Ten Ways to Apply Machine Learning in Earth and Space Sciences

Machine learning is gaining popularity across scientific and technical fields, but it's often not clear to researchers, especially young scientists, how they can apply these methods in their work.

The Earth and space sciences are poised for a revolution centered around the application of existing and rapidly emerging machine learning techniques to large and complex data sets. This image was adapted from a NASA illustration using artificial intelligence tools from Deep Dream Generator. Credit: Enrico Camporeale

By Jacob Bortnik and Enrico Camporeale 29 June 2021

The Earth and space sciences present ideal use cases for machine learning (ML) applications because the problems being addressed are globally important and the data are often freely available, voluminous, and of high quality.

Machine learning (ML), loosely defined (https://ieeexplore.ieee.org/abstract/document/5392560) as the "ability of computers to learn from data without being explicitly programmed," has become tremendously popular in technical disciplines over the past decade or so, with applications including complex game playing (https://www.theverge.com/2019/11/27/20985260/ai-go-alphago-lee-se-dol-retired-deepmind-defeat) and image recognition (https://openaccess.thecvf.com/content_iccv_2015/html/He_Delving_Deep_into_ICCV_2015_paper.html) carried out with superhuman capabilities. The Earth and space sciences (ESS) community has also increasingly adopted ML approaches to help tackle pressing questions and unwieldy data sets. From 2009 to 2019, for example, the number of studies involving ML published in AGU journals approximately doubled (https://eos.org/science-updates/advancing-ai-for-earth-science-a-data-systems-perspective).

In many ways, ESS present ideal use cases for ML applications because the problems being addressed —like climate change (https://eos.org/articles/teaching-machines-to-detect-climate-extremes), weather forecasting (https://eos.org/opinions/weathering-environmental-change-through-advances-in-ai), and natural hazards assessment (https://www.nature.com/articles/s41598-020-69233-2)—are globally important; the data are often freely available, voluminous, and of high quality; and computational resources required to develop ML models are steadily becoming more affordable. Free computational languages and ML code libraries are also now available (e.g., scikit-learn, PyTorch, and TensorFlow), contributing to making entry barriers lower than ever. Nevertheless, our experience has been that many young scientists and students interested in applying ML techniques to ESS data do not have a clear sense of how to do so.

The Tools of the Trade

An ML algorithm can be thought of broadly as a mathematical function containing many free parameters (thousands or even millions) that takes inputs (features) and maps those features into one or more outputs (targets). The process of "training" an ML algorithm involves optimizing the free parameters to map the features to the targets accurately.

There are two broad categories of ML algorithms relevant in most ESS applications: supervised and unsupervised learning (a third category, reinforcement learning, is used infrequently in ESS). Supervised learning, which involves presenting an ML algorithm with many examples of input-output pairs (called the "training set"), can be further divided, according to the type of target that is being learned, as either categorical (classification; e.g., does a given image show a star cluster or not?) or continuous (regression; e.g., what is the temperature at a given location on Earth?). In unsupervised learning, algorithms are not given a particular target to predict; rather, an algorithm's task is to learn the natural structure in a data set without being told what that structure is.

Supervised learning is more commonly used in ESS, although it has the disadvantage that it requires labeled data sets (in which each training input sample must be tagged, or labeled, with a corresponding output target), which are not always available. Unsupervised learning, on the other

hand, may find multiple structures in a data set, which can reveal unanticipated patterns and relationships, but it may not always be clear which structures or patterns are “correct” (i.e., which represent genuine physical phenomena).

Applications in Earth and Space Sciences

Books and classes about ML often present a range of algorithms but leave people to imagine specific applications of these algorithms on their own.

Books and classes about ML often present a range of algorithms that fall into one of the above categories but leave people to imagine specific applications of these algorithms on their own. However, in practice, it is usually not obvious how such approaches (some seemingly simple) may be applied in a rich variety of ways, which can create an imposing obstacle for scientists new to ML. Below we briefly describe various themes and ways in which ML is currently applied to ESS data sets (Figure 1), with the hope that this list—necessarily incomplete and biased by our personal experience—inspires readers to apply ML in their research and catalyzes new and creative use cases.

