10], Ciaran J. Harman[11], Martyn Clark[12], Matthew Farthing[13], Dapeng Feng[1], Praveen Kumar[6,14], Doaa Aboelyazeed[1], Farshid Rahmani[1], Hylke E. Beck[15], Tadd Bindas[1], Dipankar Dwivedi[16], Kuai Fang[17], Marvin Höge[7], Chris Rackauckas[18], Tirthankar Roy[19], Chonggang Xu[20], Kathryn Lawson[1]

[1] Civil and Environmental Engineering, The Pennsylvania State University, University Park, PA, USA.
[2] U.S. Geological Survey, Reston, VA, USA
[3] National Science Foundation Science and Technology Center for Learning the Earth with Artificial Intelligence and Physics (LEAP), Columbia University, New York, NY USA
[4] Life and Environmental Science Department, University of California, Merced, CA, USA
[5] Hydrology and Atmospheric Sciences, The University of Arizona, Tucson, AZ, USA.
[6] Civil and Environmental Engineering, University of Illinois, Urbana Champaign, IL, USA
[7] Eawag: Swiss Federal Institute of Aquatic Science and Technology, Dübendorf, Switzerland
[8] Computer Science and Engineering, The Pennsylvania State University, University Park, PA, USA
[9] Department of Natural Resources and the Environment, University of Connecticut, Storrs, CT, USA
[10] Southern University of Science and Technology, Shenzhen, Guangdong Province, China
[11] Department of Environmental Health and Engineering, Johns Hopkins University, Baltimore, MD, USA
[12] Global Institute for Water Security, University of Saskatchewan, Canmore, Alberta, Canada
[13] US Army Engineer Research and Development Center, Vicksburg, MS, USA
[14] Prairie Research Institute, University of Illinois, Urbana Champaign, IL, USA
[15] Physical Science and Engineering Division, King Abdullah University of Science and Technology, Thuwal, Saudi Arabia
[16] Lawrence Berkeley National Laboratory, Berkeley, CA, USA
[17] Department of Earth System Science, Stanford University, Stanford, CA, USA
[18] Computer Science and Artificial Intelligence Laboratory (CSAIL), Massachusetts Institute of Technology, Massachusetts, USA
[19] Civil and Environmental Engineering, University of Nebraska-Lincoln, NE, USA
[20] Earth and Environmental Divisions, Los Alamos National Laboratory, NM, USA
* Corresponding author, email cshen@engr.psu.edu

Abstract

Process-Based Modeling (PBM) and Machine Learning (ML) are often perceived as distinct paradigms in the geosciences. Here we present differentiable geoscientific modeling as a powerful pathway toward dissolving the perceived barrier between them and ushering in a paradigm shift. For decades, PBM offered benefits in interpretability and physical consistency but struggled to efficiently leverage large datasets. ML methods, especially deep networks, presented strong predictive skills yet lacked the ability to answer specific scientific questions. While various methods have been proposed for ML-physics integration, an important underlying theme — differentiable modeling — is not sufficiently recognized. Here we outline the concepts, applicability, and significance of differentiable geoscientific modeling (DG). "Differentiable" refers to accurately and efficiently calculating gradients with respect to model variables, critically enabling the learning of high-dimensional unknown relationships. DG refers to a range of methods connecting varying amounts of prior knowledge to neural networks and training them together, capturing a different scope than physics-guided machine learning and emphasizing first principles. Preliminary evidence suggests DG offers increased interpretability and causality than ML, improved generalizability and extrapolation capability, and strong potential for knowledge discovery, while approaching the performance of purely data-driven ML. They require less training data while scaling favorably in performance and efficiency with increasing amounts of data. Geoscientists can now frame and investigate questions, test hypotheses, and discover unrecognized linkages.

Introduction

Geoscientific models encompass a wide range of domains, with evolving scopes and ever-increasing societal importance, especially in the face of rapid climate change. For example, hydrologic models help us manage water resources[1,2] and plan for extremes such as floods and droughts[3]; vegetation models can help predict the fate of carbon and other key biogeochemical cycles on land[4] or in the ocean[5]; agricultural models estimate crop yields and also their environmental impacts[6]; geophysical models aim to predict land surface changes via processes like landslides[7], land subsidence[8], and earthquakes; biogeochemical reactive transport models aim to understand and predict surface and subsurface water chemistry and quality[9,10]. Combining many such components, Earth System Models[11–13] and integrated assessment models[14–16] provide crucial guidance for resource managers and policy makers[17,18]. The uses of such models go beyond making predictions of the future to also facilitating communication with the stakeholders and aiding in the policy-making process[18].

Geoscientific models often share some commonalities as they describe the dynamic responses of systems to time-dependent forcings as modulated by semi-static attributes. Many such problems can be described as systems of nonlinear equations, algebraic differential equations, or ordinary and/or partial differential equations (ODE/PDEs), along with parameterizations (empirical representations) of physical processes with spatially-varying parameters. The overall system can contain multiple processes chained together, some of which are well understood while others are not. Further, many of these process representations and parameterizations are subject to considerable uncertainty, some of which is related to scale, and thus has significant room for improvement. Here we argue that differentiable implementations of geoscientific models offer a transformative approach to simultaneously advancing process representations, parameter estimation, and predictive accuracy. In particular, differentiable implementations provide an unprecedentedly seamless connection between process-based and machine-learning-based model components, enabling us to realize the value and minimize the limitations of each.

Value and limitations of process-based geoscientific modeling

The traditional process-based modeling (PBM) approach has served the geosciences well in helping to improve our understanding of system functions and behaviors. Due to their physical basis, they can be leveraged in hypothesis testing to assess system responses, and cause-effect relationships (see the *Physical Laws* row in Table 1), e.g., the impacts of land use changes on flooding trends[19] and future warming on glacial melt[20]. Further, they can simulate a wide variety of observed (e.g., discharge or leaf area index) and unobserved variables (e.g., groundwater recharge or fine-root carbon). Such an ability is critical to both advancing scientific understanding and to providing a narrative when communicating with the public and stakeholders, who are engaged in the decision-making process[21]. It is possible to ask and examine specific questions regarding processes within the modelled system, by progressively improving the representations of processes [22–25] and evaluating them using controlled experiments.

Despite these benefits, there remain important challenges with PBMs.

(1) Process-based models often cannot rapidly evolve with and fully exploit the information in "big data" due to the time needed to develop and test process representations and parameterizations[26,27]. Traditionally, the differences between model predictions and observations are first reconciled by parameter calibration, which adds significant uncertainty (more about this later)[28]. For model errors beyond parameter adjustments, modelers then hypothesize different causes, implement structural changes to the model, and iteratively confront the updated model hypotheses with the data[22]. This iterative process is highly expensive (in both labor and time) and dauntingly complex, and is dependent on developer intuition and legacy[29]. Consequently, it is common for the structural representation of a specific process in a geoscientific model to stagnate, with years or decades passing between structural updates[30–33].

(2) Process-based models are limited by knowledge gaps. Extensive physical, biological, and socioeconomic knowledge is required to achieve adequate representations and updates for processes in a geoscientific model, and any deficiencies can amplify errors and ambiguity. Another major challenge is the interactions of processes across disciplinary boundaries[34]. For instance, vegetation, human management, and socioeconomic systems all interact with each other and affect the water and carbon and other biogeochemical cycles[35–38]. While the intersections of these domains will continue to stimulate scientific discovery, we need a paradigm that enables us to make progress despite knowledge gaps.

Potential and limitations of machine-learning-based geoscientific modeling

Irrespective of the domain of application, one cannot help but notice the "*Cambrian explosion*" of purely data-driven machine learning (ML)[39] approaches, especially deep neural networks (NNs), applied to a wide range of scientific applications[34,40] (see Discussion A in Supplementary Information S2). In geosciences, NNs have shown strong accuracy in predicting crop production[41,42], precipitation fields[43,44] and clouds[45], water quality variables[46,47] such as water temperature[48–51], dissolved oxygen[52], phosphorous[53], and nitrogen[54,55], and the full hydrologic cycle[56] including soil moisture[57–59], streamflow[60–63], evapotranspiration[64–66], groundwater levels[67], and snow[68], etc. Deep networks like long short-term memory (LSTM) networks[69], graph neural networks[62], and convolutional neural networks (CNNs)[70,71] have become widely known in geosciences. Many such studies reported noticeably better performance than conventional approaches, revealing that the latter did not fully exploit the information in the data[26] (Table S1 in Supplementary Information S2).

Nevertheless, there remain important challenges with purely data-driven ML:

(1) Deep networks are data hungry. The success of deep networks relies on the availability of "big data", which can, unfortunately, be sparse for many geoscientific pr oblems[55,72], where many variables are measured at dozens, hundreds, or thousands of sites only. For example, water quality data are sparse and inconsistent in temporal, spatial, and chemical coverage[73,74]. For rare and extreme events such as mega floods, droughts, and earthquakes, available data is even scarcer.

(2) ML has difficulties with errors, incompleteness, or bias in the inputs or observations. The quality of ML models is limited by the quantity, diversity, and quality of training data[51,75]. Since a purely data-driven model can, at best, nearly-perfectly replicate the patterns in the training data, it invariably inherits various issues from the training data including implicit or explicit biases, inadequate spatiotemporal resolutions (e.g., with satellite-based observations), and the inability to account for non-stationarity in time series due to the short data record.

(3) Neural Networks remain challenging to interpret. Although explainable AI methods such as layerwise relevance backpropagation[76–78] can be highly helpful in revealing some of the internal workings of a network and should be pursued, they are not designed to flexibly query a model or identify missing physics.

(4) Purely data-driven ML models cannot predict untrained variables (those not provided as training targets). Due to their very nature, ML-based models are designed to only output the training targets. They cannot provide an account of how events unfolded, *e.g.*, the ability to state that "*the flood occurred because the soil was saturated*" in a study where soil moisture is unobserved. This hinders both formation of hypotheses and communication with stakeholders.

(5) ML algorithms are based on correlations and not causality, regarding both attributes and temporal changes. There are always confounding covarying factors in data, so that ML models can produce the "right" results for the wrong reasons, potentially making projections less reliable when circumstances are changed.

The root of deep network's success – Differentiable Programming

Considering both the exceptional successes and limitations of ML and especially NNs, one can ask:

What are the foundational strengths of NNs?

How can we maximize these strengths while overcoming the limitations associated with data?

How can we extract knowledge in an interpretable form while maintaining ML-level performance?

In answering these questions, we argue that differentiable programming (explained below) is the computing paradigm that supports the efficient training of NNs which, in turn, can deliver many philosophically and practically transformative outcomes. Traditional modeling has been dealing with optimization problems for decades (see the *Similarity* block in Table 1). However, it is argued here that only by exploiting the power of parallelized gradient-based optimization have we been able to learn from big data and train the large numbers of weights (parameters) necessary to approximate complex unknown functions.

The ability of generic NN architectures such as CNNs, LSTMs, and attention mechanisms to approximate unknown functions has achieved desirable outcomes (Figure 1 & Table 1). First, the cost of learning a few generic architectures is lower than the significant domain expertise required by traditional models, making NNs suitable for widening access to usable predictions. Second, NNs can help in the identification of previously unrecognized linkages. Third, NN training can scale up favorably with the amount of data (in terms of accuracy, generalizability, and efficiency)[75,79], in contrast with traditional modeling where the learning may saturate after some limited calibration of parameters or functions[51].

All of these features are possible only because we can now train NNs with a large number of weights, providing a large learnable function space[80,81]. The number of weights easily exceeds the optimization capabilities of conventional algorithms. The most recent computer vision model contains two billion weights[82] and LSTM models widely employed in hydrology can contain ~500,000 weights. In contrast, traditional evolutionary[83–85], or genetic[86] or particle swarm optimization methods[87] can hardly handle more than a few dozen independent parameters (Table 1).

The computing paradigm that enables efficient training of so many parameters is *Differentiable Programming*[88,89] (Figure 1), where accurate derivatives of the model outputs with respect to inputs and/or intermediate variables can be efficiently computed. Without getting into details, this paradigm is often (but not always) enabled by ML platforms, e.g., PyTorch, Tensorflow, JAX, Julia, etc., which support reverse- or forward-mode automatic differentiation (AD)[88–90] using various approaches. Models written on these platforms can, often without much effort, be *programmatically differentiable* – even where certain operations are mathematically indifferentiable (e.g., thresholding or *if* statements), the fact that they are piecewise differentiable enables gradient computations to be performed. The chain rule can be applied to efficiently accumulate the derivatives in a process called "backpropagation"[91]. Note that differentiability is normally only needed for training, not when running the model in forward mode.

Here we expand the scope and use the term *differentiable modeling* to include any method that can produce the gradients rapidly and accurately at scale. A non-AD example is that of adjoint methods, which solve accompanying equations (called adjoint equations)[92–94] for the derivatives. AD differentiates through the code in an automated manner and is independent of the problem, while adjoint methods differentiate through the mathematical model equations and thus require manual derivations of adjoint equations for each problem[95]. Many alternative gradient estimation methods, e.g., finite differences, are intractable for any reasonably-sized NNs (10,000 weights would require 10,001 forward model evaluations) and can be challenged by stiffness. Cheaply obtained gradients allows for parameter updates via various first-order gradient-descent methods[96]. Second-order methods, such as Newton Raphson, have not gained popularity for the training of NNs due to the cost of computing the Hessian matrix. The vast majority of NNs are implemented on platforms supporting differentiable programming, while most existing PBMs are not.

