Launching ClimateLearn

ClimateLearn is a Python library for accessing state-of-the-art climate data and machine learning models in a standardized, straightforward way.

AUTHORS
Jason Jewik

AFFILIATIONS
UCLA

PUBLISHED
Jan. 9, 2023

Contents

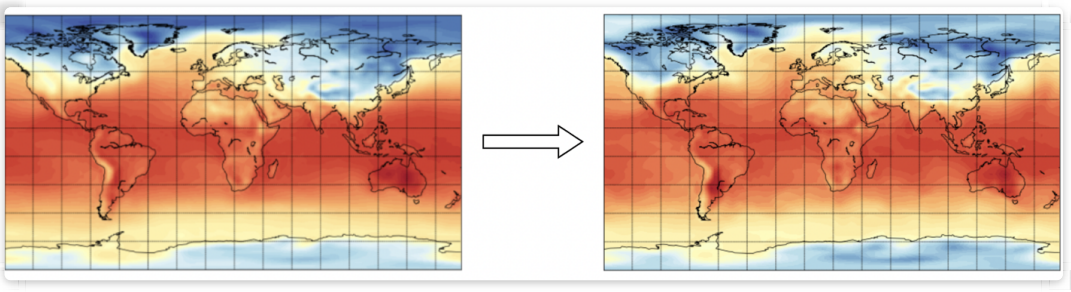
Photo by Mario von Rotz on Unsplash.

Background

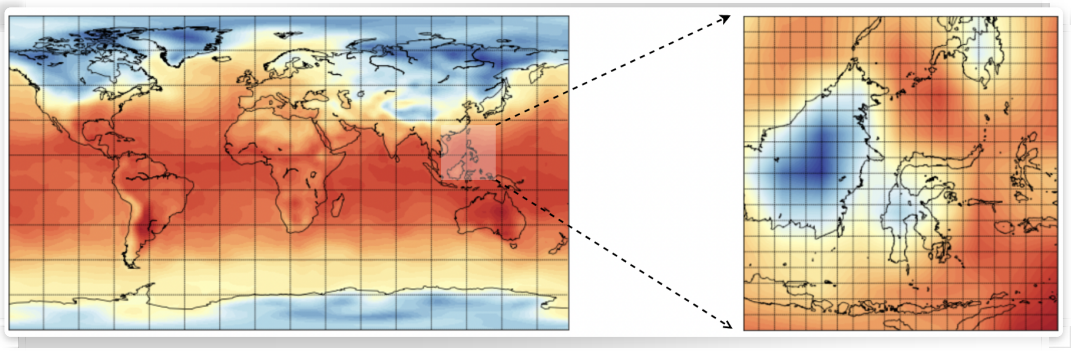
In recent years, extreme weather events have made more apparent the threat of climate change. Atlantic hurricanes slamming the eastern United States have been increasing in intensity and severity. Torrential downpours submerged much of Pakistan underwater, killing thousands of people. Unprecedented heat waves sparked wildfires that tore through swaths of Portugal and Spain. Severe droughts in the Middle East and northern Africa have devastated the region's water supplies, stirring conflicts. Depending on the response of the international

community over the next decade, Earth's average surface temperature is expected to rise anywhere from 2°C to 4°C by 2100 [1]. With this increase in temperature, climate scientists predict that extreme weather events will become much more common.

The tools used by climate scientists to make predictions of future weather and climate are called **general circulation models** (GCM). In a nutshell, GCMs are systems of differential equations that can be integrated over time to yield predictions about variables such as temperature, wind speed, and precipitation. These models are grounded in physics, their inner workings are easily interpretable, and simulating them yields reasonably accurate outputs. However, running the simulations is a computationally expensive process and it's difficult to improve the models when given more data. This is where **machine learning algorithms** step in as a promising alternative. In particular, such algorithms have demonstrated competitiveness with traditional climate models in solving two sub-problems of climate modeling called "weather forecasting" and "spatial downscaling".



Weather Forecasting: Using historical data (left) to predict future weather conditions (right).



