Statistical methods and machine learning in weather and climate modeling

Stephan Rasp

München 2018

Statistical methods and machine learning in weather and climate modeling

Stephan Rasp

Dissertation
an der Fakulät für Physik
der Ludwig–Maximilians–Universität
München

vorgelegt von
Stephan Rasp
aus München

München, 19. Dezember 2018

Erstgutachter: Prof. Dr. George C. Craig

Zweitgutachter: Prof. Dr. Bernhard Mayer

Tag der mündlichen Prüfung: 15.3.2019

Zusammenfassung

Um Wetter und Klima vorhersagen zu können, muss ein sehr komplexes und chaotisches System simuliert werden: die Atmosphäre. In den letzten Jahrzehnten wurden erstaunliche Fortschritte gemacht; dennoch verbleiben hartnäckige Fehler. Zwei Hauptursachen für diese Fehler sind die Parametrisierung von kleinskaligen Prozesses, wie zum Beispiel Wolken, und die intrinsische Unsicherheit, die durch die chaotische Evolution der Atmosphäre verursacht wird. In dieser Arbeit werden vier Studien präsentiert, die diese zwei Probleme mit statistischen Methoden und maschinellem Lernen in Angriff nehmen.

Stochastische Parametrisierungen wurden entwickelt, um die Kopplung von aufgelösten und nicht aufgelösten Prozessen zu verbessern und die Vorhersageunsicherheit präziser zu beschreiben. In der ersten Studie dieser Arbeit wird eine quantitative Theorie für zufällige Fluktuationen in einem konvektiven Wolkenfeld untersucht. Ursprünglich wurde diese Theorie für idealisierte Situationen entwickelt, in denen ein Gleichgewicht zwischen Strahlungskühlung und Konvektion besteht. In der hier vorgestellten Studie, wird diese Theorie zum ersten Mal in realistischen Wettersituationen getestet. Dazu wurden wolkenauflösende Simulationen eingesetzt, bei denen die synoptische Wetterlage identisch ist, aber die einzelnen Wolken zufällig verteilt sind. Insgesamt stimmen die Ergebnisse mit den theoretischen Vorhersagen überein. Es wurden aber auch Abweichungen festgestellt, die mit der der Organisation von Wolken zusammenhängen.

In der zweiten und dritten Studie dieser Arbeit wurde untersucht, ob traditionelle Parametrisierungen von kleinskaligen Prozessen durch ein künstliches neuronales Netz ersetzt werden könnten. Das neuronale Netz lernt von einem hochaufgelösten Datensatz, alle kleinskaligen atmosphärischen Prozesse zu modellieren. Es wird gezeigt, dass das neuronale Netz tatsächlich in der Lage ist, das komplexe Verhalten von Wolken und Strahlung zu repräsentieren. Folglich wurde die neuronale Netz-Parametrisierung in ein Klimamodell eingebaut und in mehrjährigen Simulationen getestet. Die Simulationen sind stabil und reproduzieren bei deutlich geringerem Rechenaufwand die wichtigsten Merkmal der hochaufgelösten Referenzsimulation.

Jedes Wetter- und Klimamodel hat systematische Fehler, die statistisch korrigiert werden müssen. In der letzten Studie wird ein neuronales Netz benutzt um probabilistische Temperaturvorhersagen zu kalibrieren. Das neuronale Netz lernt dabei spezifische Informationen für jede Wetterstation. Diese Methode erzielt bessere Ergebnisse im Vergleich zu bisher gängigen Methoden und kann einfach auf andere Problemstellungen angewandt werden.

Die vier Studien in dieser Arbeit zeigen wie statistische Methoden und maschinelles Lernen in der Wetter- und Klimavorhersage eingesetzt werden können. Immerfort zunehmende Rechenleistungen und Datenmengen, in Kombination mit Fortschritten in der künstlichen Intelligenz, versprechen großes Potenzial für zukünftige Forschung.

Abstract

Predicting future weather or climates requires modeling a hugely complex and chaotic system, the atmosphere. Remarkable progress has been made over the last decades but stubborn errors remain. Two main contributors to these errors are the parameterization of subgrid processes such as clouds and the uncertainty that comes from the chaotic evolution of the atmosphere. In this thesis, four studies are presented that tackle these issues with statistical and machine learning approaches.

Stochastic parameterizations have been proposed to better represent the coupling of resolved and subgrid processes as well as the forecast uncertainty. The first study in this thesis investigates a quantitative theory for the random fluctuations of a convective cloud field. Originally, the theory has been designed for idealized radiative-convective equilibrium situations. Here, it is tested, for the first time, in real weather situations. To achieve this cloud-resolving simulations are created that differ in their realization of the individual clouds but have the same large scale flow. Overall, the main assumption of the theory hold to good approximation. However, deviations were detected related to the organization of clouds which is not included in the original theory.

In the second and third paper, an approach to replace traditional subgrid parameterizations with a deep neural network is explored. The neural network learns to represent all atmospheric subgrid processes from a high-resolution simulation. It is shown that the neural network is indeed capable of capturing the complex behavior of clouds and radiation. Subsequently, the the trained deep learning parameterization is implemented in a climate model and run prognostically to create a multi-year simulation. The simulations are stable and reproduce the key features of the high-resolution simulations while being significantly faster.

Every weather and climate model exhibits systematic errors that need to be corrected statistically. In the last paper, the problem of calibrating probabilistic temperature forecasts is approached with modern machine learning techniques. In particular, a neural network that is able to learn specific information for each measurement station is used. This method outperforms previous state-of-the-art techniques and is easily adaptable to a range of problems in postprocessing.

The studies presented in this thesis show ways of using statistical and machine learning approaches in the process of creating weather or climate predictions. Ever increasing amounts of computing power and data in combination with advances in deep learning make this a promising field for future research.

List of publications for this cumulative dissertation

Includes three first-author and one co-author paper. Abbreviations P1–4 are used throughout the Introduction and Conclusion.

P1: Stephan Rasp, Tobias Selz and George C. Craig, 2018. Variability and Clustering of Midlatitude Summertime Convection: Testing the Craig and Cohen Theory in a Convection-Permitting Ensemble with Stochastic Boundary Layer Perturbations. *Journal of the Atmospheric Sciences*, 75(2), 691-706.

P2: Pierre Gentine, Michael S. Pritchard, Stephan Rasp, Gael Reinaudi and Galen Yacalis, 2018. Could machine learning break the convection parameterization deadlock? *Geophysical Research Letters*, 45, 5742–5751.

P3: Stephan Rasp, Michael S. Pritchard and Pierre Gentine, 2018. Deep learning to represent sub-grid processes in climate models. *Proceedings of the National Academy of Sciences*, 115(39), 9684-9689.

P4: Stephan Rasp and Sebastian Lerch, 2018. Neural networks for post-processing ensemble weather forecasts. *Monthly Weather Review*, 146(11), 3885-3900.

Contents

1 Introduction

Predicting the evolution of the atmosphere is an immense task. To start with, it requires modeling a wide variety of physical processes, ranging from the molecular to the planetary scale, that all are crucial to the evolution of the atmosphere (Fig. 1.1). But even if all these processes were modelled perfectly—far surpassing our current knowledge and computational capabilities—the chaotic nature of the atmosphere would still place a hard, physical limit on the predictability of weather. In light of these challenges, our current ability to predict weather and climate is even more impressive. Weather forecasts have been steadily improving by about one forecast-day per decade with no signs of slowing down (*Bauer et al.*, 2015). Climate model predictions are now routinely used to inform policy decisions, best exemplified by the work of the Intergovernmental Panel on Climate Change (IPCC; *Stocker et al.* (2013)) and the 2015 Paris agreement (*UNFCCC*, 2015). These achievements are testimony to the great minds that have been working on atmospheric modeling since its inception in the 1950s (*Charney et al.*, 1950).

Yet despite all the progress, some stubborn issues remain which limit the usefulness of today's weather and climate predictions. Weather forecasts occasionally suffer from "forecast busts"—situations where the forecasts skill across all models drops to near zero (*Rodwell et al.*, 2013). Furthermore, certain weather situations, such as summertime thunderstorms, continue to cause severe problems for all atmospheric models (*Cintineo and Stensrud*, 2013). In climate prediction, models still widely disagree on how much the Earth will warm in response to rising greenhouse gas emissions, particularly on regional scales (*Stocker et al.*, 2013; *Schneider et al.*, 2017a). These uncertainties limit the ability of end users, such as national weather services or regional planners, to make decisions based on model forecasts.

The theme of this thesis is to tackle two issues that contribute heavily to the problems in current weather and climate models: first, the representation of highly nonlinear processes that occur below the model grid scales, in particular clouds. And second, how to make useful predictions in face of the chaotic nature of the atmosphere which causes even tiny initial errors to corrupt each forecast rapidly. In particular, the papers that make up this thesis use statistical techniques and machine learning to confront these challenges. The four papers along with their abbreviations are listed in the beginning of this thesis.

The papers span a range of topics in weather and climate modeling. The goal of this introduction, therefore, is to provide context by giving a general overview of atmospheric modeling with a focus on the problems tackled in the papers. The aim is not to provide an exhaustive

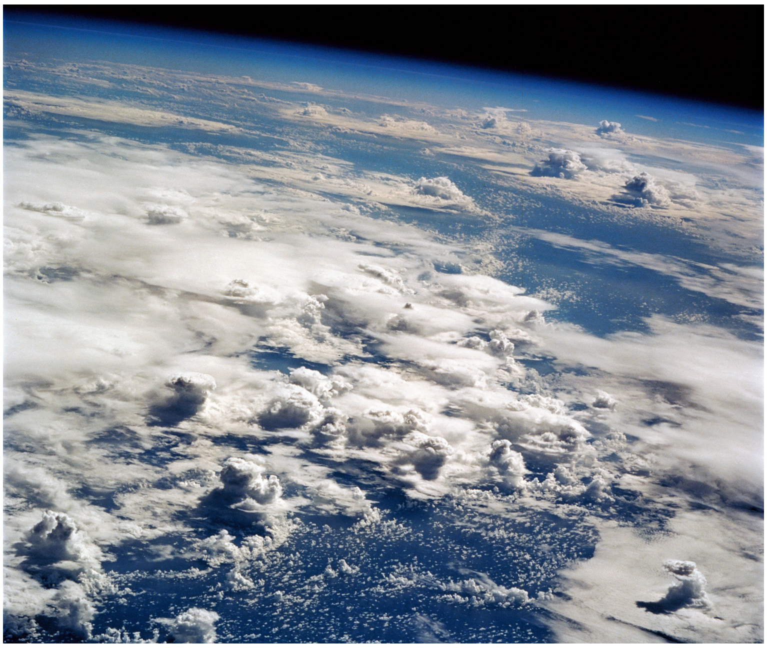


Fig. 1.1: Clouds over the Pacific Ocean as seen from Space Shuttle Discovery in 1994. This picture exemplifies the complexity of physical processes in the atmosphere. In the foreground, small shallow cumulus are visible in front of several thunderstorm towers. Thin "sheets" of ice clouds can be seen spreading out from some of the convective towers. Shadows produced by the clouds illustrate the interaction between convection and radiation. Source: NASA [1]

review of atmospheric modeling but rather to build an essential framework for readers not familiar with the key issues and jargon of atmospheric science. In the following sections, I will highlight how each topic is relevant to P1–4. Because three of the papers, P2–4, use neural networks, I will close this introduction by explaining the basic concepts of machine learning with a focus on neural networks and deep learning.

Before we start, it is useful to briefly discuss the difference between weather and climate. The term "weather" describes the short term condition of the atmosphere [2]. Weather is what we experience on a day-to-day basis and can change within minutes. "Climate", on the other hand, describes a long-term average of weather at a given location. Usually, several decades are taken for the averaging. Climate change, therefore, is a statistical deviation over long time scales. While both, weather and climate, are based on the same underlying physical system, the atmosphere, they differ in the questions we ask. This then dictates differences in modeling strategies for the two problems. The first major difference lies in time and length scales of the forecasts. Weather models typically produce forecasts for up to two weeks, whereas climate models are routinely run for 100 years or more. Because computational resources are limited, this is reflected in the resolution of weather and climate models (more in Section 1.1). The second major difference is the type of problem we are dealing with. Weather forecasting is essentially an initial condition problem that requires

[1] https://spaceflight.nasa.gov/gallery/images/shuttle/sts-64/html/sts064-83-099.html

[2] https://www.mpimet.mpg.de/en/communication/climate-faq/whats-the-difference-between-weather-and-climate/, accessed on 12 Oct 2018

a detailed knowledge about the current state of the atmosphere. For climate, the state of the atmosphere at a certain point in time is much less important. Rather we are interested in statistical drifts due to external changes to the system, for example greenhouse gas emissions. For this reason, climate modeling is mostly a boundary condition problem. In this thesis, I will cover both interchangeably as "atmospheric modeling" but highlight major differences where they appear.

1.1 The forecast chain

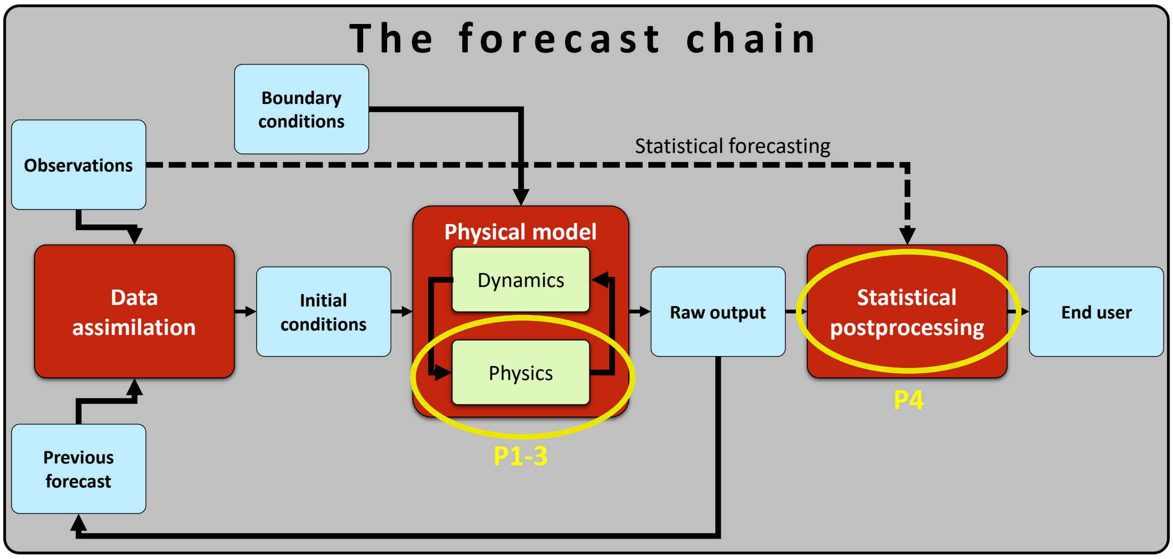


Fig. 1.2: The forecast chain. A schematic depiction of all components necessary to produce a weather or climate forecast for an end user. The blue fields indicate data while the red fields indicate methods. Yellow ellipses show where the papers in this thesis fit in.

The basic framework of this introduction is given by the "forecast chain" (Fig. 1.2) that outlines all the key steps necessary to produce an atmospheric forecast for an end user. The core of each modern forecasting system is the physical model of the atmosphere, which is the focus of P1–3 and will be described further below. But such a model also requires initial conditions, the current (or past) state of the atmosphere defined at each model grid point. Obtaining these initial conditions is not trivial because the problem is grossly underdetermined. Observations of direct model quantities like temperature are sparse in time and space, only available from surface stations, irregular weather balloon ascents and along aircraft flight tracks. Satellites provide global coverage in good temporal and spatial resolution but only take indirect measurements. To solve this problem, observations are combined with previous model forecasts in a process called “data assimilation” (*Kalnay*, 2003). Good initial conditions are essential for weather forecasts. For this reason, at most operational modeling centers nowadays more people are working in this area than on the development of the physical model. For climate predictions, it is also important to specify boundary conditions, for example greenhouse gas emissions. Because predicting these contains a lot of uncertainty, models are usually run with several emission pathways to obtain a range of solutions (*Stocker et al.*, 2013). The issue of obtaining good initial and boundary conditions is a key challenge in atmospheric modeling. In the papers presented here, however, data assimilation only plays a secondary role. P1–3 use idealized setups, while P4 uses data from an operational forecasting system that includes state-of-the-art data assimilation methods. Having obtained initial and boundary conditions, it is possible to run the atmospheric model and produce a forecast. However, the raw model output is rarely used directly by end users because it often has systematic errors. For this reason a whole field of research deals with the postprocessing of atmospheric forecasts. This is the topic of P4 and will be covered further below.

One note of caution: In this introduction and the four papers, the focus is entirely on the atmosphere, thereby ignoring the two other essential components of the Earth system, the ocean and the land. These are particularly important for climate predictions. In the Conclusion, I will discuss the applicability of the work presented here to these other components.

1.1.1 Physical model of the atmosphere

The physical model of the atmosphere is the main component of every forecasting system. Such a model tries to approximate the physical processes governing the evolution of the atmospheric state as accurately as possible. The model is made up of two core components, called, somewhat arbitrarily, “dynamics” and “physics” in atmospheric science jargon. The dynamics solve the Navier-Stokes equations, which govern the flow of air, on a three-dimensional grid using finite differencing schemes (*Holton*, 1973; *Durran*, 2010). It can be viewed as a function that maps the model state from one time step to the next:

$$\mathbf{x}_{t+1} = \mathcal{D}(\mathbf{x}_t). \tag{1.1}$$

where $\mathbf{x}$ is the model state vector that contains all model variables, e.g. temperature, winds and pressure, at every model grid point. Because the atmosphere has an exceedingly non-uniform aspect ratio, much wider than tall, the horizontal grid spacing Δx is several times the vertical grid spacing Δz. The discrete time step Δt is coupled to the spatial grid by the Courant–Friedrichs–Lewy (CFL) condition (*Courant et al.*, 1928):

$$\frac{w\Delta t}{\Delta z} + \frac{u\Delta t}{\Delta x} \leq C_{\text{max}}, \tag{1.2}$$

Model	Typical Δx	Number of vertical levels	Vertical model top	Typical Δt	References
Large eddy simulation	25–500 m	150	10–20 km	1–10 s	*Heinze et al.* (2016)
Weather (regional)	1–4 km	50	20 km	10 s–1 min	*Baldauf et al.* (2011)
Weather (global)	10–20 km	100-150	50–80 km	1–5 min	ECMWF[3]
Climate	25–200 km	100	80 km	5–30 min	*Stocker et al.* (2013), *Stevens et al.* (2013a)

Table 1.1: Typical grid dimensions and time steps for current weather and climate models. Note that vertical grids usually are fine near the surface, where vertical gradients are large and become coarser with height.

where C_{max} must be less than a certain value (that depends on the time stepping scheme) to ensure a stable time integration. Note that fast waves are typically handled by a separate integration scheme. The coupling of the time step to the grid spacing means that doubling the horizontal resolution increases the computational cost by a factor of 2^4. Because computational resources are limited, a trade-off between integration time, the geographical model extent and the model resolution has to be made. Typical resolutions for different kinds of models are listed in Table 1.1.

Many physical processes occur on scales smaller than the grid spacings of current atmospheric models, for example turbulent mixing, radiative heating and cooling and most cloud processes. These subgrid processes, however, are crucially important to the evolution of the atmosphere and their effect on the resolved scales has to be approximated. These approximations are called “parameterization”. The total set of all parameterizations is called the “physics” of an atmospheric model. A parameterization $\mathcal{P}$ predicts the effect of a subgrid process on the resolved scales $\Delta \mathbf{x}_{\text{sg}}$ as a function of the resolved state and parameterization specific parameters θ that can include tuning parameters or external forcings:

$$\Delta \mathbf{x}_{\text{sg}} = \mathcal{P}(\mathbf{x}, \theta)\Delta t \tag{1.3}$$

The total model then is a combination of the resolved advection $\mathcal{D}$ and the subgrid physics $\mathcal{P}$:

$$\mathbf{x}_{t+1} = \mathcal{D}(\mathbf{x}_t + \Delta \mathbf{x}_{\text{sg}}) \text{ or } \mathcal{D}(\mathbf{x}_t) + \Delta \mathbf{x}_{\text{sg}}. \tag{1.4}$$

The order of dynamics and physics differs from model to model.

Finding good approximations of subgrid processes turns out to be very difficult. For this reason, parameterizations are the major source of uncertainty in today’s atmospheric models

[3] `https://www.ecmwf.int/en/forecasts/documentation-and-support`, accessed on 12 Oct 2018

(*Stevens et al.*, 2013b; *Schneider et al.*, 2017a). They are the topic of P1–3 and will be covered in more detail in Section 1.2.

1.1.2 Statistical postprocessing

Statistical postprocessing is necessary for two main reasons: first, all models have systematic errors. A weather model might, for example, tend to be too warm in high pressure situations. Second, the quantity of interest might not be directly available from the raw model output but rather be a derived quantity. A wind farm provider is interested in forecasting the power output which is not a model state variable but strongly correlates with wind speed. Nomenclature can be confusing for these tasks. For this thesis, we will describe the first task as "calibration" and the second task as "statistical forecasting". Note that statistical forecasting need not include a forecast from a physical model. Instead, one can try to produce a statistical forecast directly from recently observed quantities, or anything else for that matter. It turns out that such a non-dynamical approach can be helpful for short-term forecasts, up to a few hours. For longer forecasts, however, the information from a dynamical model usually outperforms purely statistical techniques. There is a whole zoo of postprocessing techniques, specific for the requirements of each end user. This is not restricted to weather forecasts either. Climate predictions can also benefit from bias correction or a weighting of different models. One perhaps surprising finding is that the gains from postprocessing numerical weather forecasts remain constant even though the skill of the raw models continuously increase (*Hemri et al.*, 2014). The exact reasons for this are unclear but this result could indicate that improvements in the raw modeling system over time and the improvements from postprocessing are different in nature.

From a statistical point of view, a postprocessing model relates some input variables, also called predictors, $\mathbf{x}$ to the desired output $\hat{y}$ [4]. For a calibration task, $\mathbf{x}$ includes the raw model equivalent of $\hat{y}$ but may also include other "auxiliary" predictors. One could image, for example, that information about cloud cover might be useful for calibrating temperature forecasts. In almost all cases, postprocessing models learn from past forecast-observation pairs. The simplest case of postprocessing is a simple bias correction $\hat{y} = x + b$, where the bias can be estimated by comparing the past forecasts with the corresponding observations y: $b = N^{-1} \sum_i (y - x)$, where N is the number of past forecast-observation pairs. Such a forecast is then bias free, and in a statistical sense reliable, i.e. the forecast distribution is consistent with the corresponding observations (*Wilks*, 2006). A slightly more elaborate approach is to use a linear regression approach: $\hat{y} = \mathbf{a}\mathbf{x} + b$, which also allows the use of auxiliary predictors. This linear regression method is widely used operationally under the

[4] Note that $\mathbf{x}$ takes on different meanings depending on the context in this thesis. Inside the atmospheric model $\mathbf{x}$ is the model state vector. For postprocessing and machine learning $\mathbf{x}$ describes the input vector to some algorithm.

term Model Output Statistics (MOS). We will return to the more complex case of probabilistic postprocessing in Section 1.3 and P4. Postprocessing can also be viewed under the wider umbrella of machine learning which is covered in Section 1.5.

1.2 Physical processes and their parameterization

As previously mentioned, the representation of subgrid processes is a key component in every atmospheric model. Now we will look more closely at the most important atmospheric processes: turbulence, clouds and radiation. Most models also contain parameterizations for other processes, such as orographic gravity wave drag or chemistry. These will not be covered here since they are not discussed in any of the papers. The following discussion mainly follows *Stensrud* (2007), unless otherwise noted. Fig. 1.3 schematically depicts the three key processes along with their length scales and interactions between the processes. The length scales are important to appreciate how models with different grid resolutions (see Table 1.1) represent these processes.

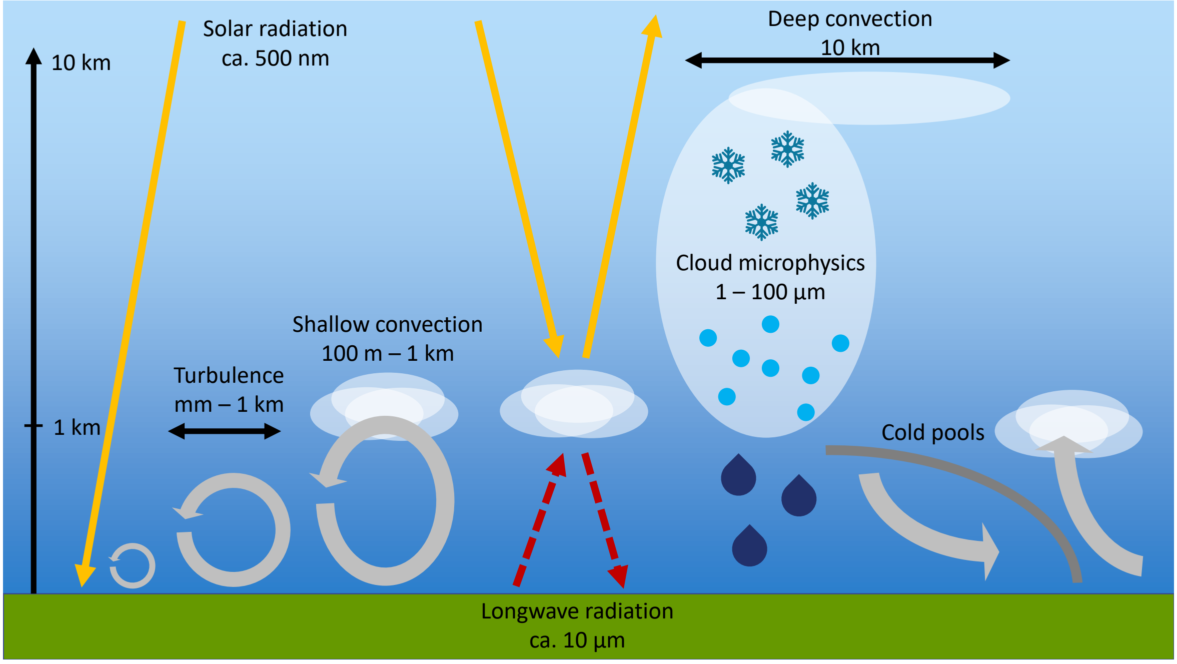


Fig. 1.3: Physical processes in the atmosphere and their interaction. Turbulence at the surface is driven by solar radiation. The turbulence in turn triggers moist convection. Clouds can absorb, emit and reflect radiation. Clouds can also impact the boundary layer structure directly by cold pool dynamics, which can lead to a self-organization of convective systems.

In general, a parameterization $\mathcal{P}$ represents the effect of a subgrid process on the grid scale. It takes as input the resolved model state $\mathbf{x}$ or a subset thereof, makes assumptions about the subgrid process and returns the grid scale tendency caused by the subgrid process $\Delta\mathbf{x}_{\text{sg}}$:

$$\Delta\mathbf{x}_{\text{sg}} = \mathcal{P}(\mathbf{x}, \theta)\Delta t \tag{1.5}$$

Typically, the assumptions and approximations of a parameterization will have several uncertain parameters θ and might also take external boundary conditions as input, such as the incoming solar radiation. The challenge of parameterization design then is to make approximations of the subgrid processes that best represent the real behavior of the atmosphere.

Traditionally, parameterization design is a mostly heuristic exercise. Knowledge about a process is gained through observations, high-resolution modeling and theoretical arguments. It is then the task of atmospheric physicists to write down approximate equations for the grid-scale mean behavior of the process. Finally, the free parameters θ are tuned to best match observations or high-resolution models. Some parameters are well constrained by observations and theory, while others have a large range of potential values. This tuning stage is usually still done manually (*Mauritsen et al.*, 2012; *Voosen*, 2016; *Hourdin et al.*, 2017). The development of automated tuning approaches is complicated by the amount of free parameters for each parameterization (typically 10–50), the nonlinearity of the processes and the large computational cost of running the full model for statistically significant periods of time (*Zhang et al.*, 2015). Experience has shown that there is no single best way to parameterize a given process. Reducing the chaotic complexity of the atmosphere to a reasonably sized set of equations—parameterization need to be significantly faster than high-resolution simulations after all—has vexed scientists every since the beginning of atmospheric model development (*Randall et al.*, 2003). For this reason there is a wide variety of parameterization approaches currently used in atmospheric models.

Some common assumptions, however, are shared by most models. First, the individual processes—turbulence, clouds and radiation in our case—are treated separately, i.e. they do not interact within a time step. Fig. 1.3 tells us that this might not necessarily be a very good assumption. We will return to this later in this section. Another common assumption shared by most parameterizations is that horizontal interactions of subgrid processes between grid columns can be ignored. In other words, each parameterization acts independently on each column. Again, we will later explore how this assumption might break down for smaller grid spacings.

