Spatially Extended Tests of a Neural Network Parametrization Trained by Coarse-Graining

Noah D. Brenowitz and Christopher S. Bretherton

1Department of Atmospheric Sciences, University of Washington, Seattle, WA, USA

Abstract General circulation models (GCMs) typically have a grid size of 25–200 km. Parametrizations are used to represent diabatic processes such as radiative transfer and cloud microphysics and account for subgrid-scale motions and variability. Unlike traditional approaches, neural networks (NNs) can readily exploit recent observational data sets and global cloud-system resolving model (CRM) simulations to learn subgrid variability. This article describes an NN parametrization trained by coarse-graining a near-global CRM simulation with a 4-km horizontal grid spacing. The NN predicts the residual heating and moistening averaged over (160 km)² grid boxes as a function of the coarse-resolution fields within the same atmospheric column. This NN is coupled to the dynamical core of a GCM with the same 160-km resolution. A recent study described how to train such an NN to be stable when coupled to specified time-evolving advective forcings in a single-column model, but feedbacks between NN and GCM components cause spatially extended simulations to crash within a few days. Analyzing the linearized response of such an NN reveals that it learns to exploit a strong synchrony between precipitation and the atmospheric state above 10 km. Removing these variables from the NN’s inputs stabilizes the coupled simulations, which predict the future state more accurately than a coarse-resolution simulation without any parametrizations of subgrid-scale variability, although the mean state slowly drifts.

1. Introduction

Current global climate and weather models cannot explicitly resolve many important physical processes because of computational expenses. Global weather and short-range climate forecast models support grid sizes of 10 to 50 km (ECMWF, 2018; NOAA, 2018), while current climate models typically have 25- to 100-km grids. Unresolved processes, including cumulus convection, turbulence, and subgrid-cloud variability, are approximated by subgrid-scale parametrizations (Palmer, 2001). These are usually designed by human physical intuition, informed by high-resolution simulations and observations.

Cumulus convection in the tropics is one of the most dynamically important processes in the atmosphere, yet it is extremely difficult to parameterize because of the multiscale nature of moist flows (Majda, 2007; Nakazawa, 1988). There is a long-standing debate about whether convective parametrizations should be based on moisture convergence or quasi-equilibrium closures (Arakawa, 2004). Moreover, important mean state biases in climate modeling such as the double Intertropical Convergence Zone bias are sensitive to convective parametrization (Woelfle et al., 2018; Zhang & Wang, 2006). Climate models also struggle to simulate observed aspects of tropical variability such as the diurnal cycle of continental precipitation (Stratton & Stirling, 2012) and the Madden Julian Oscillation (Jiang, 2017; Jiang et al., 2015). Thus, an improved convective parametrization could help improve weather and climate simulations.

With sufficient computational resources, global cloud-system resolving models (GCRMs) with sub-5-km grid spacing can be run in global domains without deep convective parametrizations (Bretherton & Khairoutdinov, 2015; Satoh et al., 2008). Recent studies have used such simulations to study the moist static energy budget (Bretherton & Khairoutdinov, 2015) and cloud feedbacks (Narapitak et al., 2017). Centennial-scale climate GCRM simulations are not yet feasible, so improving parametrizations in coarse-resolution models remains an important goal.

It remains challenging for human experts to translate the vast volume of information in new high-resolution data sets and models into better parametrizations of moist physical processes incorporating subgrid variability. On the other hand, parametrizations can also be built using powerful regression tools developed in the field of machine learning (ML; Brenowitz & Bretherton, 2018; O’Gorman & Dwyer, 2018; Rasp et al., 2018).
In particular, Brenowitz and Bretherton (2018) trained a neural network (NN) parametrization for the combined diabatic heating and moistening in a coarse-grid model by coarse-graining a GCRM simulation. They showed that coupling this parametrization to specified dynamical tendencies in a single-column model (SCM) gives stable and highly accurate simulations if the NN is trained to optimize model performance over multiple time steps.

We now extend our SCM results by coupling an NN parametrization to the dynamical core of a general circulation model (GCM). As in Brenowitz and Bretherton (2018), this unified physics scheme will predict the sources of water and heat due to latent heating, turbulence, radiation, and any other physical process beyond resolved advection. Our main goal is to produce accurate multiple day forecasts with the coupled GCM-NN model.

After a brief survey of recent work on ML parametrization in section 2, we describe our training data and GCM configuration in section 3. Section 4 describes our coarse-graining and ML strategies. Particular focus is placed on the difficult task of ensuring stability in spatially extended simulations (section 4.3). Section 5 shows the results of the coupled simulations, and we conclude in section 6.

2. Review of ML Parametrization

One attractive use of ML is to automatically tune existing GCM parametrizations, which build in physical insights and constraints that may help them apply across a range of climates. Proposed techniques include ensemble Kalman inversion (Schneider et al., 2017), adjoint-based tuning (Lyu et al., 2018), and genetic algorithms (Langenbrunner & Neelin, 2017). These techniques can tune a few free parameters but may not scale to larger numbers of parameters. Moreover, existing parametrizations may not be flexible enough to be realistic in part because they have so few parameters. For instance, adjusting the entrainment rates in a convection scheme can improve mean state bias but harm the variability (Kim et al., 2011; Mapes & Neale, 2011).

A more ambitious approach replaces an existing GCM parametrization with a ML model, which has enough parameters to capture arbitrarily complex relationships (Cybenko, 1989; Hanin, 2017) present within a training data set. ML models are typically trained by minimizing a loss function, such as the mean-square error (MSE) compared to some reference outputs from the training data; the choice of the loss function is subjective and a key to good performance. Early studies (Chevallier et al., 1998; Krasnopolsky et al., 2005) trained NNs (Goodfellow et al., 2016) to emulate the outputs of radiative transfer codes in order to decrease computational expense. Subsequent studies also trained NNs to emulate existing convective parametrizations (O’Gorman & Dwyer, 2018) and superparametrizations (SPs; Rasp et al., 2018). In each of these instances, the ML model is trained with a nearly ideal data set, where the inputs are the state of the atmosphere, and the output is a tendency that has actually driven a GCM.

Training a parametrization by coarse-graining a GCRM is more difficult because the data are not cleanly divided into coarse-grid dynamics and parametrized physics. On the other hand, GCRMs are the most realistic models available because they do not impose an arbitrary scale separation as SP does. In this setting, how does one define the target tendency? In the first study of its kind, Krasnopolsky et al. (2010, 2013) defined the output as the apparent heating and moistening (Yanai et al., 1973) of limited-area simulations. They diagnosed heating rates and cloud fractions with high accuracy but did not present any prognostic simulations. Brenowitz and Bretherton (2018) extended this work in two ways. First, their training data set was a near-global CRM with rich multiscale convective organization. Second, they found SCM simulations with NNs trained in this way diverged to infinity after just a few time steps. They avoided this divergence by minimizing the loss accumulated over several predicted time steps.