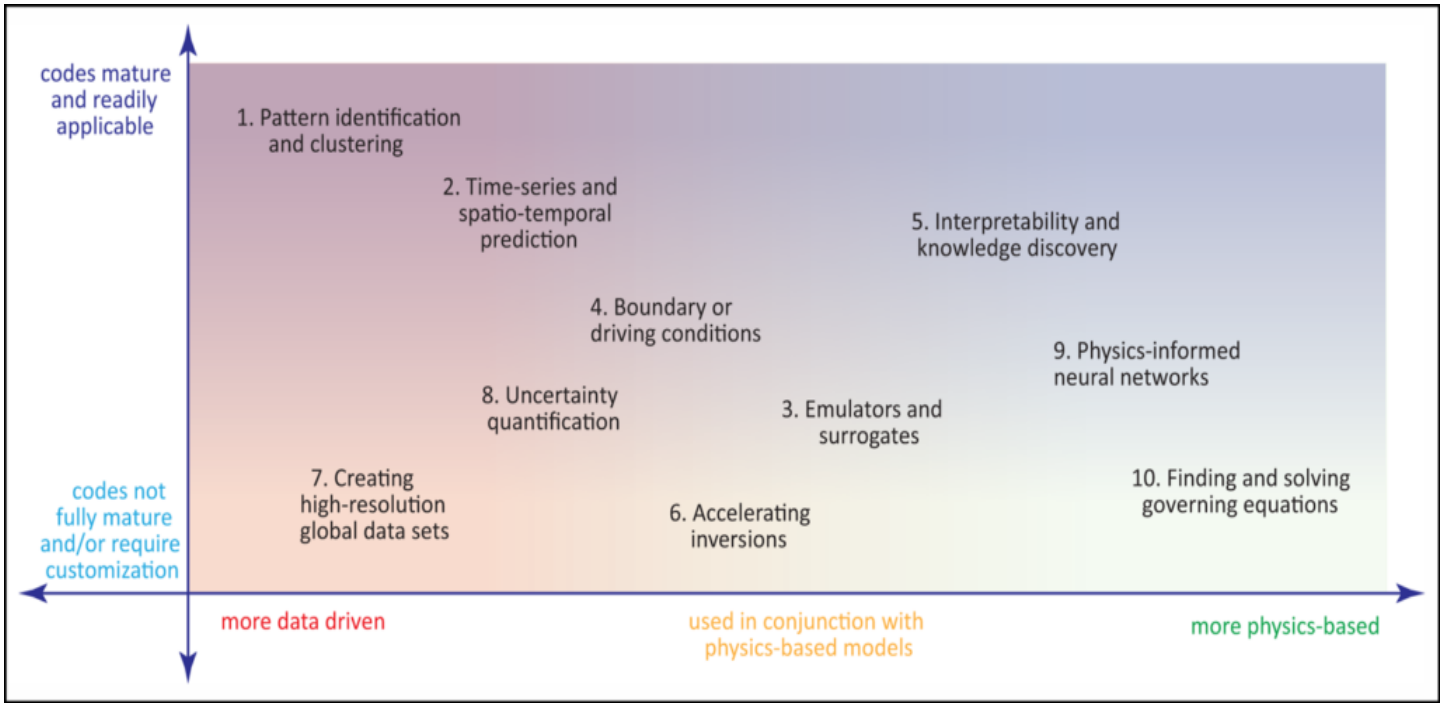

Fig. 1. Ten ideas for applying machine learning (ML) in the Earth and space sciences, roughly organized by the degree of involvement of physics-based models (horizontal scale) and the degree to which ML codes are available and readily applicable versus being in development and requiring significant customization (vertical scale). Credit: Jacob Bortnik

1. Pattern Identification and Clustering

One of the simplest and most powerful applications of ML algorithms is pattern identification, which works particularly well with very large data sets that cannot be traversed manually and in which signals of interest are faint or highly dimensional. Researchers, for example, applied ML in this way to detect signatures of Earth-sized exoplanets (https://academic.oup.com/mnras/article/474/1/478/4564439) in noisy data making up millions of light curves observed by the Kepler space telescope. Detected signals

can be further split into groups through clustering (https://link.springer.com/article/10.1007/s40745-015-0040-1), an unsupervised form of ML, to identify natural structure in a data set.