The goal of this section is to describe the three key atmospheric processes and the most common parameterization approaches used for them. P1–3 deal with the problem of subgrid parameterization directly, testing novel approaches. For this reason, a basic overview of the fundamentals of current parameterizations should prove helpful.

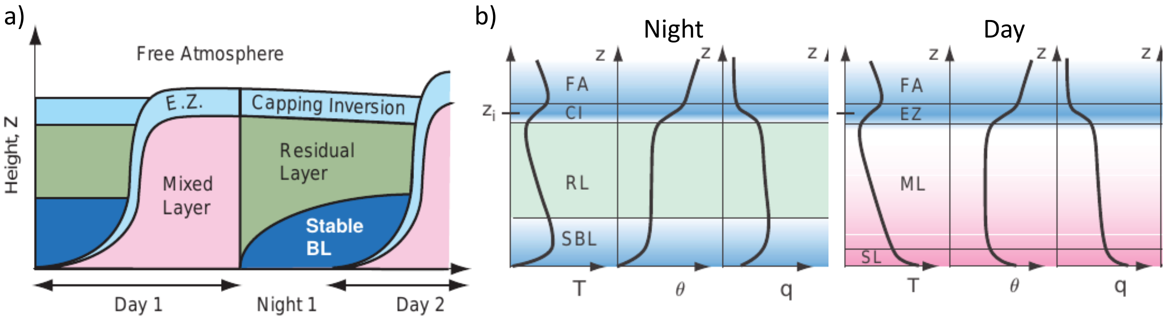


Fig. 1.4: a) Typical evolution of the planetary boundary layer in summer over land. E.Z. stands for entrainment zone. The x-axis starts at sunrise. From *Wallace and Hobbs* (2006)(p. 398). b) Profiles of temperature, potential temperature and humidity during night and day. From *Wallace and Hobbs* (2006)(p. 392).

1.2.1 The planetary boundary layer and turbulence

The planetary boundary layer is defined by its interaction with the Earth's surface. The interaction happens through the exchange of surface fluxes, both sensible and latent, and drag exerted by the surface. The depth of the boundary layer can vary between tens of meters m to several kilometers (*Stull*, 1988). It is instructive to consider the diurnal evolution of the boundary layer on a typical summer-day over land (Fig. 1.4). Taking sunrise as a starting point, the area directly above the ground is stable, indicated by an increase of the potential temperature with height.[5] As the sun heats the ground, the air directly in contact with the surface will become warmer and eventually unstable. This causes the formation of turbulent eddies, also called thermals, that mix the air vertically. The mixed layer grows throughout the day until the the inversion layer is reached, which is characterized by a sharp increase in static stability, i.e. a large gradient in potential temperature. The largest eddies inside the mixed layer span the entire depth and persist for around 10 minutes. Potential temperature and humidity are constant in the mixed layer. As eddies hit the inversion, they overshoot and entrain free atmospheric air into the boundary layer. Once the sun sets, turbulence stops. The air in contact with the surface cools because of longwave radiation. The residual layer is the remainder of the mixed layer and still exhibits near-constant potential temperature and humidity.

This is just one archetypal boundary layer structure. Maritime boundary layers behave very

[5] Potential temperature θ is defined as the temperature an air parcel would have if it were adiabatically brought to a reference pressure p_0 =1000 hPa:

$$\theta = T \left(\frac{p_0}{p} \right)^{\frac{R}{c_p}} \tag{1.6}$$

where R is the specific gas constant and c_p is the specific heat capacity under constant pressure for dry air.

differently because diurnal temperature differences are significantly smaller. Furthermore, the boundary layer can be driven by synoptic winds, where shear turbulence dominates convective turbulence. Lastly, the presence of clouds significantly influences the boundary layer. We will explore this further below.

In most weather and climate models, boundary layer turbulence happens below the grid scale and needs to be parameterized. The basic problem of modeling boundary layer turbulence is finding an expression turbulent subgrid contribution to the grid scale evolution. In the simplest case we can consider the vertical advection of a scalar ϕ:

$$\frac{\partial \phi}{\partial t} = -w \frac{\partial \phi}{\partial z} \tag{1.7}$$

where w is the vertical velocity. It is common to ignore the horizontal terms because gradients in the vertical are significantly larger and the vertical grid spacing is smaller. We can now split each variable (w and ϕ) into a mean and a perturbation component, e.g. $\phi = \bar{\phi} + \phi'$ and then average the resulting equation to obtain the temporal evolution of the mean component. In this procedure, called Reynolds averaging, the averaged products of mean and perturbation components are zero. However, the average of two perturbation components is not:

$$\frac{\partial \bar{\phi}}{\partial t} + \bar{w} \frac{\partial \bar{\phi}}{\partial z} = -w' \frac{\partial \overline{w'\phi'}}{\partial z}. \tag{1.8}$$

The left hand side represents the advection by the mean wind. The right hand side represents the turbulent advection. $\overline{w'\phi'}$ is called the Reynolds stress or subgrid flux and cannot be predicted explicitly from the mean variables. One can find a prognostic equation for the Reynolds stress terms but these then contain triple correlation terms. The system of equation is therefore unclosed. To predict the turbulent evolution, a closure assumption has to be used to relate the subgrid flux terms to known mean quantities.

There are two basic approaches for closing the turbulence equations: local and non-local. Local approaches only consider the immediate surrounding of each grid point to estimate its subgrid flux. The simplest approach is K-theory which relates the subgrid flux to the local gradient of the variable in question:

$$\overline{w'\phi'} = -K \frac{\partial \overline{\phi}}{\partial z} \tag{1.9}$$

K can be a constant or, more commonly, a function of the vertical wind shear. If K-theory is used to find an expression for the double correlation terms, as in the equation above, the scheme is called a 1st-order scheme. If K-theory is used on the triple correlation terms, the scheme is called a 2nd-order scheme (*Mellor and Yamada*, 1974). In P1, the stochastic boundary layer perturbation scheme (*Kober and Craig*, 2016) uses information from a 1.5-order scheme. This term is used for schemes that only have expressions for some higher-order moments, in this case the turbulence kinetic energy.

Non-local approaches are motivated by the observation that the most energetic eddies often span the entire vertical extent of the boundary layer. This means that transport can occur over several grid levels within a short amount of time. In these schemes, the sub-grid fluxes are directly related to the surface layer properties, such as the surface heat fluxes. There is a range of non-local approaches used in research. One commonly used hybrid approach one is the Eddy-Diffusivity Mass-Flux (EDMF) framework (*Neggers et al.*, 2009), that uses K-theory to model local mixing and a mass flux approach for non-local mixing. We will return to mass flux approaches when talking about cloud parameterizations.

Models with km-scale grid spacing nowadays typically favor a local turbulence parameterization approach, while global models usually have some non-local parameterization. The high vertical resolution and smaller time step in km-scale models allows for some explicit representation of boundary layer transport, even though such resolved turbulence is often not very realistic. Turbulent mixing also occurs outside the boundary layer, but is usually local. Nevertheless it is an important process that needs to be modeled. Local parameterizations can model turbulence throughout the atmosphere, whereas models with non-local parameterization require an additional turbulence parameterization for the free atmosphere.

1.2.2 Clouds

Clouds come in a wide variety of shapes and sizes. Fair weather cumulus clouds are several hundred meters wide and only around 100 m tall. Stratocumulus over the oceans are also thin but can cover several thousands of kilometers horizontally. Thunderstorm clouds, on the other hand, are tall towers that extend up to the tropopause but are confined in their horizontal extent. But clouds also organize into larger systems, from squall lines to tropical cyclones (*Houze*, 2004). Because of their diversity, modeling clouds is a difficult undertaking. Some cloud structures will be resolved by the model grid, other will not. For this reason it is useful to start with a basic, albeit somewhat arbitrary, taxonomy of clouds as they are represented in most models.

Clouds can be divided into stratiform and convective clouds, the latter of which can be further split into shallow and deep convective clouds (*Stevens*, 2005). Stratiform clouds stretch out over large regions horizontally. From a modeling point of view, this means that they can be represented explicitly, i.e. the clouds actually exist on the model grid. Only the microphysical processes, such as phase transitions and the growth of cloud droplets, need to be parameterized. Convective clouds are the result of localized buoyant updrafts. Shallow convection typically does not extend beyond 2 km in height and is often assumed to be non-precipitating. Deep convection, in contrast, can span the entire troposphere with heavy precipitation as a result. Individual convective clouds have a horizontal extent of 100 m to several km. In global models, therefore, they live below the grid scale and need to be parameterized. The job of a convection parameterization is to model the mean effect of the subgrid clouds on the resolved scales. As horizontal grid spacings go below 10 km, the distinction between subgrid and grid-scale convection becomes blurry. Experiments have shown that deep convection can be represented explicitly at horizontal grid spacings of 4 km and below (*Weisman et al.*, 1997). Clouds at this resolution are not realistic, however. Rather the success of explicit convection modeling at km-resolution is due to compensating errors. To achieve a realistic representation of clouds, particularly of shallow convection, grid spacings of a few hundred meters of below are required (*Craig and Dörnbrack*, 2008).

In the following discussion, I will focus on the process and the parameterization of deep convection, for two reasons: first, deep convection is the main topic of P1 and plays a central role in P2 and P3; and second, the parameterization of deep convection is a major stumbling block for many current atmospheric models (*Stevens et al.*, 2013a). Note, however, that the representation of shallow clouds, which is typically separate from the parameterization of deep convection, is probably equally as important because of their interaction with radiation (*Bony et al.*, 2015).

A good starting point to discuss atmospheric convection is to consider the adiabatic ascent of an air parcel through the atmosphere (Fig. 1.5). Starting close to the surface, the buoyant parcel follows a dry adiabat, along which the potential temperature θ is conserved. At the inversion layer, where the environment is stably stratified, the parcel will become negatively buoyant. At some point, called the lifting condensation level (LCL), the parcel becomes saturated. As a result of the latent heat released from condensation (and later freezing), the parcel now follows a moist adiabat. Depending on the environmental temperature profile and the starting conditions of the parcel, the parcel may become warmer than the environment and can rise freely—the level at which this happens is called the lifting condensation level (LFC). The free ascent then continues until the parcel and environmental temperatures become equal again at the level of neutral buoyancy (LNB).

The buoyancy force B can formally be defined as

$$B = g\left(\frac{\theta'}{\theta}\right) \tag{1.10}$$

where θ' is the temperature perturbation of a parcel $\theta_{\text{parcel}} - \theta_{\text{env}}$. From this two important quantities can be defined from this parcel view: the convective available potential temperature (CAPE) and the convective inhibition (CIN). CAPE represents the maximum energy available to the parcel throughout its ascent:

$$\text{CAPE} = g\int_{\text{LFC}}^{\text{LNB}} \frac{\theta_{\text{parcel}} - \theta_{\text{env}}}{\theta_{\text{env}}} dz \tag{1.11}$$

Conversely, CIN is the energy a parcel has to overcome from its starting position z_0 before it reaches the LFC:

$$\text{CIN} = -g\int_{z_0}^{\text{LFC}} \frac{\theta_{\text{parcel}} - \theta_{\text{env}}}{\theta_{\text{env}}} dz \tag{1.12}$$

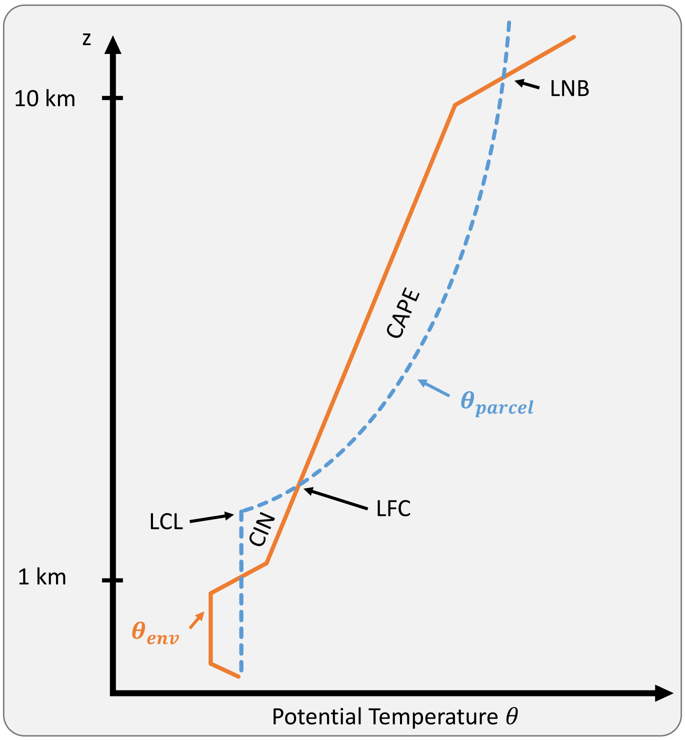


Fig. 1.5: Schematic depiction of parcel ascent. The parcel (in blue, starting at the surface) starts rising adiabatically until the lifting condensation level (LCL) is reached. From here on the parcel follows a moist adiabat. Once the parcel temperature is higher than the environmental temperature (orange), the parcel can rise freely from the lifting condensation level (LCL) to the level of neutral buoyancy (LNB).

Because the LCL and therefore also the LFC depend on the saturation of the parcel, CAPE and CIN are function of the initial parcel humidity as well as the temperature.

Convection can only occur if CAPE > 0 J kg^{-1}. However, not all potential energy will also be realized during the ascent because the parcel will mix with the environmental air, a process called entrainment, and lose some buoyancy. But even if CAPE is large, convection might not occur if CIN is too large to be overcome. There are two basic convective regimes: equilibrium and non-equilibrium convection (*Done et al.*, 2006). CAPE is created by the warming and moistening of the surface and a cooling of the free troposphere by longwave radiation. Eventually, these processes will result in an unstable stratification. The net effect of deep convection on the environment is a warming and drying. The warming is initially caused by the release of latent heat which then quickly spreads through gravity waves (*Bretherton and Smolarkiewicz*, 1989). Deep convection can, therefore, be viewed as a mechanism that restores a stable stratification and thereby destroys CAPE. Without external forcings the creation and destruction of CAPE reach an equilibrium, called the radiative-convective equilibrium (RCE). Many theories of deep convection are based on this equilibrium assumption. The earliest convection parameterizations, for example, were convective adjustment schemes that simply relaxed the temperature profile to a statically neutral sounding (*Manabe et al.*, 1965). Perhaps surprisingly, given their complete disregard for any dynamical processes inside the clouds, these schemes and their subsequent refinements provide a solid representation of the behavior of the atmosphere (*Arakawa and Schubert*, 1974).

But such an equilibrium does not always occur. If CIN is large, large CAPE values can build up which are then released suddenly rather than continuously. Such non-equilibrium convection is particularly common over land, where the surface properties are strongly modified by the diurnal cycle. In these non-equilibrium cases, the onset of deep convection depends on triggering processes that reduce and overcome CIN. One such triggering process is boundary layer turbulence, but large-scale forcings or topographic lifting can also act as triggers. To be able to represent such complex processes, most atmospheric models now have parameterizations that approximate the dynamical processes of convection itself.

The most prominent framework is the mass flux approach (*Tiedtke*, 1989), in which each grid box is conceptually divided into updrafts, downdrafts and environment. The updrafts and downdrafts then are modeled as plumes that start with a certain mass flux $M = \rho w$ which becomes modified by entrainment E, mixing of environmental air into the plume, and detrainment D, mixing of plume air out of the plume:

$$\frac{\partial M}{\partial z} = E - D \tag{1.13}$$

The challenge then is to find the initial mass flux at cloud base for the updrafts and the entrainment and detrainment coefficients. A so-called closure assumption us used to relate the cloud base mass flux to known quantities such as CAPE or the large scale moisture convergence. Finding good parameterizations for entrainment and detrainment rates is one of the key challenges in convection parameterization research today.

1.2.3 Radiation

Radiation is the main driver of the atmospheric circulation by heating the Earth's surface unequally, more a the equator and less at the poles. But radiation also interacts with molecules in the air, most famously the greenhouse gases and ozone (*Wallace and Hobbs*, 2006, p.113 ff.). The Earth's climate is highly sensitive to even small changes in the radiation budget, but radiation also acts on much shorter time and length scales. The differential heating of land and water or on mountain slopes, for example, can cause circulations that produce clouds within hours (*Lin*, 2007). An accurate representation of radiation is therefore crucial for both, weather and climate models.

Radiation can be divided into two components: shortwave radiation, emitted by the sun, and longwave radiation, emitted by the Earth and the atmosphere themselves. Since there is little overlap between the spectra and their interaction with matter differs, short and longwave radiation are treated separately in radiation parameterizations. The parameterization's job is to estimate the radiative flux F at each atmospheric grid point. This can then be used to compute the radiative heating:

$$\frac{\partial T}{\partial t} = \frac{1}{\rho c_p} \frac{\partial F}{\partial z} \tag{1.14}$$

As with turbulence and convection parameterizations, horizontal fluxes are typically ignored. In the commonly-used two-stream approximation, the fluxes are further divided into an upward and a downward flux. There are several processes that interact with the radiation as is travels through an atmospheric column. Molecules can absorb short and longwave radiation radiation—how much depends on the properties of the molecules and the wavelength—but also emit longwave blackbody radiation. For shortwave radiation, scattering also plays a crucial role. For clear sky conditions, radiation parameterization are very accurate albeit computationally expensive. Problems arise when clouds interact with radiation which we will come to shortly.

1.2.4 Process-interactions and shortcomings of current parameterization approaches

In the discussion above, we considered the three processes as independent. This is also how they are represented in most current models. Nature, however, cares little about our efforts to categorize the atmosphere into different components. There are a myriad of ways in which sub-grid processes can interact. One interaction we already touched upon is between boundary layer turbulence and convection. First, boundary layer thermals can act as triggers for deep convection because their vertical momentum can overcome the convective inhibition. Here it is important to realize that this triggering is due to the most intense boundary layer eddies, i.e. the tail of the distribution (*Gentine et al.*, 2013). In models that parameterize deep convection, the onset of precipitation is often predicted too early because the removal of CIN by boundary layer turbulence is not taken into account (*Couvreux et al.*, 2015; *Hohenegger et al.*, 2015). But the interaction between clouds and turbulence goes in both ways. Precipitating clouds create cold pools in the boundary layer, density currents caused by the evaporation of rain in a storm's downdraft. These cold pools spread out from existing clouds and have the potential to trigger new clouds, thereby giving clouds the chance to organize themselves (*Rotunno et al.*, 1988; *Tompkins*, 2001). Large convective systems, such as super-cells and squall lines, are one example of such self-organization. These systems can also propagate horizontally from grid column to grid column. Connected with this organization by cold pools is the notion of convective memory, which describes the idea that there are temporal correlations beyond the time-step that are stored in the sub-grid state. Again, most current parameterizations do not include cold pool-processes, which are crucial to represent global circulation patterns and climate change (*Bony et al.*, 2015; *Tan et al.*, 2015).

The second crucial sub-grid interaction is between clouds and radiation, and again it works in both directions. A glance at the shadows in Fig. 1.1 shows that clouds strongly modify how much radiation reaches the surface. Over land, this immediately leads to surface flux gradients that can give rise to boundary layer circulations, which in turn can impact the future cloud development (*Jakub and Mayer*, 2017). Furthermore, radiation can directly influence the microphysical processes in clouds (*Hartman and Harrington*, 2005). In global models, these interactions typically happen below the grid scale. In order for the radiation scheme to know about sub-grid clouds, a diagnostic or prognostic sub-grid cloud cover parameterization is usually introduced that links the model clouds to the radiation below the grid scale. However, most models do not allow for feedbacks between the processes since cloud and turbulence parameterizations know nothing about radiative effects.

The result of these shortcomings are biases that have plagued weather and climate models for decades. In global weather models with parameterized convection, for example, errors in the timing and intensity of deep convection have been a long-standing problem, which has only been alleviated recently (*Bechtold et al.*, 2014). In km-scale models, problems arise because the triggering processes, most notably boundary layer thermals, occur below the grid scale and cannot be appropriately represented (*Barthlott and Hoose*, 2015). The mean mixing effect of turbulence is captured by the boundary layer parameterization, but for convective initiation it is the variability that counts. This mismatch can lead to a delayed onset and underestimation of summertime convection. The physically-based stochastic perturbation scheme (PSP; *Kober and Craig* (2016)) that is used in P1 (described in the appendix) proposes one possible solution for this issue.

Climate models often struggle with the tropical circulation. Many climate models predict a "double intertropical convergence zone" (ITZC). The ITCZ, a band near the Equator where the trade winds meet, is characterized by intense deep convection (*Wallace and Hobbs* (2006); Fig. 1.6a). In nearly all climate models the ITCZ is too strong and a secondary maximum appears in the western Pacific (*Oueslati and Bellon*, 2015). Partly this bias is caused by atmosphere-ocean interactions, but the ITCZ has been shown to be very sensitive to the choice of convection parameterization. Another tropical dynamics bias is a missing Madden-Julian-Oscillation (*Madden and Julian* (1971); MJO; Fig. 1.6b). The MJO is a sub-seasonal (30-90 day) variability over the Indian Ocean and western Pacific, where a conglomeration of deep convection forms over the western Indian Ocean and then travels eastwards. The processes causing the MJO are not yet fully understood but are assumed to be strongly related to the interaction between moist convection and the large-scale dynamics (*Zhang*, 2005). In many current climate models, the MJO is too weak (*Peters et al.*, 2017; *Arnold and Randall*, 2015), a telltale sign that convection parameterizations fail to appropriately interact with the resolved flow.

The accumulation of errors in subgrid parameterization also causes uncertainty about how much the Earth will warm in response to an increase in greenhouse gases. A standard metric for all climate models is the equilibrium climate sensitivity (ECS); that is the mean increase in temperature in response to a doubling in CO_2. This is vividly illustrated in Fig. 1.7. The ECS in current generation climate models ranges from 2 to almost 5°C. This spread has

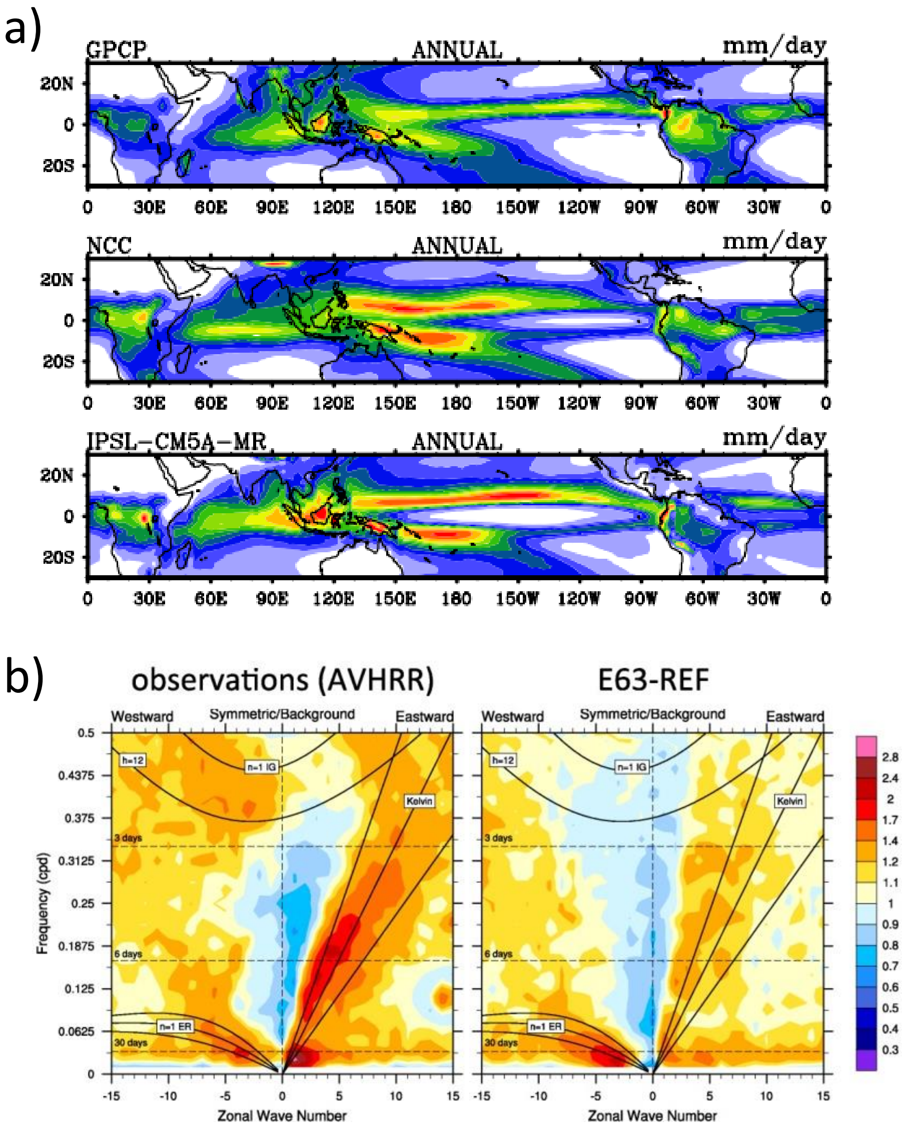


Fig. 1.6: a) Annual mean precipitation over the Indian and Pacific Oceans. Top panel (GPCP) indicates satellite observations. The other panels show two current climate models. From *Oueslati and Bellon* (2015). b) Wavenumber-frequency spectra of outgoing longwave radiation (which is approximately proportional to precipitation) in the tropics after *Wheeler and Kiladis* (1999). Figure shows the equatorially symmetric component divided by the background spectrum. Negative (positive) values denote westward (eastward) propagation. Left panel shows observations, the right panel shows a current climate model. The MJO shows up at $>$ 30 days and a zonal wave number of 1–2. From *Peters et al.* (2017).

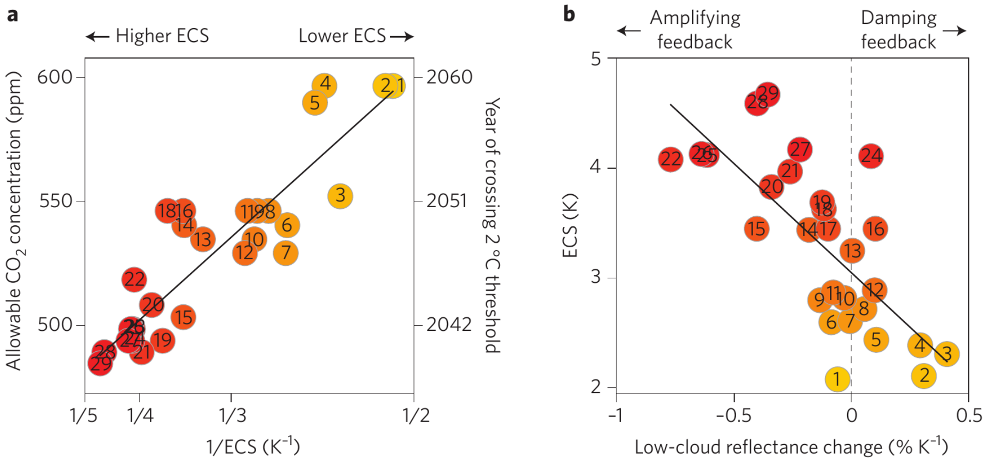


Fig. 1.7: a) Allowable CO_2 before the 2°warming threshold (and year of crossing the threshold in a high-emission scenario) is plotted against the equilibrium climate sensitivity (ECS) for 29 current climate models. b) ECS plotted against the change in the amount of sunlight reflected by low clouds over tropical oceans. From *Schneider et al.* (2017a).

remained remarkably consistent ever since the first inter-comparison of climate models in 1979 (*Charney et al.*, 1979). The majority of this uncertainty can be traced back to how low clouds are represented in the models (*Bony and Dufresne*, 2005; *Sherwood et al.*, 2014). Results like these make it more difficult to make policy decisions based on climate modeling.

Resolving deep convection explicitly improves many of the issues of coarse models (*Sun and Pritchard*, 2016; *Leutwyler et al.*, 2017; *Muller and Bony*, 2015). But problems still arise because the triggering processes, most notably boundary layer thermals, are smaller in scale and, therefore, not appropriately represented (*Barthlott and Hoose*, 2015). The mean mixing effect of turbulence is captured by the boundary layer parameterization, but for convective initiation it is the variability that counts. This mismatch can lead to a delayed onset and underestimation of summertime convection. The physically-based stochastic perturbation scheme (PSP; *Kober and Craig* (2016)) that is used in P1 (described in the appendix) proposes one possible solution for this issue.