In summary, when coarse-graining a GCRM simulation, there is no clear target to emulate. While Brenowitz and Bretherton (2018) developed an approach for ensuring stability in a SCM with prescribed dynamics, they did not test it within a full three-dimensional GCM where the dynamics interact with the NN parametrization.

3. Training Data and Atmospheric Model Configuration

3.1. Training Data

We use the same training data set as Brenowitz and Bretherton (2018): a near-global aquaplanet simulation (NG-Aqua) using the System for Atmospheric Modeling (SAM) version 6.10 (Khairoutdinov & Randall,
SAM is run in a cloud-system resolving configuration with a horizontal grid spacing of 4 km in a tropical channel domain measuring 20,480 km zonally by 10,240 km meridionally. The simulation has 34 vertical levels with a spacing that increases from 75 m at the surface to 1,200 m in the troposphere and stratosphere. The atmospheric circulation is driven by a zonally symmetric sea surface temperature (SST) profile which peaks at 300.15 K at the equator ($y = 5,120$ km) and decreases to 278.15 K at the poleward boundaries. Because SAM was originally designed for small-scale modeling, the grid is Cartesian and no spherical effects are included, except for a meridionally varying Coriolis parameter. Despite the simplifications described above, NG-Aqua features realistic multiscale organization of tropical and midlatitude systems.

The simulation uses the radiation scheme from version 3 of the Community Atmosphere Model with a zonally symmetric diurnal cycle and a bulk microphysics scheme (Khairoutdinov & Randall, 2003). It was initialized with small random perturbations to the temperature, spun up on a 20-km grid, then interpolated to 4 km and run for 100 days, storing the full 3-D outputs every 3 hr. To allow for the circulation to statistically equilibrate at 4 km, all analyses described below are performed on the final 80 days. More details about the model configuration are described by Brenowitz and Bretherton (2018) and Narenpitak et al. (2017).

### 3.2. Coarse-Resolution Model Configuration

We test the NN schemes within an atmospheric model with a grid resolution of 160 km, which is representative of coarse-resolution GCMs. This scale is coarse enough that the coarse-grid moist-convective dynamics have sufficient memory to be resolved by the 3-hr sampling rate of the NG-Aqua data; the grid mean precipitation has an autocorrelation of 0.54 at a lag of 3 hr.

We use SAM rather than a more traditional GCM as our coarse-grid model to ensure that it has the same geometry and dry dynamics as the training data. This model, coarse-SAM (cSAM), is run with the same vertical grid as the NG-Aqua simulation. Its default microphysics scheme has the same three prognostic thermodynamic variables as SAM: the total nonprecipitating water mixing ratio $q_T$; the precipitating water mixing ratio $q_p$, which is a fast variable that we will neglect in the parametrizations below; and the liquid-ice static energy $s_L$. See Khairoutdinov and Randall (2003, Appendix A) for more details.

SAM is typically used for cloud-resolving simulations so running it efficiently and stably with coarse resolution required several modifications. Most importantly, horizontal hyper-diffusion $-K_4 \nabla^4$ of $s_L$, $q_T$, and the three wind components, $u$, $v$, and $w$, was added to suppress grid-scale oscillations and model blow-up (divergence of the solutions to machine infinity in a finite amount of time). With a value of $K_4 = 1 \times 10^{15}$ m$^4$/s, simulations have more near-grid-scale power in kinetic energy than NG-Aqua, and maps of humidity and other tracers appear grainy compared to NG-Aqua. Increasing $K_4$ to $1 \times 10^{16}$ m$^4$/s damps these grid-scale oscillations, so we hereafter use that value.

For NN simulations, we use a simplified momentum damping rather than SAM’s default turbulence closure, which severely limits the time step on the 160 km grid. This forcing damps the velocity toward 0 at a rate which decreases linearly from $k_f = 1$ day$^{-1}$ at the surface to 0 at a $\sigma$-level of 0.7 (Held & Suarez, 1994). Because this is different than NG-Aqua, it leads to an inevitable drift of cSAM simulations away from NG-Aqua over the course of a few days. In the future, we hope to use learned coarse-grained momentum tendencies from NG-Aqua in place of this approach.

We compare simulations with cSAM coupled to our NN parametrization with a base version of cSAM with the microphysics and radiation tendencies calculated from grid-scale mean thermodynamics. Unlike the NN simulations, the surface fluxes are computed interactively and coupled to SAM’s default turbulence scheme. The base cSAM will only precipitate when a coarse-resolution grid cell achieves saturation, so we expect it will produce noisy simulations. Ideally, we would have preferred to compare the NN parametrizations against a traditional GCM parametrization suite, as did Rasp et al. (2018) and Brenowitz and Bretherton (2018), but no such scheme has been implemented in SAM. In this article, we assess the accuracy of 10-day weather simulations. They are initialized with the coarse-grained outputs from a particular time, Day 100.625, of the NG-Aqua simulation. Our goal is to obtain cSAM simulations that best match the actual evolution of NG-Aqua over the following 10 days. All of the simulations presented in this paper use a 120-s time step. The thermodynamic variables and vertical velocity from the 4-km NG-Aqua data are averaged over 40$^2$ grid cell blocks, but the horizontal velocities must be averaged along the interfaces of these blocks to respect the anelastic mass conservation equation.
4. ML Parametrization

4.1. Coarse-Graining Problem

An NN will parameterize the unknown sources of the thermodynamic variables $q_T$ and $s_L$. Their coarse-resolution budgets are given by

$$\frac{\partial \overline{s_L}}{\partial t} = \left( \frac{\partial \overline{s_L}}{\partial t} \right)_{GCM} + Q_1$$

$$\frac{\partial \overline{q_T}}{\partial t} = \left( \frac{\partial \overline{q_T}}{\partial t} \right)_{GCM} + Q_2$$

where $\overline{X}$ is the horizontal average of any variable $X$ over the coarse-grid boxes. The first term on the right-hand side of these budgets are the GCM’s tendencies, which include advection and any other explicitly treated process. The apparent heating $Q_1$ and moistening $Q_2$ are defined as a budget residual from these GCM tendencies (Yanai et al., 1973). Because they are residuals, they contain the effects of all unresolved and untreated physics including latent heating, turbulent mixing, and radiation. This parametrization neglects precipitating water $q_p$, which evolves on a much faster time scale than $q_T$ and $s_L$.