In addition, historical differences in the training of geoscientists vs ML practitioners (*Education* row in Table 1) may give the impression that ML and process-based modeling are fundamentally unrelated, but the perceived divide is more of a legacy issue. In reality, both ML and parametric physical models can be expressed in nearly identical mathematical forms (*Mathematical form* row in Table 1), especially through the differentiable programming lens.

This leads us to the conclusion of this section: ***differentiable programming is the core distinguishing feature of neural networks, and differentiable modeling can serve as the basis for unifying NN and process-based geoscientific modeling.*** As we will discuss in the following sections, this unification requires only minor modifications to our conceptual modeling and implementation strategies, but opens new doors to scientific discovery.

Table 1. Similarities and differences between purely data-driven NNs and purely process-based models. [Pro] annotates the comparative strengths, also shown in green text. In the equations, W stands for weights of the neural network g; θ stands for the physical parameters of the process-based model f; x, u and A are dynamic forcings, state variables, and semi-static attributes, respectively; and L represents the loss function which quantifies the difference between simulation outputs and observations.

	Purely data-driven NNs	**Purely process-based models**
	Similarities	

Mathematical form	$y = g^W(u, x, A)$ $W = argmin(L(y, y^*))$	$y = f^\theta(u, x, A)$ $\theta = argmin(L(y, y^*))$
Programmatically differentiable	Yes	Traditionally no, but could be reimplemented on differentiable platforms or supported by new libraries
	Differences	
Ease of use	[Pro] Generic model architecture – Easy to develop even without domain expertise.	Specialized domain knowledge
Architecture	[Pro] Generic structure with a large number of weights that allow the model to approximate a wide range of functions.	Specific structural priors representing human understanding of physics, with a small number of parameters
Data	[Pro] Capable of efficiently learning from large data and obtain scaling benefits from big data.	Typically calibrated at a few sites, or a few parameters are calibrated in a regionalization equation. Learning saturates at a small data quantity. [Pro] The potential to overcome data limitations in accuracy, resolution, and availability.
Training/ Calibration	[Pro] Trained using gradient descent, supported by differentiable programming.	Calibrated using various small-scale algorithms. Normally code does not support DP.
Unknown processes	[Pro] Data can be used to make up for processes we are not certain about. This also means we can learn unrecognized connections and expand knowledge.	We must specify the processes to be used in the model, even if they are only assumptions.
Outputs	Output trained variables only.	[Pro] Output many intermediate variables that facilitate providing an interpretable full narrative.
Physical laws	May not fully respect physical laws.	[Pro] Respect physical laws. Help us to assess cause-effect relationships.
Interpretation	Difficult to interpret	[Pro] Elucidate physical processes, allowing us to ask specific science questions.
Education	Taught in computer science or data science curricula.	Taught in engineering or science curricula.

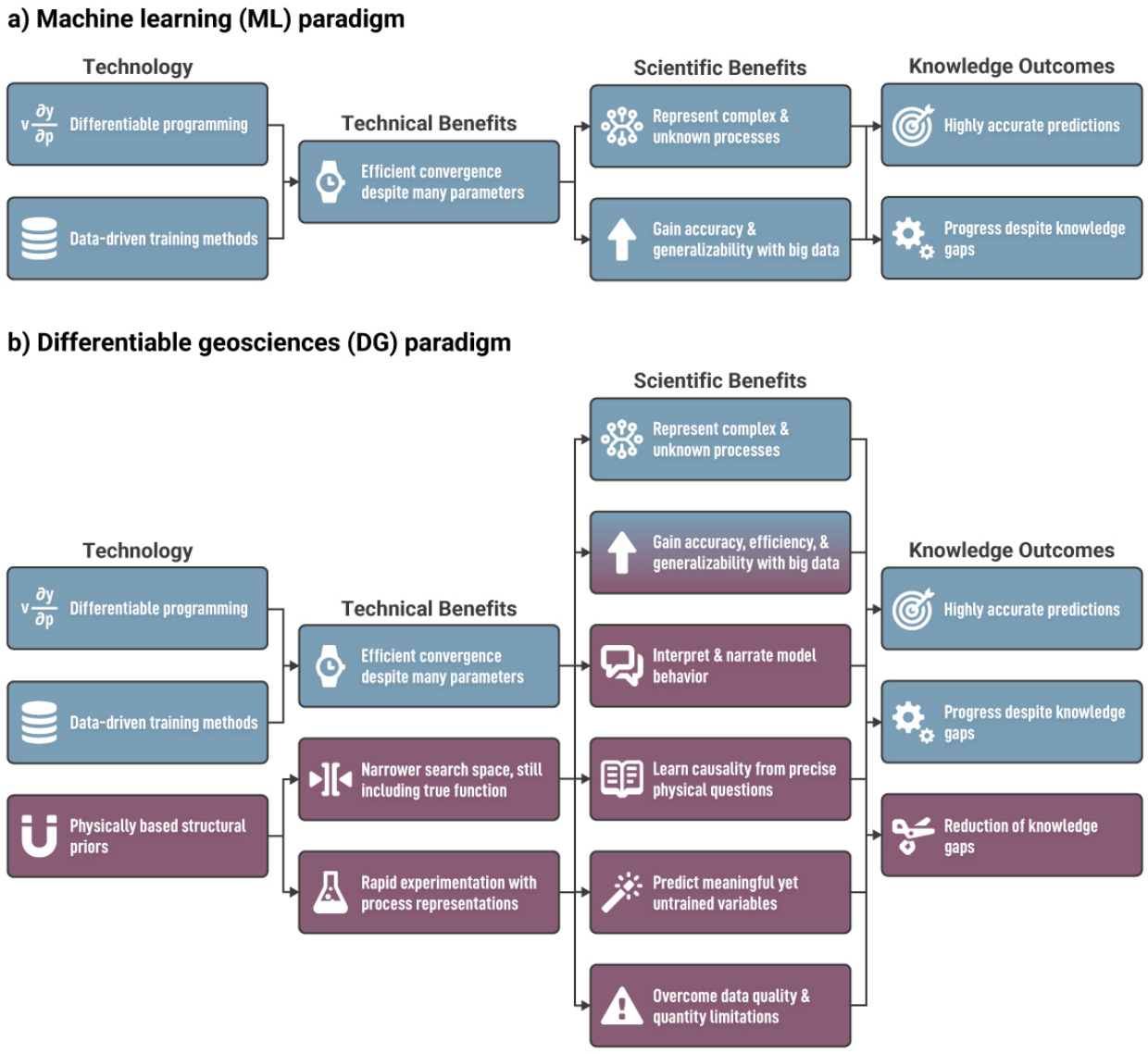

Figure 1. (a) ML (blue boxes) gives us great results with easy-to-use models, resulting from the complexity of neural networks (many parameters) and the technologies that make it feasible to train such complex models. The most fundamental of these technologies is differentiable programming. (b) In the DG paradigm, which incorporates differentiable non-ML model components (physically based structural priors), we can now obtain additional great features (plum boxes) while retaining and augmenting the old ones (blue and blue-plum boxes, respectively).

Differentiable Geosciences: Absorbing the core power of scientific ML into geoscientific domains

What is differentiable modeling in the geosciences?

Here we advocate for a new modeling paradigm: "*Differentiable Geoscientific modeling*", or simply "*Differentiable Geosciences*" (DG). DG refers to the use of models intermingling process-based descriptions and NNs to simulate geoscientific processes, update our physical process representations, learn physically meaningful parameters, quantify uncertainty, etc. DG allows us to replace poorly-understood or low-accuracy process-based model components with ML components that may be more accurate, while

retaining those process-based model components that we already trust or want to improve. DG may also exploit gradients for other purposes such as sensitivity analysis or trajectory optimization. A distinct feature of DG is its full programmatical differentiability – that is, the whole model needs to support gradient calculation from the start to the end of the workflow – to ensure that we can incorporate neural network units that can adapt to and evolve from data. The process-based descriptions retained in the model can be called the structural priors. DG seeks to marry the core of NN models – their optimizing and learning capabilities – to geoscientific process descriptions.

DG can be considered a branch of scientific machine learning[97,98] that emphasizes improving process representations and understanding. With DG, we trade the model genericity for physical interpretability, with minimal compromises to accuracy. DG reduces the cost (in terms of data) of finding good solutions because the structural priors serve to constrain the model. Meanwhile it also scales well with data quantity and can reap the benefits of big data, just as does purely data-driven ML. There are two perspectives from which we can view DG models (Figure 2):

(a) they are ML models constrained to a smaller searchable space by the structural priors.

(b) they are PBMs augmented with learnable and adaptable components (and thus an expanded searchable space) provided by NNs.

In DG, NNs can be commissioned in a wide variety of ways, ranging from learning parameters[99] to updating assumptions used in the model[75], and from estimating time-dependent forcing terms to describing the whole space-time solution[100]. The next section provides some forms of use cases, and examples are provided in *Classes of DG methods with examples* section below. DG is different from previous concepts of physics-guided machine learning (PGML) or not-fully-differentiable models in the methodology (must be fully differentiable), mission (to advance process understanding), and philosophy (whether treating physical law as truth or not). Please see Supplementary Information S2, Discussion C.

From technical breakthrough to philosophical change – why will DG be transformative?

While efficient gradient calculation may appear to be merely a technical change, it is likely to transform our modeling philosophy and scientific objectives. First, the ability to approximate complex, unknown functions greatly broadens the type of questions we can ask, by enabling us to treat trusted components as priors and focus on improving uncertain model components, one at a time. To explain this idea in concise mathematical terms, let us consider a physics-based model $y=g(u, x, \theta)$ where u, x, θ represent state variables, dynamic forcings, and physical parameters, respectively (This representation encompasses differential equations, i.e., $\partial u/\partial t= g(u, x, \theta)$, but is more generic). Traditional inversion algorithms only estimate the parameters, i.e., asking "θ =?") while requiring that the functional form g be assumed *a priori* (except for some rigid methods, e.g., nonparametric regression, which require complicated derivations and specialized training algorithms, and thus have not gained popularity). However, differentiable models allow us to ask questions about the functional form, i.e., "g =?", by training a neural network (*NN*) (or parameterized functions) to replace g: $y = NN^W(u, x, \theta)$ where W is the high-dimensional weights. Hence, with DG, we now can place our question mark precisely in the model. The functions to estimate could be

(i) a parameterization scheme, as done in differentiable parameter learning[99]: $y = g(u, x, \theta = NN^W(A))$;
(ii) a module in a model, e.g., where we can replace g_3 in $y = g(g_1, g_2, g_3(u, x, \theta))$ with *NN*: $y = g(g_1, g_2, NN^W(u, x, \theta))$, as Feng et al.[101] optionally replaced the runoff function; or
(iii) a part of a governing equation or constitutive laws, *e.g.*, we can estimate NN^W in $\partial u/\partial t = g(g_1, g_2, NN^W(u, x, \theta))$[102,103].

In the above equations, physical process equations provide a backbone for the overall model; e.g., in (i) the physical backbone is g; in (ii) and (iii) the physical backbone is g, g_1, g_2 and g_3. The unchanged parts (structural priors), i.e., g, g_1, g_2 in (ii) and (iii), critically serve as physical constraints, allowing us to isolate and focus our attention (and data) on the most unknown model components. We may gain insights by simply visualizing the relationships learned by NN^W [62,104] or applying knowledge distillation methods[105]. We are also able to evolve better process representations for some model components like g_3 mentioned above, e.g., the relation between soil moisture and effective rainfall in conceptual hydrologic models, without needing a full understanding of all the processes. This precision of questioning is opposed to some popular off-the-shelf interpretive AI approaches, e.g., layerwise relevance propagation[76,106], Shapley additive explanations[107], or local surrogate methods[108], that are limited to only asking a few fixed questions, e.g., *which parts of the inputs caused this result?* Moreover, in geoscientific modeling, directly interpreting the trained sensitivities may be risky – with only limited measurement sites, the trained relationship related to the spatial attributes tends to be overfitted.

DG provides a framework for combining deductive reasoning and inductive learning. Purely data-driven models are inductive and seek to derive almost all relationships from data, whereas process-based models first posit hypotheses and then test those hypothesis using data, albeit facing many challenges in doing so. The DG paradigm posits a user-defined number of structural priors, and then identifies many other parts of the model from data. This design follows the traditional scientific approach that identifies parsimonious models to reflect the general properties of the phenomenon, along with a quantification of the predictable aspects that are not yet understood[109]. Moreover, differentiable, learnable models can and have obtained state-of-the-art performance that can match fully data-driven models (Supplementary Information S2, Discussion B).