1.2.5 Unified parameterizations and super-parameterization

In response to the problems that arise from the separation of sub-grid processes, the last decade has seen the development of a range of unified parameterizations, especially of turbulence and convection. So far, most of them are being tested purely in an academic environment. One popular approach is the aforementioned EDMF approach that splits vertical sub-grid transport into a local component and convective plumes. Recently, EDMF has been

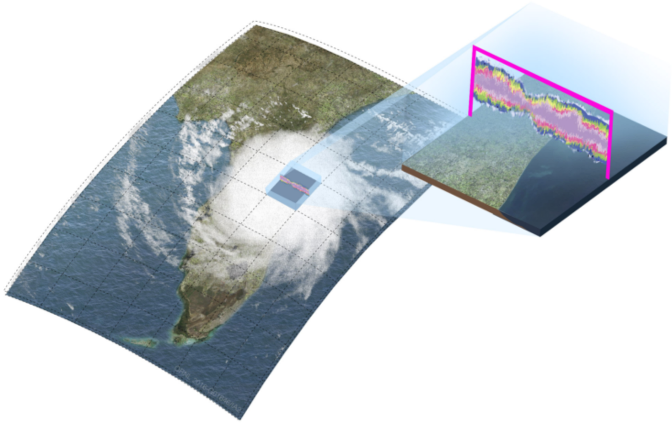


Fig. 1.8: Schematic representation of super-parameterizations. A two-dimensional cloud-resolving model is embedded in each GCM grid column. [6]

extended to include deep convection (*Tan et al.*, 2018). One problem with unified parameterizations is that the ways in which processes can interact have to be specified a priori. As we have seen above, however, the interactions can be complex and might possible go beyond the framework of a given unified parameterization. These limitations are a primary motivator for the more data-driven approach that is explored in P2–3.

Another approach of modeling subgrid processes is super-parameterization (SP; *Khairoutdinov et al.* (2005)). In SP, every global circulation model (GCM) grid-column contains a 2D higher-resolution CRM (Fig. 1.8). This allows for the explicit representation of clouds at reduced computational cost compared to a global CRM. The computational speed-up is due to the reduced dimensionality and the lack of communications between the CRMs which allows for efficient parallelization. In P2–3, we used SP to resolve deep convection. This approach has several advantages over implicit cloud parameterizations. In particular, the interaction of convection with the large-scale circulation is improved, alleviating some of the biases discussed above (*Benedict and Randall*, 2009; *Arnold and Randall*, 2015; *Kooperman et al.*, 2018). However, there are also downsides. First, the 2D nature of SP makes it difficult to model momentum transport by convection (*Woelfle et al.*, 2018; *Tulich*, 2015). Second, the lack of horizontal information exchange prohibits the propagation of mesoscale convective systems.

1.3 Chaos and ensembles

Even if all physical processes were modeled perfectly, forecasting future states of the atmosphere would still remain a challenge. The reason is the chaotic nature of the atmosphere that places a hard limit on how far into the future we can make useful predictions. In this

[6] `http://hannahlab.org/what-is-super-parameterization/`

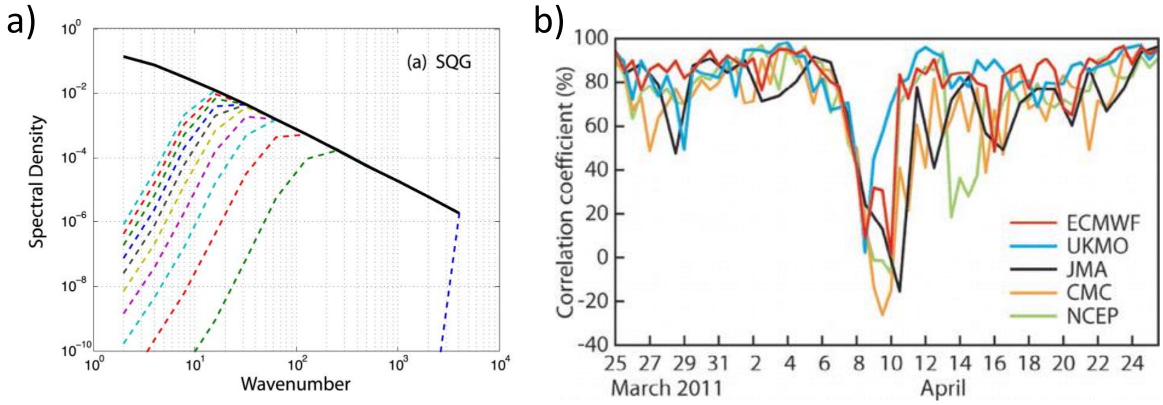


Fig. 1.9: a) Evolution of the perturbation kinetic energy spectrum for two simulations with the *Lorenz* (1969) model with tiny initial condition differences. The blue, dashed line on the left represents the initial condition perturbation at time $t = 0$. The errors then grow upscale, indicated by a movement of lines to the left for subsequent simulation times. From *Durran and Gingrich* (2014). b) Forecast skill for 6 day forecasts of 500 hPa geopotential over Europe from several weather centers. From 8-11 April all models have essentially no skill. From *Rodwell et al.* (2013)

section, we will review the basic concepts of atmospheric predictability and then explore how probabilistic forecast, in the form of Monte-Carlo simulations, can help make sense of the uncertainty of the atmosphere.

1.3.1 Sensitivity to initial conditions and upscale error growth

Chaos in the atmosphere was first discovered by *Lorenz* (1963) who noticed that in his simplified model of atmospheric circulation, two simulations with only tiny deviations in their initial conditions would eventually diverge to the point where all resemblance is lost. This phenomenon is called sensitive dependence to initial conditions, or colloquially the "Butterfly effect". In further work, *Lorenz* (1969) discovered that the deviations start growing rapidly on small scales and then propagate more slowly to larger scales, in a process called upscale error growth (Fig. 1.9a).

More recently, Lorenz's idealized experiments have been repeated in simulations of the real atmosphere. *Zhang et al.* (2007) performed error growth experiments with a cloud-resolving model where two simulations with small random perturbations in their initial conditions were compared. They found that the errors first grow rapidly in regions of active convection, then spread to the scale of baroclinic waves, where they grow more slowly with the background baroclinic instability. *Selz and Craig* (2015a) established that the first stage of rapid convective error growth is drastically underestimated in models with parameterized convection. However, a stochastic convection scheme, which we will explore later and is the core of P1, was able to reproduce rates of upscale error growth that matched cloud-resolving simulations. In global tests (T. Selz, personal communication) experiments showed that all predictability was lost after around two weeks, even though the initial condition errors were minuscule in amplitude. These estimates only consider purely atmospheric predictability. Slower processes, involving the ocean, such as ENSO or the MJO, can provide some predictability on longer time scales (*Vitart and Robertson*, 2018).

One important fact about atmospheric predictability is that it is not always the same. First of all, because errors grow upscale, the size of the feature to be predicted matters. Small features, such as individual thunderstorms, are only predictable on time scales of hours, if at all. Larger scale features, such as low or high-pressure systems can be reasonably well predicted up to one week or so. But these estimates of predictability also depend on the weather situation. Some, for example high pressure blocking situations, are very predictable whereas in others predictability is lost very quickly. In fact, there are situations where forecast skill rapidly drops to zero for all weather models. Such situations are called forecasts busts (*Rodwell et al.*, 2013). These are thought to occur when the atmospheric flow undergoes a bifurcation in phase space, often in connection with regions of large latent heating (*Grams et al.*, 2013; *Riemer and Jones*, 2014).

1.3.2 Ensemble forecasts

How then to make predictions in the face of such uncertainty? The question has led to the development of probabilistic forecasting methods. The rationale being that a single deterministic forecast only tells half the story and needs to be supplemented by a flow-dependent uncertainty estimate. For simple systems the evolution of the state-space probability evolution is described by the Liouville equation but the dimensionality and nonlinearity of an atmospheric model prohibits its use in real-world application (*Palmer*, 2000). For this reason, Monte Carlo methods have been favored in atmospheric science. Here, several forecasts are run with slightly different initial conditions and, sometimes, model configurations (Fig. 1.10). In atmospheric modeling, this is called ensemble forecasting and has been operational in most operational weather services since the turn of the century (*Lewis*, 2005). The dispersion of the ensemble is then an estimate of the real, flow-dependent uncertainty attached with a forecast

To create an ensemble forecast, initial conditions are sampled from a distribution that is unfortunately not perfectly known (*Leutbecher and Palmer*, 2008). Again, it is the task of data assimilation to produce the best guess of the initial condition uncertainty. The ensemble then generates a forecast distribution at prediction time, that again is just an approximation for the real uncertainty. The reasons for this are imperfect initial condition perturbations and models that can only provide approximations of the real evolution of the atmosphere. For

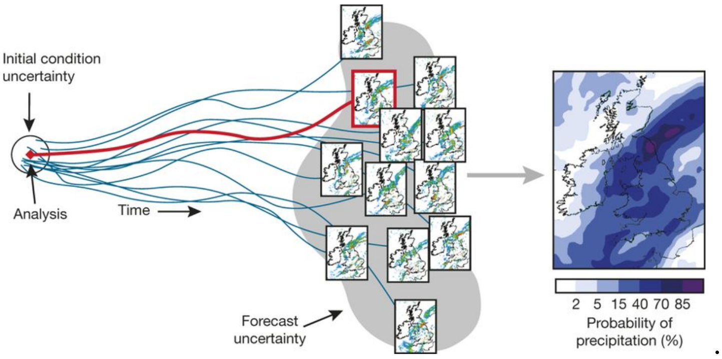

Initial condition uncertainty
Analysis
Time
Forecast uncertainty
2 5 15 40 70 85
Probability of precipitation (%)

Fig. 1.10: Schematic of an ensemble forecast. In this example, a rainfall probability map is created from an ensemble of 36 h predictions. From *Bauer et al.* (2015)

this reason, just as deterministic forecasts, probabilistic forecast exhibit systematic errors.

For statistical postprocessing, the addition of uncertainty adds another requirement to the problem. Not only should a reliable forecast be bias free but its distribution also has to match the real uncertainty of the forecasting system (*Gneiting et al.*, 2005). In a simple example, for all forecasts for which the ensemble predicts a 70% change of rain, rain should also occur in 70% of the cases. In most cases, ensembles are underdispersive, i.e. the observations fall outside the forecast range more often than statistically justified. A real example from P4 is shown in Fig. 1.11.

Aside from postprocessing, there are several ways to make an ensemble more reliable. First, one can hope to improve the initial uncertainty estimate, an active area of research in data assimilation. Another approach is to introduce stochasticity in the model which is the topic of the following section.

1.4 Stochastic parameterizations

Stochastic parameterizations, that is subgrid approximations that contain some random or quasi-random components, were first developed as an *ad hoc* solution to a lack in ensemble spread. However, there is also a more mathematical reason for why stochastic parameterizations might be necessary (*Berner et al.*, 2017). Most traditional parameterizations are build on the assumption that there is a large number of sub-grid elements, for example clouds or turbulent eddies, in each grid box. From this it follows that even though the actual sub-grid realization might differ for slightly different large-scale states, the fluctuations are negligible. For very coarse grids, say several hundred km, this might be a reasonable assumption. For current climate and weather model grids, however, this assumption breaks down (*Jones and Randall*, 2011). In climate models with horizontal grid spacings of around 50 km, the number of deep convective clouds in a grid cell can be small (Fig. 1.12a). The same applies to turbulent boundary layer eddies in km-scale weather models. In these cases, because of the chaotic nature of the atmosphere, the fluctuations of the sub-grid response can be as large

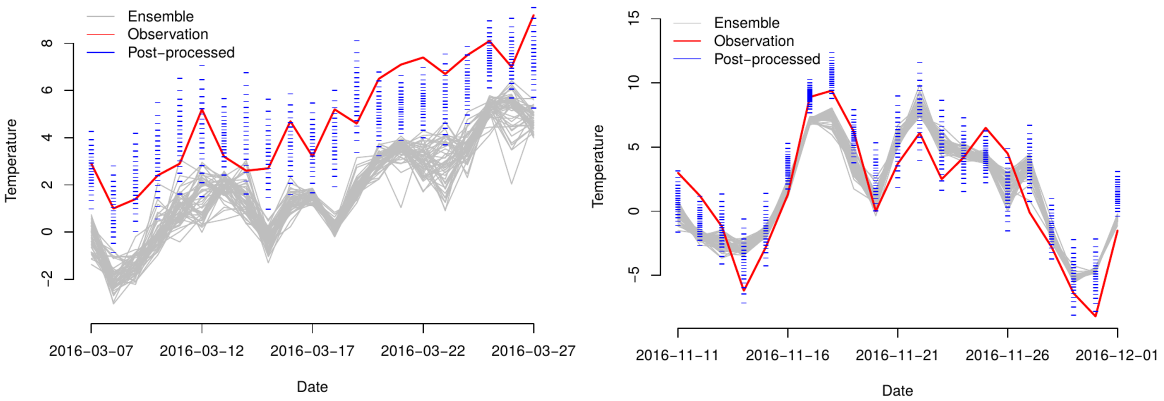


Fig. 1.11: Two examples for ensemble forecasts and their error. In the left figure, the ensemble continuously predicts temperature that are too low compared to the corresponding observations. The blue lines represent the deciles of the postprocessed distribution function. We can see that the postprocessing from P4 is able to correct the bias. On the right, for a different measurement station, the ensemble does not seem to have a mean bias. But still, the observations routinely fall outside the ensemble range, i.e. the ensemble is underdispersive. Again, postprocessing helps by increasing the ensemble spread, so that for almost all forecast times the observations lie inside the range. Figures courtesy of Sebastian Lerch.

as the mean.

Apart from causing underdispersive ensemble forecasts, neglecting these fluctuations can also significantly change the mean evolution of the atmosphere. A simple example is shown in Fig. 1.12b. In this system, described by a double-well potential, adding noise can cause a temporary transition from one state to another. The associated PDF of the entire system therefore changes as well. There is evidence that the atmosphere contains several such phenomena. In particular, large-scale atmospheric oscillations, such as the MJO or El Niño Southern Oscillation (ENSO), seem to be sensitive to the introduction of stochastic parameterizations (*Weisheimer et al.*, 2014; *Christensen et al.*, 2017; *Wang and Zhang*, 2016). ENSO, in fact, is usually too active in climate models. Adding stochastic perturbations actually decreases the variability associated with ENSO. This underlines the non-linear complexity of the atmosphere.

While there is mounting evidence that stochastic parameterizations can increase the skill of atmospheric models, designing a stochastic parameterization is non-trivial. Ideally, these parameterizations should accurately represent the true uncertainty of the sub-grid process. Similarly to the problems with designing traditional parameterizations, described above, this task is complex. In the following, we will present two basic approaches, one more pragmatic, used to create the ensemble in P4, the other based on physical reasoning that is explored in P1. Interestingly, the deep learning parameterization in P2 and P3 is not stochastic, which is

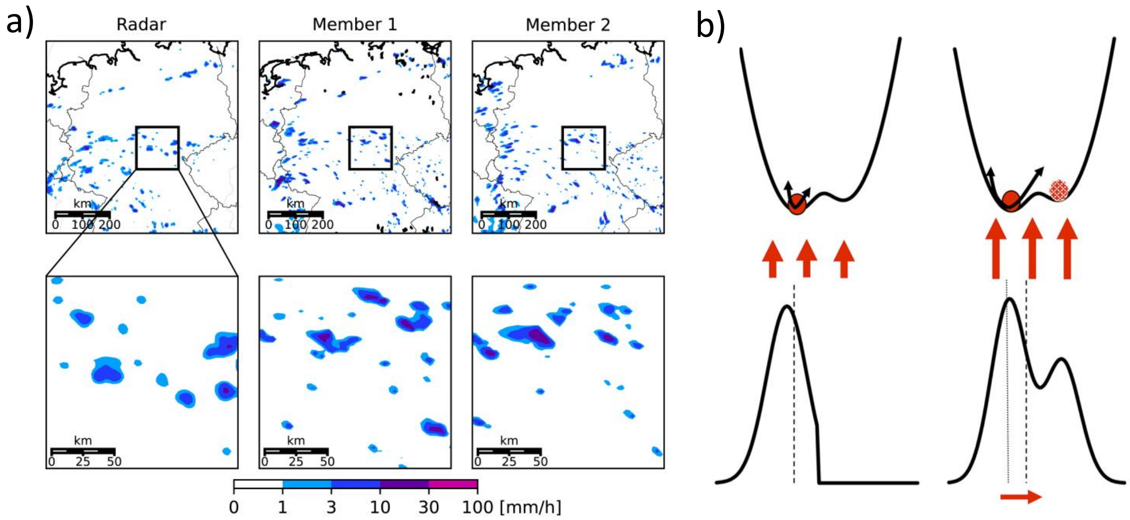


Fig. 1.12: a) Precipitation from observations and two convection-resolving simulations over Germany with the same large-scale conditions. The zoomed regions show that the individual convective elements are essentially uncorrelated. From P1. b) Hypothetical system characterized by a double-potential in top row. Bottom row shows the associated state distribution function. In the left column the stochastic noise is small and the system remains in the global minimum. In the right column the noise is larger, so that the systems intermittently jumps into the second local minimum. This also modifies the state PDF. From *Berner et al.* (2017).

one of the shortcomings mentioned in these papers. In the conclusion, I will discuss potential solutions to this issue.

1.4.1 Multiplicative methods

The first stochastic approach, called "stochastically perturbed physical tendencies" (SPPT), was developed at the European Center for Medium-Range Weather Forecasts (ECMWF) to fix the lack in ensemble spread (*Buizza et al.*, 1999). In this approach the sub-grid tendencies are multiplied by a random number $\eta \sim \mathcal{N}(1, \sigma)$ with a standard deviation σ that is usually tuned to achieve the desired result and is horizontally correlated and temporally evolving:

$$\Delta \mathbf{x}_{\text{sg}}|_{\text{stoch}} = \eta \Delta \mathbf{x}_{\text{sg}} \tag{1.15}$$

The underlying assumption for such multiplicative methods is that the standard deviation of the sub-grid tendencies is proportional to their mean (*Shutts and Palmer*, 2007). Whether this is true is one of the questions addressed in P1. It has to be said, however, that SPPT is an *ad hoc* approach that is designed as a bulk methods to account for all model uncertainties rather than specific physical processes. SPPT has been successfully used in the ECMWF ensemble system for almost two decades. Only in recent years have improved version been explored, where the individual subgrid processes are treated separately (*Christensen et al.*, 2017).

1.4.2 Physically-based stochastic parameterizations

Ideally, a stochastic parameterization produces tendencies based on a random draw from the true sub-grid distribution. For this, a model of the underlying physical process is required. Several approaches have been proposed for this purpose (*Berner et al.*, 2017). Mathematical approaches model the system of equations as stochastic differential equations. However, these approaches have so far only been applied to simplified systems. A different method is to model the sub-grid state evolution on a simple grid using conditional Markov chains (*Khouider et al.*, 2010) or cellular automata (*Bengtsson et al.*, 2013). These approaches have the added benefit of including memory from one time step to the next. In P1, two further approaches are presented: one represents the sub-grid fluctuations in the boundary layer in km-scale models, the other uses statistical mechanics to frame a stochastic model for deep convection. The first approach, originally developed by *Kober and Craig* (2016), is an additive approach

$$\Delta \mathbf{x}_{\mathrm{sg}}|_{\mathrm{stoch}} = \Delta \mathbf{x}_{\mathrm{sg}} + \eta \langle \mathbf{x}'^2 \rangle \tag{1.16}$$

where $\langle \mathbf{x}'^2 \rangle$ represents the physically-based, flow dependent uncertainty of the sub-grid process. This scheme is described in detail in the Appendix of P1. The second method was developed by *Craig and Cohen* (2006) and is based on assumptions resembling an ideal gas, where individual cloud elements are independent and their mass fluxes follow an exponential distribution. These assumptions are described in the introduction of P1. The *Craig and Cohen* (2006) theory has been used to build a stochastic parameterization (*Plant and Craig*, 2008) that has been successfully applied in numerical weather models (*Kober et al.*, 2015), climate models (*Wang et al.*, 2016; *Wang and Zhang*, 2016) and for error growth experiments (*Selz and Craig*, 2015b). In P1, the assumptions of *Craig and Cohen* (2006), which have been developed in an idealized setting, are time tested in realistic weather situations.

1.5 Machine learning

Three of the four papers of this thesis apply machine learning to problems in the forecast chain. This section is a brief introduction to the topic which aims to provide the necessary background for the papers. The term machine learning describes that learn to perform a task from data rather than being explicitly programmed (*Goodfellow et al.*, 2016). Specifically, we will look at supervised learning algorithms, in which each input sample $\mathbf{x}$ in the training

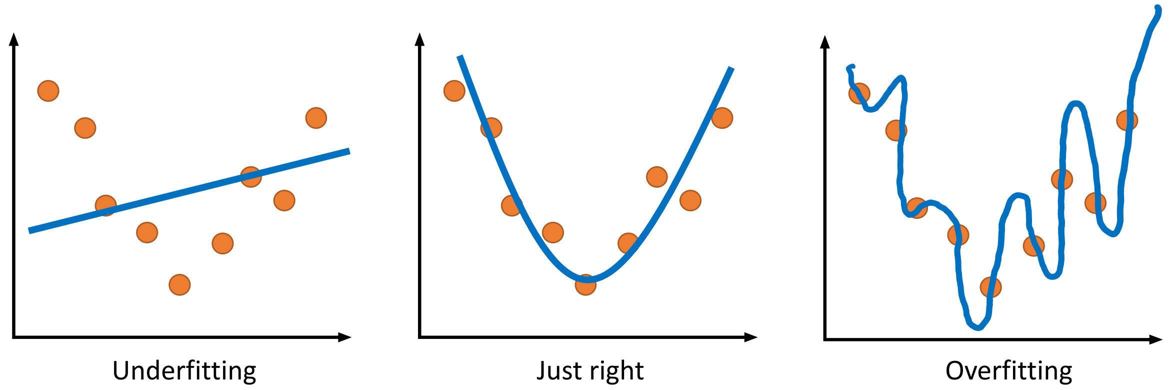


Fig. 1.13: Examples for under- and overfitting. A simple linear regression is not complex enough to capture the behaviour of the data in the left panel. On the right panel, the model perfectly models all the data samples, but is obviously not a good approximation for the "real" function. The best fit lies somewhere in-between.

dataset has a corresponding output, or target, $\mathbf{y}$. The task then is to learn a function f that makes a prediction $\hat{\mathbf{y}}$ given $\mathbf{x}$,

$$\hat{\mathbf{y}} = f(\mathbf{x}), \tag{1.17}$$

so that some error metric or loss function $\mathcal{L}(\mathbf{y}, \hat{\mathbf{y}})$ is minimized. The simplest example of a supervised learning algorithm is linear regression: $\hat{y} = \mathbf{a}\mathbf{x} + b$, where the goal is to find the parameters $\mathbf{a}$ and b that minimize the mean squared error between the predictions $\hat{y}$ and the targets y: $\mathcal{L} = \mathrm{MSE} = 1/N_{\mathrm{samples}} \sum_i (y_i - \hat{y}_i)^2$. For this problem, finding the exact solution is possible by using the normal equation. For many, more complex problems, however, iterative methods are required.

Achieving a low score $\mathcal{L}$ on the training dataset is not enough, however. The purpose of training a supervised learning algorithm is to make predictions for new inputs $\mathbf{x}$ that were not available during training. Therefore, it is essential to also monitor the performance of the trained algorithm on a test dataset that was not used for training. There are two ways machine learning algorithms can fail: underfitting and overfitting (Fig. 1.13). This is also called the bias-variance trade-off. Underfitting occurs when the algorithm is not complex enough to capture the behavior of the training data. Overfitting describes the situation when the model fits to the training data too well, but fails to generalize to the test data. Finding a balance between these two phenomena is one of the main challenges in machine learning.

1.5.1 Neural networks and deep learning

Artificial neural networks, also called feedforward networks or multilayer perceptrons, were originally inspired by the nonlinear signal processing in biological neurons. Over the last

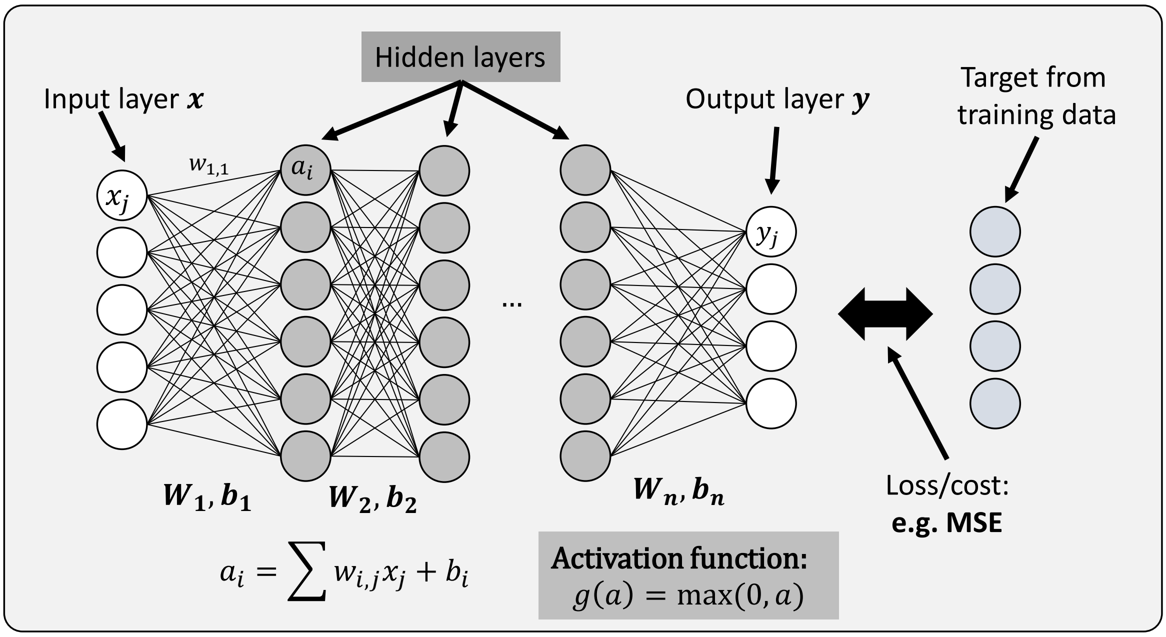

Hidden layers
Input layer $\boldsymbol{x}$
Output layer $\boldsymbol{y}$
Target from training data
$w_{1,1}$
x_j
a_i
y_j
...
$\boldsymbol{W_1}, \boldsymbol{b_1}$
$\boldsymbol{W_2}, \boldsymbol{b_2}$
$\boldsymbol{W_n}, \boldsymbol{b_n}$
Loss/cost:
e.g. MSE
$a_i = \sum w_{i,j} x_j + b_i$
Activation function:
$g(a) = \max(0, a)$

Fig. 1.14: Schematic of a neural network

decade they have become the machine learning algorithm of choice, especially in the fields of computer vision and natural language processing (*LeCun et al.*, 2015).

Neural networks consist of several layers of nodes (Fig 1.14). The value, or activation, a in each of the nodes $i = \{1, \ldots, I\}$ of one layer is a weighted sum of all the activations from the previous layer's nodes $j = \{1, \ldots, J\}$ plus a bias term b, modified by an activation function g:

$$a_i = g\left(\sum_j w_{i,j} a_j + b_i\right) \tag{1.18}$$

The activation function can be any nonlinear function. Popular choices are sigmoid function or hyperbolic tangents. For the papers here, we used rectified linear units (ReLU): $g(z) = \max(\alpha z_i, z_i)$. If α is non-zero (usually a small positive values, e.g. 0.3), this function is called a Leaky ReLU. These nonlinear activation functions enable the neural network to perform nonlinear computations. The first layer contains all input values $\mathbf{x}$, while the last layer represents the output values $\hat{\mathbf{y}}$. Layers in-between are called hidden layers. Such networks, where all nodes of one layer are connected to all nodes of the next are called fully-connected.

The goal then is to minimize some loss function $\mathcal{L}(\mathbf{y}, \hat{\mathbf{y}})$ by changing the weights w and biases b, jointly called the parameters θ. For the optimization one computes the gradient of the loss function with respect to the parameters of the model $\nabla \mathcal{L}_\theta$ for a random subset of the training data or batch, and then takes a step down the gradient direction. This algorithm is called stochastic gradient descent (SGD) and the efficient computation of the gradients through all layers is called backpropagation (*Nielsen*, 2015). More sophisticated versions of SGD have been developed since, for example Adam (*Kingma and Ba*, 2014), which is used for the papers in this thesis. Because SGD only uses one small batch of the data at one time, neural networks can be trained with very large data amounts which would be prohibitive for many other machine learning algorithms.

Deep learning describes the use of neural networks with several hidden layers. More hidden layers lead to increased nonlinear computing power and make the algorithm more parameter-efficient. This means that for the same number of total network weights, deep networks generally outperform shallow networks, something we also found in P2.

Neural networks are very flexible and can be applied to a wide variety of tasks. What is required is a training dataset of sufficient size. For the parameterization problem, the training data can be generated using high-resolution models, as done in P2–3. In postprocessing, the targets are the observations. This means that the sample size is limited by the actual passing of time. Small sample sizes, and therefore the danger of overfitting, are a major challenge, which we also encountered in P4. The other large data processing step in the forecast chain, data assimilation, could also be a promising field for the application of advanced machine learning techniques. So far, however, no promising attempts have been made.