The goal of any parametrization is to approximate $Q_1$ and $Q_2$ as functions of the coarse-resolution variables alone. However, Palmer (2001) argues that stochastic and nonlocal effects must be included for certain problems. Because radiative heating and moist physical processes couple the atmosphere more strongly in the vertical than the horizontal direction, we follow the assumption of horizontal locality typically made in moist physics parametrizations; this reduces the dimensionality of the training problem, allowing robust results to be obtained from our NG-Aqua training data set. For simplicity, we also treat $Q_1$ and $Q_2$ as deterministic functions of the inputs.

The inputs are the vertical profiles of the prognostic variables $q_T$ and $s_L$, as well as the SST and downwelling insolation (SOLIN) at the top of the atmosphere, which are external parameters. Unlike Brenowitz and Bretherton (2018), we do not use the sensible or latent heat fluxes because they depend on the surface winds, which tend to be inaccurate because they are forced by the simplified damping scheme described in section 3 rather than a coarse-grained source from NG-Aqua.

Let the vector $\mathbf{x}_o^i$ concatenate the observed prognostic variables $\overline{q_T}$ and $\overline{s_L}$ over all coarse-grained grid levels in grid column $i$, and let $\mathbf{y}_o^i$ contain the auxiliary variables SST and SOLIN at this location. With these assumptions, (1) and (2) can be combined:

$$\frac{d\mathbf{x}_o^i}{dt} = g(\mathbf{x}_o^i, \mathbf{y}_o^i) + f(\mathbf{x}_o^i, \mathbf{y}_o^i; \theta) + \epsilon_i.$$  

where the subscript $i$ is the index of the given horizontal grid cell and the superscript $o$ indicates that these are observed quantities. The function $f$ and its parameters $\theta$ are invariant for all locations. Here, $g$ is the GCM tendency from coarse-grid advection (which involves other grid columns and hence is nonlocal); $f$ is the portion of the apparent sources $Q_1$ and $Q_2$ that can be modeled deterministically, which the NN will be used to parameterize; and $\epsilon_i$ includes stochastic and structural error. As with Brenowitz and Bretherton (2018), $f$ will be parameterized as an NN where the parameters $\theta$ include the weights and biases of the network’s layers.

4.2. Loss Functions for Stable Parametrizations

A key challenge for ML parameterizations is that the predicted state should not diverge to machine infinity after multiple steps. Traditionally, there are two ways this type of instability can arise. First, numerical instability arises from the spatial and/or temporal discretization of an otherwise stable dynamical system. Second, dynamical instability arises from the dynamical system itself. Since the parameterization $f$ represents a time-continuous processes, but is trained and evaluated with discrete data, it is hard to classify into these categories. With that acknowledgement, we simply define a “stable” scheme as one which does not produce diverging solutions in time.

Brenowitz and Bretherton (2018) trained a stable NN for use in SCM simulations by minimizing the multiple step prediction error (MSPE) over a 1- to 3-day prediction window $T$. SCM dynamics decouple the dynamics of an atmospheric column from its surroundings and are described mathematically by
\[
\frac{d\mathbf{x}^{\text{SCM}}}{dt} = g_i(\mathbf{x}^i(t), \mathbf{y}^i(t)) + f(\mathbf{x}^{\text{SCM}}_i, \mathbf{y}^i(t); \theta),
\]

where \( \mathbf{x}^{\text{SCM}}(t; t_0, \theta) \) is the single-column time series for a given location \( i \), starting prediction time \( t_0 \), and set of parameters \( \theta \). Compared to (3), the global state \( \mathbf{x}^i \) and the auxiliary variables \( \mathbf{y}^i \) are prescribed functions of time, which are decoupled from the local dynamics.

Brenowitz and Bretherton (2018) define the MSPE as

\[
J_{\text{MSPE}}(\theta) = \sum_{t_0} \frac{1}{T} \int_{t_0}^{t_0+T} \left\| \mathbf{x}^{\text{SCM}}(t; t_0, \theta) - \mathbf{x}^i(t) \right\|^2_M dt
\]

where the sum ranges over all possible prediction start times \( t_0 \) and spatial locations \( i \). The inner norm is given by

\[
\left\| \mathbf{s} \right\|^2_M = \frac{\lambda_q}{M} \int q^2(z) \rho_0(z) dz + \frac{\lambda_s}{M} \int s^2(z) \rho_0(z) dz.
\]

where \( \rho_0 \) is SAM's reference density profile. \( M = \int \rho_0 dz \) is the mass of an atmospheric column, while \( \lambda_s \) and \( \lambda_q \) are weight constants for the temperature and humidity, respectively. In this study, we find that when setting \( \lambda_q = 1.0 \text{ kg}^2/\text{g}^2 \) and \( \lambda_s = 1.0 \text{ K}^{-2} \) each variable contributes comparably to the overall loss. This resulted in a highly accurate single-column time-stepping scheme which closely replicated the \( q_T \) and \( s_L \) time series of the NG-Aqua data.

Unfortunately, MSPE training produces NN schemes which implicitly depend on the temporal discretization and time step used to simulate (4). The main symptom of this is a “spin-up” error where the SCM only produces accurate tendencies after a few time steps of prediction (see supporting information Figure S1): The spin-up predictions (panels c and d) more closely resemble the truth (panel a) than the instantaneous prediction (b) does. Because this spin-up process occurs after 3 hr, it did not cause large errors in the single-column simulations of Brenowitz and Bretherton (2018). However, for spatially extended simulations, the coupling between the GCM dynamics and the NN occurs at every 120-s time step, so the NN must produce accurate predictions from the start. Thus, MSPE-trained NNs are not suitable for use within cSAM because they spin up on a time scale longer than the cSAM time step.

Our goal in this article is to develop a loss function which produces an NN that gives spatially extended coarse-grid simulations that are stable, unbiased, and insensitive to the model time step. To encourage the NN to both predict \( Q_1 \) and \( Q_2 \) accurately and remain stable in time, this loss includes the two terms given by

\[
J_{\text{total}}(\theta) = J_{\text{instant}}(\theta) + J_{\text{stability}}(\theta).
\]

The first term, \( J_{\text{instant}} \), is the mass-weighted MSE of the predicted \( Q_1 \) and \( Q_2 \), which is given by

\[
J_{\text{instant}}(\theta) = \sum_{t_0} \left\| f(\mathbf{x}^i(t^n), \mathbf{y}^i(t^n); \theta) - \left[ \frac{Q_1}{Q_2} \right]_M \right\|^2_M,
\]

\[
= \sum_{t_0} \left\| \frac{dx^i}{dt}(t^n) - \left[ g_i(\mathbf{x}^i(t^n), \mathbf{y}^i(t^n)) + f(\mathbf{x}^i(t^n), \mathbf{y}^i(t^n); \theta) \right]_M \right\|^2_M.
\]

In practice, the storage terms \( \frac{dx^i}{dt} \) are estimated from the 3-hourly sampled data using finite differences. Thus, minimizing \( J_{\text{instant}} \) demands that \( f \) approximates the apparent heating and moistening.