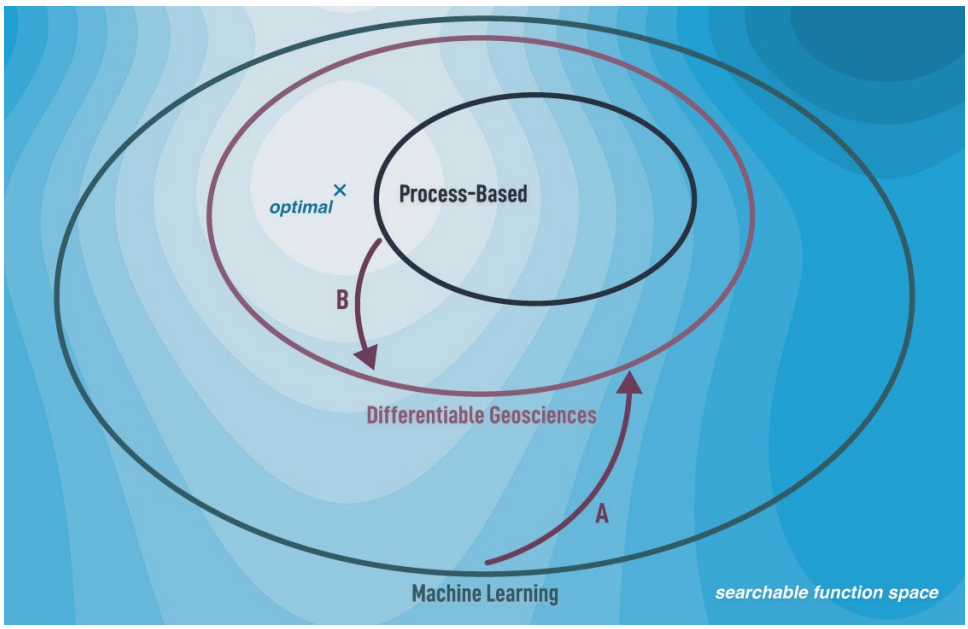

Figure 2. Differentiable models can be viewed as (A) machine learning models guided into a smaller searchable space by structural priors or (B) process-based models with expanded search space supported by learnable units. The background fill colors indicate model optimality, related to the cost function if we had infinite data.

Why is differentiable modeling particularly valuable for the geosciences?

First of all, geoscientific data are strongly imbalanced in spatial extent, temporal coverage, and in terms of variables of interest. While satellites can measure leaf area index[110] or coarse-resolution surface soil moisture[111,112] all over the world, there are a limited number of sites measuring photosynthesis rates[113] or streamflow, especially in Africa and Asia[114]; and there is very limited knowledge of subsurface properties. Purely data-driven ML may be biased or stymied by these data limitations, which may be overcome by the inclusion of physics. Indeed, preliminary analysis shows that differentiable models with a physical model as the backbone can outperform LSTM in regional extrapolation[115].

The second major motivation is system nonstationarity induced by climate change, which could drive many systems out of the previously observed range of variability[116]. Data-driven methods are tailored to the training data and may not maintain accuracy in the face of strongly changing conditions (this is a nuanced statement as models like LSTM may have highly competitive scores even in long-term projection tests[61,115], but nonetheless experience large declines in accuracy when faced with nonstationary processes). Careful testing suggests adding stronger priors may lead to better future projections[115].

As DG models can also output any diagnostic (latent) variable available from the process-based equations within the DG model. we can perform model conditioning and/or data assimilation operations with sparse and scattered data. By conditioning, we mean constraining the model using an observable to learn more realistic parameters or processes so that the overall model dynamics are better. For example, satellite-based soil moisture data can condition a hydrologic model to better predict vegetation water use[99] or primary productivity; streamflow can constrain a model to better simulate snow water equivalent[117]. For data assimilation, the model can use recent observations of B to improve the short-term forecast of A, as B can also help to update our model state variables.

Physical parameters play key roles in geoscientific models in modulating the behaviors of the system. Parameter estimation transfers information from either (i) raw observable physiographical variables or (ii) fine-scale dynamics to parameters. Quite often, we have no ground truth information for the parameters and they require inversion using observations or high-resolution simulations. Parameter estimation has, for decades, been fraught with uncertainty, ambiguity, and frustration. Due to different parameters producing very similar output and their sensitivity to spatiotemporal resolutions, calibration at a geographic location can often lead to nonunique inference (sometimes referred to as "equifinality")[118–120]. Extending parameters to unmonitored locations requires "regionalization", which also introduces uncertainty. Because of increasing geospatial data availability, parameter estimation is an area where machine learning is well-poised to make significant progress. A novel aspect is that, as with purely data-driven ML, DG methods provide favorable scaling relationships – more training data leads to improved performance, efficiency, and generalizability[99] (discussed in Supplementary Information Text S1).

What are the promises of differentiable modeling in geosciences?

We hope to evolve differentiable models so that we can gain process knowledge while improving the model predictions. Success can be claimed if we obtain models with the following features:

(i) Predictive accuracy and transferability equal to or superseding purely data-driven models for extensively measured variables;
(ii) Models capable of structural evolution, i.e., we can improve the parameterization and formulation of the processes;
(iii) Accurate generalizability to data-sparse regions or into long-term future;
(iv) Conservation of mass/energy/momentum and consistency of internal physical fluxes and states that can provide a full narrative of the events and full support to downstream processes;

(v) Permits efficient isolation of one uncertain model component at a time to learn physics with less ambiguity.

This wish list is ambitious and yet partial. However, as shown below, some examples already demonstrated the plausibility of these goals.

Motivating questions for DG

With differentiable geosciences models, we hope to ask and answer the following types of questions:

a. What is the relationship between variable x and variable y?
b. What is the missing physics as part of the differential equation?
c. What should have been the assumption or function here?
d. How does factor A influence parameter β?
e. Which process is causing phenomenon P?
f. What will happen in new environmental conditions?
g. What is the information content of datasets, either input or target data for training?

Most domains in geosciences could benefit from DG (Figure 3). To provide more concrete motivating examples, we now list one example question that DG is primed to answer from each of the domains below (ordered alphabetically):

Agriculture: *Can we predict crop phenology dynamics (e.g., planting, shooting, flowering, harvesting) and assess potential production risk under future climate change (type f), which involves interconnected biotic, abiotic, and human influences?*

Climate: *Can we predict cloud processes and ocean eddies and their impact on climate sensitivity?* NNs can help to improve cloud representations.

Ecosystem: *Should we parameterize ecosystem models regarding carbon and nutrient cycles on the plant functional type level or the trait level (type c)?* Testing the configurations of differentiable parameter learning schemes could answer this question.

Cryosphere: *Can we leverage both physics and data to create more accurate models for ice dynamics within the cryosphere and better constrain its fate under climate change(type f)?* For example, the plumbing system for melted water and its influence on ice-basal bed rock friction are two of the key components for ice mass movement[121,122], with increasingly available data.

Coastal: *Can we better leverage emerging sensing platforms while improving our model representations of sediment transport and nonlinear wave-wave interactions in order to infer nearshore bathymetry at large scales (type g)?*

Geohazards: *Can we use space-based observations of geohazards, e.g., landslides*[123]*, to quantify subsurface properties (type d) so we can better predict future events (type f)?* Space-based observations and differentiable parameter learning provide an opportunity to inversely estimate properties like soil cohesion and friction angle which are challenging and expensive to measure.

Hydraulics: *How do we estimate floodplain hydraulic parameter values efficiently at large scales using new sensing data (type a, d)?* Estimation and inversion are most difficult problems facing the hydraulics research community, e.g., Manning's *n* for flow resistance and sediment transport rate. Another example is bathymetry which is required to run any hydraulics model but hard to observe.

Hydrology: *How does global groundwater-dominated baseflow respond to climate change (type a)? What is a proper, scale-appropriate way to parameterize groundwater storage and flow at the global scale (type c)?* For this question, we cannot answer it using a purely data-driven method, but could leverage differentiable models for the diagnosis.

Soil science: *Can we find functional forms to express soil hydraulic properties (water retention and hydraulic conductivity function) that describes non-equilibrium flow (type c or b)?*

Water quality: *How and to what extent do denitrification rates vary across gradients of climate, vegetation, land use, and geology conditions (type d) and thus how do they change under different climates.* Nitrate is one of the most widespread and persistent contaminants. Denitrification removes nitrate from water but the rates and extent of denitrification however depend on an array of entangled environmental factors.

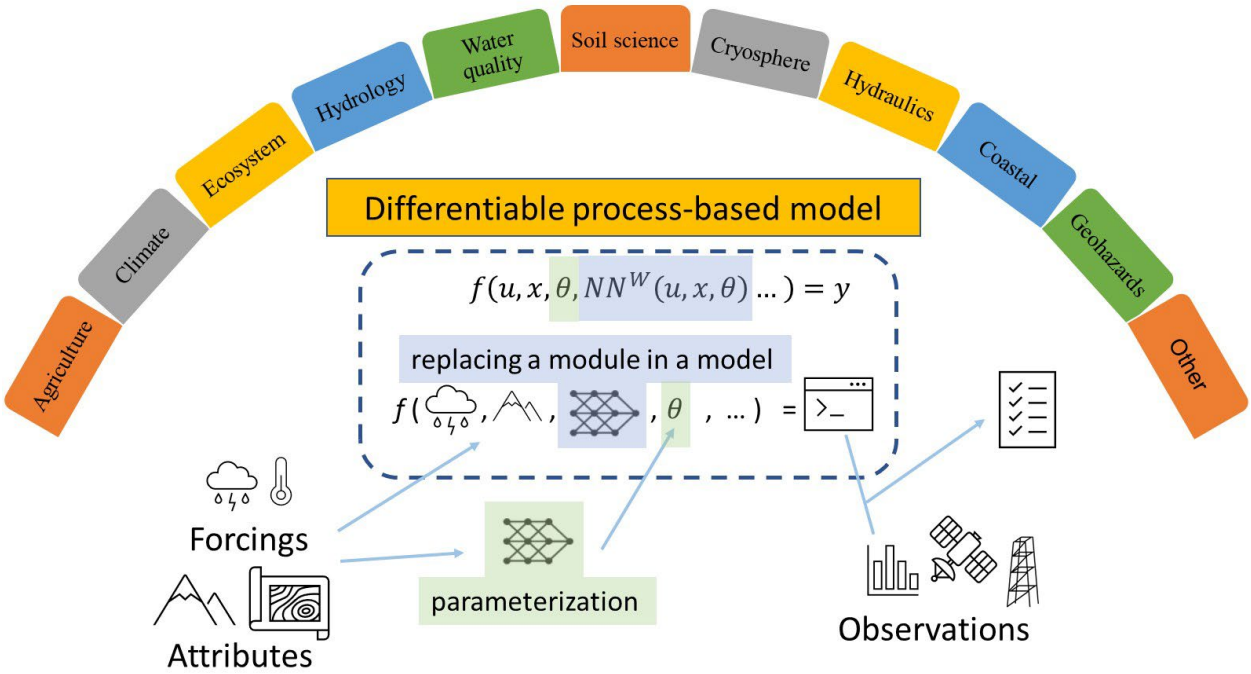

Figure 3. Differentiable Geosciences can help almost all geoscientific domains in knowledge discovery and improving simulation quality. Green and blue highlighting is used to show how there can be multiple uses for neural networks within a single model.

Classes of DG methods with examples.

DG is a young modeling paradigm and we call for wider participation. This section briefly describes early explorations of DG, categorized by how gradients are computed and employed. This section also gives examples, which are by no means exhaustive, to explain the concepts and to inspire more innovation.

I. Directly differentiating through numerical models and connecting them to NNs.

Among the several options, directly differentiating numerical models is the most straightforward method and is most similar to traditional models. Utilizing the AD functionality provided by modern ML platforms like PyTorch or Julia, one can reimplement an existing model to obtain a differentiable model version (and ensure reproducibility). Then the differentiable model is connected to NNs as discussed in the section "*Why will DG be transformative*". Because the model being trained is the same one for the forward simulation, the physics is clearly enforced, and the user obtains an efficient forward simulator for any initial, boundary and forcing conditions. They can also migrate the learned relationships to existing implementations, e.g., the national water model, to immediately support operations. However, reimplementing a model does incur non-trivial initial development cost. Mathematical changes may be required to adapt previously non-differentiable mathematical operations to be mathematically differentiable, e.g., by replacing indexing with convolutions, and to improve parallel efficiency. While DG models may not always have to run on Graphical Process Units (GPUs), enabling GPUs will improve the computational efficiency by orders of magnitudes, notwithstanding some current challenges (described in the *Challenges to address for DG* section). Our position is that in most cases, the cost is well justified due to the potential to interrogate into the model, make changes, and learn physics. The reimplementation may provide a "reset" opportunity to reexamine many of the habitually-made assumptions.

As an example, Feng et al.[101] implemented the conceptual hydrologic model HBV (a system of ODEs) on PyTorch and used coupled NNs for parameterization and optionally replaced processes with NNs (Figure 4a). Strikingly, they approached the performance level of LSTM, giving a median Nash Sutcliffe model Efficiency coefficient (NSE) of 0.732 for the CAMELS streamflow benchmark, compared to LSTM's 0.748 for the same dataset, or 0.715 vs. 0.722 for another forcing dataset (Figure 4b). They also output untrained variables such as evapotranspiration and baseflow, which agreed well with alternative estimates (Figure 4e). Moreover, in spatial extrapolation test cases, the differentiable model outperformed LSTM with respect to daily metrics and decadal trends[115] (Figure 4 c-d) due to the structural constraints, demonstrating its potential for global hydrologic modeling. Similarly, Jiang et al.[117] encoded the hydrologic model EXP-HYDRO as a recurrent NN architecture and coupled it with fully connected NNs which served as the parameterization pipeline as well as postprocessor to improve runoff. They showed that a symbiotic integration between NN and physics led to robust transferability and that snow water equivalent was well captured. In the Biogeosciences or ecosystem modeling, differentiable models found improved parameters for photosynthesis[124] at large scales.

Apart from models similar to ODEs, direct differentiation can also be applied to models operating on graphs representing the natural systems such as river networks. Bindas et al.[125] created a differentiable river routing model that was trained on daily discharge at a gauge downstream of a river network (with pretrained LSTM producing runoff as inputs to the graph) to learn a parameterization scheme for Manning's roughness coefficient (n). They obtained a power-law-like curve between n and catchment area that was consistent with the expected n behavior. Similarly, Bao et al.[126] implemented an advective dispersion equation on the river graph to simulate stream water temperature and found that the model performed better in data-sparse situations.