1.5.2 Applications of machine learning in physics

The initial successes of deep learning came from image recognition and natural language processing, and more recently game playing (*Silver et al.*, 2016, 2017). In recent years, however, many researchers have applied these techniques to problems in more traditional fields of science, such as computational biology (*Angermueller et al.*, 2016) and chemistry (*Goh et al.*, 2017). In physics, the number of machine learning-related publications is sky-rocketing. Here, I will present some recent highlights as examples.

In particle physics, huge amounts of data are generated by experiments such as the Large Hadron Collider at CERN. Machine learning is used in this environment to sift through the data and identify interesting collision signatures *Radovic et al.* (2018). Further machine learning techniques are used to calibrate the detected signals, which also helped in finding the Higgs boson. Most of these achievements used more traditional machine learning techniques, such as boosted decision trees, but more recently deep learning has been used to find neutrinos and better identify beauty quarks.

In quantum physics, computations for many-body systems are very expensive with traditional techniques. Recently, researchers have used neural networks to learn approximate descriptions of the wave function in order to find the ground state of the systems *Carleo and Troyer* (2017). They found that neural networks are more memory efficient than traditional techniques. In a different study, machine learning was used to create quantum experiments (*Melnikov et al.*, 2018). Specifically, the algorithm is tasked with finding experimental configurations that produce entangled multi-photon states.

Iten et al. (2018) use a deep neural network to distill important parameters of a physical systems which can then be used to make general predictions. For example, their algorithm learned the spring constant and damping coefficient for a damped harmonic oscillator from observing the behavior of the system. In their study, they only used simple systems for which the exact equations are known. It would be interesting to test a similar approach for a complex system, such as the subgrid cloud evolution.

Finally, a number of studies have been published on learning the evolution systems described by partial differential equations (*Bar-Sinai et al.*, 2018; *Pathak et al.*, 2018; *Raissi*, 2018; *Kim et al.*, 2018). Typically, the goal is to improve the accuracy by learning from expensive high-resolution simulations or to speed up the computations. Similarly to the challenges for deep learning atmospheric parameterizations described in the Conclusion of this thesis, the key obstacles in this line of research are stability and adherence to physical constraints.

2 Papers

2.1 P1: Verification of a theory for convective variability

Variability and clustering of mid-latitude summertime convection: Testing the Craig and Cohen (2006) theory in a convection-permitting ensemble with stochastic boundary layer perturbations

Stephan Rasp, Tobias Selz and George C. Craig, 2018.
Journal of the Atmospheric Sciences, 75(2), 691–706.

Context In this paper, a theory for the random fluctuations of an ensemble of convective clouds is tested in realistic weather situations. The theory is the foundation for a popular stochastic convection parameterization but has so far only been verified in idealized situations. Here we use cloud-resolving simulations with a stochastic perturbation scheme for boundary layer turbulence to create randomized cloud fields for the same large-scale synoptic situation. The paper shows the general applicability of the *Craig and Cohen* (2006) theory, thereby challenging the variance scaling assumption in the widely used SPPT scheme, but also highlights that the organization of clouds—not included in the theory—is an important factor.

Author contribution GC designed research. I conducted model simulations and analysis with help from TS and led the writing of the manuscript. All authors discussed results and manuscript drafts.

Variability and Clustering of Midlatitude Summertime Convection: Testing the Craig and Cohen Theory in a Convection-Permitting Ensemble with Stochastic Boundary Layer Perturbations

STEPHAN RASP, TOBIAS SELZ, AND GEORGE C. CRAIG

Meteorological Institute, Ludwig-Maximilians-Universität München, Munich, Germany

(Manuscript received 31 August 2017, in final form 23 November 2017)

ABSTRACT

The statistical theory of convective variability developed by Craig and Cohen in 2006 has provided a promising foundation for the design of stochastic parameterizations. The simplifying assumptions of this theory, however, were made with tropical equilibrium convection in mind. This study investigates the predictions of the statistical theory in real-weather case studies of nonequilibrium summertime convection over land. For this purpose, a convection-permitting ensemble is used in which all members share the same large-scale weather conditions but the convection is displaced using stochastic boundary layer perturbations. The results show that the standard deviation of the domain-integrated mass flux is proportional to the square root of its mean over a wide range of scales. This confirms the general applicability and scale adaptivity of the Craig and Cohen theory for complex weather. However, clouds tend to cluster on scales of around 100 km, particularly in the morning and evening. This strongly impacts the theoretical predictions of the variability, which does not include clustering. Furthermore, the mass flux per cloud closely follows an exponential distribution if all clouds are considered together and if overlapping cloud objects are separated. The nonseparated cloud mass flux distribution resembles a power law. These findings support the use of the theory for stochastic parameterizations but also highlight areas for improvement.

1. Introduction

Stochastic parameterizations have the potential to increase forecast skill and decrease model biases by capturing the inherently turbulent nature of many subgrid processes [for a comprehensive overview, see Berner et al. (2016)]. In the case of atmospheric deep convection, the fluctuations around the mean state within a grid box become significant for model grid spacing less than 100 km (Jones and Randall 2011). This subgrid noise can feed back onto the resolved scales, impacting tropical oscillations (Wang et al. 2016; Christensen et al. 2017) and the upscale growth of forecast errors (Selz and Craig 2015b). Designing a stochastic parameterization requires some model of the subgrid-scale variability. Simple approaches include perturbing parameterized model tendencies in a multiplicative way (Buizza et al. 1999) or perturbing parameters in the parameterizations (Ollinaho et al. 2017). More complex schemes have been designed based on subgrid Markov chains (Dorrestijn et al. 2013) and cellular automata (Bengtsson et al. 2013).

In this study, we focus on a theory of convective variability based on statistical physics developed by Craig and Cohen (2006; the theory is hereafter abbreviated CC06). Its application in a stochastic parameterization framework (Plant and Craig 2008) proved beneficial in a number of ways: it produces scale-aware fluctuations in a mesoscale model (Keane et al. 2014); forecast errors grow upscale realistically, as opposed to deterministic parameterizations in which errors grow too slowly (Selz and Craig 2015a); and in global climate simulations, precipitation variability and tropical wave activity, such as the Madden–Julian oscillation, are improved (Wang et al. 2016; Wang and Zhang 2016). Recently, Sakradzija et al. (2015) extended the approach to shallow convection, which improves coupling the resolved

Denotes content that is immediately available upon publication as open access.

Supplemental information related to this paper is available at the Journals Online website: https://doi.org/10.1175/JAS-D-17-0258.s1.

Corresponding author: Stephan Rasp, s.rasp@lmu.de

DOI: 10.1175/JAS-D-17-0258.1

model dynamics (Sakradzija et al. 2016). For all its benefits, the CC06 theory is based on strongly simplifying assumptions about how convection behaves. The main purpose of this study is to test the applicability of the CC06 theory outside of its comfort zone.

a. The CC06 theory

The aim of the CC06 theory is to derive a minimally simple model of convective variability. Under the quasi-equilibrium assumption, the large-scale state prescribes the mean mass flux in a certain domain $\langle M \rangle$[1]—through a closure assumption in a parameterization. The mean mass flux of an individual cloud $\langle m \rangle$ is determined solely by local properties such as boundary layer turbulence or entrainment and is therefore independent of $\langle M \rangle$. The average number of clouds is then $\langle N \rangle = \langle M \rangle / \langle m \rangle$. The individual clouds are assumed to be pointlike, randomly distributed in space, and noninteracting. The most likely state of such a cloud ensemble is characterized by an exponential distribution of the cloud mass flux m:

$$p(m) = \frac{1}{\langle m \rangle} e^{-m/\langle m \rangle}; \qquad (1)$$

and a Poisson distribution of the cloud number in the domain

$$p(N) = \frac{\langle N \rangle^n}{n!} e^{-\langle N \rangle} \quad \text{for} \quad n = 0, 1, \ldots \qquad (2)$$

Combining these equations yields a distribution of the domain-total mass flux $p(M)$ as a function of its mean $\langle M \rangle$ and the mean cloud mass flux $\langle m \rangle$. In principle, it is possible and interesting to investigate the full distribution function or several higher-order moments. For this study, however, we focus on the second moment. This is done for two main reasons: first, higher moments require a larger sample size to yield statistically significant results, and second, deviations in the variances allow for clear and physical interpretations. The second moment can be expressed in terms of the normalized variance

$$\frac{\langle (\delta M)^2 \rangle}{\langle M \rangle^2} = \frac{2}{\langle N \rangle} \qquad (3)$$

or in terms of the unnormalized standard deviation

$$\langle (\delta M)^2 \rangle^{1/2} = \sqrt{2 \langle M \rangle \langle m \rangle}. \qquad (4)$$

b. Previous tests of the CC06 theory

So far, few studies have directly tested the assumptions and predictions of CC06. Cohen and Craig (2006) used a convection-permitting model in a radiative–convective equilibrium setup with different large-scale forcing and vertical wind shear strengths and found that $p(m)$ was well approximated by an exponential distribution for all settings but that $p(N)$ was broader than predicted by Eq. (2) because of cloud clustering. Despite this, the simulated mass flux variability was up to 20% lower than predicted, which they attributed to compensating errors. Davoudi et al. (2010) confirmed this result in their simulations with a diurnal cycle of radiation. Scheufele (2014) tested the sensitivity of the Cohen and Craig (2006) results to model resolution and found that clouds are more strongly clustered at higher resolutions. Furthermore, to reproduce the exponential cloud mass flux distributions, a separation of connected clouds into individual updrafts becomes necessary. Davies (2008) focused on the variation of the CC06 predictions in an idealized diurnal cycle setup. Cloud clustering and convective variability were strongest shortly after convective initiation and during the decline of convective activity. Studying the variability scaling in trade wind shallow cumulus, Sakradzija et al. (2015) saw a drastic increase in cloud organization once precipitation started to form. Consequently, the mass flux variability was many times larger than the predictions of the statistical theory.

c. Motivation and aims of this paper

The studies mentioned above have two things in common: first, they show that the cloud field is organized in contradiction with the CC06 assumptions, and second, all the studies were conducted with simplified, idealized setups. These setups generally aim to represent tropical equilibrium convection or a diurnal cycle over land in the absence of any large-scale changes in forcing or other complications such as land surface variations or orography. Stochastic parameterizations in general circulation models, however, must be able to cope with a wide variety of convection around the globe. It remains unclear if, and to what extent, the CC06 theory and the stochastic parameterizations based on it are useful for representing complex real-world weather situations.

In this study, we aim to test the assumptions and predictions of CC06 in simulations of midlatitude summertime convection over land. In particular, we ask the following research questions:

- How well do the CC06 predictions hold up for complex, nonequilibrium convection?
- Are there systematic deviations, and can we explain them?
- What role does convective organization play?

[1] Note that the angle brackets used throughout the text describe ensemble means.

To answer these questions, we set up ensemble simulations (section 2) that allow us to quantitatively measure convective variability in real-world case studies (section 3). The analysis details are described in section 4. In section 5, we compare the predictions of the CC06 theory to our simulation results and identify systematic deviations. We then focus on cloud organization (section 6) and the cloud mass flux distribution (section 7). A discussion of the implications for stochastic parameterizations follows in section 8 before a summary in section 9.

2. Numerical simulations, observations, and computational reproducibility

a. COSMO model and ensemble setup

The numerical simulations are done with the Consortium for Small-Scale Modeling (COSMO) model (Baldauf et al. 2011). The horizontal grid spacing Δx is 0.025°, roughly 2.8 km, with 50 levels in the vertical. There is no parameterization of deep convection, but shallow convection is parameterized using the Tiedtke scheme [for complete information on the parameterizations used, see Doms et al. (2011)]. The domain spans 357 grid points in either horizontal direction centered over Germany at 50°N, 10°E. For the analysis, a 256×256 gridpoint subdomain, roughly 717 km × 717 km, at the center of the simulation domain is considered to avoid boundary spinup effects. The 50-member ensemble simulations are started at 0000 UTC on each of the 12 consecutive days (see section 3) with a simulation time of 24 h. For initial and boundary conditions, we use hourly interpolated deterministic COSMO European version (COSMO-EU) analyses, which have a horizontal resolution of 7 km. Each ensemble member differs only in the random seed used for the stochastic perturbation scheme, described below, which has the effect of randomly shuffling the convective cells. Additionally, one deterministic simulation is run without the stochastic perturbation scheme. The output frequency is 60 min. The model name lists are saved in the online repository accompanying this paper (see section 2e).

Given unlimited computational resources, it would be desirable to run the model at a higher resolution to actually resolve the cloud features. With our computational constraints, however, a trade-off between resolution, ensemble size, and length of the simulation period is necessary. To ensure the simulations create realistic cloud features, we compare the model to radar observations (see section 2c). We will also discuss the potential impact of resolution on our results in section 9.

b. Stochastic boundary layer perturbations

The physically based stochastic perturbation (PSP) scheme was first proposed by Kober and Craig (2016). It represents a general framework for adding process-specific perturbations to reintroduce missing variability on the grid scale of convection-permitting models that is associated with unresolved processes. The process considered here is boundary layer turbulence. In the appendix, we present an updated formulation of the scheme to clarify the physical rationale. In this study, the stochastic scheme enables us to obtain many different realizations of the convective cloud fields for the same large-scale flow. One limitation of this approach is the fact that only subgrid turbulence is considered in the stochastic perturbations. Other subgrid processes such as orography might preferentially trigger convection in particular locations, thereby violating the CC06 assumptions.

c. Radar-derived precipitation observations

To validate our simulations we use the Radar Online Aneichung (RADOLAN) radar-weighted (RW) product provided by the German Weather Service (DWD; DWD 2017). These data contain estimates of 1-hourly precipitation accumulations based on radar reflectivities adjusted using rain gauges. Even though the precipitation values are not direct observations, we use the term *observation* for the RADOLAN RW product in the rest of the text. The original data, which have a spatial resolution of 1 km, are adapted to the COSMO model grid. For all model–observation comparisons, only grid points with observation data are used. If daily time series are computed, a joint mask is used only including grid points for which observation data are available at all times throughout the day. The hourly RADOLAN RW products are valid at 10 min to the full hour, whereas our model precipitation accumulations are written to the full hour. For the purposes of this study, we think it is reasonable to neglect this difference. Last, because of the automatic adjustment of the radar observations with rain gauge data, some unrealistically high precipitation values occur, sometimes exceeding 300 mm h^{-1} (K. Stephan 2017, personal communication). Since we are not computing forecast scores in this study, an ad hoc measure of removing all grid points with hourly precipitation accumulations larger than 100 mm h^{-1} turned out to be sufficient to remove any significant artifacts.

d. Displacement growth of stochastic perturbations

Figure 1a shows precipitation snapshots of the observations and two ensemble members. The large-scale

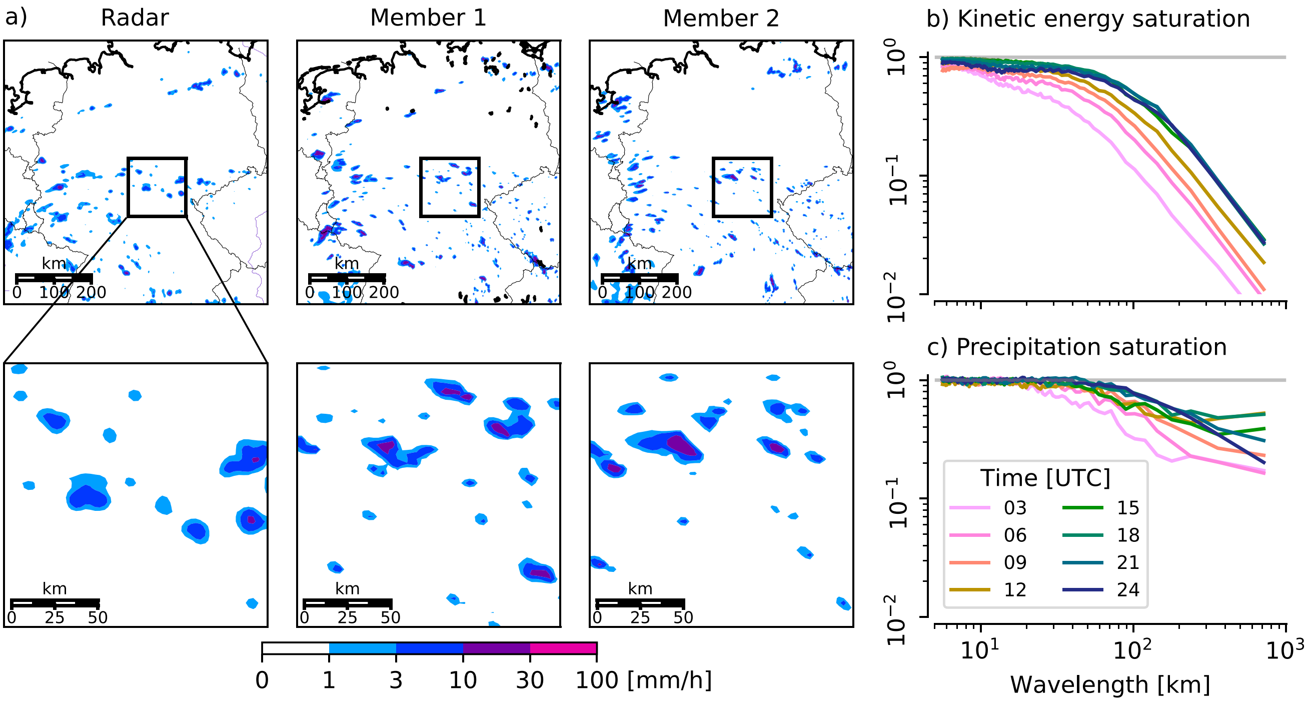


FIG. 1. (a) Hourly precipitation accumulations at 1400 UTC 4 Jun 2016 for the observations and two ensemble members. (top) Entire analysis domain; (bottom) zoom in on a smaller region. (b),(c) Ratio of difference to twice the difference spectrum of (b) horizontal kinetic energy in the troposphere, omitting the top 15 model levels, and (c) hourly precipitation for different times of day. For details on the calculation, see Selz and Craig (2015b). Values of one indicate a full displacement at this scale. Lines represent composites over all days.

precipitation pattern is very similar, but the individual convective elements appear to be completely uncorrelated. To quantitatively assess the displacement at different scales, we look at the saturation of the ensemble difference spectra of kinetic energy and precipitation [Figs. 1b,c; for a detailed description of the calculation of the spectra, see Selz and Craig (2015b)]. The kinetic energy spectrum shows typical signs for upscale error growth: small scales saturate first, followed by saturation at increasingly larger scales. At the scale of large precipitation patterns, around 300 km, the difference between the ensemble members is below 20%. This confirms that the large-scale forcing is similar between all ensemble members. In contrast, the precipitation spectrum at the scale of individual convective elements, approximately 50 km, is almost fully decorrelated after 6 h. The combination of similar large-scale conditions and displaced convection allows us to investigate convective variability in real-weather case studies.

e. *Computational details and reproducibility*

This subsection closely follows the guidelines on publishing computational results proposed by Irving (2016). The analysis and plotting of model and observation data were done using Python. The Python libraries Numerical Python (NumPy; van der Walt et al. 2011) and Scientific Computing Tools for Python (SciPy; Jones et al. 2001) were used heavily. The raw data were read with the Python module cosmo_utils (code available upon request). The figures were plotted using the Python module Matplotlib (Hunter 2007). Plotting colors were chosen according to the hue–chroma–luminance color space (Stauffer et al. 2015). Some plots were postprocessed using the vector graphics program Inkscape.

To enable reproducibility of the results, this paper is accompanied by a version-controlled code repository (https://github.com/raspstephan/convective_variability_analysis) and a Figshare repository (Rasp 2017), which contains a snapshot of the code repository at the time of submission and supplementary log files for each figure. These log files contain information about the computational steps taken from the raw data to the generation of the plots. While the model code and initial data are not openly available, a detailed technical description of the model simulations can be found in the cosmo_runscripts directory of the code repository. The Jupyter notebooks (Kluyver et al. 2016) mentioned in the text are stored in the directory jupyter_notebooks of the repository. Links to noninteractive versions of the notebooks can be found on the front page of the Github repository; rendered PDF versions are also added to the supplement of this paper.

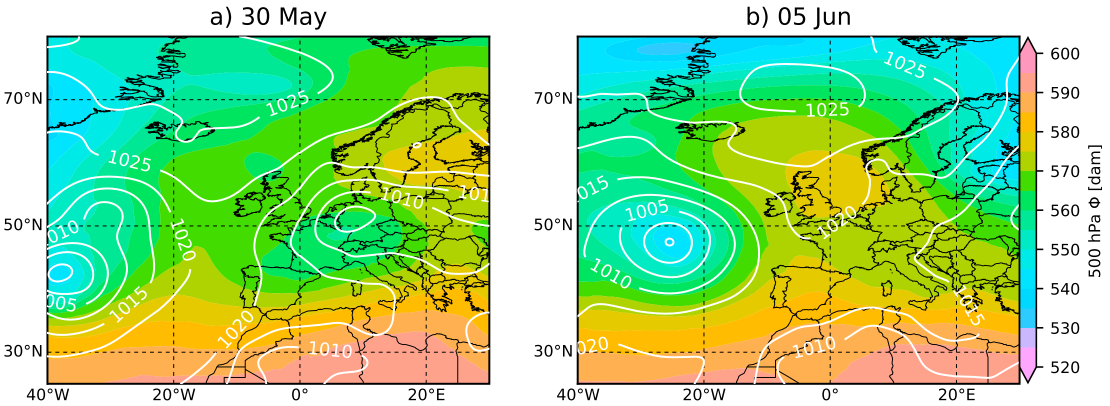


FIG. 2. Synoptic charts at (a) 0000 UTC 30 May and (b) 0000 UTC 5 Jun 2016. White lines represent mean sea level pressure (hPa). Colors represent 500-hPa geopotential (dam). Figures created from ECMWF analyses.

3. Weather situation and precipitation in model and observations

a. Synoptic situation and convective regime

The period from 26 May to 9 June 2016 was characterized by extraordinary extreme weather over central Europe and, in particular, Germany (Piper et al. 2016). Heavy precipitation, exceeding a 200-yr return period in some regions of southern Germany, caused flash floods that, together with hail measuring up to 5 cm in diameter and 12 confirmed tornadoes, resulted in damages of over EUR 5 billion. A persistent heavy-precipitation period of similar length is unprecedented in a 55-yr climatology. For this study, we selected 12 contiguous days from 28 May to 8 June.

The period can be roughly divided into two phases (Fig. 2). In the first, from 28 May to approximately 3 June, an upper-level trough dominated European weather, subsequently developing into a cutoff low. This upper-level feature caused strong synoptic lifting and several weak surface low pressure systems. These were accompanied by cyclonic circulation centered over the Alpine region and southeasterly advection of moist air over central Europe. This lead to a destabilization of the atmosphere, particularly over southern Germany. In the second phase, from 4 to 8 June, the cutoff gave way to a stationary upper-level ridge, typical for an omega-blocking situation. This caused very persistent weather with large instability building up over southern Germany.

The synoptic instability, combined with strong surface heating, provided a favorable environment for the development of deep convection. Precipitation followed a diurnal pattern on most days (Fig. 3) but was modulated by synoptic lifting on several days. This is most noticeable in the night of 29–30 May, in which a mesoscale convective system in association with large-scale ascent covered most of southern Germany (domain-mean precipitation plots for each individual day can be found in the supplement). The precipitation lags the convective available potential energy (CAPE) by around 4 h. The chosen period represents a variety of nonequilibrium convection over land and is therefore well suited to test the CC06 theory outside of the regime for which it was originally designed.

b. Precipitation in simulations and observations

Finally, we want to compare the mean precipitation in our simulations with the PSP scheme (ens), without the PSP scheme (det), and in the observation (obs) in Fig. 3. The PSP scheme causes an earlier onset and a higher maximum in precipitation on several days, which is the expected systematic effect (Kober and Craig 2016). The maximum precipitation increase caused by the stochastic perturbations is 10%, indicating that the model behavior is not drastically altered.

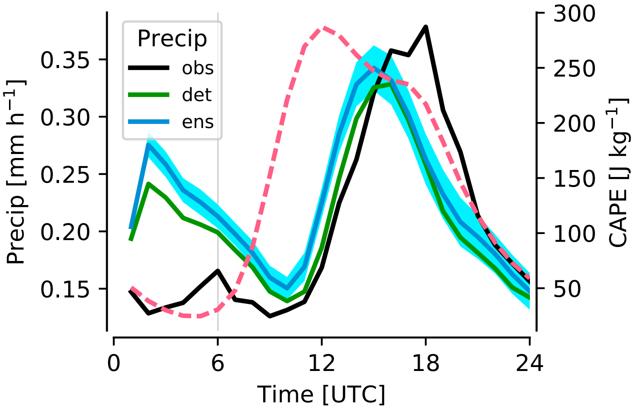


FIG. 3. Composite of domain-averaged precipitation for observations (black), deterministic (green), and ensemble simulations (blue). The blue shaded region represents plus and minus one standard deviation of the ensemble. Additionally, ensemble-mean domain-averaged CAPE (J kg^{-1}) is shown in red. The gray vertical line indicates the time at which the analysis starts.

Larger differences can be seen between the model simulations and the observations. It is important to note that we are not interested in obtaining simulations that are as close to the observations as possible. Rather, our aim is to test whether the general weather situation is reproduced and if there are any systematic differences. On most days, the precipitation amounts in the simulations and observations match well. There is, however, a systematic early decline of precipitation in the late afternoon. Inspecting the precipitation stamps (available in the supplement) suggests that, in some situations, the model is not able to reproduce lasting organized convection when the forcing becomes weaker (an example for this can be seen in Fig. 8). While we are confident that the general characteristics of the convection are well captured in our simulations, it is important to keep this systematic deviation in mind for the subsequent analysis. Furthermore, there is a spinup peak in the first few forecast hours caused by the stochastic perturbations, which is typical when starting from a downscaled analysis. To avoid this effect and to allow the perturbations to develop, we start all our subsequent analyses from 0600 UTC.

4. Cloud identification and computation of statistics

To test the CC06 predictions and assumptions, we need to identify the individual clouds and compute statistics of the modeled updraft mass flux field. The process of identifying cloud objects and computing the radial distribution function is illustrated in an accompanying Jupyter notebook (called cloud_identification_and_rdf.ipynb). We identify convective updrafts by using a vertical velocity threshold $w > 1\,\mathrm{m\,s^{-1}}$ combined with a positive cloud water content at model level 30. This corresponds to a height above ground level of around 3000 m in the terrain-following COSMO grid. Ideally, one would choose the cloud base since this is what many convection parameterizations use, but this is difficult to determine. Previous studies have shown that the mass flux statistics are relatively insensitive to changes in the analysis height in a reasonable range around the chosen level [see, e.g., Fig. 13 of Davoudi et al. (2010) or Fig. 5.6 in Davies (2008)]. From the resulting binary field, objects are classified as pixels that share an edge. Visual inspection suggested that many "objects" are in fact conglomerates of several touching updrafts [this has also been found by Scheufele (2014)]. We therefore separate the cloud objects using a local maximum filter in combination with a watershed algorithm (Beucher and Meyer 1992). For the local maximum filter, we use a search footprint of 3 × 3 grid points. All subsequently presented analysis is done using the separated objects unless otherwise stated. For further information and sensitivity tests of the cloud separation algorithm, see the aforementioned Jupyter notebook cloud_identification_and_rdf.ipynb.

Computation of statistics

For each identified cloud $k = 1, \ldots, N_{\mathrm{cld},i}$ in each ensemble member $i = 1, \ldots, N_{\mathrm{ens}}$, the cloud size, defined as the horizontal area, σ_k is computed:

$$\sigma_k = N_{\mathrm{px}} \Delta x^2, \tag{5}$$

where N_{px} is the number of pixels for each cloud k. The mass flux per cloud m_k is given by

$$m_k = \Delta x^2 \sum_{l}^{N_{\mathrm{px}}} w_l \rho_l, \tag{6}$$

where ρ is density.

To compute the domain statistics, a coarse graining is applied to create coarse boxes $j = 1, \ldots, N_{\mathrm{box},n}$ with edge lengths $n \in \{256, 128, 64, 32, 16, 8\}\Delta x$, where the number of coarse boxes for each analysis time step is $N_{\mathrm{box},n} = (256/n)^2$. No smaller neighborhoods are considered, since these would be below the effective resolution of the model and the sample size of clouds within the coarse boxes becomes too small. The total mass flux per box j per member i, denoted $M_{i,j,n}$, is given by

$$M_{i,j,n} = \sum_{k=1}^{N_{\mathrm{cld}\,i,j,n}} m_{k,i,j,n}, \tag{7}$$

where $N_{\mathrm{cld}\,i,j,n}$ is simply the number of clouds that fall into each box. To avoid splitting clouds at the boundaries of the coarse fields, the centers of mass for each cloud are first identified. Then m_k is attributed to that one point in space. Therefore, the coarse box that contains the center of mass also contains the entire cloud, while the other box does not contain any of the cloud (Cohen and Craig 2006).