On the other hand, stability requires that the small perturbations to the inputs \( \mathbf{x} \) do not grow under the dynamics given by (4). In practice, we enforce this using a heuristic penalty given by

\[
J_{\text{stability}} = \sum_{t_0} \left\| \mathbf{x}^{\text{mean}}(t_0 + T; t_0, \theta) - \langle \mathbf{x}^i \rangle_M \right\|^2_M,
\]

where \( \langle \cdot \rangle \) is the time average operator. \( \mathbf{x}^{\text{mean}}(t_0) \) is the single-column simulation under the action of the time-mean forcing, whose dynamics are given by
Figure 1. Linearized response of the neural network to input perturbations. Stippling and hatching indicate where the absolute value exceeds 10 and 100, respectively. The figure shows the sensitivity of the output at a pressure level $p_{ou}$ to input perturbations at another level $p_{in}$. The panels show the sensitivity of $Q_2$ to $q_T$ (a), $Q_2$ to $s_L$ (b), $Q_1$ to $q_T$ (c), and $Q_1$ to $s_L$ (d).

\[
\frac{dx_{\text{mean}}^i}{dt} = f(x_{\text{mean}}^i, y_i^o(t); \theta) + \langle g_i(x^o(t)) \rangle. \tag{11}
\]

Its initial condition is $x_{\text{mean}}^i(t_0, t_0; \theta) = x_i^0(t_0)$. This penalty demands that the SCM simulation does not diverge too quickly for a prediction of length $T$ (2.5 days). Because $J_{\text{stability}}$ uses constant forcing, it is less sensitive than the MSPE to the temporal discretization of (4). In practice, NNs trained to minimize $J_{\text{total}}$ avoid spin-up errors and accurately predict $Q_1$ and $Q_2$ at the initial time. Like MSPE-trained NNs, they do produce stable, albeit less accurate, SCM simulations. Since our main goal is to produce accurate coupled GCM-NN simulations, this deteriorated SCM performance is acceptable.

4.3. Coupled Instability

The methods discussed still do not prevent model blow-up when coupled to cSAM, as we will show in section 5. This occurs because the single-column training strategy does not account for feedbacks between cSAM and the NN so corresponding instabilities are not penalized.

A similar feedback causes grid-scale storms when a GCM is coupled to a moisture-convergence closure (Arakawa, 2004). Emanuel et al. (1994) argue this closure fundamentally confuses cause and effect and that precipitation drives moisture convergence rather than vice versa. In fact, such misleading correlations are common in nonlinear dynamical systems such as the atmosphere, a phenomenon known as synchronization (Pecora et al., 1997), so data-driven models must be taught to ignore inputs that are synchronized to the desired outputs.

4.3.1. Interpreting the Trained Networks Using Saliency Maps

What input variables do our NNs depend on most sensitively? Saliency maps are a common tool for interpreting the dependency structure of NN. They measure the sensitivity of the output variables to small perturbations in the inputs (Simonyan et al., 2013). Mathematically, they are given by the jacobian matrix

\[
J(x, y) = \nabla_{xy} f(x, y; \theta),
\]

where the derivative is with respect to the phase space variables $x$. For ease of discussion, we defer the model architecture and training details to section 4.4. We first train an NN using the stability-penalized
Our NN predicts that heating and moistening at all heights are very sensitive to the humidity near and above the tropopause (200 mbar). A 1-g/kg increase of humidity at this level leads to more than 10 g kg\(^{-1}\) day\(^{-1}\) of drying at all lower levels. While this result is potentially consistent with formation of high clouds, it is not desirable that the NN depends so strongly on these upper atmospheric inputs, especially because it does not accurately predict \(Q_1\) and \(Q_2\) at these levels (cf. Figure 4). Thus, the NN depends most sensitively on the inputs that it performs worst on. To demonstrate the robustness of this sensitivity to changes in the base-state profile, Figure S2 shows \(\nabla qT Q\) for 20 profiles randomly drawn from between 3\(^\circ\)S and 3\(^\circ\)N. While the saliency maps do vary from sample to sample, the sensitivity to the upper atmospheric humidity is robust.

The saliency maps are analogous to the so-called linearized response functions (LRFs) of Kuang (2018), which were computed by perturbing the steady forcing of a CRM. The LRFs have an upward dependence structure (a mainly upper-triangular structure in the panels of Figure 2 of Kuang, 2018), where the heating and, to a lesser extent, the moistening at a given vertical level depend only on data from below. This is physically realistic since cumulus convection initiates at lower elevations and deepens into cumulonimbus; evaporation of precipitation leads to some low-triangular structure as well. Unfortunately, we cannot compare our saliency maps directly to the LRFs of Kuang (2018) for several reasons: they use different thermodynamic variables (e.g., temperature and humidity); they ignore moisture inputs above 150 mbar; and they disaggregate radiation from other heat sources. Nonetheless, some loose comparisons are possible. For instance, both our saliency maps and the LRFs of Kuang (2018) show that decreasing the lower tropospheric temperature, which destabilizes the atmosphere, leads to heating/drying throughout the column (Figure 1d). Likewise, the lower tropospheric moisture induces convective heating above (Figure 1c). Future studies will explore the comparison with LRFs more completely.

The NN is likely sensitive to upper-level humidity because the latter is synchronized to the convective processes below. This occurs because upper-level total water is (1) strongly damped and (2) forced by supply of moisture from advection. Figure 2a shows the effective damping rate given by

\[
\gamma = -\langle qT qT \rangle / \langle qT^2 \rangle
\]  

where \(\langle \cdot \rangle\) is the zonal and time averaging operator. Near the tropopause, the moisture has a fast damping time scale of 3 hr, which is the sampling rate of the NG-Aqua data. A conceptual mathematical model for this behavior is given by

\[
\partial_t qT = -\gamma qT + F_{\text{adv}}
\]

where \(F_{\text{adv}}\) is the supply of moisture by advection, which we treat as an external forcing. When the right-hand side of this equation dominates over the left, which is the case for the upper atmospheric humidity, the total water is approximately given by \(qT \approx \gamma^{-1} F_{\text{adv}}\). In other words, the total water in the upper atmosphere becomes synchronized to the supply of moisture from below, as shown in Figure 2b. Kuang (2018) justified ignoring the humidity variables above 150 mbar on these grounds. Likewise, this synchronization could harm our ML procedure by polluting the inputs with information about the desired outputs.

If used as a closure assumption, the synchronization of the upper atmospheric total water to the convective processes below could cause a dynamical instability as follows. A positive moisture anomaly in the tropical upper troposphere will increase the predicted precipitation and heating. This will then increase the upward velocity, which closes the feedback loop by increasing the supply of moisture to the upper atmosphere. Similarly to moisture-convergence closures, this feedback can lead to grid-scale storms.