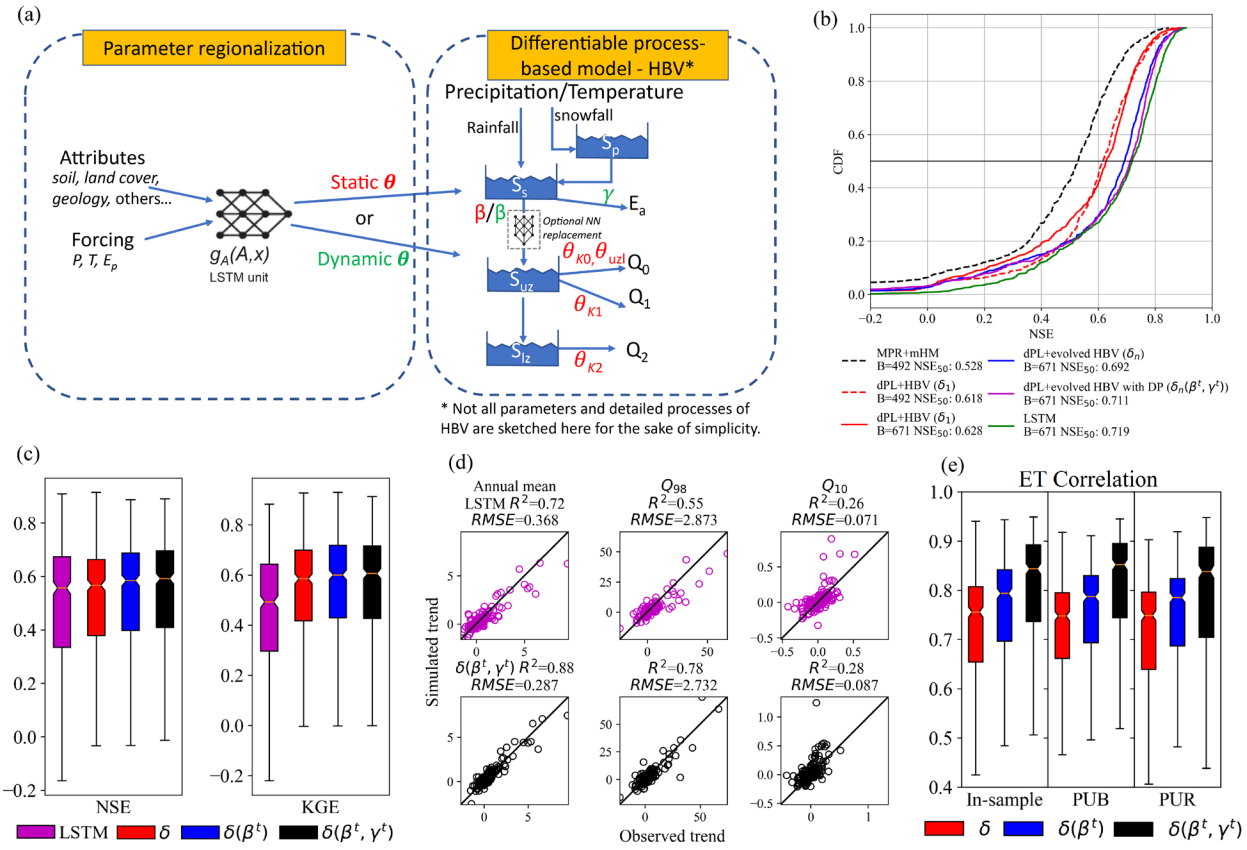

Figure 4. (From Feng et al.[101,115]. Reprint permission obtained). (a) Sketch of a differentiable hydrologic model using process-based hydrologic model HBV as a backbone (b) For temporal test using NLDAS forcings, δ models can approach the performance of LSTM and greatly outperform traditional approaches; (c) For prediction in ungauged regions (PUR; train in some regions and test in another large ungauged region), δ models can surpass the performance of LSTM; (d) For the PUR test, δ models can predict long-term trends of annual flow percentiles more reliably than LSTM. (d) δ models can predict high-quality evapotranspiration estimate (not used in training) compared to a satellite product for both in-sample and spatial generalization tests.

II. Connecting NN with PBM through surrogate models.

If we train a neural network as a surrogate for a PBM and faithfully reproduce its behavior, we can connect this surrogate to other NN components in a DG framework because the surrogate is programmatically differentiable. This idea has driven many studies to train surrogate models for hydrologic, hydraulic[127,128], and reactive transport models, and then further using inversion[129] and optimization. For another example, Tsai et al.[99] proposed differentiable parameter learning which connected a physics-based model or its surrogate model to a neural network (g) that estimated the physical parameters of the VIC ($\boldsymbol{\theta}$) hydrologic model, using some widely available attributes ($\boldsymbol{A}$): $\boldsymbol{\theta} = g(\boldsymbol{A})$, and trained the network on all sites. They found myriad benefits and a favorable scaling relationships (with more data). The full example is described in Supplementary Information S1. The initial effort of the surrogate approach is low, but one often needs to continuously train the surrogate models as the optimization goes to different regions of the parameter space. Furthermore, a surrogate model does not permits changes to the model. As a result, we only recommend it for highly complex and computationally expensive models that are intractable for reimplementation.

Such cases may arise in hydraulics or subsurface modeling where governing PDEs of fluid dynamics and sediment transport must be solved with high spatial and temporal resolutions. Solving PDEs using neural networks has obtained increasing attention, with many studies seeking NNs that can approximate the numerical solution[130–132], e.g., for the Richards equation[133]. For another example, for 2D hydraulic simulations, a differentiable surrogate model can be employed for the inversion of bathymetry[134]. While not directly the philosophical theme of DG, and so far many studies are at the proof-of-accuracy state (as opposed to being used in optimization), surrogate models can certainly accelerate and aid the mission of DG.

III. Adjoint solvers for gradients

Adjoint methods have been used to solve optimization problems governed by PDEs, with the adjoint equations derived either manually[135], or, more rarely, by automated programs.[136] Both AD and adjoint are accurate computation of derivatives given a numerical model that implements the physical law [137]. Once the models become differentiable, whichever way we enable it, we can also use the gradients for other purposes, e.g., performing sensitivity analysis[138] or PDE-constrained optimization. Adjoint methods might be more computationally efficient than AD for certain problems by exploiting the structure of the mathematical model. Adjoint solvers have long been employed in numerical weather prediction, e.g., 4DVar[139] and groundwater modeling[140], for the purpose of efficient data assimilation or calibration. However, these adjoint solvers were traditionally not merged with neural network training, perhaps because the role of differentiable modeling was not clear at the time.

In Mitusch (2021), unknown functions or operators in a PDE were replaced by NNs, while the PDE is discretized by a finite element method and the gradient provided by the adjoint method. To overcome the challenge facing Newton iteration convergence due to the incorporation of NN and the lack of a preconditioner, they used an operator-splitting approach to discretize the PDE into two subproblems. The first subproblem only has differential operators of the PDE, not NNs, while the other subproblem with NNs can be solved by integrating NNs by a Gaussian quadrature rule. They could recover a non-linear coefficient in the Poisson equation and the heat equation. The approach can similarly apply to equations in geosciences.

IV. PINN method for learning parameters and constitutive relationships

In the physics-informed neural networks (PINN) method[100,103,141] parameterization schemes can be learned by modeling the space-dependent properties of a system (e.g., hydraulic conductivity of porous media) and unknown constitutive relationships (e.g., pressure-dependent permeability of the unsaturated porous media and strain-dependent effective viscosity of non-Newtonian fluids). To make the hydrological model fully differentiable, in the PINN method, the states of the system are also modeled with neural networks. Then, all neural networks are jointly trained using the system state measurements and the fundamental conservation law constraints added as penalty terms to the joint loss function. As a result, the PINN method allows for learning systems parameters and constitutive relationships using the measurements of the system states that are easier to collect than the direct measurements of the parameters. The latter would be needed for learning parameters using data only. An example of the application of PINN for learning the constitutive relationship in the unsaturated flow model is given in the Supplementary Information S1.

V. ML-dominant hybrid models with limited physics.

Another class of models applicable in the data-rich realm employs NNs for the majority of modeling but inserting physical operators for imposing limited physics. For example, Kraft et al.[142] proposed an architecture that intends to use LSTM to estimate physical fluxes including evaporation, runoff and recharge, and mass balance equations were added. Since the only supervising signal for the fluxes is discharge (they were completely based on LSTM), it is uncertain whether the terms maintain their physical meaning. Authors later constrained the system using more observations[143] which improved the simulations but data limitations mentioned earlier may still apply. Liu et al.[144] trained LSTM soil moisture models at 9-km resolution, whose solution was fed into an averaging operation to obtain output at 36-km, and computed loss function at both resolutions against in-situ and satellite-based observations, respectively. They found the model learning from both data sources to outperform those learning from one. Overall, ML-dominant systems can be potent predictors and a beneficial option in DG, but one needs to carefully assess the interpretability and physical significance of the terms.

Challenges to address for DG

Vanishing gradient is a major issue in NN training, which can occur if there are too many calculation steps: in some cases, the parameters in deeper layers have very small gradients, so they become exceedingly sensitive to machine precision and thus difficult to train[145,146]. Vanishing gradient can happen with recurrent NNs, which are similar to differentiable models solving dynamical system problems, or direct differentiation through numerical solvers. We anticipate new issues to emerge and new solutions to address them.

While numerical solvers for ordinary differentiable equations (ODEs) can be readily accommodated by current differentiable computing platforms like PyTorch or Julia, partial differential equations (PDEs) may still be challenging. This is first because solving PDEs requires substantial computation, which makes training by a batch of examples expensive in terms of both compute and memory usage. The architecture suitable for big-data ML training tends to prefer massive parallelism, which reduces the range of suitable numerical algorithms. Algorithms solving PDEs may not connect well with ML computing infrastructure, and may encounter issues with vanishing gradients[147,148] due to too many operations over many time steps, as discussed below. Nevertheless, some differentiable numerical solvers to PDEs have been proposed and tested in computational fluid mechanics, and appear to be alternative to standard solvers in Fortran or C/C++[149].

Since differentiable modeling allows us to learn processes, it is to be expected that we may run into "process non-uniqueness", i.e., "process equifinality". In traditional hydrologic modeling, "multiple working hypotheses" has long been proposed as a viable way to test different model formulations coupled together[30]. While the involvement of simultaneous constraints from big data should alleviate these problems, multiple options are needed to address the non-uniqueness issue and reduce uncertainty. First, we can be aided by systematic development approaches that allow us to solve a part of the problem or determine one process at a time to reduce the intertwining of issues, yet also avoiding falling into local minima as "greedy" algorithms sometimes do. Second, we need more mature uncertainty quantification techniques, i.e., going beyond ensemble methods[150–153], to help assess the success and failures of hypotheses.

Finally, we need large and multivariate benchmarks and extrapolation tests that match the intended use cases to verify the validity and realism of physical outputs. For example, models intended for climate change impact assessment must be tested for long term projection fidelity; models for global-scale applications must pass rigorous spatial extrapolation tests --- when we withhold a large region from training (similar to the global-scale application), the error is larger than on training sites or randomly held-out sites.

These are typical tests relevant to climate-scale geoscientific problems but the uncertainty was often not fully realized[114].

Concluding remarks

Throughout our examples, we demonstrated that the DG is a novel framework that allows varying amounts of structural priors to be flexibly employed along with NNs, ranging from having just a few physically-based operators to significant physically-based structures. We can perceive DG as a continuum of methods where we choose how much structural prior information to insert into the model. The divide is dissolved between ML and PBM. Understanding the core strength of deep networks and the role of differentiable programming allows us to break free from thinking about fixed methods or approaches for their integration – we can now focus on physical processes, uncertainty, unknown relationships and data. With differentiable modeling serving as a unifying framework, one stops paying attention to the divide between ML and PBM, but thinks in terms of what priors we insert and how the gradients were constructed. As long as we can obtain gradients end-to-end, the system can learn from available multisource, multiscale datasets, and leverage the benefits of big or small data. Differentiable modeling, while emerging as a technological breakthrough, can lead to philosophical changes – we are liberated to ask new types of questions and utilize data in new ways. Recognizing the power and philosophy of DG will usher in a paradigm shift in geoscientific modeling.

Acknowledgements

We attribute many ideas of the paper to a discussion in the HydroML symposium, University Park, PA, May 2022, https://bit.ly/3g3DQNX, sponsored by National Science Foundation EAR #2015680 and Penn State Institute for Computational and Data Sciences is a Computational Research. Content related to this paper was also presented in some presentations, including AI4ESP talk online https://bit.ly/3etm5aI in Nov 2021. Shen was supported by National Science Foundation EAR-2221880 and Office of Science, US Department of Energy under award DE-SC0016605. Gentine acknowledges funding from the National Science Foundational Science and Technology Center, Learning the Earth with Artificial intelligence and Physics (LEAP), award #2019625 and USMILE European Research Council grant. Marty Wernimont at USGS greatly improved the presentation of Figures 1 and 2; Wernimont and Appling were supported by the USGS Water Mission Area, Water Availability and Use Science Program. Any use of trade, firm, or product names is for descriptive purposes only and does not imply endorsement by the U.S. Government.