Conversely, atypical signals may be teased out of data by first identifying and excluding typical signals, a process called anomaly or outlier detection. This technique is useful, for example, in searching for signatures of new physics (https://iopscience.iop.org/article/10.1088/1742-6596/368/1/012032/meta) in particle collider experiments.

2. Time Series and Spatiotemporal Prediction

An important and widespread application of supervised ML is the prediction of time series data from instruments or from an index (https://agupubs.onlinelibrary.wiley.com/doi/full/10.1029/2018SW001898) (or average value) that is intended to encapsulate the behavior of a large-scale system. Approaches to this application often involve using past data in the time series itself to predict future values; they also commonly involve additional inputs that act as drivers of the quantities measured in the time series. A typical example of ML applied to time series in ESS is its use in local weather prediction (https://keras.io/examples/timeseries/timeseries_weather_forecasting/), with which trends in observed air temperature and pressure data, along with other quantities, can be predicted.

In many instances, however, predicting a single time series of data is insufficient, and knowledge of the temporal evolution of a physical system over regional (or global) spatial scales is required. This spatiotemporal approach is used, for example, in attempts to predict weather (https://eos.org/research-spotlights/boosting-weather-prediction-with-machine-learning) across the entire globe as a function of time and 3D space in high-capacity models such as deep neural networks.

3. Emulators and Surrogates

Physics-based simulations can take days or weeks to run on even the most powerful computers. An alternate solution is to train ML models to act as emulators for physics-based models.
Traditional, physics-based simulations (e.g., global climate models) are often used to model complex systems, but such models can take days or weeks to run on even the most powerful computers, limiting their utility in practice. An alternate solution is to train ML models to act as emulators for physics-based models or to replicate computationally intensive portions within such models. For example, global climate models that run on a coarse grid (e.g., 50- to 100-kilometer resolution) can include subgrid processes (https://www.nature.com/articles/s41467-020-17142-3), like convection, modeled using ML-based parameterizations. Results with these approaches are often indistinguishable from those produced by the original model alone but can run millions or billions (https://www.sciencemag.org/news/2020/02/models-galaxies-atoms-simple-ai-shortcuts-speed-simulations-billions-times) of times faster.

4. Boundary or Driving Conditions

Many physics-based simulations proceed by integrating a set of partial differential equations (PDEs) that rely on time-varying boundary conditions and other conditions that drive interior parts of the simulation. The physics-based model then propagates information from these boundary and driver conditions into the simulation space—imagine, for example, a 3D cube being heated at its boundary faces with time-varying heating rates or with thermal conductivity that varies spatiotemporally within the cube. ML models can be trained to reflect the time-varying parameterizations both within and along the simulation boundaries of a physical model, which again may be computationally cheaper and faster.

5. Interpretability and Knowledge Discovery

If a spatiotemporal ML model of a physical system can be trained to produce accurate results under a variety of input conditions, then the implication is that the model implicitly accounts for all the physical processes that drive that system, and thus, it can be probed to gain insights into how the system works. Certain algorithms (e.g., random forests) can automatically provide a ranking of "feature importance," giving the user a sense of which input parameters affect the output most and hence an intuition about how the system works.

More sophisticated techniques, such as layerwise relevance propagation, can provide deeper insights into how different features interact to produce a given output (https://agupubs.onlinelibrary.wiley.com/doi/full/10.1029/2019MS002002) at a particular location and time. For example, a neural network trained to predict the evolution of the El Niño–Southern Oscillation (ENSO), which is predominantly associated with changes in sea surface temperature in the equatorial Pacific Ocean, revealed that precursor conditions for ENSO events occur in the South Pacific and Indian Oceans.

6. Accelerating Inversions

A ubiquitous challenge in ESS is to invert observations of a physical entity or process into fundamental information about the entity or the causes of the process (e.g., interpreting seismic data to determine rock properties). Historically, inverse problems are solved in a Bayesian framework requiring multiple runs of a forward model, which can be computationally expensive and often inaccurate. ML offers alternative methods to approach inverse problems, either by using emulators to speed up forward models or by using physics-informed machine learning to discover hidden physical quantities directly. ML models trained on prerun physics-based model outputs (https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2019JA027135) can be used for rapid inversion.