This instability might be less problematic had we been able to customize the 4-km NG-Aqua simulations for the training process so as to avoid this synchronization issue, for example, by using higher output time resolution and calculation of instantaneous heating and moistening rates. However, we were forced to work around it, as described in the next section.

4.3.2. Stabilizing the NN by Ignoring Upper-Atmospheric Inputs

We regularize the NN by not using the humidity and temperature above certain heights as predictors. This is motivated by (1) the synchronization argument above (Figure 2) and (2) the NN’s poor skill at predicting the upper atmospheric \(Q_1\) and \(Q_2\). In any event, we obtain a stable simulation without grid-scale storms by
training the NN without including \( q_T \) and \( s_L \) at or above levels 15 (450 mbar) and 19 (225 mbar), respectively, as inputs. However, the NN still predicts the apparent heating and moistening at these levels. We have run cSAM simulations with this configuration for up to 100 days without model blow-up or grid point storms. Admittedly, this is a crude approach that future iterations of this work should improve.

Figure 3 shows the saliency map of the modified NN. This modification did not greatly alter sensitivity of the NN to inputs below the ignored levels. Overall, the modified NN is slightly more sensitive to tropospheric \( s_L \) and \( q_T \), but the structure is similar to Figure 1. That said, there are some strong oscillations in the response of \( Q_1 \) and \( Q_2 \) to \( q_T \) near 500 mbar. Thus, removing upper atmospheric inputs did not harm the physically plausible characteristics of the original model.

4.4. Network Architecture and Training

NNs are nonlinear regression models which are parametrized by sequentially passing an input vector through alternating linear transformations and nonlinear function applications. The action of each layer on an input vector \( x \) is given by \( \sigma(Ax + b) \), where the matrix \( A \) and vector \( b \) are free parameters which will be tuned through the training process. The nonlinear “activation function” \( \sigma(\cdot) \) is applied element-wise to the output

Figure 3. Same as Figure 1, but for a model which ignores the humidity and temperature above the dashed lines at \( p = 450 \) and 225 mbar, respectively.
of the linear transformation. For problems with continuous output spaces such as parameterization, the final layer will have the identity as its activation function. We refer the interested reader to Goodfellow et al. (2016) for more details.

Rasp et al. (2018) found that an NN needed at least nine hidden layers to be stable when coupled to a dynamical core. Unfortunately, we could not replicate this result. Increasing the depth of the our NNs had no impact on stability. However, since we are training with global data, we use a slightly deeper network than the single layer model Brenowitz and Bretherton (2018) used for the tropics alone. Each model trained in this paper has three densely connected hidden layers with 256 nodes each and rectified linear unit activations. Altogether, this architecture has 160,266 free parameters.

Prior to input, each input variable is normalized by subtracting the global mean and dividing by the standard deviation plus a small number \(10^{-7}\) for each level independently. The small number ensures that the denominator is nonzero. This level-by-level normalization is a departure from past studies (Brenowitz & Bretherton, 2018; Rasp et al., 2018) and avoids prejudicing the training procedure against input variables with small variance. That said, the choice of normalization has little impact upon our results. In particular, NNs trained with inputs normalized as Brenowitz and Bretherton (2018) also learn to depend strongly on the upper atmospheric humidity and suffer from coupled instabilities discussed above.

The networks weights and biases are optimized by stochastic gradient descent (SGD) with the popular Adam (Kingma & Ba, 2014) optimizer with an initial learning rate of 0.005. If a sample is defined as an individual atmospheric column for a given horizontal location and time step, then there are 5,242,880 samples in the training data set. These samples are randomly in horizontal space and formed into batches of 64 time series, each containing the full 80 days of data. Five epochs (i.e., passes through the entire training data set) of training are performed. In practice, the SGD converges quickly, but time and zonal mean bias of the predictions fluctuates from epoch to epoch. This occurs because mean state biases in predicted heating and moistening only account for a small proportion of the overall error because these are such noisy fields. To mitigate this bias, we select for further analysis the epoch which minimizes the global mean-square bias of the predicted net precipitation. Future work should attempt to improve the convergence of SGD.

**Figure 4.** The \(R^2\) of the \(Q_1\) (a) and \(Q_2\) (b) predicted by NN-lower, which ignores upper atmospheric inputs. (c) The accuracy \((R^2)\) for the mass-weighted vertical integrals of \(Q_1\) (heating) and \(Q_2\) (moistening). The \(R^2\) coefficient measures the accuracy of a prediction and is defined as the ratio of the variance explained by the NN to the total variance. NN = neural network.
Figure 5. Comparison of the true $Q_1$ and $Q_2$ and instantaneous predictions with NN-Lower for $x = 0$ km at Day 101.875. Dashed lines indicate the levels above which the NN ignores $s_L$ and $q_T$. (a) $Q_1$ truth, (b) $Q_1$ NN-Lower, (c) $Q_2$ truth, and (d) $Q_2$ NN-Lower. NN = neural network.

The NN training and linear regression are implemented in Python using PyTorch (Paszke et al., 2017). The NN models are then coupled to cSAM using a python interface. The predicted tendencies are disabled within three grid cells (480 km) of the poleward boundaries because the NN is very inaccurate there as measured by its $R^2$ coefficient compared to $Q_1$ and $Q_2$ (cf. Figure 4). We suspect this inaccuracy occurs because the flow in these regions is greatly influenced by the no-normal flow boundary conditions there. In particular, these boundary conditions could either introduce nonlocal dependencies that violate the horizontal locality assumptions of the parameterization or simply inject harmful noise into the training data.

5. Results

We now present our spatially extended simulations. As described in section 3, the simulations are initialized with the NG-Aqua data at Day 100.625, and our discussion focuses on the prediction accuracy in the first 10 forecast days. Two NN parametrizations will be compared with the training data and the base simulation with resolved-scale microphysics and radiation. The first NN simulation, labeled as “NN-All,” is performed using the NN which includes all levels as input. The discussion in section 4.3 indicates that this setup leads to instability. The NN in the second simulation, known as “NN-Lower,” only uses humidities below level 16 (375 mbar) and temperature below level 19 (226 mbar) as described in section 4.3.2. When evaluated directly on the NG-Aqua data, NN-Lower has an $R^2$ value of around 60% for $Q_1$ and $Q_2$ in the lower troposphere (Figure 4). Interestingly, Figure 5 shows that the predicted heating and moistening for a single time point are less noisy than the “true” apparent heating $Q_1$ and apparent moistening $Q_2$. Thus, the NN-Lower parametrization performs well in an instantaneous sense.