Additionally, we compute statistics for the mean heating rate Q for each coarse box j at the same model level. Note that, unlike M, Q can be negative. The variables M and Q are well correlated, with a correlation coefficient of 0.8 across all scales.

Ensemble statistics of $\Phi \in \{M, N, Q\}$ are then calculated for each box j. The sample variance is computed as

$$\langle (\delta\Phi)^2 \rangle_{j,n} = \frac{1}{N_{\mathrm{ens}} - 1} \sum_{i=1}^{N_{\mathrm{ens}}} (\Phi_{i,j,n} - \langle \Phi \rangle_{j,n})^2, \tag{8}$$

where the ensemble mean is given by

$$\langle\Phi\rangle_{j,n} = \frac{1}{N_{\text{ens}}}\sum_{i=1}^{N_{\text{ens}}}\Phi_{i,j,n}. \tag{9}$$

To compute statistics for m, a different approach is taken. Here, the clouds in all ensemble members for each box are considered together to calculate the variance and mean:

$$\langle(\delta m)^2\rangle_{j,n} = \frac{1}{N_{\text{cldtot}}-1}\sum_{k=1}^{N_{\text{cldtot}}}(m_{k,j,n} - \langle m\rangle_{j,n})^2, \tag{10}$$

where the mean is given by

$$\langle m\rangle_{j,n} = \frac{1}{N_{\text{cldtot}}}\sum_{k=1}^{N_{\text{cldtot}}} m_{k,j,n}, \tag{11}$$

and $N_{\text{cldtot}} = \sum_{i=1}^{N_{\text{ens}}} N_{\text{cld}\,i,j,n}$ is the total number of clouds in all ensemble members. If N_{cldtot} becomes too small, the statistics are severely affected by sampling issues. After inspecting numerical tests, we decided to drop all data where the total number of clouds across all ensemble members is less than 11 (see supplementary Jupyter notebook beta_sample_size_dependency.ipynb).

5. Comparison of CC06 predictions with simulations

a. *Scaling of standard deviation with the mean*

Our first research question is whether the CC06 theory is applicable for complex real-weather situations. Assuming that the variation of the mean cloud mass flux $\langle m\rangle$ is small, an assumption we will revisit later, the CC06 theory states that the domain-total mass flux standard deviation increases with the square root of its mean [Eq. (4)]. The combined data, including all coarse boxes for all scales n for all days and time steps, show that, over more than three orders of magnitude in $\langle M\rangle$ and almost two orders of magnitude in horizontal scale, the mass flux standard deviation is well described by the proposed square root relation (Fig. 4a). The fit parameter b is $9.62 \times 10^7\,\text{kg}\,\text{s}^{-1}$. The variability drops off at both ends, which is reflected in a change in b if the curve is fitted to each scale individually—smaller scales have a steeper slope.

The heating rate Q, being a horizontally averaged quantity, needs to be multiplied by the area of the coarse box to test for the scaling; $Q \times A$ behaves similarly to M (Fig. 4b). The square root scaling applies over an even larger range of scales, but the uncertainty increases toward small values of $Q \times A$. In contrast to the updraft mass flux, which is by definition positive, the heating rate can also be negative. Therefore, the heating rate standard deviation is not constrained to go to zero as its mean goes to zero, which is a potential cause for the deviations at smaller scales.

Additionally, we fit a linear relation, which corresponds to multiplicative noise as in the stochastically perturbed parameterization tendencies (SPPT) scheme (Shutts and Palmer 2007). Note that changing b does not change the apparent slope on a log–log plot but only displaces the line. The linear relation is not able to capture the standard deviation scaling of $\langle M\rangle$ or $Q \times A$ in our simulations. This is in line with the findings of Shutts and Pallarès (2014), who argue that, for convection, the standard deviation is better described by a square root scaling.

This first general test of the main prediction of the CC06 theory confirms its applicability even in the complex situations in our simulations. One key feature of the theory is its scale awareness. This can be seen in the slope of the fit b, which varies by a factor of 2 as the mean mass flux increases by more than three orders of magnitude (see insert of Fig. 4a). A small variation of the slope indicates that the CC06 scaling describes the variability for a range of different coarsening resolutions without the need for retuning of the parameters. This resolution independence is a desirable trait for stochastic parameterizations, particularly as the gray zone is approached. Multiplicative perturbations, in contrast, appear to be inherently resolution dependent. We discuss the implications of our findings for SPPT in section 8.

b. *Deviations from the CC06 theory*

While the overall variability scaling is reasonably described by the CC06 theory, the analysis above implies differences between the coarsening scales. In this section, we focus on the systematic deviations from the CC06 theory as a function of the coarsening scale and time of day—our second research question. Since most of the 12 simulation days show similar characteristics (see supplement), we concentrate on composites of all days. To test the CC06 variance predictions, the ratio of simulated to predicted variance is defined as

$$R_V = \frac{1}{N_{\text{box},n}}\sum_{j}^{N_{\text{box},n}}\frac{\langle(\delta M)^2\rangle_j}{2\langle M\rangle_j\langle m\rangle_j}. \tag{12}$$

This metric is similar to the one used by Davies (2008) and Davoudi et al. (2010) and describes whether the simulated variance is less (values smaller than one) or more (values larger than one) compared to the variance predicted by the CC06 theory given the same values of $\langle M\rangle$ and $\langle m\rangle$. In Fig. 5a R_V is shown for three representative scales. Two trends jump out. First, there is a diurnal cycle with lower variability in the early

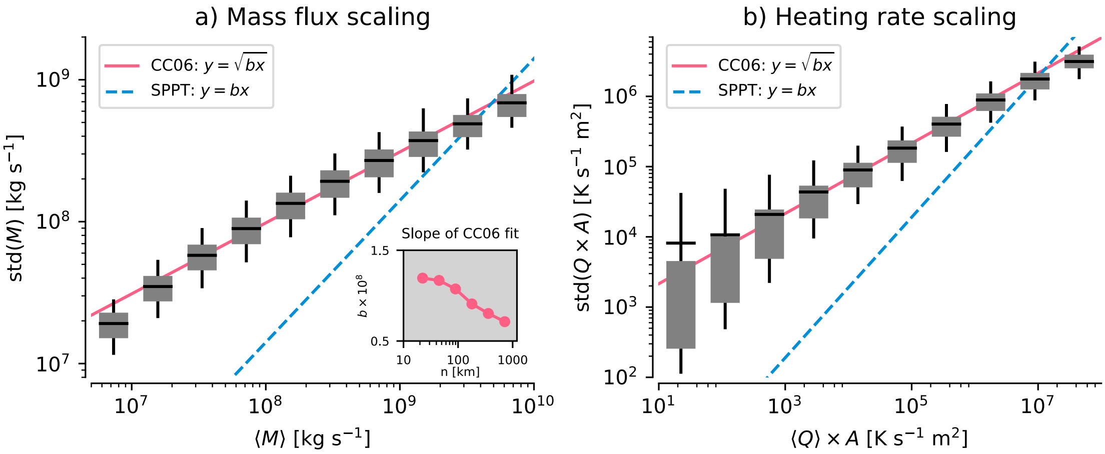


FIG. 4. Standard deviation plotted against mean for (a) domain-total mass flux M (kg s^{-1}) and (b) heating rate Q multiplied by domain size (K m^2 s^{-1}). The data are binned in logarithmic bins. The horizontal line represents the mean for each bin; the gray box indicates the 25th–75th-percentile range, and the vertical line represents the 5th–95th-percentile range. A least squares fit for a square root relation through all data points (without binning) is shown in red and for a linear relation in dashed blue. The inset shows the square root fit parameter b if fitted against each coarsening scale n individually.

afternoon, from 1200 to 1500 UTC, and increased variability in the morning and, particularly, the evening. The trend becomes stronger with scale. Second, the mean R_V is largest for scales around 100 km. Our subsequent analysis aims to identify the causes of these systematic deviations.

Recall that the CC06 variance prediction arises from two distributions: a Poisson distribution of the cloud number N and an exponential distribution of the cloud mass flux m. To test these distributions in our simulations, we define parameters describing the width of the

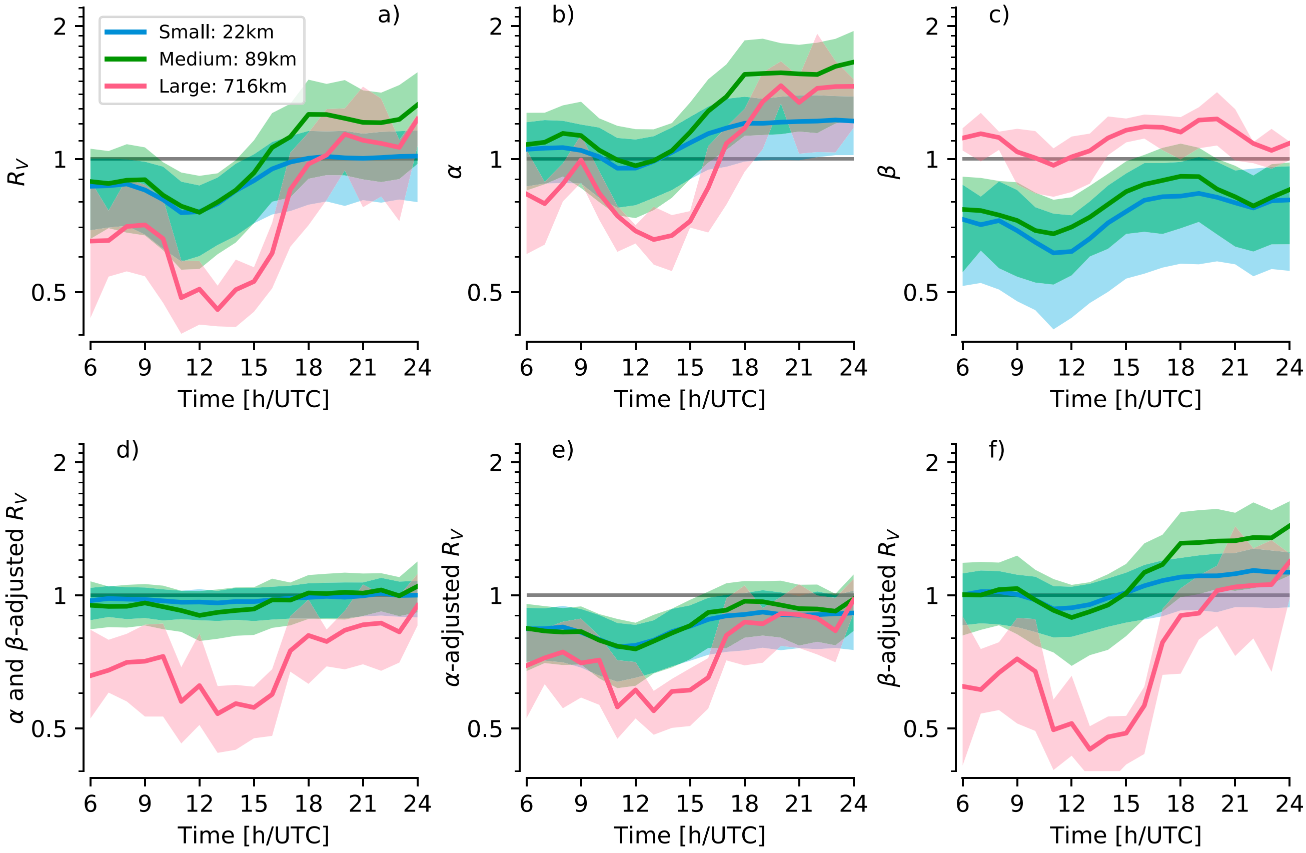


FIG. 5. Composites of (a) R_V, (b) α, (c) β, (d) α- and β-adjusted R_V, (e) α-adjusted R_V, and (f) β-adjusted R_V for three selected coarsening scales. Description of the parameters can be found in section 5b. Shaded regions represent the 25th–75th-percentile range.

distributions with respect to their mean. Starting with the cloud number distribution,

$$\alpha = \frac{1}{N_{\mathrm{box},n}} \sum_{j}^{N_{\mathrm{box},n}} \langle(\delta N)^2\rangle_j / \langle N\rangle_j \qquad (13)$$

describes whether clouds are more clustered, $\alpha > 1$, or more regularly spaced, $\alpha < 1$, compared to a completely random distribution in space [see appendix A of Davoudi et al. (2010)]. Figure 5b indicates that the cloud clustering varies significantly throughout the diurnal cycle and also depends on the coarsening scale. We will investigate these aspects further in section 6. The changes in α closely resemble the behavior of R_V. In fact, we can remove the deviations in R_V caused by the deviations in α in each coarse grid box j. The CC06 theory states that the factor 2 in Eq. (3) comes in equal parts from the normalized variance of N and m. Therefore, we can correct this factor by taking into account the simulated variance of one of the two (or later both) distributions:

$$\alpha - \text{adjusted } R_V = \frac{1}{N_{\mathrm{box},n}} \sum_{j}^{N_{\mathrm{box},n}} \frac{\langle(\delta M)^2\rangle_j}{(1+\alpha_j)\langle M\rangle_j \langle m\rangle_j}. \qquad (14)$$

Doing so halves the amplitude of the diurnal variation for the medium and large scales (Fig. 5e). In other words, changes in cloud clustering in the convective life cycle are responsible for around 50% of the deviations from the CC06 variance predictions. The small and medium scales are now closely aligned, which suggests that cloud organization constitutes the main difference between these two scales (discussed further in section 6).

Similarly, for the cloud mass flux distribution,

$$\beta = \frac{1}{N_{\mathrm{box},n}} \sum_{j}^{N_{\mathrm{box},n}} \langle(\delta m)^2\rangle_j / \langle m\rangle_j^2 \qquad (15)$$

indicates whether the distribution is narrower or broader compared to an exponential distribution with the same mean. Figure 5c primarily shows a strong scale dependence but also a weaker diurnal cycle. Small and medium scales consistently have narrower distributions than large scales, for which β is around one. Again, we can account for the β deviations in R_V (Fig. 5f):

$$\beta - \text{adjusted } R_V = \frac{1}{N_{\mathrm{box},n}} \sum_{j}^{N_{\mathrm{box},n}} \frac{\langle(\delta M)^2\rangle_j}{(1+\beta_j)\langle M\rangle_j \langle m\rangle_j}. \qquad (16)$$

This shifts the means of the small and medium scales upward but also slightly decreases the diurnal amplitude. The large scales are hardly affected.

Finally, we remove both α and β deviations in R_V (Fig. 5d):

$$\alpha \text{ and } \beta - \text{adjusted } R_V = \frac{1}{N_{\mathrm{box},n}} \sum_{j}^{N_{\mathrm{box},n}} \frac{\langle(\delta M)^2\rangle_j}{(\alpha_j+\beta_j)\langle M\rangle_j \langle m\rangle_j}. \qquad (17)$$

According to the theory of random sums (Taylor and Karlin 1998, p. 72), the variance of M should be fully described by the variances of its underlying distributions N and m, assuming that the distribution of m, conditionally averaged on N, does not depend on N. This applies well to the small and medium scales, where R_V fluctuates around one after adjusting for differences in α and β. For the large scales, however, a significant deviation remains. This may be due to a correlation between the cloud number $N_{i,j,n}$ and the mean cloud mass flux $(1/N_{\mathrm{cld}\,i,j,n})\sum_k^{N_{\mathrm{cld}\,i,j,n}} m_{k,i,j,n}$ in the ensemble dimension i. Figure 6b shows the interensemble correlation coefficient between these two quantities. For the large scales, the two are negatively correlated, most strongly in the early afternoon. This corresponds well to the residual lack in variance and suggests that there is a large-scale constraint on the mass flux variance—ensemble members with more clouds have a lower mean cloud mass flux. This contradicts the CC06 theory, which states that m should be independent of the large-scale forcing. Cohen and Craig (2006) found that, in their experiments, changes in M were primarily accomplished by a response in N but m was also affected. It is possible that the large-scale constraint is artificially strong in the present simulations because of one-way nesting and identical initial and boundary conditions, which impairs the large-scale response to convective variability. Curiously, the medium and small scales show a strong positive correlation. This is likely an artifact of small sample sizes. An ensemble member with more clouds is also more likely to contain a larger cloud in a given region even if the underlying distributions are identical.

The analysis above highlights some systematic deviations of our simulation results from the CC06 predictions. The cloud number distribution $p(N)$, an indicator of cloud organization, varies with the time of day, predominantly affecting larger scales. Furthermore, the cloud mass flux distribution $p(m)$ is narrower for smaller scales. In the following two sections, we explore the physical processes behind these deviations.

6. Cloud clustering

The parameter α indicates that cloud clustering strongly affects convective variability. Now we

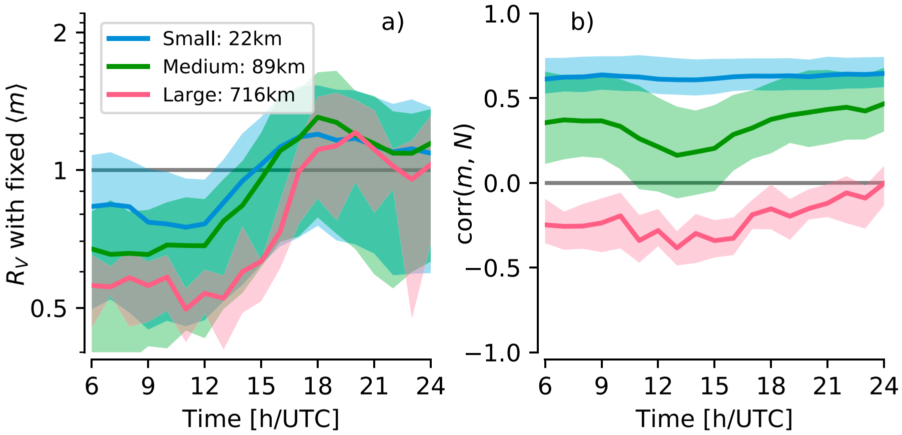


FIG. 6. (a) R_V computed with a prescribed $\langle m \rangle = 5.07 \times 10^7$ kg s^{-1}. (b) Interensemble correlation coefficient of N and m.

introduce another, independent metric of cloud clustering, the radial distribution function (RDF), which allows us to better understand how clouds organize. The RDF measures how many times more likely than random a cloud is located at a certain distance r to another cloud. A completely random distribution would give RDF = 1 for all r. The mathematical and computational details of the algorithm are explained in the accompanying Jupyter notebook cloud_identification_and_rdf.ipynb. To enable comparison with observation data, we use the hourly precipitation field for the RDF analysis. The results for the mass flux field are qualitatively similar. The RDF shows some variability between simulation days (see supplement). In particular, the first, synoptically dominated phase shows relatively constant RDFs, while the second, locally forced phase shows a stronger diurnal variation. This diurnal cycle also shows up in the composite in Fig. 7. Figure 7a shows the RDF as a function of the radius for two times, 1400 and 2100 UTC. The simulations with and without stochastic perturbations are very similar, confirming that the PSP scheme does not drastically alter the behavior of clouds. The observations have their peak RDF value at larger distances. This agrees with the visual impression that clouds in the model are more intermittent than in the observations, a characteristic of cloud-resolving models also observed in other studies (Hanley et al. 2015; Nguyen et al. 2017). In general, the RDF indicates that cloud clusters have a scale of around 50–100 km (25–50 km is the radius at which the RDFs drop to half their peak value) and that the clustering is stronger in the evening.

Since the RDF changes mostly in amplitude, not in shape, we use the maximum RDF value as a proxy of clustering strengths. Figure 7b therefore illustrates the diurnal variation in clustering. The results strongly correspond to the evolution of α: clouds are more clustered in the morning and evening. Similar results were found by Davies (2008) in her idealized diurnal cycle experiments. The scale dependence of α may be explained by the typical cluster size of around 50–100 km. The small scales are close to the typical cloud separation distance, around 10 km. Consequently, the clustering does not fully impact these scales. The medium scales correspond to the typical clustering size and, therefore, experience the strongest impact resulting in larger values of α. Finally, the large scales can contain many cloud clusters. During the day, these clusters appear to be more regularly spaced, leading to values of α smaller than one. This could be related to orography, the land surface, or synoptic variations within the domain. Toward the evening, the convective activity drops off rapidly. The few remaining active clusters are less constrained by the large-scale forcing and rely more on internal processes to maintain convection. Supporting evidence for this argument can be found in the ensemble precipitation variability (Fig. 3), which stays approximately constant even as the total precipitation amount rapidly declines.

To understand the nature of cloud clustering, it is important to note that the RDF measures the cloud clustering relative to all clouds in the domain. In fact, the absolute cloud number density around existing clouds changes little in the diurnal cycle. The variation in the RDF stems primarily from the increased isolation of the existing clusters in the morning and evening. Figure 8 illustrates a typical sequence of these events. In the morning, convection is clustered because of two processes: larger precipitation patches from the previous day in the south and spatially confined regions of early convective cells in the north. In the early afternoon, when convection is strongest, convective cells are

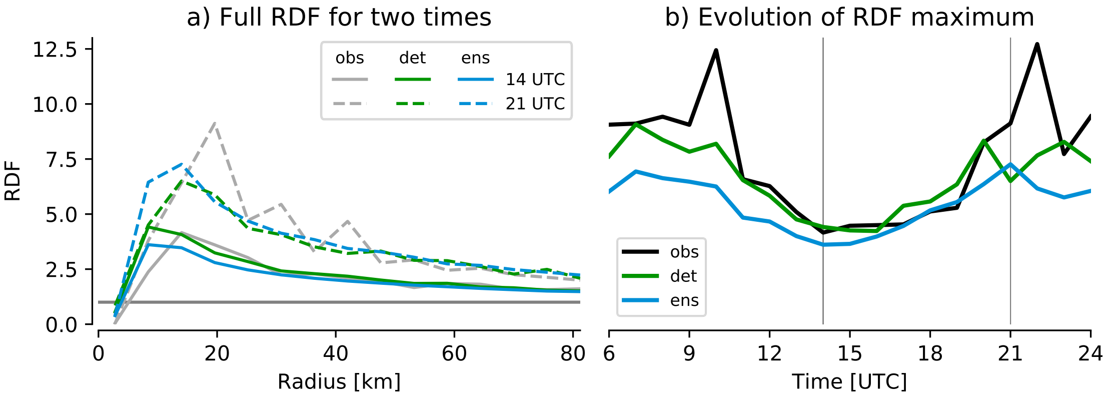


FIG. 7. (a) Radial distribution function of ensemble and deterministic simulations and observations for two times (1400 and 2100 UTC) as a composite over all 12 simulation days. The maximum search radius is $30\Delta x$ with a search step of $2\Delta x$. (b) Evolution of the RDF maximum as a function of time. The two vertical lines correspond to the times in (a).

distributed over most of the domain. Local clustering still occurs, but the clusters are simply more numerous. In the evening, as the forcing subsides, only strong, congregated convection survives. On this particular day, the formation of a squall line over western Germany is missed by all of the simulations. As mentioned in section 3, this failure of the model to produce larger organized features is apparent on several days.

7. Cloud mass flux distribution

The parameter β shows a substantial dependence on the coarse-graining scale and a weaker systematic diurnal cycle. In this section, we further investigate the cloud mass flux distribution. Specifically, we inspect the variations of the mean mass flux, the geographical variation of $\langle m \rangle$, and the impact of the cloud separation algorithm.

a. Temporal and spatial variations of $\langle m \rangle$

Figure 9 shows how the mean cloud mass flux and size varies with the time of day. Note that $\langle m \rangle$ increases as the total convective activity picks up, but its peak is delayed around 3 h behind $\langle M \rangle$. Also $\langle \sigma \rangle$ is smallest during the increase of convection around 1200 UTC and then increases toward the evening. The difference between the separated and nonseparated $\langle m \rangle$ are also strongest at that time; $\langle m \rangle$ fluctuates by about ±20% throughout the diurnal cycle, while $\langle M \rangle$ varies by a factor of 4. This is in line with previous findings of how the mean cloud mass flux changes with the forcing (Cohen and Craig 2006; Scheufele 2014): the primary effect of increasing the large-scale cooling is an increase in the number of clouds; the changes in the cloud properties are secondary. For this reason, $\langle m \rangle$ is often a prescribed constant, for example, in the Plant and Craig (2008) parameterization. To assess the impact of using a fixed $\langle m \rangle$, we compute R_V using the overall mean cloud mass flux of $5.07 \times 10^7\,\mathrm{kg\,s^{-1}}$ (Fig. 6a). This increases the diurnal variation for the small and medium scales. Additionally, the spread of these scales is increased. These changes indicate that the variations in $\langle m \rangle$ impact the CC06 variance predictions, but the magnitude is secondary compared to other systematic biases.

In section 5b, we found that small and medium scales seem to have a narrower than expected cloud mass flux distribution. The precipitation snapshots (Figs. 1 and 8 and supplemental material) lead us to hypothesize that $\langle m \rangle$ varies geographically. There appear to be regions with mostly larger clouds and other regions with mostly smaller clouds. This could result from differences in the synoptic situation, orography, cloud–cloud interactions, or land surface variations leading to changes in the Bowen ratio, which was found to be important for the shallow cumulus mass flux distribution (Sakradzija and Hohenegger 2017). On the domain scale, regions with different mean cloud sizes are included, potentially increasing the width of cloud mass flux distribution.

b. Overall cloud mass flux distribution—The impact of cloud separation

Next, we take a closer look at the overall cloud size distribution for all dates, times, and ensemble members (Fig. 10). If the clouds are separated using the local maximum method, the distribution is close to exponential. Clouds collect near the grid scale, an observation also made by Scheufele (2014). Without the cloud separation, the distribution is closer to a power law. Windmiller (2017) argues that a power-law cloud size distribution is the result of cloud clustering when individual clouds drawn from an exponential distribution overlap. It is curious, however, that this effect already shows up at resolutions of 2.8 km. Previous kilometer-scale studies in idealized setups (Cohen and Craig 2006; Scheufele 2014) found that cloud overlap only became significant at much higher resolutions. This

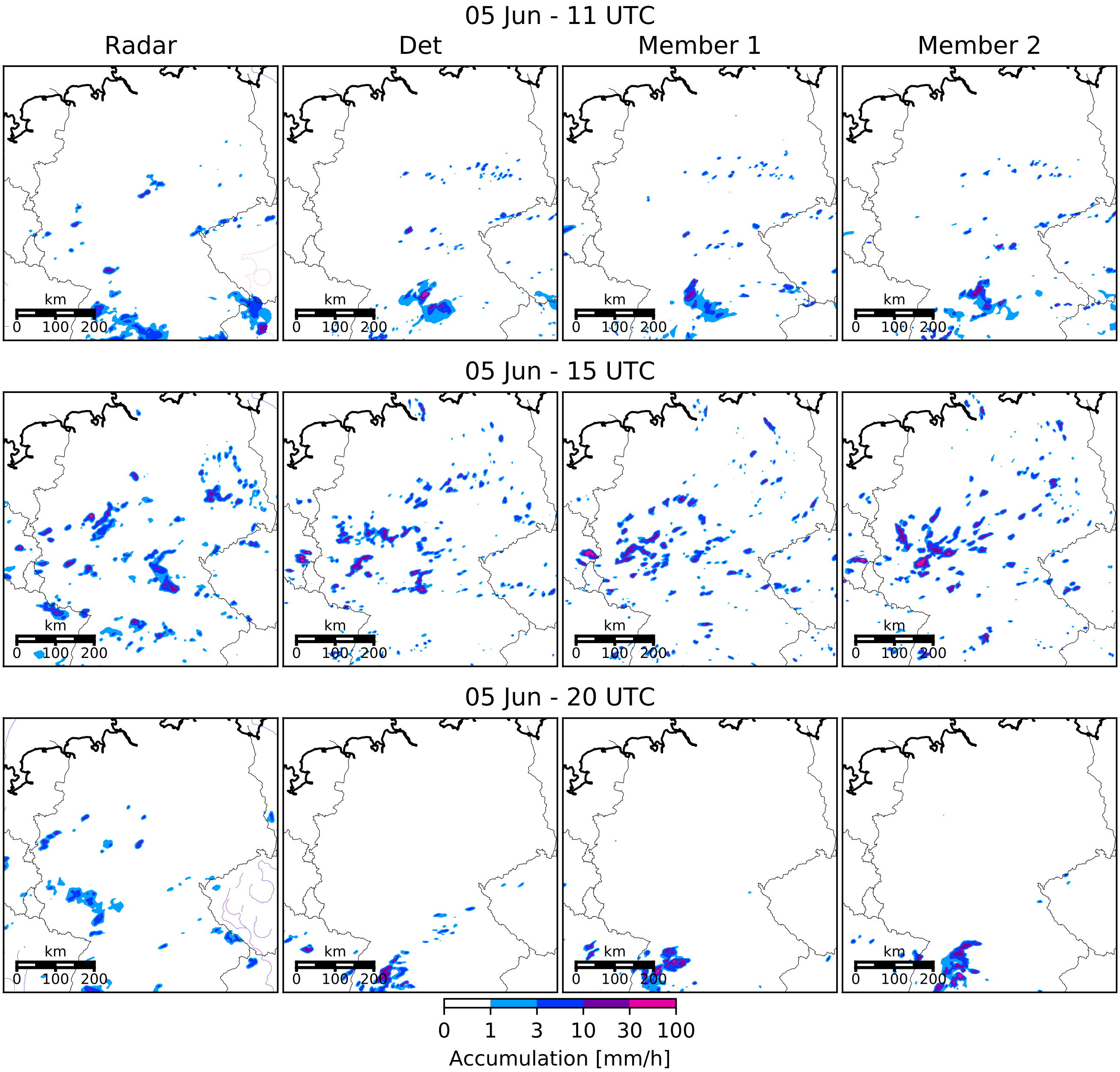


FIG. 8. Hourly precipitation snapshots at 1100, 1500, and 2000 UTC 5 Jun 2016 for radar observations, deterministic, and two ensemble simulations. Contours in the radar snapshots indicate radar coverage.

seems to suggest that in our real-weather case studies, clouds tend to congregate more. The deterministic run and the stochastically perturbed ensemble agree well for smaller mass fluxes. The differences for the larger bins are most likely caused by the much smaller sample size of the deterministic run.