Competing Interests

KL and CS have financial interests in HydroSapient, Inc., a company which could potentially benefit from the results of this research. This interest has been reviewed by the University in accordance with its Individual Conflict of Interest policy, for the purpose of maintaining the objectivity and the integrity of research at The Pennsylvania State University.

Main Bibliography

1. Ajami, N. K., Gupta, H., Wagener, T. & Sorooshian, S. Calibration of a semi-distributed hydrologic model for streamflow estimation along a river system. *Journal of Hydrology* **298**, 112–135 (2004).
2. van Griensven, A. & Meixner, T. A global and efficient multi-objective auto-calibration and uncertainty estimation method for water quality catchment models. *Journal of Hydroinformatics* **9**, 277–291 (2007).
3. Barendrecht, M. H. *et al.* The value of empirical data for estimating the parameters of a sociohydrological flood risk model. *Water Resour. Res.* **55**, 1312–1336 (2019).
4. Post, H., Vrugt, J. A., Fox, A., Vereecken, H. & Franssen, H.-J. H. Estimation of Community Land Model parameters for an improved assessment of net carbon fluxes at European sites. *Journal of Geophysical Research: Biogeosciences* **122**, 661–689 (2017).
5. Aumont, O., Ethé, C., Tagliabue, A., Bopp, L. & Gehlen, M. PISCES-v2: An ocean biogeochemical model for carbon and ecosystem studies. *Geoscientific Model Development* **8**, 2465–2513 (2015).
6. Ahmed, M. *et al.* Calibration and validation of APSIM-Wheat and CERES-Wheat for spring wheat under rainfed conditions: Models evaluation and application. *Computers and Electronics in Agriculture* **123**, 384–401 (2016).
7. Lepore, C., Arnone, E., Noto, L. V., Sivandran, G. & Bras, R. L. Physically based modeling of rainfall-triggered landslides: a case study in the Luquillo forest, Puerto Rico. *Hydrology and Earth System Sciences* **17**, 3371–3387 (2013).
8. Shirzaei, M. *et al.* Measuring, modelling and projecting coastal land subsidence. *Nat Rev Earth Environ* **2**, 40–58 (2021).
9. Lee, A., Aubeneau, A., Liu, X. & Cardenas, M. B. Hyporheic Exchange in Sand Dunes Under a Freely Deforming River Water Surface. *Water Resources Research* **57**, e2020WR028817 (2021).
10. Li, B. *et al.* Flexible and Modular Simultaneous Modeling of Flow and Reactive Transport in Rivers and Hyporheic Zones. *Water Resources Research* **56**, e2019WR026528 (2020).

11. Flato, G. M. Earth system models: an overview. *WIREs Climate Change* **2**, 783–800 (2011).
12. Danabasoglu, G. *et al.* The Community Earth System Model Version 2 (CESM2). *Journal of Advances in Modeling Earth Systems* **12**, e2019MS001916 (2020).
13. Eyring, V. *et al.* Overview of the Coupled Model Intercomparison Project Phase 6 (CMIP6) experimental design and organization. *Geosci. Model Dev.* **9**, 1937–1958 (2016).
14. Calvin, K. *et al.* GCAM v5.1: representing the linkages between energy, water, land, climate, and economic systems. *Geoscientific Model Development* **12**, 677–698 (2019).
15. ISIMIP. The Inter-Sectoral Impact Model Intercomparison Project (ISIMIP). *ISIMIP* https://www.isimip.org/ (2022).
16. Lange, S. Trend-preserving bias adjustment and statistical downscaling with ISIMIP3BASD (v1.0). *Geoscientific Model Development* **12**, 3055–3070 (2019).
17. Weyant, J. *et al.* Integrated assessment of climate change: An overview and comparison of approaches and results. in 367–396 (1996).
18. IPCC. *Climate Change 2021: The Physical Science Basis. Contribution of Working Group I to the Sixth Assessment Report of the Intergovernmental Panel on Climate Change*. (Cambridge University Press, 2021).
19. Rogger, M. *et al.* Land use change impacts on floods at the catchment scale: Challenges and opportunities for future research. *Water Resources Research* **53**, 5209–5219 (2017).
20. Biemans, H. *et al.* Importance of snow and glacier meltwater for agriculture on the Indo-Gangetic Plain. *Nat Sustain* **2**, 594–601 (2019).
21. Hood, R. R. *et al.* The Chesapeake Bay program modeling system: Overview and recommendations for future development. *Ecological Modelling* **456**, 109635 (2021).
22. Fatichi, S. *et al.* An overview of current applications, challenges, and future trends in distributed process-based models in hydrology. *Journal of Hydrology* **537**, 45–60 (2016).
23. Fan, Y. *et al.* Hillslope hydrology in global change research and earth system modeling. *Water Resources Research* **55**, 1737–1772 (2019).

24. van Kampenhout, L. *et al.* Improving the representation of polar snow and firn in the community earth system model. *Journal of Advances in Modeling Earth Systems* **9**, 2583–2600 (2017).
25. Medlyn, B. E. *et al.* Using ecosystem experiments to improve vegetation models. *Nature Clim Change* **5**, 528–534 (2015).
26. Nearing, G. S. *et al.* What role does hydrological science play in the age of machine learning? *Water Resources Research* **57**, e2020WR028091 (2021).
27. Shen, C. *et al.* HESS Opinions: Incubating deep-learning-powered hydrologic science advances as a community. *Hydrology and Earth System Sciences* **22**, 5639–5656 (2018).
28. Hunt, R. J., Fienen, M. N. & White, J. T. Revisiting "An Exercise in Groundwater Model Calibration and Prediction" After 30 Years: Insights and New Directions. *Groundwater* **58**, 168–182 (2020).
29. Addor, N. & Melsen, L. A. Legacy, rather than adequacy, drives the selection of hydrological models. *Water Resources Research* **55**, 378–390 (2019).
30. Clark, M. P., Kavetski, D. & Fenicia, F. Pursuing the method of multiple working hypotheses for hydrological modeling. *Water Resources Research* **47**, (2011).
31. Jakeman, A. J. & Hornberger, G. M. How much complexity is warranted in a rainfall-runoff model? *Water Resources Research* **29**, 2637–2649 (1993).
32. Wagener, T., Wheater, H. S. & Gupta, H. V. Identification and Evaluation of Watershed Models. in *Calibration of Watershed Models* 29–47 (American Geophysical Union (AGU), 2003). doi:10.1029/WS006p0029.
33. Young, P., Jakeman, A. & McMurtrie, R. An instrumental variable method for model order identification. *Automatica* **16**, 281–294 (1980).
34. Shen, C. A transdisciplinary review of deep learning research and its relevance for water resources scientists. *Water Resources Research* **54**, 8558–8593 (2018).
35. Abbott, B. W. *et al.* Human domination of the global water cycle absent from depictions and perceptions. *Nat. Geosci.* **12**, 533–540 (2019).

36. Lemordant, L., Gentine, P., Swann, A. S., Cook, B. I. & Scheff, J. Critical impact of vegetation physiology on the continental hydrologic cycle in response to increasing CO2. *PNAS* **115**, 4093–4098 (2018).

37. Trancoso, R., Larsen, J. R., McVicar, T. R., Phinn, S. R. & McAlpine, C. A. CO2-vegetation feedbacks and other climate changes implicated in reducing base flow. *Geophysical Research Letters* **44**, 2310–2318 (2017).

38. Yu, D. *et al.* Socio-hydrology: an interplay of design and self-organization in a multilevel world. *Ecology and Society* **25**, (2020).

39. Leopold, G. Nvidia's Huang Sees AI 'Cambrian Explosion'. *Datanami* https://www.datanami.com/2017/05/24/nvidias-huang-sees-ai-cambrian-explosion/ (2017).

40. LeCun, Y., Bengio, Y. & Hinton, G. Deep Learning. *Nature* **521**, 436–444 (2015).

41. Khaki, S. & Wang, L. Crop yield prediction using deep neural networks. *Frontiers in Plant Science* **10**, (2019).

42. Wang, A. X., Tran, C., Desai, N., Lobell, D. & Ermon, S. Deep Transfer Learning for Crop Yield Prediction with Remote Sensing Data. in *Proceedings of the 1st ACM SIGCAS Conference on Computing and Sustainable Societies* 1–5 (Association for Computing Machinery, 2018). doi:10.1145/3209811.3212707.

43. Pan, B. *et al.* Improving Seasonal Forecast Using Probabilistic Deep Learning. *Journal of Advances in Modeling Earth Systems* **14**, e2021MS002766 (2022).

44. Shi, X. *et al.* Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting. in *Advances in Neural Information Processing Systems* vol. 28 (Curran Associates, Inc., 2015).

45. Bhowmik, M., Singh, M., Rao, S. & Paul, S. DeepClouds.ai: Deep learning enabled computationally cheap direct numerical simulations. Preprint at https://doi.org/10.48550/arXiv.2208.08956 (2022).

46. Lin, G.-Y., Chen, H.-W., Chen, B.-J. & Yang, Y.-C. Characterization of temporal PM2.5, nitrate, and sulfate using deep learning techniques. *Atmospheric Pollution Research* **13**, 101260 (2022).

47. Varadharajan, C. *et al.* Can machine learning accelerate process understanding and decision-relevant predictions of river water quality? *Hydrological Processes* **36**, e14565 (2022).
48. Jia, X. *et al.* Physics-Guided Recurrent Graph Model for Predicting Flow and Temperature in River Networks. in *Proceedings of the 2021 SIAM International Conference on Data Mining (SDM)* 612–620 (Society for Industrial and Applied Mathematics, 2021). doi:10.1137/1.9781611976700.69.
49. Rahmani, F. *et al.* Exploring the exceptional performance of a deep learning stream temperature model and the value of streamflow data. *Environ. Res. Lett.* (2021) doi:10.1088/1748-9326/abd501.
50. Rahmani, F., Shen, C., Oliver, S., Lawson, K. & Appling, A. Deep learning approaches for improving prediction of daily stream temperature in data-scarce, unmonitored, and dammed basins. *Hydrological Processes* **35**, e14400 (2021).
51. Read, J. S. *et al.* Process-guided deep learning predictions of lake water temperature. *Water Resources Research* **55**, 9173–9190 (2019).
52. Zhi, W. *et al.* From hydrometeorology to river water quality: Can a deep learning model predict dissolved oxygen at the continental scale? *Environ. Sci. Technol.* **55**, 2357–2368 (2021).
53. He, M., Wu, S., Huang, B., Kang, C. & Gui, F. Prediction of Total Nitrogen and Phosphorus in Surface Water by Deep Learning Methods Based on Multi-Scale Feature Extraction. *Water* **14**, 1643 (2022).
54. Hrnjica, B., Mehr, A. D., Jakupović, E., Crnkić, A. & Hasanagić, R. Application of Deep Learning Neural Networks for Nitrate Prediction in the Klokot River, Bosnia and Herzegovina. in *2021 7th International Conference on Control, Instrumentation and Automation (ICCIA)* 1–6 (2021). doi:10.1109/ICCIA52082.2021.9403565.
55. Xiong, R. *et al.* Predicting Dynamic Riverine Nitrogen Export in Unmonitored Watersheds: Leveraging Insights of AI from Data-Rich Regions. *Environ. Sci. Technol.* **56**, 10530–10542 (2022).
56. Shen, C., Chen, X. & Laloy, E. Editorial: Broadening the use of machine learning in hydrology. *Front. Water* **3**, (2021).

57. Fang, K., Shen, C., Kifer, D. & Yang, X. Prolongation of SMAP to spatiotemporally seamless coverage of continental U.S. using a deep learning neural network. *Geophys. Res. Lett.* **44**, 11,030-11,039 (2017).

58. Fang, K., Pan, M. & Shen, C. The value of SMAP for long-term soil moisture estimation with the help of deep learning. *IEEE Trans. Geosci. Remote Sensing* **57**, 2221–2233 (2019).

59. Fang, K. & Shen, C. Near-real-time forecast of satellite-based soil moisture using long short-term memory with an adaptive data integration kernel. *J. Hydrometeor.* **21**, 399–413 (2020).

60. Feng, D., Fang, K. & Shen, C. Enhancing streamflow forecast and extracting insights using long-short term memory networks with data integration at continental scales. *Water Resources Research* **56**, e2019WR026793 (2020).

61. Kratzert, F. *et al.* Towards learning universal, regional, and local hydrological behaviors via machine learning applied to large-sample datasets. *Hydrology and Earth System Sciences* **23**, 5089–5110 (2019).

62. Sun, A. Y., Jiang, P., Mudunuru, M. K. & Chen, X. Explore spatio-temporal learning of large sample hydrology using graph neural networks. *Water Resources Research* **57**, e2021WR030394 (2021).

63. Xiang, Z. & Demir, I. Distributed long-term hourly streamflow predictions using deep learning – A case study for State of Iowa. *Environmental Modelling & Software* **131**, 104761 (2020).

64. Alemohammad, S. H. *et al.* Water, Energy, and Carbon with Artificial Neural Networks (WECANN): a statistically based estimate of global surface turbulent fluxes and gross primary productivity using solar-induced fluorescence. *Biogeosciences* **14**, 4101–4124 (2017).

65. Jung, M. *et al.* The FLUXCOM ensemble of global land-atmosphere energy fluxes. *Sci Data* **6**, 74 (2019).

66. Zhao, W. L. *et al.* Physics-constrained machine learning of evapotranspiration. *Geophysical Research Letters* **46**, 14496–14507 (2019).