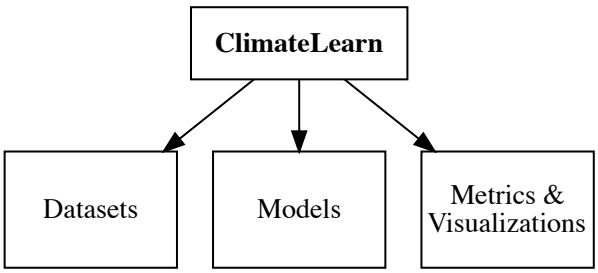
Spatial Downscaling: Refining low resolution global climate model (left) to high resolution (right).

Weather forecasting is the problem of predicting climate variables into the future. For example, given daily surface temperature in Los Angeles, California over the past week, what will daily surface temperatures look like over the next week? Answering questions like this is analogous to the problem of video frame prediction in computer vision. **Spatial downscaling** is the problem of refining spatially-coarse climate model predictions (*e.g.*, from a grid of 100 km $\times$ 100 km cells to a grid of 1 km $\times$ 1 km cells). This is similar to another computer vision problem called super resolution (SR), where the goal is to upsample low-resolution images. A key difference between forecasting/downscaling and frame-prediction/SR is that we can use additional signals to constrain the space of possible predictions. For instance, in video frame

additional signals to constrain the space of possible predictions. For instance, in video frame prediction, the machine learning model is given a sequence of images as input and produces a sequence of images as output. The input and output modalities are the same. In weather forecasting, the machine learning model can make use of exogenous variables in different modalities. Suppose that the model is predicting surface temperature. Future surface temperatures are not influenced only by past surface temperatures. Factors such as humidity and wind speed also play a role, and they can be provided as inputs to the model in addition to temperature.

Thus, as deep learning research has exploded in recent years, machine learning and climate scientists alike have begun exploring the application of deep learning methods to solving weather forecasting and spatial downscaling. However, the two communities approach the problem of applying machine learning in different ways. Climate scientists know what physical equations should be respected and what evaluation metrics are most important. Meanwhile, machine learning scientists know what architectures are best suited for what problems and how to process data in a way that is amenable to modern machine learning methods. Progress is impeded by confounding terminology (*e.g.*, "bias" in climate modeling versus "bias" in machine learning), a lack of standardization in applying machine learning for climate science problems (*e.g.*, defining appropriate train and held-out datasets, data augmentation strategies), and unfamiliarity with how to interpret climate data (*e.g.*, reanalysis vs. simulated datasets, file formats such as NetCDF). This lack of a "lingua franca" is what motivated us to launch **ClimateLearn**.

We believe that good research needs to be supported by good infrastructure. In that spirit, ClimateLearn is a Python package for accessing state-of-the-art climate data and machine learning models in a standardized, straightforward way. In this package, we provide access to multiple datasets, a zoo of state-of-the-art baseline models, and a suite of metrics and visualizations for large-scale benchmarking of weather forecasting and spatial downscaling methods.



ClimateLearn's features.

Datasets

ClimateLearn supports loading data from **ERA5** [2] [3], the fifth generation ECMWF (European Centre for Medium-Range Weather Forecasts) reanalysis for the global climate and weather from the past four to seven decades. A reanalysis dataset is one that combines historical observations into global estimates using modeling and data assimilation systems. This combination of real data and modeling allows reanalysis products to have complete global

data at fairly high accuracy. However, the process of creating the reanalysis is time-consuming. ERA5 data is published within 3 months of real-time, motivating the necessity for computationally cheap methods via machine learning. Besides the raw ERA5 data, ClimateLearn supports loading preprocessed ERA5 data from **WeatherBench** [4], a benchmark dataset for data-driven weather forecasting. In either case, ClimateLearn provides the data in a format that is easily used by today's deep learning architectures.

Models

ClimateLearn implements a variety of baseline machine learning algorithms so that users can quickly get a sense of how machine learning can be applied to forecasting and downscaling problems. These include simple statistical methods such as linear regression, persistence, and climatology as well as state-of-the-art deep learning implementations for **residual convolutional neural networks**, **U-nets**, and **vision transformers**. Our baseline models have been well-tuned for the climate tasks, and are easy to extend for other downstream pipelines in climate science.