Figure 6b demonstrates that the NN-Lower configuration is stable and produces a reasonable prediction of the precipitable water (PW) after 5.5 days, albeit with a visible loss in large-scale moisture variability in the tropics. On the other hand, NN-All shows a narrow line of moisture concentrated at the equator, coinciding with intense precipitation and ascent (not shown). Within this line are grid-scale artifacts devoid of humidity, and this simulation became unstable and crashed immediately after this snapshot. Likewise, the base simulation suffers from grid point storms in the tropics and frontal regions in the extratropics, and the simulation crashes (i.e., produces infinite vertical velocity) after 9 days with a 120-s time step. It is possible that using a smaller time step could stabilize this simulation, but 9 days is sufficient to evaluate the accuracy of its forecasts.
5.1. Short-Term Forecast Accuracy of the Coupled Simulations

Figure 7 shows the root-mean-square error (RMSE) of $s_L$, $q_T$, and the vertical component of vorticity for each configuration. The RMSE values are vertically integrated and mass weighted and averaged over different meridional bands. The tropics ($11.5^\circ$S to $11.5^\circ$N) lie within 1,280 km of the equator, the subtropics ($11.5^\circ$–23$^\circ$) are the 1,280-km-wide bands lying north and south of the tropics, and the extratropics ($23^\circ$–46$^\circ$) are the outermost 2,560 km of the domain. For all regions and variables, NN-Lower has the lowest RMSE. This improvement is most striking for the thermodynamic variables $q_T$ and $s_L$, which have parameterized source terms. Compared to the base simulation, NN-Lower has 50% lower errors in both $s_L$ and $q_T$. The vorticity error improves less, which is not surprising since we do not attempt to correctly represent the source of momentum ($Q_3$). However, in the extratropics, both NN-Lower and NN-All outperform the base simulation. Overall, NN-Lower produces the most accurate predictions.

The NN-Lower simulation also predicts the net precipitation rate more accurately than the base simulation does. The net precipitation rate is given by

$$-\int \rho Q_2 \, dz = P - E,$$

where $P$ is precipitation and $E$ is the evaporation (Yanai et al., 1973). Net precipitation is the closest analog to precipitation that our NNs can predict because they predict the combined effect of precipitation and evaporation.

The net precipitation is much less noisy than in NN-Lower than the base run. Figure 8 shows maps of net precipitation after 2 days of prediction (Day 102.625) for the NG-Aqua training data and the NN-Lower and base simulations. Overall, the NN-Lower simulation produces the correct extratropical pattern of net precipitation but fails to produce enough variability in the tropics. In NG-Aqua, the tropical precipitation occurs in small-scale clusters and in a larger-scale Kelvin wave centered around $x = 15,000$ km at Day 102.625; elsewhere there is column moistening due to evaporation. On the other hand, the base simulation produces far too strong and noisy precipitation, as expected for a scheme which only rains when a large-scale grid cell becomes saturated.

Does the NN predict overly smooth net precipitation at the initial time or does this smoothness only appear as the coupled simulation evolves? The NN’s “semiprognostic” net precipitation (i.e., predicted from the true coarse-grained NG-Aqua state at that time) is also shown in Figure 8b. Even in this instantaneous sense, the NN-Lower prefers to lightly rain throughout the tropics, rather than moisten in certain regions and strongly
Figure 7. Root-mean-square error (RMSE) evolution versus forecast lead time for $s_L$ (a–c), $q_T$ (d–f), and vertical relative vorticity (g–i). Separate curves are shown for each region (column) and cSAM configuration (indicated by color).

dry in others. This suggests that ML parametrizations are still susceptible to the same biases as traditional schemes.

Figure 9 shows the pattern correlation compared to NG-Aqua of net precipitation for the NN-Lower and base simulations. NN-Lower has higher pattern correlations than the base simulation at all times and retains a correlation of 0.5 up to Day 101.5 in the tropics and beyond Day 102.5 in the extratropics and subtropics. By this measure, the base simulation has little predictive skill in the tropics.

5.2. Mean State Bias

Short-term predictions with NN-Lower are reasonably accurate, but the mean state drifts away from the true climate. While the short-term prediction results above are robust, the details of this drift are sensitive to the noise in the SGD process used to train the NN. For example, the biases of an NN saved after four and five epochs of training will differ. With that caveat, Figure 10 shows time evolution of the zonal mean PW and net precipitation for this particular NN-Lower simulation. The fields are shown for Days 100–120 for NG-Aqua and the NN-Lower simulation; the base simulation crashed in Day 110. The tropical net precipitation and PW in NN-Lower are reasonably similar to NG-Aqua, but the width of the rainy/moist region shrinks throughout the simulation. Also, there is much less negative evaporation (i.e., negative net precipitation) in the subtropical regions. On the other hand, the net precipitation in the base simulation becomes too concentrated at the equator and atmosphere dries out significantly. Overall, NN-Lower outperforms the base case.

Figure 11 shows the zonal mean bias versus NG-Aqua for the zonal winds, meridional stream function, humidity $q_T$, and temperature $s_L$ in the NN-Lower simulation, time-averaged over forecast Days 5–10. The humidity bias is relatively small with an amplitude of less than 2 g/kg. The bias of $s_L$ is also small and peaks near the meridional boundaries of the domain where the NN is not active. On the other hand, the meridional circulation substantially weakens from a peak mass transport of $80 \times 10^{12}$ to $35 \times 10^{12}$ kg/s. This is associated with the decreasing $P - E$ in the tropics seen in Figure 10. The zonal mean zonal winds also dif-
Figure 8. Comparison of forecasted and actual net precipitation after two forecast days (time = 102.625 days). (a) NG-Aqua, (b) NN-Lower semiprognostic, (c) NN-Lower simulation, and (d) base simulation.

The surface westerlies and eddy-driven-jets become weaker, likely due to the Held-Suarez momentum damping. At the same time, the jets shift southward and superrotation develops near 250 mbar, likely as a transient response to the weakening Hadley circulation. Overall, the small biases in the thermodynamic variables $q_T$ and $s_L$ belie growing biases in the circulation and zonal wind. In addition, the biases in $q_T$ and $s_L$ fluctuate and are sensitive to the NN training, but the biases in circulation and zonal wind are quite robust.

Several aspects of our coarse-graining procedure could cause the mean state to drift, which Rasp et al. (2018) did not observe with their SP data. First, unlike in SP, NG-Aqua contains many subgrid-scale sources of momentum, which we neglected altogether. This seems consistent with the large biases in the zonal winds and the meridional stream function. Second, SP assumes a perfect separation between resolved and unresolved scales (Majda, 2007), which is only approximately true when coarse-graining GCRM data like NG-Aqua. For instance, it is unknown how precisely the apparent heating and moistening should approximate the coarse-grid average of the radiative and microphysical processes. Third, our training procedure depends implicitly on the resolution of the training data, but cSAM has both explicit dissipation (e.g., hyperdiffusion and damping) and implicit numerical dissipation which the training data lack. Thus, the effective resolution of the coupled simulations is less than that of the training data, so developing a scheme which adapts across an array of resolutions could ameliorate these biases. Finally, the biases could develop because the parameterization is deterministic rather than stochastic.