7. Creating High-Resolution Global Data Sets

Satellite observations often provide global, albeit low-resolution and sometimes indirect (i.e., proxy-based), measurements of quantities of interest, whereas local measurements provide more accurate

and direct observations of those quantities at smaller scales. A popular and powerful use for ML models is to estimate the relationship between global proxy satellite observations and local accurate observations, which enables the creation of estimated global observations (https://eos.org/articles/aquasat-gives-water-quality-researchers-new-eyes-in-the-sky) on the basis of localized measurements. This approach often includes the use of ML to create superresolution images (https://iopscience.iop.org/article/10.3847/2041-8213/ab9d79/meta) and other data products.

8. Uncertainty Quantification

Typically, uncertainty in model outputs is quantified using a single metric such as the root-mean-square of the residual (the difference between model predictions and observations). ML models can be trained to explicitly predict the confidence interval (https://agupubs.onlinelibrary.wiley.com/doi/full/10.1029/2018SW002026), or inherent uncertainty, of this residual value, which not only serves to indicate conditions under which model predictions are trustworthy (or dubious) but can also be used to generate insights about model performance. For instance, if there is a large error at a certain location in a model output under specific conditions, it could suggest that a particular physical process is not being properly represented in the simulation.

9. Physics-Informed Neural Networks

Domain experts analyzing data from a given system, even in relatively small quantities, are often able to extrapolate the behavior of the system—at least conceptually—because of their understanding of and trained intuition about the system based on physical principles. In a similar way, laws and relationships that govern physical processes and conserved quantities can be explicitly encoded into neural network algorithms (https://www.sciencedirect.com/science/article/pii/S0021999118307125), resulting in more accurate and physically meaningful models that require less training data.

10. Finding and Solving Governing Equations

In certain applications, the values of terms or coefficients in PDEs that drive a system—and thus that should be represented in a model—are not known. Various ML algorithms were developed recently that automatically determine (https://www.pnas.org/content/113/15/3932) PDEs that are consistent with the available physical observations (https://science.sciencemag.org/content/324/5923/81), affording a new and powerful discovery tool.

In still newer work, ML methods are being developed to directly solve PDEs (https://www.technologyreview.com/2020/10/30/1011435/ai-fourier-neural-network-cracks-navier-stokes-and-partial-differential-equations/). These methods offer accuracy comparable to traditional numerical integrators but can be dramatically faster, potentially allowing large-scale simulations of complex sets of PDEs that have otherwise been unattainable.

Addressing Urgent Challenges

The Earth and space sciences are poised for a revolution centered around the application of existing and rapidly emerging ML techniques to large and complex ESS data sets being collected. These techniques have great potential to help scientists address some of the most urgent challenges and questions about the natural world facing us today. We hope the above list sparks creative and valuable new applications of ML, particularly among students and young scientists, and that it becomes a community resource to which the ESS community can add more ideas.

Acknowledgments

We thank the AGU Nonlinear Geophysics section for promoting interdisciplinary, data-driven research, for supporting the idea of writing this article, and for suggesting *Eos* as the ideal venue for dissemination. The authors gratefully acknowledge the following sources of support: J.B. from subgrant 1559841 to the University of California, Los Angeles, from the University of Colorado Boulder under NASA Prime Grant agreement 80NSSC20K1580, the Defense Advanced Research Projects Agency under U.S. Department of the Interior award D19AC00009, and NASA/SWO2R grant 80NSSC19K0239 and E.C. from NASA grants 80NSSC20K1580 and 80NSSC20K1275. Some of the ideas discussed in this paper originated during the 2019 Machine Learning in Heliophysics (https://ml-helio.github.io/2019/index_2019.html) conference.

Author Information

Jacob Bortnik (jbortnik@gmail.com (mailto:jbortnik@gmail.com)), University of California, Los Angeles; and Enrico Camporeale, Space Weather Prediction Center, NOAA, Boulder, Colo.; also at Cooperative Institute for Research in Environmental Sciences, University of Colorado Boulder

Citation: Bortnik, J., and E. Camporeale (2021), Ten ways to apply machine learning in Earth and space sciences, *Eos, 102,* https://doi.org/10.1029/2021EO160257. Published on 29 June 2021.

This article does not represent the opinion of AGU, *Eos,* or any of its affiliates. It is solely the opinion of the author.