8. Implications for (stochastic) parameterizations of convection

Much of the interest in a theory for convective variability such as CC06 comes from the application to stochastic convective parameterization. The theory provides guidance for how the stochastic variability should change with meteorological situation and model resolution. The evaluation of the theory presented here has both positive and negative implications for a convection scheme, like that of Plant and Craig (2008), which is based on the CC06 theory. Most importantly, the basic square root scaling of mass flux standard deviation with the total mass flux in any finite region holds to a reasonable degree of approximation. Although the convective cases considered here are much more complex than the radiative–convective equilibrium environment for which the theory was first developed and

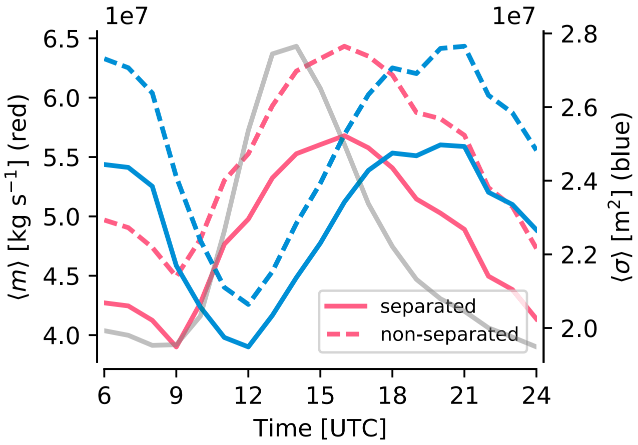


FIG. 9. Evolution of mean cloud mass flux (red) and mean cloud size (blue) for separated (solid) and unseparated (dashed) clouds as a function of time as a composite over all days. The gray line indicates the evolution of $\langle M \rangle$.

tested, the additional complexity does not completely alter the behavior. Since it is the variability scaling that relates the amplitude of convective variability to both the convective closure and the model resolution, its use in stochastic parameterization is supported. In contrast, the most commonly used stochastic parameterization method in numerical weather prediction, SPPT, scales the standard deviation linearly proportional to the convective amount, which is not supported by our results. This does not imply that SPPT should not be used since it is not a convection parameterization but an "all inclusive" method to account for model errors and primarily increase ensemble spread. Furthermore, SPPT must also represent other processes that may scale differently. However, to the extent that small-scale variability is associated with convective clouds, the incorrect scaling implies that the scheme will need to be retuned whenever factors such as model resolution are changed.

However, the significant deviations from CC06 theory found in this study should be accounted for in parameterization. This is not simple, since the dominant deviation was related to convective clustering on a characteristic scale of order 100 km, varying in intensity throughout the diurnal cycle. Unlike the simple model of unorganized convection, the clustering couples different grid columns in the large-scale model—an effect that can be difficult to implement. The model dynamics can only represent organization on scales larger than the effective resolution, approximately $5\Delta x$ (Skamarock 2004). Furthermore, even within a grid box, convective clustering may impact other parameterized processes including radiation and surface fluxes, which would need to be accounted for in the relevant schemes. The changes in $\langle m \rangle$ are technically easier to implement in a stochastic convection scheme, but a good understanding of what determines $\langle m \rangle$ is still lacking. Additionally, we note that it is important to consider the individual updrafts rather than overlapping features to construct distributions that agree with the theory.

Finally, it should be emphasized that we focused on spatial variability in this study and did not address the important question of temporal structures or convective memory, both of which are important for parameterization. The Plant and Craig (2008) parameterization has some memory by giving each random cloud a fixed cloud lifetime larger than the convective time step. Sakradzija et al. (2016) further related the cloud lifetime to the mass flux. No attempt has been made, however, to include the effect of organization on convective memory.

9. Conclusions

In this study, we tested a minimally simple theory of convective variability developed by Craig and Cohen (2006) in complex summertime weather over land.

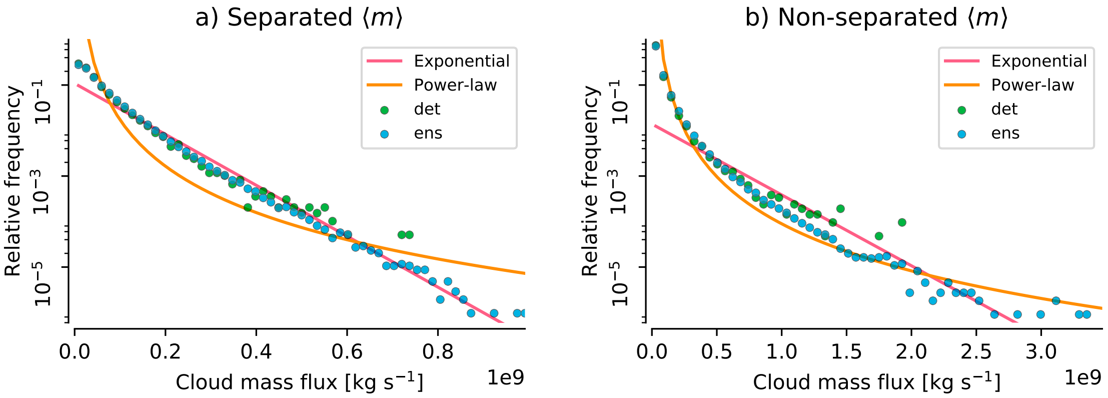


FIG. 10. Histogram of cloud mass flux $\langle m \rangle$ for all dates and times: (a) separated and (b) nonseparated. An exponential and a power-law curve are fitted using a least squares algorithm. The bin widths are $1.67 \times 10^7\,\mathrm{kg\,s^{-1}}$ in (a) and $5.93 \times 10^6\,\mathrm{kg\,s^{-1}}$ in (b).

The theory's core building blocks, a random distribution of clouds in space and an exponential mass flux distribution, were mainly developed with tropical maritime convection in mind. We chose 12 consecutive, high-impact weather days over Germany to test the theory's prediction outside of its comfort zone. To quantify the variability of convection in real-weather case studies, we set up a 50-member ensemble in which all members had the same large-scale conditions but the convective cells were displaced using stochastic boundary layer perturbations.

In general, the mass flux standard deviation scales with the square root of its mean, in accordance with the theory. We did find systematic deviations, however. First, clouds tend to be clustered, violating the no cloud–cloud interaction assumption, more strongly so in the morning and evening. The typical cluster size is around 50–100 km. Second, the mean mass flux per cloud varies geographically—clouds of a certain size appear to congregate—and temporally, by around ±20% in the diurnal cycle. This indicates that the cloud properties are not entirely independent of the large-scale conditions, as assumed by the theory.

Our findings support the applicability of the CC06 theory for stochastic parameterizations in global models but also highlight areas for improvement. Particularly, the organization of clouds, most likely caused by cold pool dynamics, remains an outstanding issue. The incorporation of cloud organization in convection parameterization requires either a diagnostic grid-scale indicator of subgrid clustering or a prognostic subgrid variable.

While the 12-day period in our study presents a variety of nonequilibrium convections, they still only represent a narrow selection of all possible convective situations. In particular, it would be interesting to investigate how well the theory holds up for larger organized systems such as mesoscale convective systems or even tropical cyclones. Furthermore, the realism of our simulations is somewhat limited by our grid spacing. As shown in comparison with radar observations, our kilometer-scale model produces a large number of gridcell storms and is, on several days, not able to create lasting organization into the night. A more detailed investigation of the resolution dependence of cloud statistics would certainly be desirable, particularly as recent studies (Craig and Dörnbrack 2008; Scheufele 2014; Hanley et al. 2015; Heath et al. 2017) show a lack of convergence even at horizontal grid spacings of around 100 m.

Acknowledgments. The research leading to these results has been done within the subproject "Representing forecast uncertainty using stochastic physical parameterizations" of the Transregional Collaborative Research Center SFB/TRR 165 "Waves to Weather" funded by the German Research Foundation (DFG). We thank Robert Redl for technical support and the German Weather Service (DWD) for providing the COSMO model and the initial and boundary conditions, as well as ECMWF for providing the data for our synoptic charts. Last, we would like to thank the three anonymous reviewers for their help in improving this paper.

APPENDIX

Rationale and Formulation of the Stochastic Boundary Layer Perturbations

In the atmosphere, the boundary layer is characterized by turbulent eddies, which occur on a wide range of scales from millimeters to approximately the height of the boundary layer, typically around 1 km. Convection-permitting models with horizontal grid spacings of order 1 km are not able to represent these eddies. Therefore, parameterizations are necessary to describe the effect of the unresolved turbulence on the grid scale. Usually, these boundary layer parameterizations are assumed to represent the average effect of many eddies and are deterministic in nature: given a certain grid-scale condition, they always produce the same mean response. The only variance thus comes from the resolved boundary layer circulation. On a scale equivalent to the grid length of a kilometer-scale model, however, the turbulent response can vary significantly from realization to realization. Therefore, the variability in the model, expressed by the joint probability density function (PDF) of boundary layer quantities, can be much smaller than the corresponding variability in nature. While the mean boundary evolution might still be adequately represented, this lack of variability can drastically alter the grid-scale behavior if nonlinear convection occurs.

On typical summer days, turbulence is driven by surface heating, leading to a growth of the boundary layer after sunrise until the capping inversion is reached. At this point, only parcels with enough momentum as well as positive humidity and temperature perturbations can break through the inversion layer to eventually trigger deep convection. Parcels that have these properties originate from the extreme end of the joint boundary layer PDF. Reduced variability accordingly reduces the probability of such parcels existing. This can lead to systematic biases in model behavior. In other words, there is a disparity between the convection, which is assumed to be resolved, and the process responsible for triggering it, namely, boundary layer turbulence, which is not resolved. Note that operational models such as the COSMO model, which is specifically designed for precipitation forecasts, are often tuned to produce the correct diurnal cycle of precipitation at the expense of other biases.

The PSP scheme aims to reintroduce the missing variability on the smallest model-resolved scale, around $\Delta x_{\mathrm{eff}} = 5\Delta x$ (Bierdel et al. 2012). In other words, perturbations with a scale equal to the effective model resolution Δx_{eff} and with an amplitude proportional to the subgrid standard deviation $\sqrt{\overline{\Phi'^2}}$ of each variable $\Phi \in \{T, q_v, w\}$ are added to resolved flow. The perturbations are introduced as a forcing term in the model equations that persists over a representative eddy lifetime τ_{eddy}. Mathematically, this can be expressed as

$$\partial_t \Phi|_{\mathrm{PSP}} = \alpha_{\mathrm{tuning}} \eta \frac{1}{\tau_{\mathrm{eddy}}} \frac{l_{\mathrm{eddy}}}{\Delta x_{\mathrm{eff}}} \sqrt{\overline{\Phi'^2}}. \qquad \text{(A1)}$$

The subgrid standard deviation $\sqrt{\overline{\Phi'^2}}$ is taken directly from the turbulence scheme. In the case of the COSMO model, this is a 1.5-order closure (Raschendorfer 2001) based on level 2.5 of Mellor and Yamada (1982). Here, the second moments are diagnostically computed based on the turbulence kinetic energy and the vertical gradient of the variable in question. The factor $l_{\mathrm{eddy}}/\Delta x_{\mathrm{eff}}$ scales the amplitude of the perturbations to the grid length; $l_{\mathrm{eddy}} = 1\,\mathrm{km}$ is the typical size of an eddy spanning the daytime convective boundary layer. Therefore, this ratio is equal to $1/\sqrt{N_{\mathrm{eddy}}}$, where N_{eddy} is the number of eddies in a square with edge length Δx_{eff}. This follows the CC06 theory for convective variability by assuming that the variability of the domain total depends on the number of elements in question. A larger domain relative to the eddy size contains more eddies and the variability is, therefore, reduced. We define η as a two-dimensional random field with mean zero and standard deviation one, which is horizontally correlated using a Gaussian kernel with half width $2.5\Delta x$. The random field is kept constant for $\tau_{\mathrm{eddy}} = 10\,\mathrm{min}$, after which a completely new random field is drawn. Finally, α_{tuning} is a tuning factor, which is set to 7.2. Note that the value of the tuning factor is different than in Kober and Craig (2016) because of the changes in the formulation. The tuning factor was chosen so that the effects of the PSP scheme were noticeable but reasonable (Kober and Craig 2016).

REFERENCES

Baldauf, M., A. Seifert, J. Förstner, D. Majewski, M. Raschendorfer, and T. Reinhardt, 2011: Operational convective-scale numerical weather prediction with the COSMO model: Description and sensitivities. *Mon. Wea. Rev.*, **139**, 3887–3905, https://doi.org/10.1175/MWR-D-10-05013.1.

Bengtsson, L., M. Steinheimer, P. Bechtold, and J.-F. Geleyn, 2013: A stochastic parametrization for deep convection using cellular automata. *Quart. J. Roy. Meteor. Soc.*, **139**, 1533–1543, https://doi.org/10.1002/qj.2108.

Berner, J., and Coauthors, 2016: Stochastic parameterization: Toward a new view of weather and climate models. *Bull. Amer. Meteor. Soc.*, **98**, 565–588, https://doi.org/10.1175/BAMS-D-15-00268.1.

Beucher, S., and F. Meyer, 1992: The morphological approach to segmentation: The watershed transformation. *Mathematical Morphology in Image Processing*, E. R. Dougherty, Ed., Marcel Dekker, 433–481.

Bierdel, L., P. Friederichs, and S. Bentzien, 2012: Spatial kinetic energy spectra in the convection-permitting limited-area NWP model COSMO-DE. *Meteor. Z.*, **21**, 245–258, https://doi.org/10.1127/0941-2948/2012/0319.

Buizza, R., M. Milleer, and T. N. Palmer, 1999: Stochastic representation of model uncertainties in the ECMWF ensemble prediction system. *Quart. J. Roy. Meteor. Soc.*, **125**, 2887–2908, https://doi.org/10.1002/qj.49712556006.

Christensen, H. M., J. Berner, D. R. B. Coleman, and T. N. Palmer, 2017: Stochastic parameterization and El Niño–Southern Oscillation. *J. Climate*, **30**, 17–38, https://doi.org/10.1175/JCLI-D-16-0122.1.

Cohen, B. G., and G. C. Craig, 2006: Fluctuations in an equilibrium convective ensemble. Part II: Numerical experiments. *J. Atmos. Sci.*, **63**, 2005–2015, https://doi.org/10.1175/JAS3710.1.

Craig, G. C., and B. G. Cohen, 2006: Fluctuations in an equilibrium convective ensemble. Part I: Theoretical formulation. *J. Atmos. Sci.*, **63**, 1996–2004, https://doi.org/10.1175/JAS3709.1.

——, and A. Dörnbrack, 2008: Entrainment in cumulus clouds: What resolution is cloud-resolving? *J. Atmos. Sci.*, **65**, 3978–3988, https://doi.org/10.1175/2008JAS2613.1.

Davies, L., 2008: Self-organisation of convection as a mechanism for memory. Ph.D. thesis, University of Reading, 157 pp.

Davoudi, J., N. A. McFarlane, and T. Birner, 2010: Fluctuation of mass flux in a cloud-resolving simulation with interactive radiation. *J. Atmos. Sci.*, **67**, 400–418, https://doi.org/10.1175/2009JAS3215.1.

Doms, G., and Coauthors, 2011: A description of the nonhydrostatic regional COSMO model. Part II: Physical parameterization. COSMO Tech. Rep., 161 pp.

Dorrestijn, J., D. T. Crommelin, J. A. Biello, and S. J. Böing, 2013: A data-driven multi-cloud model for stochastic parametrization of deep convection. *Philos. Trans. Roy. Soc. London*, **371A**, 20120374, https://doi.org/10.1098/rsta.2012.0374.

DWD, 2017: Analysen radarbasierter stündlicher (RW) und täglicher (SF) Niederschlagshöhen. DWD, https://www.dwd.de/DE/leistungen/radolan/radolan.html.

Hanley, K. E., R. S. Plant, T. H. M. Stein, R. J. Hogan, J. C. Nicol, H. W. Lean, C. Halliwell, and P. A. Clark, 2015: Mixing-length controls on high-resolution simulations of convective storms. *Quart. J. Roy. Meteor. Soc.*, **141**, 272–284, https://doi.org/10.1002/qj.2356.

Heath, N. K., H. E. Fuelberg, S. Tanelli, F. J. Turk, R. P. Lawson, S. Woods, and S. Freeman, 2017: WRF nested large-eddy simulations of deep convection during SEAC^{4}RS. *J. Geophys. Res. Atmos.*, **122**, 3953–3974, https://doi.org/10.1002/2016JD025465.

Hunter, J. D., 2007: Matplotlib: A 2D graphics environment. *Comput. Sci. Eng.*, **9**, 90–95, https://doi.org/10.1109/MCSE.2007.55.

Irving, D., 2016: A minimum standard for publishing computational results in the weather and climate sciences. *Bull.*

Amer. Meteor. Soc., **97**, 1149–1158, https://doi.org/10.1175/BAMS-D-15-00010.1.

Jones, E., and Coauthors, 2001: SciPy: Open source scientific tools for Python. Accessed 28 August 2017, http://www.scipy.org.

Jones, T. R., and D. A. Randall, 2011: Quantifying the limits of convective parameterizations. *J. Geophys. Res.*, **116**, D08210, https://doi.org/10.1029/2010JD014913.

Keane, R. J., G. C. Craig, C. Keil, and G. Zängl, 2014: The Plant–Craig stochastic convection scheme in ICON and its scale adaptivity. *J. Atmos. Sci.*, **71**, 3404–3415, https://doi.org/10.1175/JAS-D-13-0331.1.

Kluyver, T., and Coauthors, 2016: Jupyter notebooks—A publishing format for reproducible computational workflows. Positioning and Power in Academic Publishing: Players, *Agents and Agendas*, IOS Press, 87–90, https://doi.org/10.3233/978-1-61499-649-1-87.

Kober, K., and G. C. Craig, 2016: Physically based stochastic perturbations (PSP) in the boundary layer to represent uncertainty in convective initiation. *J. Atmos. Sci.*, **73**, 2893–2911, https://doi.org/10.1175/JAS-D-15-0144.1.

Mellor, G. L., and T. Yamada, 1982: Development of a turbulence closure model for geophysical fluid problems. *Rev. Geophys.*, **20**, 851–875, https://doi.org/10.1029/RG020i004p00851.

Nguyen, H., A. Protat, H. Zhu, and M. Whimpey, 2017: Sensitivity of the ACCESS forecast model statistical rainfall properties to resolution. *Quart. J. Roy. Meteor. Soc.*, **143**, 1967–1977, https://doi.org/10.1002/qj.3056.

Ollinaho, P., and Coauthors, 2017: Towards process-level representation of model uncertainties: Stochastically perturbed parametrizations in the ECMWF ensemble. *Quart. J. Roy. Meteor. Soc.*, **143**, 408–422, https://doi.org/10.1002/qj.2931.

Piper, D., M. Kunz, F. Ehmele, S. Mohr, B. Mühr, A. Kron, and J. Daniell, 2016: Exceptional sequence of severe thunderstorms and related flash floods in May and June 2016 in Germany—Part 1: Meteorological background. *Nat. Hazards Earth Syst. Sci.*, **16**, 2835–2850, https://doi.org/10.5194/nhess-16-2835-2016.

Plant, R. S., and G. C. Craig, 2008: A stochastic parameterization for deep convection based on equilibrium statistics. *J. Atmos. Sci.*, **65**, 87–105, https://doi.org/10.1175/2007JAS2263.1.

Raschendorfer, M., 2001: The new turbulence parameterization of LM. *COSMO Newsletter*, No. 1, DWD, Offenbach, Germany, 89–97, http://www.cosmo-model.org/content/model/documentation/newsLetters/newsLetter01/newsLetter_01.pdf.

Rasp, S., 2017: Variability and clustering of mid-latitude summertime convection—Metadata. Figshare, accessed 23 November 2017, https://doi.org/10.6084/m9.figshare.c.3864490.

Sakradzija, M., and C. Hohenegger, 2017: What determines the distribution of shallow convective mass flux through a cloud base? *J. Atmos. Sci.*, **74**, 2615–2632, https://doi.org/10.1175/JAS-D-16-0326.1.

——, A. Seifert, and T. Heus, 2015: Fluctuations in a quasi-stationary shallow cumulus cloud ensemble. *Nonlinear Processes Geophys.*, **22**, 65–85, https://doi.org/10.5194/npg-22-65-2015.

——, ——, and A. Dipankar, 2016: A stochastic scale-aware parameterization of shallow cumulus convection across the convective gray zone. *J. Adv. Model. Earth Syst.*, **8**, 786–812, https://doi.org/10.1002/2016MS000634.

Scheufele, K., 2014: Resolution dependence of cumulus statistics in radiative-convective equilibrium. Ph.D. thesis, Ludwig–Maximilian University of Munich, 121 pp.

Selz, T., and G. C. Craig, 2015a: Simulation of upscale error growth with a stochastic convection scheme. *Geophys. Res. Lett.*, **42**, 3056–3062, https://doi.org/10.1002/2015GL063525.

——, and ——, 2015b: Upscale error growth in a high-resolution simulation of a summertime weather event over Europe. *Mon. Wea. Rev.*, **143**, 813–827, https://doi.org/10.1175/MWR-D-14-00140.1.

Shutts, G. J., and T. N. Palmer, 2007: Convective forcing fluctuations in a cloud-resolving model: Relevance to the stochastic parameterization problem. *J. Climate*, **20**, 187–202, https://doi.org/10.1175/JCLI3954.1.

——, and A. C. Pallarès, 2014: Assessing parametrization uncertainty associated with horizontal resolution in numerical weather prediction models. *Philos. Trans. Roy. Soc. London*, **372A**, 20130284, https://doi.org/10.1098/rsta.2013.0284.

Skamarock, W. C., 2004: Evaluating mesoscale NWP models using kinetic energy spectra. *Mon. Wea. Rev.*, **132**, 3019–3032, https://doi.org/10.1175/MWR2830.1.

Stauffer, R., G. J. Mayr, M. Dabernig, and A. Zeileis, 2015: Somewhere over the rainbow: How to make effective use of colors in meteorological visualizations. *Bull. Amer. Meteor. Soc.*, **96**, 203–216, https://doi.org/10.1175/BAMS-D-13-00155.1.

Taylor, H. M., and S. Karlin, 1998: *An Introduction to Stochastic Modeling.* Academic Press, 646 pp.

van der Walt, S., S. C. Colbert, and G. Varoquaux, 2011: The NumPy array: A structure for efficient numerical computation. *Comput. Sci. Eng.*, **13**, 22–30, https://doi.org/10.1109/MCSE.2011.37.

Wang, Y., and G. J. Zhang, 2016: Global climate impacts of stochastic deep convection parameterization in the NCAR CAM5. *J. Adv. Model. Earth Syst.*, **8**, 1641–1656, https://doi.org/10.1002/2016MS000756.

——, ——, and G. C. Craig, 2016: Stochastic convective parameterization improving the simulation of tropical precipitation variability in the NCAR CAM5. *Geophys. Res. Lett.*, **43**, 6612–6619, https://doi.org/10.1002/2016GL069818.

Windmiller, J. M., 2017: Organization of tropical convection. Ph.D. thesis, Ludwig–Maximilian University of Munich, 137 pp., https://edoc.ub.uni-muenchen.de/21245/.

Supplementary Figures for *Variability and clustering of mid-latitude summertime convection: Testing the Craig and Cohen (2006) theory in a convection-permitting ensemble with stochastic boundary layer perturbations*

Stephan Rasp, Tobias Selz and George C. Craig

Description of `prec_stamps_stamps_nens_2_all_days.pdf/.gif`

Hourly precipitation snapshots of radar observations, deterministic run and two ensemble members for the entire simulation period. As Fig. 10 of main paper.

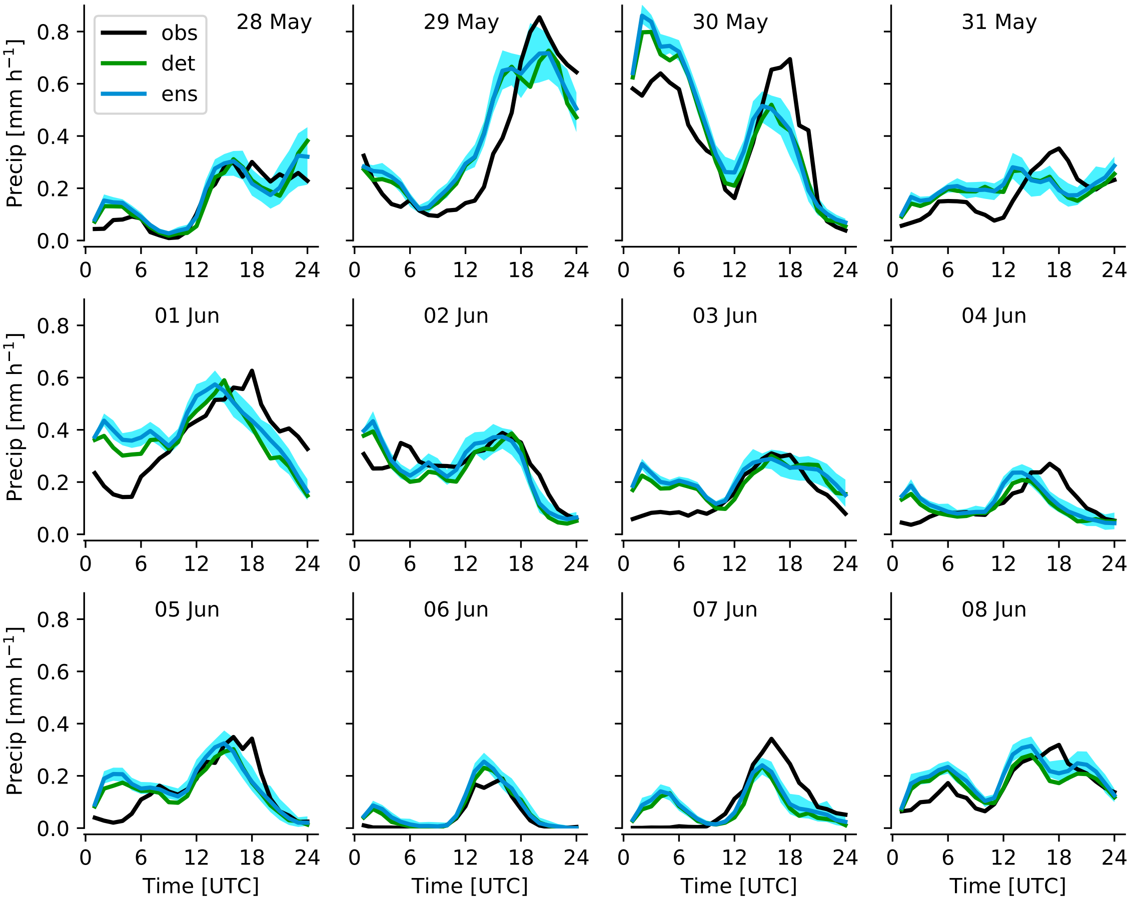


Figure 1: Domain-averaged hourly precipitation accumulation in mm h^{-1} for observations (black), deterministic run (green) and ensemble (blue). Shaded area indicates ensemble range. Within one day only grid points are considered where radar data is available for all time steps.