Figure 9. Hovmoller diagram of the net precipitation pattern correlation between the truth and the (a) NN-Lower and (b) base simulations. The white contour indicates a correlation of 0.5.
To elaborate on this final point, Figure 12 shows that the joint distribution of PW and net precipitation has much less variance in NN-Lower (panel b) than in NG-Aqua (panel a). Furthermore, the predicted net precipitation has too little variance even when evaluated on the true state (panel c). Such lack of variability could lead to mean state bias, because the expected precipitation depends exponentially on PW (Bretherton et al., 2004; Rushley et al., 2018), so a reduction in the zonal variance of PW (e.g., fewer extremely moist or dry coarse grid cells) will tend to decrease the zonal mean precipitation for a given zonal mean PW. Thus,
enhancing the variance of the conditional distributions of net precipitation could ameliorate the circulation biases in these simulations.

6. Conclusions

We use a NN to build a unified physics parametrization of diabatic heating and moistening, which include the effects of unresolved subgrid processes. The NN is trained by coarse-graining a near-global aquaplanet simulation with a grid size of 4 km to a resolution of 160 km. It learns the coarse-grid heating and moistening profiles, calculated as residuals from the coarse-grid dynamics, from the temperature and moisture profiles, plus auxiliary parameters. A key challenge of the coarse-graining approach is that it lacks the natural hierarchical structure of other training data sets used in the literature, such as SP (Rasp et al., 2018). In the SP application, for instance, the target model and the training model share the same software interface for the physical process tendencies. These tendencies are predicted by a high-resolution submodel in the SP training data set and must be learned by the NN parametrization for application to the target model. This application is thus a clear target for emulation.

We extend the SCM results of Brenowitz and Bretherton (2018) to a spatially extended simulation. The coarse-grid dynamical core is the same anelastic atmospheric model that generated the 4-km training data set but is run with a 160-km resolution. This model needed additional damping in the form of horizontal hyper-diffusion and Newtonian relaxation near the surface to run stably in a base configuration with its default resolved-scale microphysics and radiation schemes, or with our NN unified-physics parametrization. We developed a deeper version of our NN capable of simulating the diabatic heating and moistening over the entire 46°S to 46°N domain of our training simulations, rather than just the tropical subdomain used by Brenowitz and Bretherton (2018).

An attempt to couple a preliminary version of this NN to this GCM caused the model to blow up. Analyzing the linearized response of that NN showed that it inadvertently learns to exploit a strong correlation between upper atmospheric humidity and precipitation. This correlation is likely an artifact of the short lifetime of water vapor at those heights and the strong forcing from atmospheric convection below.

The spatially extended simulations were stabilized by removing the upper atmospheric humidity and temperature from the NN inputs. This configuration can run stably indefinitely, without blowing up. Thus, stabilizing the parametrization required a rather crude human intervention. Future studies will need to explore automatic ways to discover true causal relationships and forestall model blow-up in a dynamically coupled setting.

The resulting NN predicted the weather of NG-Aqua with much higher forecast skill and lower bias than a base coarse-grid simulation with only resolved microphysics and radiation parametrizations, which is the only reference case we could easily use as a metric. For a fair comparison, we will need to train and implement the ML parametrization in a global model whose coarse-grid version is typically run with a traditional suite of physical parametrizations.

More work is needed to quantify and reduce the long-term biases in these simulations. While temperature and humidity biases are small in the first 10 days, the Hadley circulation is dramatically weakened. To better
References

Arakawa, A. (2004). The cumulus parameterization problem: Past, present, and future. Journal of Climate, 17(13), 2493–2525. https://doi.org/10.1175/1520-0442(2004)017<2493:CRPTPO>2.0.CO;2

Brenowitz, N. D. (2019a). Data and code for training neural network parameterizations from an near-global aqua-planet simulation. https://doi.org/10.5281/zenodo.3248382

Brenowitz, N. D. (2019b). https://doi.org/10.5281/zenodo.3248586

Brenowitz, N. D., & Bretherton, C. S. (2018). Prognostic validation of a neural network unified physics parameterization. Geophysical Research Letters, 17, 6289–6298. https://doi.org/10.1029/2018GL078510

Bretherton, C. S., Peters, M. E., & Back, L. E. (2004). Relationships between water vapor path and precipitation over the tropical oceans. Journal of Climate, 17(7), 1517–1528. https://doi.org/10.1175/1520-0442(2004)017<1517:RBWPPA>2.0.CO;2

Chevallier, F., Chéruy, F., Scott, N. A., & Chédin, A. (1998). A neural network approach for a fast and accurate computation of a longwave radiative budget. Journal of Applied Meteorology and Climatology, 37(11), 1385–1397. https://doi.org/10.1175/1520-0450(1998)037<1385:ANNAAP>2.0.CO;2

Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. Mathematics of Control, Signals, and Systems, 2(4), 303–314. https://doi.org/10.1007/BF02551274

ECMWF (2018). Part III : Dynamics and numerical procedures, IFS Documentation CY41R1 (pp. 1–31). Reading, UK: ECMW.

Emanuel, K. A., & Bretherton, C. S. (1994). On large-scale circulations in convecting atmospheres. Journal of the Atmospheric Sciences, 51(9), 1370–1383. https://doi.org/10.1175/1520-0442(1994)051<1370:OSCSIC>2.0.CO;2

Graepel, T., & Putzer, K. (2018). The 2010 International Joint Conference on Neural Networks (IJCNN) (pp. 1–6). Barcelona, Spain. https://doi.org/10.1109/IJCNN.2018.8485193

Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep learning. Cambridge, Massachusetts: The MIT Press.

Held, I. M., & Suarez, M. J. (1994). A proposal for the intercomparison of the dynamical cores of atmospheric general circulation models. Bulletin of the American Meteorological Society, 75(10), 1825–1830. https://doi.org/10.1175/1520-0477(1994)075<1825:APFTIO>2.0.CO;2

Jiang, X. (2017). Key processes for the eastward propagation of the Madden-Julian oscillation based on multimodel simulations. Journal of Geophysical Research: Atmospheres, 122, 755–770. https://doi.org/10.1002/2016JD025955

Jiang, X., Waliser, D. E., Xavier, P. K., Peixoto, J. T., Klingaman, N. P., Woolnough, S. J., et al. (2015). Vertical structure and physical processes of the Madden-Julian oscillation: Exploring key model physics in climate simulations. Journal of Geophysical Research: Atmospheres, 120, 4718–4748. https://doi.org/10.1002/2014JD023375

Khairoutdinov, M. F., & Randall, D. A. (2003). Cloud resolving modeling of the ARM summer 1997 IOP: Model formulation, results, uncertainties, and sensitivities. Journal of the Atmospheric Sciences, 60(4), 607–625. https://doi.org/10.1175/1520-0442(2003)060<0607:CRMOTA>2.0.CO;2

Kim, D., Jang, Y.-S., Kim, D.-H., Jin, Y.-H., Watanabe, M., Jin, F.-F., & Kug, J.-S. (2011). El Niño-Southern Oscillation sensitivity to cumulus entrainment in a coupled general circulation model: ENSO sensitive to convection scheme. Journal of Geophysical Research, 116, D22112. https://doi.org/10.1029/2011JD016526

Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization.