67. Afzaal, H., Farooque, A. A., Abbas, F., Acharya, B. & Esau, T. Groundwater estimation from major physical hydrology components using artificial neural networks and deep learning. *Water* **12**, 5 (2020).

68. Meyal, A. Y. *et al.* Automated cloud based long short-term memory neural network based SWE prediction. *Front. Water* **2**, (2020).

69. Hochreiter, S. & Schmidhuber, J. Long Short-Term Memory. *Neural Computation* **9**, 1735–1780 (1997).

70. Krizhevsky, A., Sutskever, I. & Hinton, G. E. ImageNet Classification with Deep Convolutional Neural Networks. in *Advances in Neural Information Processing Systems* vol. 25 1097–1105 (Curran Associates, Inc., 2012).

71. Lecun, Y. & Bengio, Y. Convolutional networks for images, speech, and time-series. in *The handbook of brain theory and neural networks* (ed. Arbib, M. A.) (MIT Press, 1995).

72. McDonnell, J. J. & Beven, K. Debates—The future of hydrological sciences: A (common) path forward? A call to action aimed at understanding velocities, celerities and residence time distributions of the headwater hydrograph. *Water Resources Research* **50**, 5342–5350 (2014).

73. Appling, A. P., Oliver, S. K., Read, J. S., Sadler, J. M. & Zwart, J. Machine learning for understanding inland water quantity, quality, and ecology. (2022).

74. Li, L. *et al.* Toward catchment hydro-biogeochemical theories. *WIREs Water* **8**, e1495 (2021).

75. Fang, K., Kifer, D., Lawson, K., Feng, D. & Shen, C. The data synergy effects of time-series deep learning models in hydrology. *Water Resources Research* **58**, e2021WR029583 (2022).

76. Bach, S. *et al.* On Pixel-Wise Explanations for Non-Linear Classifier Decisions by Layer-Wise Relevance Propagation. *PLOS ONE* **10**, e0130140 (2015).

77. Montavon, G., Samek, W. & Müller, K.-R. Methods for Interpreting and Understanding Deep Neural Networks. *Digital Signal Processing* (2017) doi:10/gcvxrb.

78. Toms, B. A., Barnes, E. A. & Ebert-Uphoff, I. Physically interpretable neural networks for the geosciences: Applications to earth system variability. *Journal of Advances in Modeling Earth Systems* **12**, e2019MS002002 (2020).

79. Fleming, S. W., Watson, J. R., Ellenson, A., Cannon, A. J. & Vesselinov, V. C. Machine learning in Earth and environmental science requires education and research policy reforms. *Nat. Geosci.* **14**, 878–880 (2021).

80. Hornik, K. Approximation capabilities of multilayer feedforward networks. *Neural Networks* **4**, 251–257 (1991).

81. Hornik, K., Stinchcombe, M. & White, H. Multilayer feedforward networks are universal approximators. *Neural Networks* **2**, 359–366 (1989).

82. Zhai, X., Kolesnikov, A., Houlsby, N. & Beyer, L. Scaling vision transformers. Preprint at https://doi.org/10.48550/arXiv.2106.04560 (2022).

83. Deb, K., Pratap, A., Agarwal, S. & Meyarivan, T. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation* **6**, 182–197 (2002).

84. Duan, Q., Sorooshian, S. & Gupta, V. Effective and efficient global optimization for conceptual rainfall-runoff models. *Water Resources Research* **28**, 1015–1031 (1992).

85. Zitzler, E., Laumanns, M. & Thiele, L. *SPEA2: Improving the strength pareto evolutionary algorithm*. *TIK Report* vol. 103 https://www.research-collection.ethz.ch/handle/20.500.11850/145755 (2001).

86. Liu, S. *et al.* A hybrid approach of support vector regression with genetic algorithm optimization for aquaculture water quality prediction. *Mathematical and Computer Modelling* **58**, 458–465 (2013).

87. Zambrano-Bigiarini, M. & Rojas, R. A model-independent Particle Swarm Optimisation software for model calibration. *Environmental Modelling & Software* **43**, 5–25 (2013).

88. Baydin, A. G., Pearlmutter, B. A., Radul, A. A. & Siskind, J. M. Automatic differentiation in machine learning: A survey. *Journal of Machine Learning Research* **18**, 1–43 (2018).

89. Innes, M. *et al.* A Differentiable Programming System to Bridge Machine Learning and Scientific Computing. Preprint at http://arxiv.org/abs/1907.07587 (2019).

90. Paszke, A. *et al.* Automatic differentiation in PyTorch. in *31st Conference on Neural Information Processing Systems (NIPS 2017)* (2017).

91. Rumelhart, D. E., Hinton, G. & Williams, R. J. Learning representations by back-propagating errors. *Nature* **323**, 533–536 (1986).
92. Errico, R. M. What Is an Adjoint Model? *Bulletin of the American Meteorological Society* **78**, 2577–2592 (1997).
93. Johnson, S. G. Notes on Adjoint Methods for 18.335. 7 (2021).
94. Pal, A., Edelman, A. & Rackauckas, C. Mixing Implicit and Explicit Deep Learning with Skip DEQs and Infinite Time Neural ODEs (Continuous DEQs). Preprint at https://doi.org/10.48550/arXiv.2201.12240 (2022).
95. Ghattas, O. & Willcox, K. Learning physics-based models from data: perspectives from inverse problems and model reduction. *Acta Numerica* **30**, 445–554 (2021).
96. Goodfellow, I., Bengio, Y. & Courville, A. Numerical Computation - Gradient-Based Optimization. in *Deep Learning* 775 (The MIT Press, 2016).
97. Baker, N. *et al. Workshop Report on Basic Research Needs for Scientific Machine Learning: Core Technologies for Artificial Intelligence*. https://www.osti.gov/biblio/1478744 (2019) doi:10.2172/1478744.
98. Rackauckas, C. *et al.* Universal differential equations for scientific machine learning. Preprint at http://arxiv.org/abs/2001.04385 (2021).
99. Tsai, W.-P. *et al.* From calibration to parameter learning: Harnessing the scaling effects of big data in geoscientific modeling. *Nat Commun* **12**, 5988 (2021).
100. Raissi, M., Perdikaris, P. & Karniadakis, G. E. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics* **378**, 686–707 (2019).
101. Feng, D., Liu, J., Lawson, K. & Shen, C. Differentiable, learnable, regionalized process-based models with multiphysical outputs can approach state-of-the-art hydrologic prediction accuracy. *Water Resources Research* **58**, e2022WR032404 (2022).

838 102. Huang, D. Z., Xu, K., Farhat, C. & Darve, E. Learning constitutive relations from indirect
839 observations using deep neural networks. *Journal of Computational Physics* **416**, 109491 (2020).
840 103. Tartakovsky, A. M., Marrero, C. O., Perdikaris, P., Tartakovsky, G. D. & Barajas-Solano, D. Physics-
841 informed deep neural networks for learning parameters and constitutive relationships in subsurface
842 flow problems. *Water Resources Research* **56**, e2019WR026731 (2020).
843 104. Padarian, J., McBratney, A. B. & Minasny, B. Game theory interpretation of digital soil mapping
844 convolutional neural networks. *SOIL* **6**, 389–397 (2020).
845 105. Udrescu, S.-M. & Tegmark, M. AI Feynman: A physics-inspired method for symbolic regression.
846 *Science Advances* **6**, eaay2631 (2020).
847 106. Montavon, G., Binder, A., Lapuschkin, S., Samek, W. & Müller, K.-R. Layer-Wise Relevance
848 Propagation: An Overview. in *Explainable AI: Interpreting, Explaining and Visualizing Deep*
849 *Learning* (eds. Samek, W., Montavon, G., Vedaldi, A., Hansen, L. K. & Müller, K.-R.) 193–209
850 (Springer International Publishing, 2019). doi:10.1007/978-3-030-28954-6_10.
851 107. Lundberg, S. M. & Lee, S.-I. A unified approach to interpreting model predictions. in *Proceedings of*
852 *the 31st International Conference on Neural Information Processing Systems* 4768–4777 (Curran
853 Associates Inc., 2017).
854 108. Molnar, C. 9.2 Local Surrogate (LIME). in *Interpretable Machine Learning* (2022).
855 109. Ma, Y., Tsao, D. & Shum, H.-Y. On the Principles of Parsimony and Self-Consistency for the
856 Emergence of Intelligence. Preprint at https://doi.org/10.48550/arXiv.2207.04630 (2022).
857 110. Myneni, Ranga, Knyazikhin, Yuri, & Park, Taejin. MCD15A2H MODIS/Terra+Aqua Leaf Area
858 Index/FPAR 8-day L4 Global 500m SIN Grid V006. *NASA EOSDIS Land Processes DAAC* (2015)
859 doi:10.5067/MODIS/MCD15A2H.006.
860 111. ESA. About SMOS - Soil Moisture and Ocean Salinity mission. *European Space Agency (ESA)*
861 https://earth.esa.int/eogateway/missions/smos (2022).

112. O'Neill, P. E. *et al.* SMAP Enhanced L3 Radiometer Global and Polar Grid Daily 9 km EASE-Grid Soil Moisture, Version 5 (SPL3SMP_E). *NASA National Snow and Ice Data Center (NSIDC) Distributed Active Archive Center (DAAC)* (2021) doi:10.5067/4DQ54OUIJ9DL.

113. Lin, Y.-S. *et al.* Optimal stomatal behaviour around the world. *Nature Climate Change* **5**, 459–464 (2015).

114. Feng, D., Lawson, K. & Shen, C. Mitigating prediction error of deep learning streamflow models in large data-sparse regions with ensemble modeling and soft data. *Geophysical Research Letters* **48**, e2021GL092999 (2021).

115. Feng, D., Beck, H., Lawson, K. & Shen, C. The suitability of differentiable, learnable hydrologic models for ungauged regions and climate change impact assessment. *Hydrology and Earth System Sciences Discussions* 1–28 (2022) doi:10.5194/hess-2022-245.

116. Wagener, T. *et al.* The future of hydrology: An evolving science for a changing world. *Water Resources Research* **46**, 1–10 (2010).

117. Jiang, S., Zheng, Y. & Solomatine, D. Improving AI system awareness of geoscience knowledge: Symbiotic integration of physical approaches and deep learning. *Geophysical Research Letters* **47**, e2020GL088229 (2020).

118. Beven, K. A manifesto for the equifinality thesis. *Journal of Hydrology* **320**, 18–36 (2006).

119. Pokhrel, P., Gupta, H. V. & Wagener, T. A spatial regularization approach to parameter estimation for a distributed watershed model. *Water Resour. Res.* **44**, (2008).

120. Wagener, T., McIntyre, N., Lees, M. J., Wheater, H. S. & Gupta, H. V. Towards reduced uncertainty in conceptual rainfall-runoff modelling: Dynamic identifiability analysis. *Hydrol. Process.* **17**, 455–476 (2003).

121. Bell, R. E. The role of subglacial water in ice-sheet mass balance. *Nature Geosci* **1**, 297–304 (2008).

122. Chen, Y., Liu, X., Gulley, J. D. & Mankoff, K. D. Subglacial conduit roughness: Insights from computational fluid dynamics models. *Geophysical Research Letters* **45**, 11,206-11,218 (2018).

123. Nagendra, S. *et al.* Constructing a large-scale landslide database across heterogeneous environments using task-specific model updates. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* **15**, 4349–4370 (2022).

124. Aboelyazeed, D. *et al.* A differentiable ecosystem modeling framework for large-scale inverse problems: demonstration with photosynthesis simulations. *Biogeosciences Discussions* 1–33 (2022) doi:10.5194/bg-2022-211.

125. Bindas, T. *et al.* Improving large-basin streamflow simulation using a modular, differentiable, learnable graph model for routing. Preprint at https://doi.org/10.1002/essoar.10512512.1 (2022).

126. Bao, T. *et al.* Partial Differential Equation Driven Dynamic Graph Networks for Predicting Stream Water Temperature. in *2021 IEEE International Conference on Data Mining (ICDM)* 11–20 (2021). doi:10.1109/ICDM51629.2021.00011.

127. Forghani, M. *et al.* Application of deep learning to large scale riverine flow velocity estimation. *Stoch Environ Res Risk Assess* **35**, 1069–1088 (2021).

128. Forghani, M. *et al.* Variational encoder geostatistical analysis (VEGAS) with an application to large scale riverine bathymetry. *Advances in Water Resources* **170**, 104323 (2022).

129. Asher, M. J., Croke, B. F. W., Jakeman, A. J. & Peeters, L. J. M. A review of surrogate models and their application to groundwater modeling. *Water Resources Research* **51**, 5957–5973 (2015).

130. Blechschmidt, J. & Ernst, O. G. Three ways to solve partial differential equations with neural networks — A review. *GAMM-Mitteilungen* **44**, e202100006 (2021).

131. Lu, L., Meng, X., Mao, Z. & Karniadakis, G. E. DeepXDE: A deep learning library for solving differential equations. *SIAM Rev.* **63**, 208–228 (2021).

132. Takamoto, M. *et al.* PDEBENCH: An Extensive Benchmark for Scientific Machine Learning. Preprint at https://doi.org/10.48550/arXiv.2210.07182 (2022).

133. Maxwell, R. M., Condon, L. E. & Melchior, P. A physics-informed, machine learning emulator of a 2D surface water model: What temporal networks and simulation-based inference can help us learn about hydrologic processes. *Water* **13**, 3633 (2021).