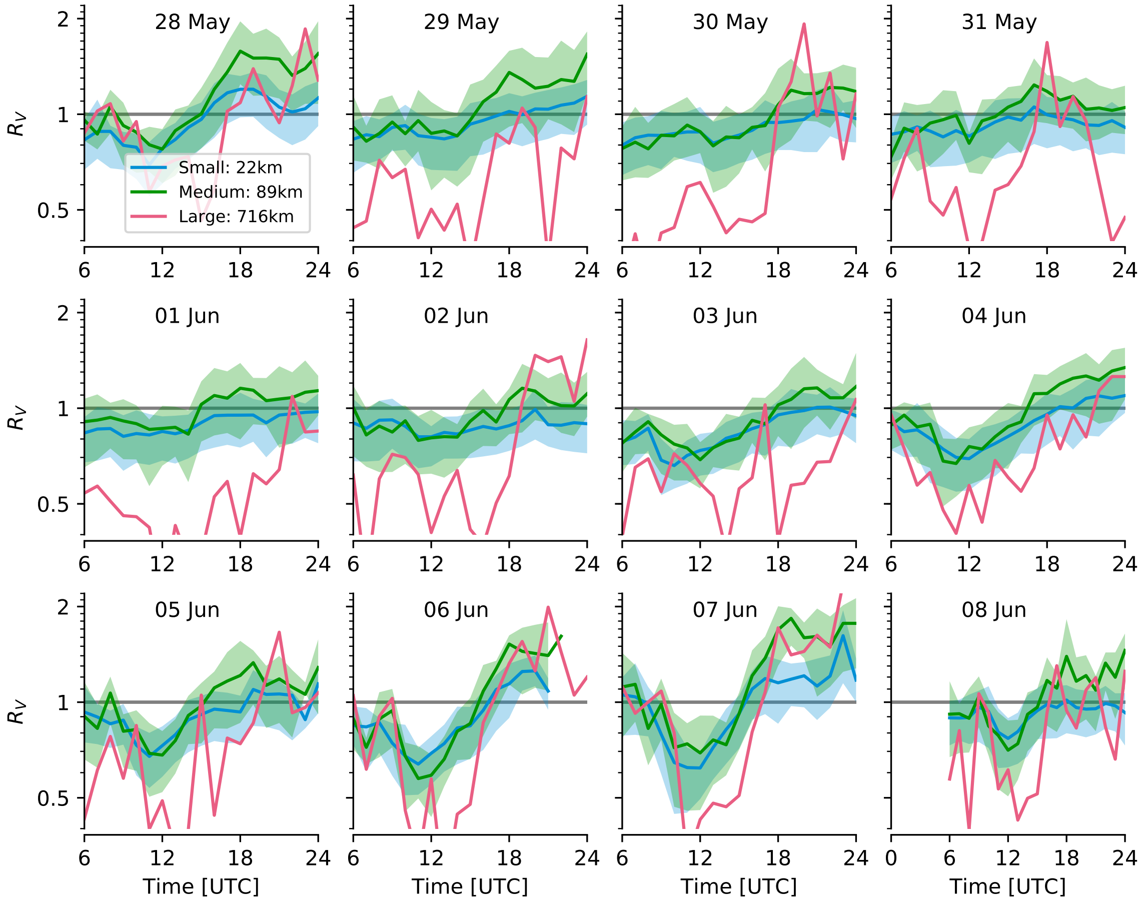


Figure 2: R_V for each day individually. Colors and scales as in Fig. 5a of main paper.

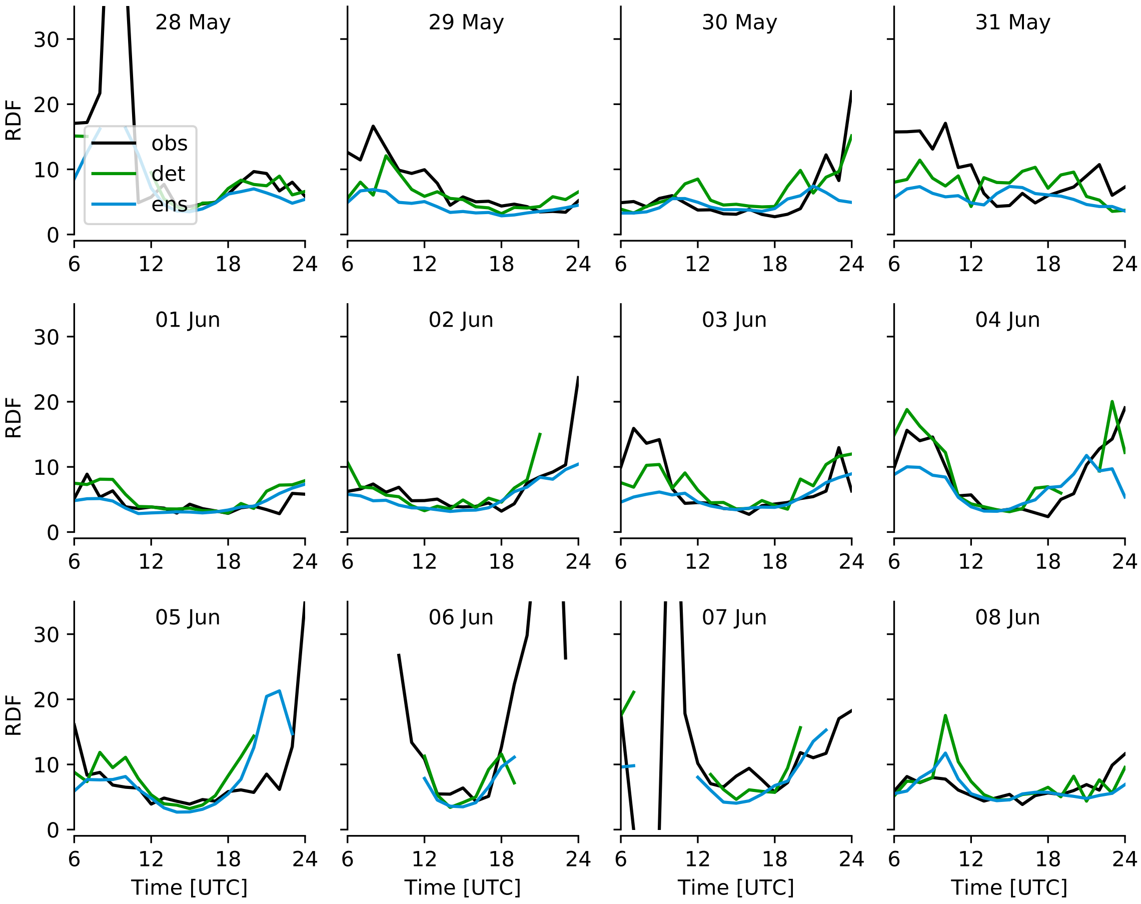


Figure 3: RDF for each day individually. As Fig. 7b of main paper.

2.2 P2: Feasibility study for a unified deep learning parameterization

COULD MACHINE LEARNING BREAK THE CONVECTION PARAMETERIZATION DEADLOCK?

Pierre Gentine, Michael S. Pritchard, **Stephan Rasp**, Gael Reinaudi and
Galen Yacalis, 2018.
Geophysical Research Letters, 45, 5742–5751.

Context In this paper, we show that a deep neural network can learn to predict subgrid tendencies from a cloud-resolving dataset. In this paper, we focus on offline tests, i.e. the neural network parameterization is not yet run in prognostic mode. The paper contains sensitivity studies for the required amount of training data and neural network architecture.

Author contribution PG and MP designed research. MP created the training dataset. PG and GR designed the initial neural network with help from GY. I trained the neural networks and conducted analysis for the results in the paper. PG led the writing of the paper with input from MP, GY and myself.

ADVANCING EARTH AND SPACE SCIENCE

Geophysical Research Letters

RESEARCH LETTER

10.1029/2018GL078202

Key Points:

- We use a global atmospheric model with embedded cloud resolving model, as training data set for a machine learning algorithm of convection
- The machine learning algorithm can reproduce most of the key features of the embedded cloud resolving model
- The machine learning algorithm is much more computationally efficient than a super parameterization, but does not behave stochastically

Supporting Information:

- Supporting Information S1

Correspondence to:
P. Gentine,
pg2328@columbia.edu

Citation:
Gentine, P., Pritchard, M., Rasp, S., Reinaudi, G., & Yacalis, G. (2018). Could machine learning break the convection parameterization deadlock? *Geophysical Research Letters*, *45*, 5742–5751. https://doi.org/10.1029/2018GL078202

Received 6 APR 2018
Accepted 18 MAY 2018
Accepted article online 23 MAY 2018
Published online 12 JUN 2018

Could Machine Learning Break the Convection Parameterization Deadlock?

P. Gentine[1], M. Pritchard[2], S. Rasp[3], G. Reinaudi[1], and G. Yacalis[2]

[1]Earth and Environmental Engineering, Columbia University, New York, NY, USA, [2]Earth System Science, University of California, Irvine, CA, USA, [3]Faculty of Physics, LMU Munich, Munich, Germany

Abstract Representing unresolved moist convection in coarse-scale climate models remains one of the main bottlenecks of current climate simulations. Many of the biases present with parameterized convection are strongly reduced when convection is explicitly resolved (i.e., in cloud resolving models at high spatial resolution approximately a kilometer or so). We here present a novel approach to convective parameterization based on machine learning, using an aquaplanet with prescribed sea surface temperatures as a proof of concept. A deep neural network is trained with a superparameterized version of a climate model in which convection is resolved by thousands of embedded 2-D cloud resolving models. The machine learning representation of convection, which we call the Cloud Brain (CBRAIN), can skillfully predict many of the convective heating, moistening, and radiative features of superparameterization that are most important to climate simulation, although an unintended side effect is to reduce some of the superparameterization's inherent variance. Since as few as three months' high-frequency global training data prove sufficient to provide this skill, the approach presented here opens up a new possibility for a future class of convection parameterizations in climate models that are built "top-down," that is, by learning salient features of convection from unusually explicit simulations.

Plain Language Summary The representation of cloud radiative effects and the atmospheric heating and moistening due to moist convection remains a major challenge in current generation climate models, leading to a large spread in climate prediction. Here we show that neural networks trained on a high-resolution model in which moist convection is resolved can be an appealing technique to tackle and better represent moist convection in coarse resolution climate models.

1. Introduction

Convective parameterization remains one of the main roadblocks to weather and climate prediction (Bony et al., 2015; Medeiros et al., 2014; Sherwood et al., 2014; Stevens & Bony, 2013). In fact, most of the intermodel spread in equilibrium climate sensitivity can be traced back to the representation of clouds (Schneider et al., 2017). Convective schemes exhibit systematic biases in the vertical structure of heating and moistening, precipitation intensity, and cloud cover (Daleu et al., 2015, 2016). These errors, in turn, feed back onto larger-scale circulations, deteriorating general circulation model (GCM) simulations, and prediction skill (Bony et al., 2015). A challenge in current convective schemes is representing the transitions between different types of convection, such as the transition from shallow to deep convection (Couvreux et al., 2015; D'Andrea et al., 2014; Khouider et al., 2003, 2010; Khouider & Majda, 2006; Peters et al., 2013; Rochetin, Couvreux, et al., 2014; Rochetin, Grandpeix, et al., 2014), which is especially crucial to predicting both continental precipitation and modes of climate variability (Arnold et al., 2014). In addition, most convective parameterizations do not represent important processes, such as convective aggregation, which are essential to accurately predicting the response of clouds and precipitation to global warming, as well as modes of climate variability (Arnold & Randall, 2015; Bony et al., 2015; Bretherton & Khairoutdinov, 2015; Coppin & Bony, 2015; Jeevanjee & Romps, 2013; Muller & Bony, 2015; Wing & Emanuel, 2014).

Current generation climate models (and typical weather forecast models) with parameterized convection do not capture much of the degree of organization, nor do they represent mesoscale convective systems (MCS; Hohenegger & Stevens, 2016). MCS and the impact of shear are, however, crucial for correct representation of rainfall and radiative feedback (Cao & Zhang, 2017; Houze, 2004; Moncrieff, 2010; Moncrieff et al., 2012, 2017; Tan et al., 2015). Finally, another challenge is that climate sensitivity is strongly related to the interaction between deep and shallow convection (Bony et al., 2015), and the coupling between clouds, convection, and the large-scale circulation, which is currently poorly captured by parameterized convection (Bony et al., 2015; Daleu et al., 2016; Hohenegger & Stevens, 2016; Nie et al., 2016).

Many of the previously mentioned problems related to the representation of convection are partly alleviated when using convective-permitting resolutions, that is, at horizontal grid spacing of ~2 km or less. For instance, the transition between shallow and deep convection can be correctly captured at convective permitting scale (Khairoutdinov et al., 2009; Khairoutdinov & Randall, 2006). Convective aggregation is observed at convective permitting scale (Hohenegger & Stevens, 2016) so that cloud resolving models (CRMs) have been the tool of choice to understand convective aggregation (Arnold & Randall, 2015; Bony et al., 2015; Bretherton & Khairoutdinov, 2015; Coppin & Bony, 2015; Jeevanjee & Romps, 2013; Muller & Bony, 2015; Wing & Emanuel, 2014). CRMs (at convective permitting scales <2 km) also correctly reproduce MSCs and squall lines (Moncrieff & Liu, 2006; Taylor et al., 2009), as well as extreme precipitation events driven by larger scale anomalies. Convective-permitting simulations better represent modes of tropical climate variability (Arnold et al., 2014), shallow to deep convection (Guichard et al., 2004), and mesoscale propagation (Hohenegger et al., 2015).

Therefore, modeling at convective-permitting scales is transformative to the representation of convection. It is, however, impractical at present to use convective resolving resolution at global scale for climate prediction given its computational requirements (Satoh et al., 2008). While global cloud resolving models (GCRMs) can be run easily for months, multidecadal simulations are computationally challenging. To alleviate this problem, an interesting approach has been to use cloud "superparameterization (SP)," which computes the subgrid vertical heating and moistening profiles within a GCM grid cell by sampling a curtain of an embedded 2-D CRM that uses convective permitting resolution (Grabowski, 1999; Khairoutdinov et al., 2005). This has led to many successes such as the possibility to rectify the diurnal continental cycle, to improve the representation of the MJO, and to represent both some MCS propagation and some degree of aggregation, and reduce overly strong land-atmosphere coupling (Benedict & Randall, 2009; Grabowski, 2001; Holloway et al., 2012, 2015; Khairoutdinov et al., 2005; Kooperman et al., 2016a, 2016b; Pritchard & Somerville, 2009; Pritchard et al., 2011; Qin et al., 2018; Randall, 2013; Sun & Pritchard, 2016).

While promising, SP is not without its own idealizations that also limit its predictive ability and usefulness for climate simulation. For instance, restricting explicit convection to two dimensions makes it difficult to represent momentum transport (Arakawa, 2011; Jung & Arkawa, 2010; Tulich, 2015; Woelfle et al., 2018), and the limited CRM domain extent artificially constrains vertical mixing efficiency (Pritchard et al., 2014). Meanwhile, the typical use of 1–4 km CRM horizontal resolution and 250-m vertical resolution cannot resolve important boundary layer turbulence, lower tropospheric inversions, and associated entrainment that are critical to low cloud dynamics (Parishani et al., 2017).

In light of this ongoing deadlock, we propose to use an alternative approach to convective parameterization in which convection is represented using a machine-learning algorithm based on artificial neural networks (ANNs), trained on superparameterized simulations, called Cloud Brain (CBRAIN). ANNs can approximate any nonlinear deterministic function, a property called the universal approximation theorem (Schmidhuber, 2015). Clearly, parameterizing convection appears as an ideal problem for the use of machine learning algorithms and especially ANNs. Indeed, machine-learning algorithms have been used in many applications where a clear physically based algorithm could not be defined. Applications have included self-driving cars, society games (chess and go; Silver et al., 2016), speech recognition (Hinton et al., 2012), object recognition and detection, medical detection of cancers (Karabatak & Ince, 2009; Khan et al., 2001; Zhou et al., 2002), and genomics. There are also applications of ANNs to the geosciences, such as for rainfall prediction (Miao et al., 2015; Moazami et al., 2013; Tao et al., 2016), weather forecast, soil moisture (Kolassa et al., 2013, 2016; Kolassa, Gentine, et al., 2017; Kolassa, Reichle, & Draper, 2017), and surface turbulent flux retrievals (Alemohammad et al., 2017; Jimenez et al., 2009; Jung et al., 2011). Specifically, the development of deep learning and deep neural networks, that is, those with multiple hidden layers, has led to important developments in many different fields such as object detection or game strategy learning (Dahl et al., 2011; Hinton et al., 2012; LeCun et al., 2015; Silver et al., 2016; Tao et al., 2016). One of the advantages of ANNs is that once trained, they are computationally efficient, as most of the computational burden is dedicated to the training phase.

Our aim here is to use such ANN techniques to better parameterize convection in coarse-scale climate simulations by learning from cloud-permitting SP-simulations, while trying to minimize the computational cost compared to those cloud-permitting simulations, which are still computationally prohibitive.

2. Data

2.1. SuperParameterized Community Atmosphere Model

To evaluate this idea, we use a well-validated version of the SuperParameterized Community Atmosphere Model (SPCAM3) in a simplified aquaplanet configuration with zonally symmetric SSTs following a realistic meridional distribution (Andersen & Kuang, 2012). The global model uses a spectral dynamical core with approximately two-degree horizontal resolution (T42 triangular truncation) and 30 levels in the vertical. The CRM uses a simplified bulk one-moment microphysics scheme and a Smagorinsky 1.5-order subgrid scale turbulence closure as described by (Khairoutdinov et al., 2003) and shares the host GCM's vertical grid. For computational efficiency and convenience we use the "micro-CRM" (8-column) CRM domain discussed by Pritchard et al. (2014) for this proof of concept. Following a three-month spin-up period, we save global data at the host global model time step frequency (every 30 min) representing arterial inputs to (and outputs from) each of 8,192 cloud-resolving arrays embedded SPCAM. The simulation is run for two years, yielding around 140 million training samples per year. One year of data represents 375 Gb.

3. Neural Network Setup

We are using an ANN to predict SPCAM's total physics package tendencies, that is, the cumulative tendency produced by turbulence, convection, and radiation. Rather than purely isolating any of the above subtendencies from the CRM or GCM parameterizations, we chose a holistic approach in representing their sum—that is, the arterial total heating and moistening profiles that ultimately link a GCM's subgrid physics to its dynamical core. This has practical advantages in that the individual physical subprocesses—turbulence, convection, microphysics, and radiation—can interact in complex, nonlinear ways. Approximating the net effect of such interactions is one of the big strengths of ANNs. The ANN is not interacting with the dynamical core and uses the same inputs as SPCAM at each time step.

The ANN is written using the Python library Keras (https://keras.io), a high-level wrapper around TensorFlow (http://www.tensorflow.org). The code for the ANN training as well as for the validation and analysis below can be found at https://github.com/raspstephan/CBRAIN-CAM. Training took on the order of 12 hr on a graphical processing unit (Nvidia GTX 970). The first year of SP-CAM data was used for training, while the second year was used for independent validation.

The feedforward ANNs consist of interconnected layers, each of which has a certain number of nodes (Figure S1). The input and output variables are listed in Table 1. The first layer is the input layer, which in our case is a stacked vector containing the input variables including their vertical variation for a specific column. No latitude or longitude information is specifically passed to the neural network, meaning that we train a single neural network to be used for every column. The last layer is the output layer, which again is a stacked vector of the four output vertical profile variables. All layers in between are called hidden layers. Deep neural networks have more than one hidden layer. The values in the nodes of each layer are weighted sums of all node values in the previous layer plus a bias, passed through a nonlinear activation function. Here we used the Leaky Rectified Linear Unit (LeakyReLU) $a(x) = \max(0.3x, x)$, which resulted in better scores compared to other common activation functions such as tanh, sigmoid, or regular ReLU. The output layer is purely linear without an activation function.

Training an ANN means optimizing the weight matrices and bias vectors that define it, to minimize a loss function—in our case the mean squared error—between the ANN outputs and the truth for a given input. The loss is computed for a shuffled (in space and time) minibatch of the training data with a batch size of 1,024 samples. To reduce the loss, the gradient of the loss function with respect to all weights and biases is computed using a backpropagation algorithm, followed by a step down the gradient—that is, stochastic gradient descent. In particular, we use a version of stochastic gradient descent called Adam (Kingma & Ba, 2014). How much to step down the gradient is determined by the learning rate. We started with a learning rate of 10^{-3}, dividing it by 5 every 5 epochs (i.e., five passes through the entire training data set). In total, we trained for 30 epochs. Regularization techniques were not necessary because we did not see any signs of overfitting given the large number of training samples. Despite the random initialization of the ANN weights and biases, the final result proved robust between training realizations.

Table 1
List of Input and Output Variables Used for the Neural Network

Input variables	Vertical levels	Output variables	Vertical levels
Temperature at beginning of time step	30	Convective and turbulent temperature tendency	30
Humidity at beginning of time step	30	Convective and turbulent humidity tendency	30
Surface pressure	1	Longwave heating tendency	30
Sensible heat flux	1	Shortwave heating tendency	30
Latent heat flux	1		
Temperature tendency from dynamics	30		
Humidity tendency from dynamics	30		
Incoming solar radiation	1		
Size of stacked array	124		120

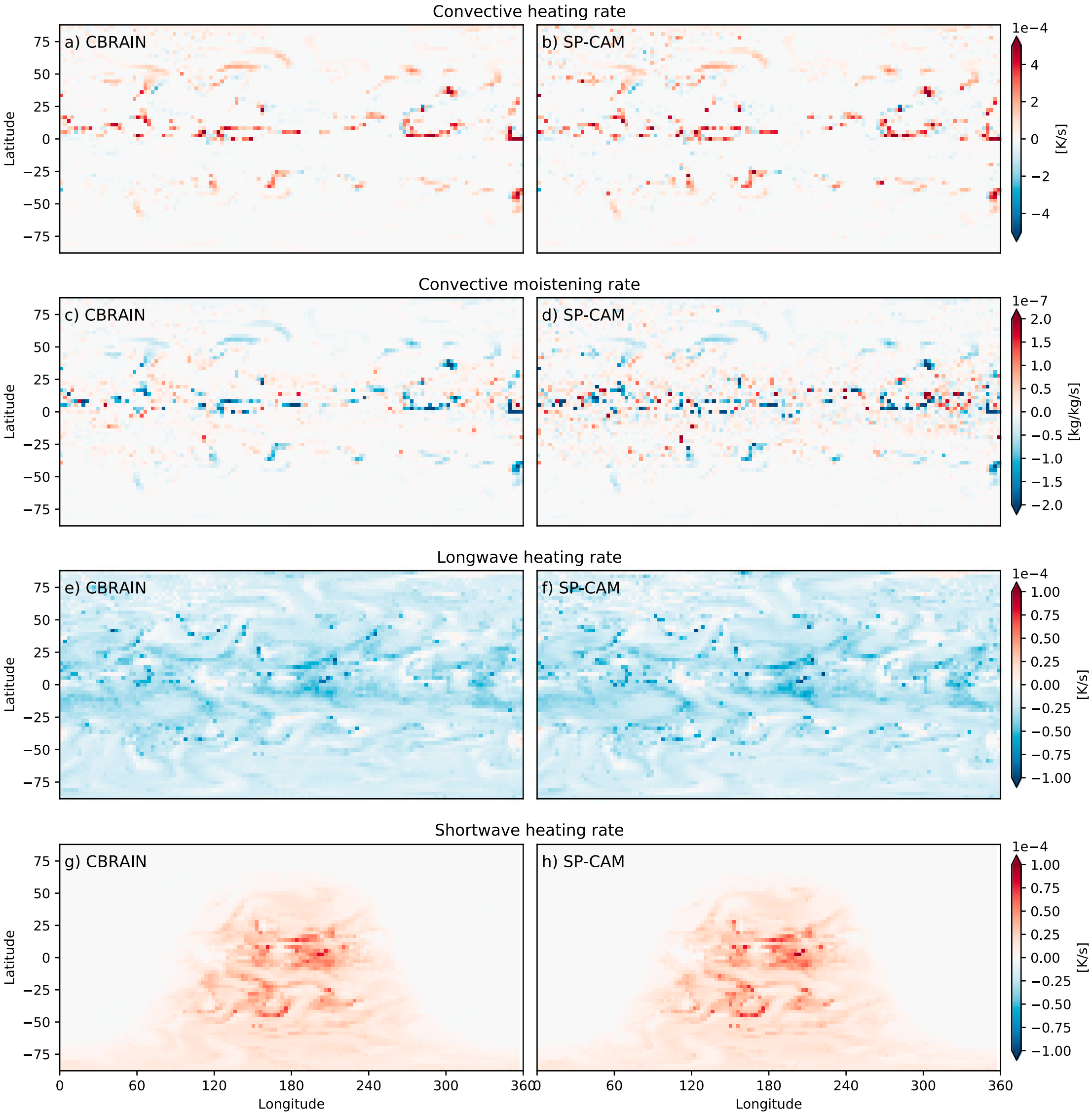


Figure 1. Latitude-longitude snapshot of neural network predictions and the corresponding SP-CAM truth at model level 20 (roughly 700 hPa) for one time step in the validation set.

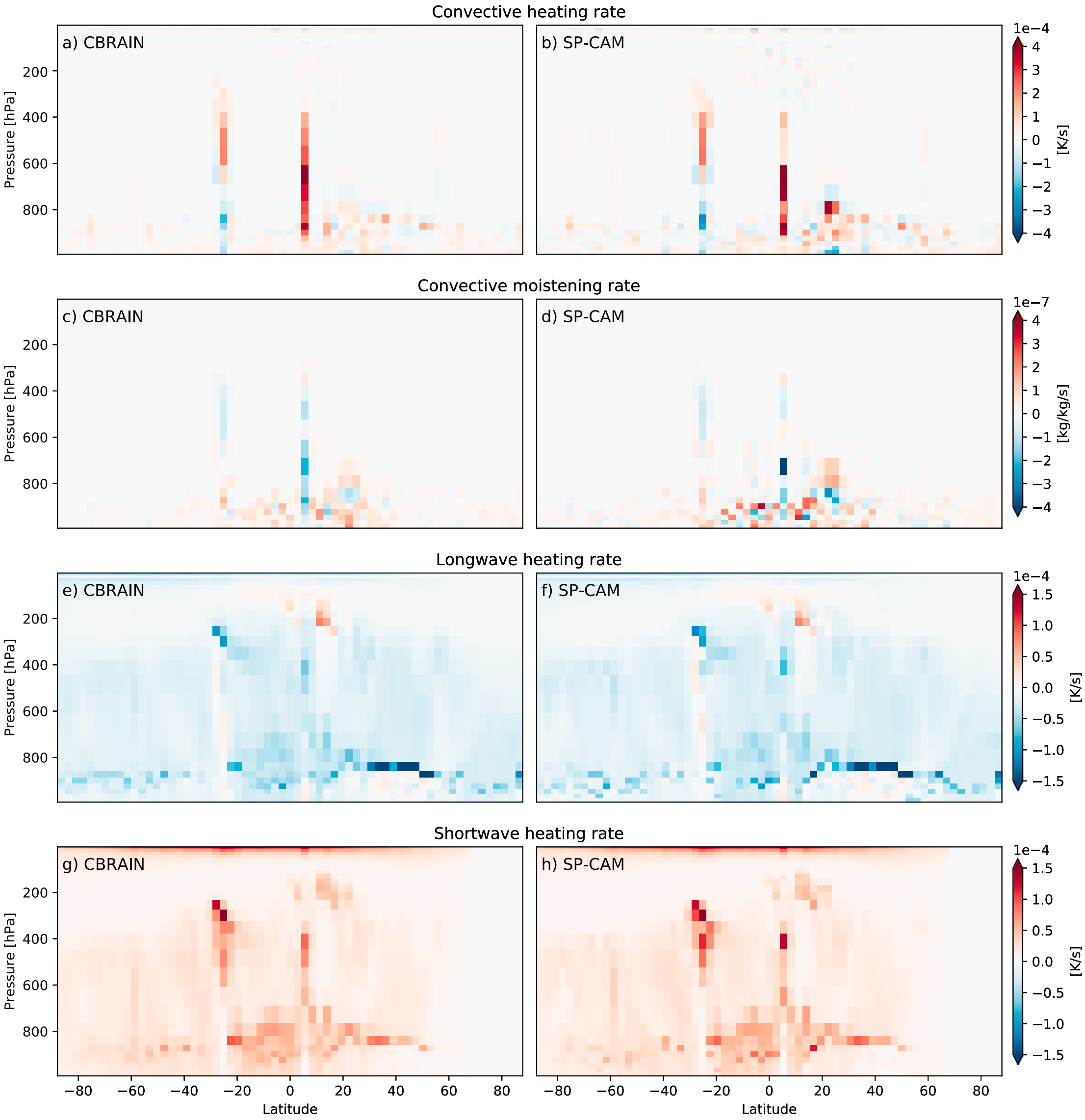


Figure 2. Pressure-latitude snapshot at 180° longitude corresponding to Figure 3.