Krasnopolsky, V. M., Fox-Rabinovitz, M. S., & Belochitski, A. A. (2010). Development of neural network convection parameterizations for numerical climate and weather prediction models using cloud resolving model simulations. In The 2010 International Joint Conference on Neural Networks (IJCNN) (pp. 1–8). Barcelona, Spain. https://doi.org/10.1109/IJCNN.2010.5596766

Krasnopolsky, V. M., Fox-Rabinovitz, M. S., & Belochitski, A. A. (2013). Using ensemble of neural networks to learn stochastic convection parameterizations for climate and numerical weather prediction models from data simulated by a cloud resolving model. Advances in Artificial Neural Systems, 2013, e485913. https://doi.org/10.1155/2013/485913

Krasnopolsky, V. M., Fox-Rabinovitz, M. S., & Chalikov, D. V. (2005). New approach to calculation of atmospheric model physics: Accurate and fast neural network emulation of longwave radiation in a climate model. Monthly Weather Review, 133(5), 1370–1383. https://doi.org/10.1175/MWR2923.1

Kuang, Z. (2018). Linear stability of moist convecting atmospheres part I: From linear response functions to a simple model and applications to convectively coupled waves. Journal of the Atmospheric Sciences, 75, 2889–2907. https://doi.org/10.1175/JAS-D-18-0092.1

Langenbrunner, B., & Neelin, J. D. (2017). Pareto-optimal estimates of California precipitation change. Geophysical Research Letters, 44, 12,436–12,446. https://doi.org/10.1002/2017GL075226

Lyu, G., Köhl, A., Matei, I., & Stammer, D. (2018). Adjoint-based climate model tuning: Application to the planet simulator. Journal of Advances in Modeling Earth Systems, 10, 207–222. https://doi.org/10.1002/2017MS001194

Majda, A. J. (2007). Multiscale models with moisture and systematic strategies for superparameterization. Journal of the Atmospheric Sciences, 64(7), 2726–2743. https://doi.org/10.1175/JAS3976.1

Mapes, B., & Neale, R. (2011). Parameterizing convective organization to escape the entrainment dilemma. Journal of Advances in Modeling Earth Systems, 3, M06004. https://doi.org/10.1029/2011MS000042

NOAA (2018). Strategic implementation plan for evolution of NGGPS to a national unified modeling system.

Nakazawa, T. (1988). Tropical super clusters within intraseasonal variations over the western Pacific. Journal of the Meteorological Society of Japan. Ser. II, 66(6), 823–839.
Narenpitak, P., Bretherton, C. S., & Khairoutdinov, M. F. (2017). Cloud and circulation feedbacks in a near-global aquaplanet cloud-resolving model: Cloud feedbacks in a near-global CRM. Journal of Advances in Modeling Earth Systems, 9, 1069–1090. https://doi.org/10.1002/2019MS001711

O’Gorman, P. A., & Dwyer, J. G. (2018). Using machine learning to parameterize moist convection: Potential for modeling of climate, climate change, and extreme events. Journal of Advances in Modeling Earth Systems, 10, 2548–2563. https://doi.org/10.1029/2018MS001351

Palmer, T. N. (2001). A nonlinear dynamical perspective on model error: A proposal for non-local stochastic–dynamic parametrization in weather and climate prediction models. Quarterly Journal of the Royal Meteorological Society, 127(572), 279–304. https://doi.org/10.1002/qj.49712757202

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., et al. (2017). Automatic differentiation in PyTorch.

Pecora, L. M., Carroll, T. L., Johnson, G. A., Mar, D. J., & Heagy, J. F. (1997). Fundamentals of synchronization in chaotic systems, concepts, and applications. Chaos, 7(4), 520–543. https://doi.org/10.1063/1.166278

Rasp, S., Pritchard, M. S., & Gentile, P. (2018). Deep learning to represent subgrid processes in climate models. Proceedings of the National Academy of Sciences of the United States of America, 115(39), 9684–9689. https://doi.org/10.1073/pnas.1810286115

Rushley, S. S., Kim, D., Bretherton, C. S., & Ahn, M.-S. (2018). Reexamining the nonlinear moisture-precipitation relationship over the tropical oceans. Geophysical Research Letters, 45, 2017GL076296. https://doi.org/10.1002/2017GL076296

Satoh, M., Matsuno, T., Tomita, H., Miura, H., Nasuno, T., & Iga, S. (2008). Nonhydrostatic icosahedral atmospheric model (NICAM) for global cloud resolving simulations. Journal of Computational Physics, 227(7), 3486–3514. https://doi.org/10.1016/j.jcp.2007.02.006

Schneider, T., Lan, S., Stuart, A., & Teixeira, J. (2017). Earth system modeling 2.0: A blueprint for models that learn from observations and targeted high-resolution simulations. Geophysical Research Letters, 44, 12,396–12,417. https://doi.org/10.1002/2017GL076101

Simonyan, K., Vedaldi, A., & Zisserman, A. (2013). Deep inside convolutional networks: Visualising image classification models and saliency maps.

Stratton, R. A., & Stirling, A. J. (2012). Improving the diurnal cycle of convection in GCMs. Quarterly Journal of the Royal Meteorological Society, 138(666), 1121–1134. https://doi.org/10.1002/qj.991

Woelfle, M. D., Yu, S., Bretherton, C. S., & Pritchard, M. S. (2018). Sensitivity of coupled tropical Pacific model biases to convective parameterization in CESM1. Journal of Advances in Modeling Earth Systems, 10, 126–144. https://doi.org/10.1002/2017MS001176

Yanai, M., Esbensen, S., & Chu, J.-H. (1973). Determination of bulk properties of tropical cloud clusters from large-scale heat and moisture budgets. Journal of the Atmospheric Sciences, 30(4), 611–627. https://doi.org/10.1175/1520-0469(1973)030<0611:DOBOP>2.0.CO;2

Zhang, G. J., & Wang, H. (2006). Toward mitigating the double ITCZ problem in NCAR CCSM3. Geophysical Research Letters, 33, L06709. https://doi.org/10.1029/2005GL025229