134. Liu, X., Song, Y. & Shen, C. Bathymetry inversion using a deep-learning-based surrogate for shallow water equations solvers. Preprint at https://doi.org/10.48550/arXiv.2203.02821 (2022).
135. Mitusch, S. K., Funke, S. W. & Kuchta, M. Hybrid FEM-NN models: Combining artificial neural networks with the finite element method. *Journal of Computational Physics* **446**, 110651 (2021).
136. Farrell, P. E., Ham, D. A., Funke, S. W. & Rognes, M. E. Automated derivation of the adjoint of high-level transient finite element programs. *SIAM J. Sci. Comput.* **35**, C369–C393 (2013).
137. Wilcox, L. C., Stadler, G., Bui-Thanh, T. & Ghattas, O. Discretely exact derivatives for hyperbolic pde-constrained optimization problems discretized by the Discontinuous Galerkin Method. *J Sci Comput* **63**, 138–162 (2015).
138. Isaac, T., Petra, N., Stadler, G. & Ghattas, O. Scalable and efficient algorithms for the propagation of uncertainty from data through inference to prediction for large-scale problems, with application to flow of the Antarctic ice sheet. *Journal of Computational Physics* **296**, 348–368 (2015).
139. Fisher, M. & Andersson, E. *Developments in 4D-Var and Kalman Filtering*. https://www.ecmwf.int/sites/default/files/elibrary/2001/9409-developments-4d-var-and-kalman-filtering.pdf (2001).
140. Neupauer, R. M. & Wilson, J. L. Adjoint-derived location and travel time probabilities for a multidimensional groundwater system. *Water Resources Research* **37**, 1657–1668 (2001).
141. He, Q., Barajas-Solano, D., Tartakovsky, G. & Tartakovsky, A. M. Physics-informed neural networks for multiphysics data assimilation with application to subsurface transport. *Advances in Water Resources* **141**, 103610 (2020).
142. Kraft, B., Jung, M., Körner, M. & Reichstein, M. Hybrid modeling: Fusion of a deep learning approach and a physics-based model for global hydrological modeling. in *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* vol. XLIII-B2-2020 1537–1544 (Copernicus GmbH, 2020).
143. Kraft, B., Jung, M., Körner, M., Koirala, S. & Reichstein, M. Towards hybrid modeling of the global hydrological cycle. *Hydrology and Earth System Sciences* **26**, 1579–1614 (2022).

144. Liu, J., Rahmani, F., Lawson, K. & Shen, C. A multiscale deep learning model for soil moisture integrating satellite and in situ data. *Geophysical Research Letters* **49**, e2021GL096847 (2022).
145. Hochreiter, S. The vanishing gradient problem during learning recurrent neural nets and problem solutions. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* **06**, 107–116 (1998).
146. Hochreiter, S., Bengio, Y., Frasconi, P., & Jürgen Schmidhuber. Gradient Flow in Recurrent Nets: The Difficulty of Learning Long-Term Dependencies. in *A Field Guide to Dynamical Recurrent Neural Networks* (eds. Kremer, S. C. & Kolen, J. F.) 237–244 (IEEE Press, 2001).
147. Basodi, S., Ji, C., Zhang, H. & Pan, Y. Gradient amplification: An efficient way to train deep neural networks. *Big Data Mining and Analytics* **3**, 196–207 (2020).
148. Hochreiter, J. Untersuchungen zu dynamischen neuronalen Netzen. (Institut f. Informatik, Technische Univ. Munich, 1991).
149. Kochkov, D. *et al.* Machine learning–accelerated computational fluid dynamics. *Proceedings of the National Academy of Sciences* **118**, e2101784118 (2021).
150. Fang, K., Kifer, D., Lawson, K. & Shen, C. Evaluating the potential and challenges of an uncertainty quantification method for long short-term memory models for soil moisture predictions. *Water Resources Research* **56**, e2020WR028095 (2020).
151. Li, D., Marshall, L., Liang, Z., Sharma, A. & Zhou, Y. Bayesian LSTM with stochastic variational inference for estimating model uncertainty in process-based hydrological models. *Water Resources Research* **57**, e2021WR029772 (2021).
152. Tabas, S. S. & Samadi, S. Variational Bayesian dropout with a Gaussian prior for recurrent neural networks application in rainfall–runoff modeling. *Environ. Res. Lett.* **17**, 065012 (2022).
153. Krapu, C. & Borsuk, M. A Differentiable Hydrology Approach for Modeling With Time-Varying Parameters. *Water Resources Research* **58**, e2021WR031377 (2022).
154. Wang, N., Zhang, D., Chang, H. & Li, H. Deep learning of subsurface flow via theory-guided neural network. *Journal of Hydrology* **584**, 124700 (2020).

155. Karniadakis, G. E. *et al.* Physics-informed machine learning. *Nat Rev Phys* **3**, 422–440 (2021).

156. Fleming, S. W., Garen, D. C., Goodbody, A. G., McCarthy, C. S. & Landers, L. C. Assessing the new Natural Resources Conservation Service water supply forecast model for the American West: A challenging test of explainable, automated, ensemble artificial intelligence. *Journal of Hydrology* **602**, 126782 (2021).

157. Li, L. *et al.* Developing machine learning models with multi-source environmental data to predict wheat yield in China. *Comput. Electron. Agric.* **194**, (2022).

158. Paudel, D. *et al.* Machine learning for regional crop yield forecasting in Europe. *Field Crops Research* **276**, 108377 (2022).

159. Shahhosseini, M., Hu, G., Huber, I. & Archontoulis, S. V. Coupling machine learning and crop modeling improves crop yield prediction in the US Corn Belt. *Sci Rep* **11**, 1606 (2021).

160. Chen, S., Zwart, J. A. & Jia, X. Physics-guided graph meta learning for predicting water temperature and streamflow in stream networks. in *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* 2752–2761 (Association for Computing Machinery, 2022). doi:10.1145/3534678.3539115.

161. Rahmani, F. *et al.* Data Release: Deep learning approaches for improving prediction of daily stream temperature in data-scarce, unmonitored, and dammed basins: U.S. Geological Survey data release. *U.S. Geological Survey* https://doi.org/10.5066/P9VHMO56 (2021).

162. Daraio, J. A., Bales, J. D. & Pandolfo, T. J. Effects of land use and climate change on stream temperature II: Threshold exceedance duration projections for freshwater mussels. *JAWRA Journal of the American Water Resources Association* **50**, 1177–1190 (2014).

163. van Vliet, M. T. H. *et al.* Coupled daily streamflow and water temperature modelling in large river basins. *Hydrol. Earth Syst. Sci.* **16**, 4303–4321 (2012).

164. He, X. *et al.* Improving predictions of evapotranspiration by integrating multi-source observations and land surface model. *Agricultural Water Management* **272**, 107827 (2022).

165. Talib, A. *et al.* Evaluation of prediction and forecasting models for evapotranspiration of agricultural lands in the Midwest U.S. *Journal of Hydrology* **600**, 126579 (2021).
166. Seibert, J., Vis, M. J. P., Lewis, E. & Meerveld, H. J. van. Upper and lower benchmarks in hydrological modelling. *Hydrological Processes* **32**, 1120–1125 (2018).
167. Mohamoud, Y. M. & Parmar, R. S. Estimating Streamflow and Associated Hydraulic Geometry, the Mid-Atlantic Region, USA1. *JAWRA Journal of the American Water Resources Association* **42**, 755–768 (2006).
168. Merritt, A. M., Lane, B. & Hawkins, C. P. Classification and Prediction of Natural Streamflow Regimes in Arid Regions of the USA. *Water* **13**, (2021).
169. Stefan, H. G. & Fang, X. Dissolved oxygen model for regional lake analysis. *Ecological Modelling* **71**, 37–68 (1994).
170. Heddam, S. Simultaneous modelling and forecasting of hourly dissolved oxygen concentration (DO) using radial basis function neural network (RBFNN) based approach: a case study from the Klamath River, Oregon, USA. *Modeling Earth Systems and Environment* **2**, 135 (2016).
171. Keshtegar, B. & Heddam, S. Modeling daily dissolved oxygen concentration using modified response surface method and artificial neural network: a comparative study. *Neural Computing and Applications* **30**, 2995–3006 (2018).
172. Chen, R. T. Q., Rubanova, Y., Bettencourt, J. & Duvenaud, D. Neural Ordinary Differential Equations. Preprint at https://doi.org/10.48550/arXiv.1806.07366 (2019).
173. Haber, E. & Ruthotto, L. Stable Architectures for Deep Neural Networks. *Inverse Problems* **34**, 014004 (2018).
174. Shen, C. Deep learning: A next-generation big-data approach for hydrology. *Eos* vol. 99 (2018).
175. Karpatne, A. *et al.* Theory-guided data science: A new paradigm for scientific discovery from data. *IEEE Transactions on Knowledge and Data Engineering* **29**, 2318–2331 (2017).
176. Khandelwal, A. *et al.* Physics Guided Machine Learning Methods for Hydrology. Preprint at https://doi.org/10.48550/arXiv.2012.02854 (2020).

177. Pawar, S., San, O., Aksoylu, B., Rasheed, A. & Kvamsdal, T. Physics guided machine learning using simplified theories. *Physics of Fluids* **33**, 011701 (2021).
178. Bennett, A. & Nijssen, B. Deep learned process parameterizations provide better representations of turbulent heat fluxes in hydrologic models. *Water Resources Research* **57**, e2020WR029328 (2021).
179. Schaap, M. G., Leij, F. J. & van Genuchten, M. Th. Rosetta: a Computer Program for Estimating Soil Hydraulic Parameters With Hierarchical Pedotransfer Functions. *Journal of Hydrology* **251**, 163–176 (2001).
180. Rasp, S., Pritchard, M. S. & Gentine, P. Deep learning to represent subgrid processes in climate models. *Proceedings of the National Academy of Sciences of the United States of America* **115**, 9684–9689 (2018).
181. Zhu, Y. *et al.* Physics-informed deep-learning parameterization of ocean vertical mixing improves climate simulations. *National Science Review* **9**, nwac044 (2022).
182. Koppa, A., Rains, D., Hulsman, P., Poyatos, R. & Miralles, D. G. A deep learning-based hybrid model of global terrestrial evaporation. *Nat Commun* **13**, 1912 (2022).
183. Liu, B. *et al.* Physics-Guided Long Short-Term Memory Network for Streamflow and Flood Simulations in the Lancang–Mekong River Basin. *Water* **14**, 1429 (2022).
184. Frame, J. M. *et al.* Post-Processing the National Water Model with Long Short-Term Memory Networks for Streamflow Predictions and Model Diagnostics. *JAWRA Journal of the American Water Resources Association* **57**, 885–905 (2021).
185. Sun, A. Y., Jiang, P., Yang, Z.-L., Xie, Y. & Chen, X. A graph neural network approach to basin-scale river network learning: The role of physics-based connectivity and data fusion. *Hydrology and Earth System Sciences Discussions* (2022) doi:10.5194/hess-2022-111.
186. Reichstein, M. *et al.* Deep learning and process understanding for data-driven Earth system science. *Nature* **566**, 195–204 (2019).

Supplementary Information

S1. Details for some examples.

Example 1. Part of the effort in Tsai et al.[99], which proposed differentiable parameter learning (dPL), connected the Variable Infiltration Capacity (VIC) process-based hydrologic model to a neural network (g) that estimates physical parameters of VIC ($\boldsymbol{\theta}$) using some widely available attributes ($\boldsymbol{A}$): $\boldsymbol{\theta} = g(\boldsymbol{A})$. In an "end-to-end" workflow, $\boldsymbol{\theta}$ is then sent to VIC, whose outputs are compared with observations, effectively turning the parameter calibration problem into a machine learning problem, trained on all sites simultaneously using backpropagation and gradient descent (Figure S1a). As a result of this global loss function, dPL exhibits advantages over traditional calibration on multiple fronts, for three different datasets (soil moisture, CAMELS streamflow, and global headwater runoff). The parameter sets are spatially coherent (Figure S1b-c) and extrapolate better in space (Figure S1d-e). dPL is hyper efficient: a job that normally takes a 100-CPU cluster 2-3 days now takes a single Graphical Processing Unit (GPU) one hour. dPL allows the combined model to output unobserved variables while addressing the notorious problem of parameter equifinality[118]. As summarized earlier, these are the great advantages we expect to harness with differentiable modeling.