Metrics & Visualizations

The predictions of such models can be easily evaluated and visualized using ClimateLearn's support for commonly used metrics such as (latitude-weighted) root mean squared error, anomaly correlation coefficient, Pearson's correlation coefficient, and mean bias. ClimateLearn also supports visualizations of ground truth, model predictions, and the difference between the two. Visual inspection of the predicted variables is a natural way of gaining an intuition about model performance and important outcomes.

Conclusion

Today we are launching ClimateLearn, a package that can bridge the gap between the climate science and machine learning communities through the provision of straightforward dataset access, baseline methods for easy comparison, and metrics and visualizations to understand model outputs.

Our roadmap for the future of ClimateLearn includes expanding support for **more datasets** such as CMIP6 (the sixth generation Climate Modeling Intercomparison Project) [5], which was used in the Sixth Assessment Report by the IPCC (International Panel on Climate Change) [1]. We also plan to add support for **probabilistic forecasting** with new metrics for uncertainty quantification such as continuous ranked probability score and new machine learning algorithms such as Bayesian neural networks and diffusion models. Implementing such features would create additional value for ClimateLearn. Machine learning researchers can unlock insights into model performance, expressiveness, and robustness. Climate scientists can understand how changing the values of input variables will result in different output

can understand how changing the values of input variables will result in different output distributions, which matches how modern climate studies are done: scientists provide a range of potential outcomes based on hypothetical emissions scenarios. We will also be formalizing guidelines for feature/pull requests to our Github repository and look forward to community contributions.

Our aim in building ClimateLearn is to create a tool that can accelerate research at the intersection of machine learning and climate science, and we hope you are just as excited about it as we are.

This blog post was written to accompany our ClimateLearn is available on GitHub at this link: https://github.com/aditya-grover/climate-learn. We previewed some of its key features at a spotlight tutorial in the "Tackling Climate Change with Machine Learning" Workshop at the Neural Information Processing Systems 2022 Conference.

References

1. **IPCC, 2022: Summary for Policymakers** [link]
   Pörtner, H., Roberts, D., Poloczanska, E., Mintenbeck, K., Tignor, M., Alegría, A., Craig, M., Langsdorf, S., Löschke, S., Möller, V. and (eds.), A.O., 2022. Climate Change 2022: Impacts, Adaptation, and Vulnerability, pp. 3-33. Cambridge University Press.
2. **ERA5 hourly data on single levels from 1959 to present. Copernicus Climate Change Service (C3S) Climate Data Store (CDS)** [link]
   Hersbach, H., Bell, B., Berrisford, P., Biavati, G., Horányi, A., Muñoz Sabater, J., Nicolas, J., Peubey, C., Radu, R., Rozum, I., Schepers, D., Simmons, A., Soci, C., Dee, D. and Thépaut, J., 2018.
3. **ERA5 hourly data on pressure levels from 1959 to present. Copernicus Climate Change Service (C3S) Climate Data Store (CDS)** [link]
   Hersbach, H., Bell, B., Berrisford, P., Biavati, G., Horányi, A., Muñoz Sabater, J., Nicolas, J., Peubey, C., Radu, R., Rozum, I., Schepers, D., Simmons, A., Soci, C., Dee, D. and Thépaut, J., 2018.
4. **WeatherBench: A Benchmark Data Set for Data-Driven Weather Forecasting** [link]
   Rasp, S., Dueben, P.D., Scher, S., Weyn, J.A., Mouatadid, S. and Thuerey, N., 2020. Journal of Advances in Modeling Earth Systems, Vol 12(11). American Geophysical Union (AGU). DOI: 10.1029/2020ms002203
5. **CCCma CanESM5 model output prepared for CMIP6 ScenarioMIP** [link]
   Swart, N.C., Cole, J.N., Kharin, V.V., Lazare, M., Scinocca, J.F., Gillett, N.P., Anstey, J., Arora, V., Christian, J.R., Jiao, Y., Lee, W.G., Majaess, F., Saenko, O.A., Seiler, C., Seinen, C., Shao, A., Solheim, L., von Salzen, K., Yang, D., Winter, B. and Sigmond, M., 2019. Earth System Grid Federation. DOI: 10.22033/ESGF/CMIP6.1317

An error occurred: giscus is not installed on this repository