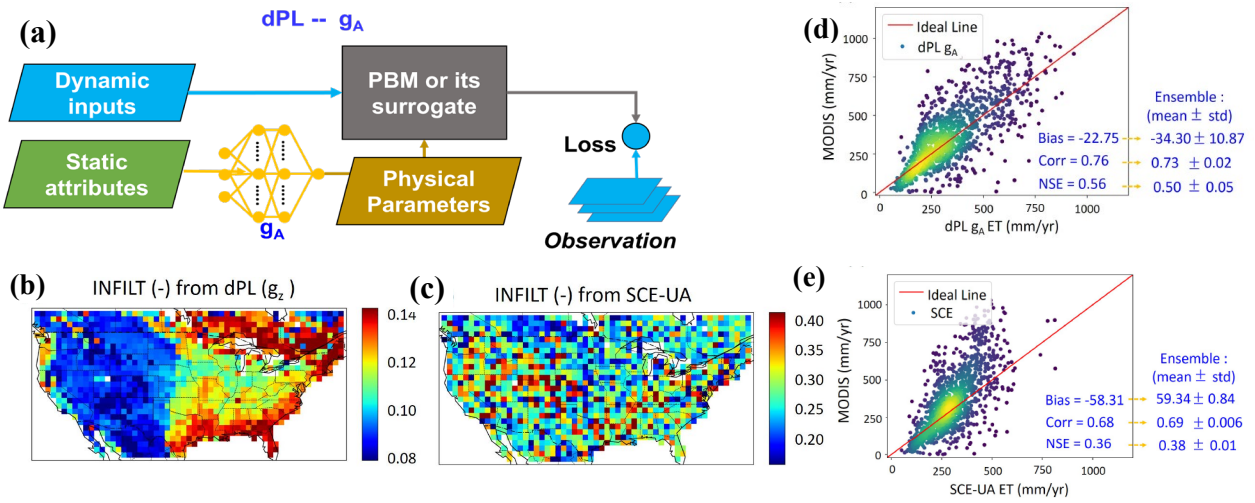

Figure S1. (From Tsai et al.[99], reprint allowed via Creative Commons Attribution 4.0 International License, http://creativecommons.org/licenses/by/4.0/) (a) Structural diagram of one of the dPL frameworks called g_A; (b & c) The estimated infiltrating curve parameter (INFILT) from dPL vs. the site-by-site calibrated shuffled complex evolutionary algorithm (SCE-UA);(d & e) dPL better matches the MODIS satellite product for uncalibrated variable ET than does SCE-UA.

Example 2. Physics-informed neural networks (PINNs)[100,141], while first published in 2017 before the existence of the term "differentiable geosciences", could be perceived as a genre of DG as the gradient information is critically employed. PINNs pose problems in a unique way, seeking to train a neural network with space-time coordinates as inputs, *h(t,x)* where *x* represents spatial coordinates and *t* is time such that (i) *h(t,x)* agrees with known data points at *(t,x)*, and (ii) the derivatives *dh/dx*, *dh/dt,* etc. agree with the governing partial differential equations. Physical parameters could also be part of the inputs to the *h* network[154]. PINNs are a highly innovative approach tested on a large variety of applications in many domains, and there have been a number of good reviews of this work[131,155]. PINNs have made enormous

strides, with novel inversion uses such data assimilation[141] and learning governing equations, but, as with other methods, there are also some limitations. Obviously, the function *h(t,x)* is tied to the initial and boundary conditions so it needs to be trained separately for different initial/boundary condition pairs, and the form of the inputs limits the neural network to certain types (multilayer perceptron network) that are not the easiest to train. However, the learned parameters and constitutive relationships can describe the system under a wide range of boundary and initial conditions. Furthermore, the fidelity of the trained network to physical equations must be carefully examined.

In geosciences, a PINN method for learning unknown parameter fields and constitutive relationships was proposed[103] (Figure S2). As an example, steady-state groundwater flow in an aquifer with an unknown conductivity field and unsaturated flow in the vadose zone with an unknown pressure-dependent conductivity were considered. In the unsaturated flow application, it was assumed that only sparse measurements of pressure head were available. The quantities of interest were the unsaturated conductivity as a function of the pressure head, and the pressure head field. Notably, it was assumed that no measurements of the unknown parameters were available. In the proposed PINN method, both quantities of interest were represented with neural networks (NNs) (with unknown parameters). This step created a differentiable model of the unsaturated flow in the vadose zone. It was also assumed that the pressure head measurements could be described by the steady-state Richards equation. Substituting the NN approximations into this equation formed the axillary residual NN, which shared the (unknown) parameters with the primary NNs. For the primary NNs to satisfy the governing equation, the residual NN should be zero everywhere in the domain – in other words, the exact measurements of the residuals are available everywhere in the domain. The NNs were trained jointly using the pressure head measurements. Since the conductivity and residual NNs share the same parameters, estimating parameters in the residual NN also provides the parameterization of the conductivity NN. Figure 5a shows the reference pressure head field and the locations of the measurements. Figure 5b shows the point errors in the estimated pressure head field. The reference and estimated unsaturated conductivity functions are shown in Figure 5c. These figures demonstrate that the PINN method can learn both the state variable and the constitutive relationship very accurately.

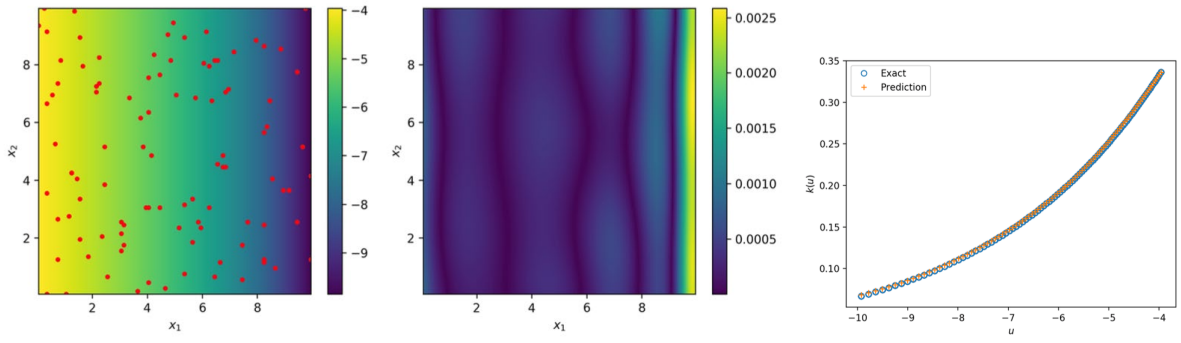

Figure S2. (from Tartakovsky et al.[103], reprint permission obtained) (a) The reference pressure head field and the locations of the measurements. (b) The point errors in the estimated head field. (c) The reference and estimated conductivities as functions of the pressure head.

S2. Supplementary Discussion

A. Recent progresses in geoscientific domains from purely data-driven machine learning.

ML has gradually but pervasively permeated the vast majority of scientific disciplines, and is transforming those sciences at an unprecedented pace. In hydrology, deep networks such as long short-term memory (LSTM) networks[69], and convolutional neural networks (CNNs)[70,71] have shown strong ability with regard to prediction of soil moisture[57–59], water supply[156], streamflow[60–63], evapotranspiration[64–66], groundwater levels[67], snow[68], and other aspects of the water cycle[56]. In water quality studies, LSTMs and CNNs have shown promise in simulating water temperature[48–51], dissolved oxygen[52], phosphorous[53], and nitrogen[54,55], among others[46,47]. In agriculture, ML approaches have been widely applied for crop production prediction[157–159]. In regional climate studies, CNN-based schemes or generative algorithms have been found to improve the forecasting of precipitation fields[43,44] and prediction of clouds (deep clouds)[45]. Often the studies have reported state-of-the-art performance when compared with conventional approaches. Typically, such high-quality predictions can be made even when a good understanding of the underlying processes is not available. We made an effort to collect a list of somewhat comparable studies with metrics for both traditional and ML models (Figure S3 and Table S1). Previous models have been highly useful in advancing science, but these results imply that they were not fully exploiting the information available in the data[26], and they can benefit from leveraging the strength of ML.

Table S1. *ML vs. traditional model performances for a number of scientific applications with data from many sites. The metrics were computed based on simulations and observations. The lower the values, the better for RMSE, while higher is better for Pearson's correlation (COR), R^2, and Nash-Sutcliffe model efficiency coefficient (NSE). This is presented with many caveats, such as the ML model is optimized to match observations while traditional models have many other constraints; a selection bias – where ML did not outperform did not get published (nevertheless, one could also argue studies where PBM outperformed were not easily found). The point of this table was not to show that ML was always better, but to support the argument that ML tends to have advantages in accuracy. Also note the limitations of ML we discussed in the Introduction.*

Variable	**Metric**	**Deep networks**	**Traditional**	**Reference**
Stream Temperature	RMSE (°C)	1.91	4.01	Chen et al.[160]
	RMSE (°C)	0.89	1.80	Rahmani et al.[161] and Daraio et al.[162]
	Pearson COR	0.99	0.91	Rahmani et al.[161] and van Vliet et al.[163]
	R^2	0.942	0.93	Rahmani et al.[161]
	NSE	0.98	0.93	Rahmani et al.[161]
Evapotranspiration	R^2	0.67	0.21	He et al.[164]
	RMSE (mm/day)	1.21	2.56	
	NSE	0.65	0.57	Talib et al.[165]
Soil Moisture	RMSE	0.027	0.085	Fang et al.[57]
	Pearson COR	0.87	0.72	
	RMSE	0.027	0.035	
	Pearson COR	0.87	0.82	
	Pearson COR	0.91	0.77	Liu et al.[144]
	RMSE	0.034	0.08	
Streamflow	NSE	0.76	0.68	Seibert et al.[166] and Kratzert et al.[61]
	NSE	0.9 /0.68	-	Mohamoud and Parmar[167]
	Mean R^2	0.71	-	Merritt et al.[168]
	NSE	0.78	-	Zhi et al.[168]
Dissolved oxygen	Median R^2	-	0.64	Stefan and Fang[169]
	CC (correlation Coefficient)	0.972	-	Heddam[170]
	Median NSE	0.760	-	Keshtegar and Heddam[171]

B. *Why can differentiable process-based models achieve state-of-the-art predictive performance?*

Purely data-driven ML architectures have set a high bar for accuracy in multiple geoscience domains, such that one would be tempted to predict a loss in accuracy when adding in less-flexible process-based components. However, here it is argued that generic ML architectures are not necessary to achieve good model accuracy. As long as some model components are adaptable and learnable, we can learn from data. If we view the model as a more strongly constrained ML model (perspective "a" in Figure 2), it is easy to see that there is a potential to achieve ML-level performance if we enlarge the searchable space of PBM to include a good approximation of the true function, directed by gradient-based training. The paths we take to upgrade the models will be expert-dependent (prior-dependent), so one should not expect a unified approach at present.

Many dynamical systems in Geosciences can be written as ordinary differential equations (ODEs), e.g., rainfall runoff in a basin, crop growth, or nutrient release. While solving these equations, we run the numerical model for many steps. This is mathematically similar to recurrent neural networks, and the time integration operation is similar to the functionality achieved by some neural networks like the Residual Networks[172,173]. It should not be surprising that learnable process-based models with some ML components can perform as well as deep networks.

As we discuss in Section S1, multiple studies have already shown that differentiable, learnable models can approach the performance of purely data-driven models, or exhibit advantages in some cases where extrapolation is key. Differentiable model formulations can maintain at least two of the three desirable features: approximating complex, previously unknown functions, and the ability to assimilate information from big data. Compared to purely data-driven ML, DG trades genericity for interpretability and the ability to ask specific questions. Deep networks like CNNs, LSTMs, and transformers will be an ingrained part of DG. Eventually, deep learning will become part of the repertoire of geoscientists, just like with numerical methods[174].

C. *How is DG related to physics-guided machine learning (PGML) and how are they different?*

Many ML-physics integration strategies with a wide variety of complexity have been proposed in the past in a seemingly scattered manner, such that a clear classification is difficult[155]. It has not been sufficiently recognized that some of these algorithms work fundamentally because they leveraged the differentiable programming tools. The scattered nature of those publications makes the landscape of ML-physics integration daunting and confusing, while hindering us from making innovations based on first principles. However, once we treat differentiability as the central tool, it serves as a compass to guide us in understanding newly proposed methods, i.e., we can ask if a method is fully (end-to-end) differentiable, how it uses gradients, how much prior information is inserted, what questions are asked, and how it scales with data. Here we outline some similarities and differences between DG and some existing methods.

DG and physics-guided (or physics-informed, theory-guided, or knowledge-guided) machine learning (PGML)[175–177] both seek to combine physics with ML, but they differ in their approaches, purposes, and philosophies. Many PGML studies seek to introduce physical constraints, e.g., as regularization or pre-training, to ML methods to gain better generalizability with less training data. PGML does not in theory need differentiable programming and partial physics could be enforced. In contrast, DG is more thorough in that it uses the numerical physical model as the backbone and demands that the entire workflow be differentiable. In terms of purposes, PGML is tasked to make the ML model more robust, while differentiable modeling seeks to update our assumptions or discover new knowledge. Relatedly, in terms of philosophies, when a physical law was introduced in PGML, it was treated as truth (albeit sometimes with some tolerance level, as in Read et al.[51]). Often, this includes all the calculations and assumptions to

support the law. In DG, we do not presume the physical laws to be correct, and, rather, are constantly looking for opportunities to update existing knowledge.

There are many not-fully-differentiable methods that could be valuable for various applications but are outside of the scope of DG for this paper. For one, it is possible to incorporate ML algorithms trained offline on datasets as part of a physical model, e.g., training a neural network on turbulent heat fluxes and inserting into a hydrologic model[178]; training pedotransfer functions to infer soil parameters from soil hydraulic data[179]; training an atmospheric parameterization network on short-term cloud-resolving simulations[180]; or training ocean-mixing parameterizations on data and physical constraints[181]. While this approach has the advantage that the physical meaning of the NN is clear and stands alone, direct training data are needed for the variable of interest (thus having issues with pure ML as discussed earlier) and the network can no longer evolve and adapt in an interactive fashion, for instance to further update the model when exposed to observations. In the future these NNs could be further incorporated into DG models. Some other offline coupling methods include providing outputs of process-based models as inputs to neural networks (this helps to integrate over spatiotemporal heterogeneity)[182,183], or training ML models to predict the PBM residuals[151,184,185]. Readers are referred to Reichstein et al.[186] which promoted a number of ways to connect physics and ML for geosciences, with a brief mention of differentiable programming.