For an ANN to train efficiently, all input values should be on the same order of magnitude. For this purpose, for each input variable, we subtracted the mean and divided by the standard deviation, independently for each vertical level; not normalizing did not modify any results but extended the duration of the training process. To make the outputs comparable, we converted the output variables (i.e., convective and radiative heating as well as convective moistening rates) to common energy units.

4. Results

4.1. Sensitivity to ANN Architecture and Amount of Training Data

We start by testing how the amount of ANN parameters and their configuration impacts the performance. Table S1 summarizes 12 separate ANN architectures tested. As a first metric of skill we assess a mean squared error statistic computed across all four output variables, all space, and all time during the second simulated year. That is, given knowledge of the inputs to each CRM, we measure the error across 143 million separate

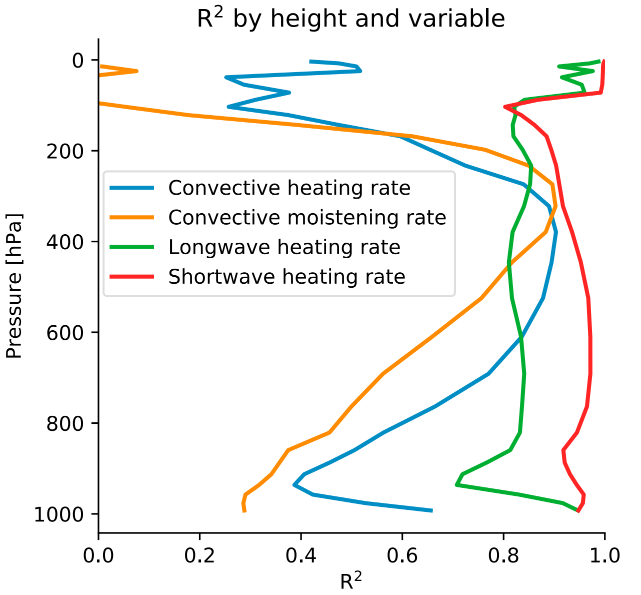


Figure 3. R^2 computed for each model pressure level and variable as described in the text.

ANN predictions of the CRM heating and moistening output profiles received by SPCAM's dynamical core, during a one-year time period that was not included in the training data set.

Figure S2a shows strong sensitivities to network architecture that underscore the importance of the ANN design—more parameters generally produce better scores and deeper networks give better results, because they also allow for more nonlinear interactions. For all subsequent analyses we thus only use our best performing network—a large, deep network with eight hidden layers of 512 nodes each.

A key question for the generalizability of our approach is how much training data is needed. For this we incrementally increase the length of continuous simulation data for training up to one year (Figure S2b). As expected, more training data do lead to better scores on the validation set. But, interestingly, three months appear to be sufficient to yield most of the information (Figure S2b). This suggests promising potential to generalize our approach beyond an SPCAM demonstration test bed to other simulation strategies that do even more justice to the true physics of moist convection. Indeed, three-month simulations are practical even for GCRMs or high-resolution, 3-D variants of SP.

4.2. Evaluation of NN Predictions

Latitude-longitude and pressure-latitude snapshots (Figures 1 and 2) provide a good qualitative starting point for evaluating the NN predictions (supplement videos). Overall, the NN predictions agree remarkably well with the SP-CAM truth in terms of horizontal and vertical structure. Lower tropospheric convective (turbulent and latent) heating and moistening associated with the intertropical convergence zone and extratropical cyclones occur at approximately the correct geographic locations (Figures 1a and 1d). The radiative heating rates show very good agreement, which is particularly impressive given the fact that there is no cloud condensate information in the input; that is, cloud-radiative feedback is all internal to the ANN. For instance, ANN skillfully predicts the geographic location of shortwave absorption by water vapor and regional cloud anomalies (Figures 1g and 1h) as well as the vertical location of longwave cooling maxima at the tops of subtropical boundary layer clouds and deep tropical clouds (Figures 2e and 2f). However, one issue for the convective heating and particularly moistening rates is that the NN predictions are smoother and do not exhibit as much of the variability as SP-CAM (internal stochastic variability). Indeed, the ANN is by definition deterministic and thus cannot reproduce any stochasticity.

To assess the quality of the predictions in more detail, we analyze R^2 averaged over both time and horizontal dimensions to yield statistics for each level and predicted variable (Figure 3). R^2 is defined as one minus the ratio of the sum of squared error to the true variance. The radiative heating rates are well represented throughout the column, particularly for shortwave heating. The convective tendencies interestingly show a distinct profile with less predictive skill in the boundary layer and the stratosphere. In the stratosphere, this lower skill is simply due to the near absence of convection at upper levels and likely not a concern. In the boundary layer, the reasons for reduced skill are discussed more below.

First, for a closer analysis of the skill in the troposphere, we also look at spatial statistics. Pressure-latitude maps of R^2 and the standard deviation (Figure 4) reveal patches of especially high skill in the midlevels at the equator and midlatitudes, which correspond to the locations of the Intertropical Convergence Zone and the midlatitude storm tracks. Since these are the locations of latent heating most fundamental to forcing the free tropospheric general circulation, this is reassuring regarding the potential of CBRAIN to reproduce important heating and moistening tendencies in future tests that could allow it to feedback with a dynamical core.

The skill in the boundary layer is significantly lower, again. One possibility is that this reflects the difficulty in representing mesoscale effects and subcloud layer organization as well as its memory (D'Andrea et al., 2014; Mapes & Neale, 2011). SPCAM does include some degree of convective aggregation (Arnold et al., 2015) and

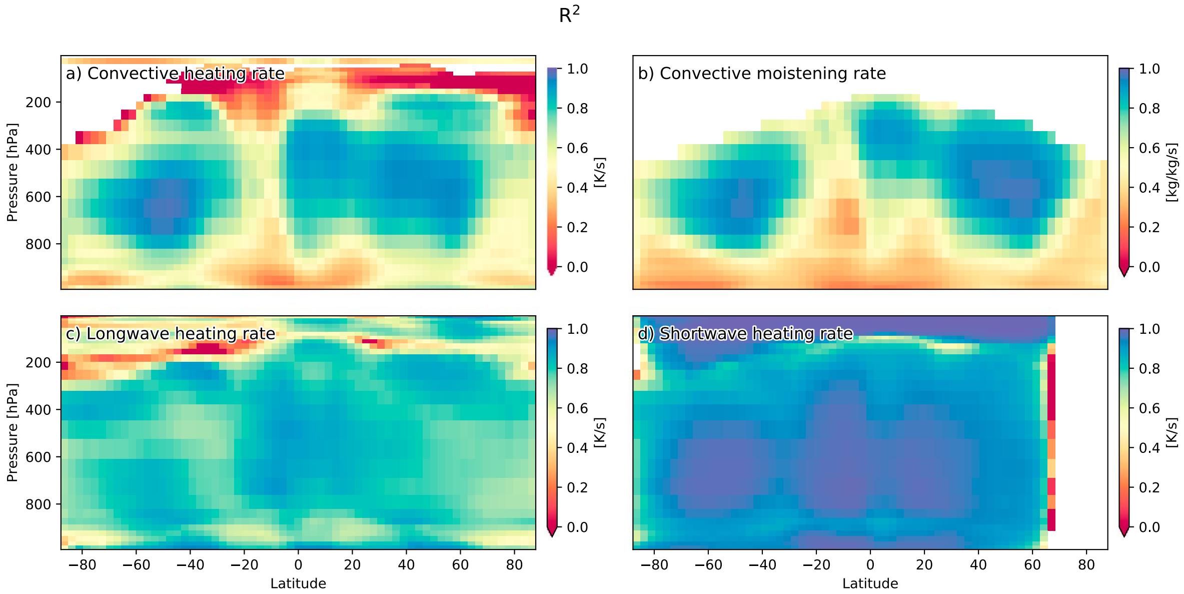


Figure 4. Pressure-latitude maps of (a and b) R^2 and (c and d) true standard deviation averaged over time and longitude. Regions where the variance was less than 0.05% of the global variance were masked out.

also carries memory of CRM organization from one-time step to the next through the embedded CRM (Pritchard et al., 2011). Our ANN does not include memory, as our objective was to mimic most current practice in convective parameterization, which is local in space and time. Another source of lower R^2 is related to the higher internal variability in SPCAM simulations compared to the ANN prediction, evident in Figures 1 and 2. This may be less of an issue in configurations that use larger, or 3-D CRMs; the small-extent 2-D CRMs used here are known to throttle deep updrafts and lead to unrealistically intense extremes (Pritchard et al., 2014). SPCAM does have some internal stochasticity (Subramanian & Palmer, 2017), which, by definition, a deterministic ANN cannot reproduce. The boundary layer and shallow convection tendencies, particularly for the moistening rate, are much noisier and thus appear much more stochastic than at higher levels. In these lower levels, the predictions here have significantly less variability in terms of its mean squared error loss function, which encourages the ANN to predict just an average value in cases where it is not certain.

5. Discussion and Conclusion

We have demonstrated that machine learning, and neural networks in particular, can skillfully represent many of the effects of unresolved clouds and convection, including their vertical transport of heat and moisture and the interaction of radiation with clouds and water vapor. The concept was proven in an idealized test bed using SPCAM over an aquaplanet. The implication of the success in this context is that an approach like CBRAIN could glean the advantages of GCRMs or high-resolution, 3-D SPs not yet practical for multidecadal climate simulations.

There are, however, important steps required for full implementation of CBRAIN in a GCM. First, neural networks do not intrinsically preserve energy and moisture. This can be fine for implementation in a weather forecast model but energy and moisture conservation are required for climate prediction. Second, neural networks are inherently deterministic. It was shown here that the resulting CBRAIN representation of heating and moistening tendencies was too smooth compared to the original SPCAM field used for training, which is more variable especially in the lower levels of the atmosphere (below 700 hPa). An important next test is to examine how CBRAIN feeds back with the GCM's resolved scale dynamics and surface fluxes. A final challenge is related to the fact that inherently a machine-learning algorithm is trained on existing data. For climate prediction, the algorithm should be able to generalize to situations that have potentially not been seen such as changes in trace gas profile and concentrations or aerosols, and should be able to represent convection over continents.

Notwithstanding the above challenges, we believe that our preliminary results motivate the case that machine learning represents a powerful alternative to GCRMs or embedded-2-D CRM parameterizations. It is computationally efficient, even for relatively large networks. For instance, without specific optimization, a preliminary test showed that CBRAIN was 10 times faster than the micro-CRM form of SP used in our study and produces tendencies of unresolved physics comparable to SP. It would thus be several orders of magnitude faster than an SP equipped with large, 3-D, high-resolution domains, or a GCRM. CBRAIN is also naturally fitted for data assimilation since computation of the adjoint is straightforward and analytical, making it a natural candidate for operational weather forecasting. CBRAIN could represent a useful alternative to current parameterizations, which have followed a "bottom-up" deterministic strategy that still exhibits too many biases for satisfying prediction of the future hydrological cycle. A "top-down" strategy that instead learns the realistic complexity of simulated convection, as captured in short multimonth simulations at convection permitting resolution, is an attractive alternative. As global temperature sensitivity to CO_2 is strongly linked to convective representation, this might also improve our estimates of future temperature.

Acknowledgments
The neural network and analysis code can be found at https://github.com/raspstephan/CBRAIN-CAM. The exact version of the code base used for this study is tagged grl_submission. The raw SP-CAM output is very large (several TB) and available upon request to Mike Pritchard. M. P. acknowledges funding from the DOE SciDac and Early Career Programs (DE-SC0012152 and DE-SC00-12548) as well as the NSF (AGS-1734164). Stephan Rasp was funded by the German Research Foundation (DFG) Transregional Collaborative Research Center SFB/TRR 165 "Waves to Weather". Computational resources for our SPCAM3 simulations were provided through the NSF Extreme Science and Engineering Discovery Environment (XSEDE) under allocation TG-ATM120034.

References

Alemohammad, S. H., Fang, B., Konings, A. G., Aires, F., Green, J. K., Kolassa, J., et al. (2017). 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*(18), 4101.

Andersen, J. A., & Kuang, Z. (2012). Moist static energy budget of MJO-like disturbances in the atmosphere of a zonally symmetric aquaplanet. *Journal of Climate*, *25*(8), 2782–2804. https://doi.org/10.1175/JCLI-D-11-00168.1

Arakawa, A., Jung, J. H., & Wu, C. M. (2011). Toward unification of the multiscale modeling of the atmosphere. *Atmospheric Chemistry and Physics*, *11*(8), 3731–3742.

Arnold, N. P., Branson, M., Burt, M. A., Abbot, D. S., Kuang, Z., Randall, D. A., & Tziperman, E. (2014). Effects of explicit atmospheric convection at high CO_2. *Proceedings of the National academy of Sciences of the United States of America*, *111*(30), 10,943–10,948. https://doi.org/10.1073/pnas.1407175111

Arnold, N. P., Branson, M., Kuang, Z., & Randall, D. A. (2015). MJO intensification with warming in the superparameterized CESM. *Journal of Climate*, *28*(7), 2706–2724. https://doi.org/10.1175/JCLI-D-14-00494.1

Arnold, N. P., & Randall, D. A. (2015). Global-scale convective aggregation: Implications for the Madden-Julian Oscillation. *Journal of Advances in Modeling Earth Systems*, *7*(4), 1499–1518. https://doi.org/10.1002/2015MS000498

Benedict, J. J., & Randall, D. A. (2009). Structure of the Madden-Julian oscillation in the superparameterized CAM. *Journal of the Atmospheric Sciences*, *66*(11), 3277–3296.

Bony, S., Stevens, B., Frierson, D. M. W., Jakob, C., Kageyama, M., Pincus, R., et al. (2015). Clouds, circulation and climate sensitivity. *Nature Geoscience*, *8*(4), 261–268. https://doi.org/10.1038/ngeo2398

Bretherton, C. S., & Khairoutdinov, M. F. (2015). Convective self-aggregation feedbacks in near-global cloud-resolving simulations of an aquaplanet. *Journal of Advances in Modeling Earth Systems*, *7*(4), 1765–1787. https://doi.org/10.1002/2015MS000499

Cao, G., & Zhang, G. J. (2017). Role of vertical structure of convective heating in MJO simulation in NCAR CAM 5.3. *Journal of Climate*, *30*(18), 7423–7439. https://doi.org/10.1175/JCLI-D-16-0913.1

Coppin, D., & Bony, S. (2015). Physical mechanisms controlling the initiation of convective self-aggregation in a general circulation model. *Journal of Advances in Modeling Earth Systems*, *7*(4), 2060–2078. https://doi.org/10.1002/2015MS000571

Couvreux, F., Roehrig, R., Rio, C., Lefebvre, M. P., Caian, M., Komori, T., et al. (2015). Representation of daytime moist convection over the semi-arid tropics by parametrizations used in climate and meteorological models. *Quarterly Journal of the Royal Meteorological Society*, *141*(691), 2220–2236. https://doi.org/10.1002/qj.2517

Dahl, G. E., Yu, D., Deng, L., & Acero, A. (2011). Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition. *IEEE Transactions on Audio, Speech and Language Processing*, *20*(1), 30–42. https://doi.org/10.1109/TASL.2011.2134090

Daleu, C. L., Plant, R. S., Woolnough, S. J., Sessions, S., Herman, M. J., Sobel, A., et al. (2015). Intercomparison of methods of coupling between convection and large-scale circulation: 1. Comparison over uniform surface conditions. *Journal of Advances in Modeling Earth Systems*, *7*(4), 1576–1601. https://doi.org/10.1002/2015MS000468

Daleu, C. L., Plant, R. S., Woolnough, S. J., Sessions, S., Herman, M. J., Sobel, A., et al. (2016). Intercomparison of methods of coupling between convection and large-scale circulation: 2. Comparison over nonuniform surface conditions. *Journal of Advances in Modeling Earth Systems*, *8*(1), 387–405. https://doi.org/10.1002/2015MS000570

D'Andrea, F., Gentine, P., Betts, A. K., & Lintner, B. R. (2014). Triggering deep convection with a probabilistic plume model. *Journal of the Atmospheric*, *71*(11), 3881–3901. https://doi.org/10.1175/JAS-D-13-0340.1

Grabowski, W. W. (1999). A parameterization of cloud microphysics for long-term cloud-resolving modeling of tropical convection. *Atmospheric research*, *52*(1–2), 17–41.

Grabowski, W. W. (2001). Coupling cloud processes with the large-scale dynamics using the cloud-resolving convection parameterization (CRCP). *Journal of the Atmospheric Sciences*, *58*(9), 978–997.

Guichard, F., Petch, J. C., Redelsperger, J. L., Bechtold, P., Chaboureau, J. P., Cheinet, S., et al. (2004). Modelling the diurnal cycle of deep precipitating convection over land with cloud-resolving models and single-column models. *Quarterly Journal of the Royal Meteorological Society*, *130*(604), 3139–3172. https://doi.org/10.1256/qj.03.145

Hinton, G., Deng, L., Yu, D., Dahl, G., Mohamed, A. R., Jaitly, N., et al. (2012). Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Processing Magazine*, *29*(6), 82–97. https://doi.org/10.1109/MSP.2012.2205597

Hohenegger, C., Schlemmer, L., & Silvers, L. (2015). Coupling of convection and circulation at various resolutions. *Tellus Series A-Dynamic Meteorology And Oceanography*, *67*(0). https://doi.org/10.3402/tellusa.v67.26678

Hohenegger, C., & Stevens, B. (2016). Coupled radiative convective equilibrium simulations with explicit and parameterized convection. *Journal of Advances in Modeling Earth Systems*, *8*(3), 1468–1482. https://doi.org/10.1002/2016MS000666

Holloway, C. E., Woolnough, S. J., & Lister, G. M. S. (2012). Precipitation distributions for explicit versus parametrized convection in a large-domain high-resolution tropical case study. *Quarterly Journal of the Royal Meteorological Society*, *138*(668), 1692–1708.

Holloway, C. E., Woolnough, S. J., & Lister, G. M. S. (2015). The effects of explicit versus parameterized convection on the MJO in a large-domain high-resolution tropical case study. Part II: Processes leading to differences in MJO development. *Journal of the Atmospheric Sciences*, *72*(7), 2719–2743. https://doi.org/10.1175/JAS-D-14-0308.1

Houze, R. A. (2004). Mesoscale convective systems. *Reviews of Geophysics*, *42*, RG4003. https://doi.org/10.1029/2004RG000150

Jeevanjee, N., & Romps, D. M. (2013). Convective self-aggregation, cold pools, and domain size. *Geophysical Research Letters*, *40*, 994–998. https://doi.org/10.1002/grl.50204

Jimenez, C., Prigent, C., & Aires, F. (2009). Toward an estimation of global land surface heat fluxes from multisatellite observations. *Journal of Geophysical Research*, *114*, D06305. https://doi.org/10.1029/2008JD011392

Jung, J. H., & Arakawa, A. (2010). Development of a Quasi-3D Multiscale Modeling Framework: Motivation, Basic Algorithm and Preliminary results. *Journal of Advances in Modeling Earth Systems*, *2*, 11. https://doi.org/10.3894/JAMES.2010.2.11

Jung, M., Reichstein, M., Margolis, H. A., Cescatti, A., Richardson, A. D., Arain, M. A., et al. (2011). Global patterns of land-atmosphere fluxes of carbon dioxide, latent heat, and sensible heat derived from eddy covariance, satellite, and meteorological observations. *Journal of Geophysical Research*, *116*, G00J07. https://doi.org/10.1029/2010JG001566

Karabatak, M., & Ince, M. C. (2009). An expert system for detection of breast cancer based on association rules and neural network. *Expert Systems with Applications*, *36*(2), 3465–3469. https://doi.org/10.1016/j.eswa.2008.02.064

Khairoutdinov, M., & Randall, D. (2006). High-resolution simulation of shallow-to-deep convection transition over land. *Journal of the Atmospheric Sciences*, *63*(12), 3421–3436. https://doi.org/10.1175/JAS3810.1

Khairoutdinov, M., Randall, D., & DeMott, C. (2005). Simulations of the atmospheric general circulation using a cloud-resolving model as a superparameterization of physical processes. *Journal of the Atmospheric Sciences*, *62*(7), 2136–2154. https://doi.org/10.1175/JAS3453.1

Khairoutdinov, M. F., Krueger, S. K., Moeng, C.-H., Bogenschutz, P. A., & Randall, D. A. (2009). Large-eddy simulation of maritime deep tropical convection. *Journal of Advances in Modeling Earth Systems*, *2*, 15. https://doi.org/10.3894/JAMES.2009.1.15.S1

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.

Khan, J., Wei, J. S., Ringner, M., Saal, L. H., & Ladanyi, M. (2001). Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks. *Nature*, *411*(6837), 522. https://doi.org/10.1038/35079160

Khouider, B., Biello, J., & Majda, A. J. (2010). A stochastic multicloud model for tropical convection. *Communications in Mathematical Sciences*, *8*(1), 187–216.

Khouider, B., & Majda, A. (2006). A simple multicloud parameterization for convectively coupled tropical waves. Part I: Linear analysis. *Journal of the Atmospheric Sciences*, *63*(4), 1308–1323. https://doi.org/10.1175/JAS3677.1

Khouider, B., Majda, A., & Katsoulakis, M. (2003). Coarse-grained stochastic models for tropical convection and climate. *Proceedings of the National Academy of Sciences of the United States of America*, *100*(21), 11,941–11,946. https://doi.org/10.1073/pnas.1634951100

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

Kolassa, J., Aires, F., Polcher, J., Prigent, C., Jimenez, C., & Pereira, J. M. (2013). Soil moisture retrieval from multi-instrument observations: Information content analysis and retrieval methodology. *Journal of Geophysical Research: Atmospheres*, *118*, 4847–4859. https://doi.org/10.1029/2012JD018150

Kolassa, J., Gentine, P., Prigent, C., & Aires, F. (2016). Soil moisture retrieval from AMSR-E and ASCAT microwave observation synergy. Part 1: Satellite data analysis. *Remote Sensing of Environment*, *173*(C), 1–14. https://doi.org/10.1016/j.rse.2015.11.011

Kolassa, J., Gentine, P., Prigent, C., Aires, F., & Alemohammad, S. H. (2017). Soil moisture retrieval from AMSR-E and ASCAT microwave observation synergy. Part 2: Product evaluation. *Remote Sensing of Environment*, *195*, 202–217. https://doi.org/10.1016/j.rse.2017.04.020

Kolassa, J., Reichle, R. H., & Draper, C. S. (2017). Merging active and passive microwave observations in soil moisture data assimilation. *Remote Sensing of Environment*, *191*(C), 117–130. https://doi.org/10.1016/j.rse.2017.01.015

Kooperman, G. J., Pritchard, M. S., Burt, M. A., Branson, M. D., & Randall, D. A. (2016a). Impacts of cloud superparameterization on projected daily rainfall intensity climate changes in multiple versions of the Community Earth System Model. *Journal of Advances in Modeling Earth Systems*, *8*(4), 1727–1750. https://doi.org/10.1002/2016MS000715

Kooperman, G. J., Pritchard, M. S., Burt, M. A., Branson, M. D., & Randall, D. A. (2016b). Robust effects of cloud superparameterization on simulated daily rainfall intensity statistics across multiple versions of the Community Earth System Model. *Journal of Advances in Modeling Earth Systems*, *8*, 140–165. https://doi.org/10.1002/2015MS000574

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436–444. https://doi.org/10.1038/nature14539

Mapes, B., & Neale, R. (2011). Parameterizing convective organization to escape the entrainment dilemma. *Journal of Advances in Modeling Earth Systems*, *3*(2), M06004. https://doi.org/10.1029/2011MS000042

Medeiros, B., Stevens, B., & Bony, S. (2014). Using aquaplanets to understand the robust responses of comprehensive climate models to forcing. *Climate Dynamics*, *44*(7–8), 1957–1977. https://doi.org/10.1007/s00382-014-2138-0

Miao, C., Ashouri, H., Hsu, K.-L., Sorooshian, S., & Duan, Q. (2015). Evaluation of the PERSIANN-CDR daily rainfall estimates in capturing the behavior of extreme precipitation events over China. *Journal of Hydrometeorology*, *16*(3), 1387–1396. https://doi.org/10.1175/JHM-D-14-0174.1

Moazami, S., Golian, S., Kavianpour, M. R., & Hong, Y. (2013). Comparison of PERSIANN and V7 TRMM Multi- satellite Precipitation Analysis (TMPA) products with rain gauge data over Iran. *International Journal of Remote Sensing*, *34*(22), 8156–8171. https://doi.org/10.1080/01431161.2013.833360

Moncrieff, M. W. (2010). The multiscale organization of moist convection and the intersection of weather and climate. In D.-Z. Sun & F. Bryan (Eds.), *Climate Dynamics: Why Does Climate Vary? Geophysical Monograph Series* (pp. 3–26). Washington, DC: American Geophysical Union. https://doi.org/10.1029/2008GM000838

Moncrieff, M. W., & Liu, C. (2006). Representing convective organization in prediction models by a hybrid strategy. *Journal of the Atmospheric Sciences*, *63*(12), 3404–3420. https://doi.org/10.1175/JAS3812.1

Moncrieff, M. W., Liu, C., & Bogenschutz, P. (2017). Simulation, modeling and dynamically based parameterization of organized tropical convection for global climate models. *Journal of the Atmospheric Sciences*, *74*(5), 1363–1380. https://doi.org/10.1175/JAS-D-16-0166.1

Moncrieff, M. W., Waliser, D. E., Miller, M. J., Shapiro, M. E., Asrar, G., & Caughey, J. (2012). Multiscale convective organization and the YOTC Virtual Global Field Campaign. *Bulletin of the American Meteorological Society*, *93*(8), 1171–1187. https://doi.org/10.1175/BAMS-D-11-00233.1

Muller, C., & Bony, S. (2015). What favors convective aggregation and why? *Geophysical Research Letters*, *42*, 5626–5634. https://doi.org/10.1002/2015GL064260

Nie, J., Shaevitz, D. A., & Sobel, A. H. (2016). Forcings and feedbacks on convection in the 2010 Pakistan flood: Modeling extreme precipitation with interactive large-scale ascent. *Journal of Advances in Modeling Earth Systems*, *8*(3), 1055–1072. https://doi.org/10.1002/2016MS000663

Parishani, H., Pritchard, M. S., Bretherton, C. S., Wyant, M. C., & Khairoutdinov, M. (2017). Toward low cloud-permitting cloud superparameterization with explicit boundary layer turbulence. *Journal of Advances in Modeling Earth Systems*, *9*(3), 1542–1571. https://doi.org/10.1002/2017MS000968

Peters, K., Jakob, C., Davies, L., Khouider, B., & Majda, A. J. (2013). Stochastic behavior of tropical convection in observations and a multicloud model. *Journal of the Atmospheric Sciences*, *70*(11), 3556–3575. https://doi.org/10.1175/JAS-D-13-031.1

Pritchard, M. S., Bretherton, C. S., & Demott, C. A. (2014). Restricting 32–128 km horizontal scales hardly affects the MJO in the Superparameterized Community Atmosphere Model v.3.0 but the number of cloud-resolving grid columns constrains vertical mixing. *Journal of Advances in Modeling Earth Systems*, *6*(3), 723–739. https://doi.org/10.1002/2014MS000340

Pritchard, M. S., Moncrieff, M. W., & Somerville, R. C. J. (2011). Orogenic propagating precipitation systems over the United States in a global climate model with embedded explicit convection. *Journal of the Atmospheric Sciences*, *68*(8), 1821–1840. https://doi.org/10.1175/2011JAS3699.1

Pritchard, M. S., & Somerville, R. C. J. (2009). Assessing the diurnal cycle of precipitation in a multi-scale climate model. *Journal of Advances in Modeling Earth Systems*, *2*, 12. https://doi.org/10.3894/JAMES.2009.1.12

Qin, H., Pritchard, M. S., Kooperman, G. J., & Parishani, H. (2018). Global Effects of Superparameterization on Hydrothermal Land-Atmosphere Coupling on Multiple Timescales. *Journal of Advances in Modeling Earth Systems*, *10*(2), 530–549.

Randall, D. A. (2013). Beyond deadlock. *Geophysical Research Letters*, *40*, 5970–5976. https://doi.org/10.1002/2013GL057998

Rochetin, N., Couvreux, F., Grandpeix, J.-Y., & Rio, C. (2014). Deep convection triggering by boundary layer thermals. Part I: LES analysis and stochastic triggering formulation. *Journal of the Atmospheric Sciences*, *71*(2), 496–514. https://doi.org/10.1175/JAS-D-12-0336.1

Rochetin, N., Grandpeix, J.-Y., Rio, C., & Couvreux, F. (2014). Deep convection triggering by boundary layer thermals. Part II: Stochastic triggering parameterization for the LMDZ GCM. *Journal of the Atmospheric Sciences*, *71*(2), 515–538. https://doi.org/10.1175/JAS-D-12-0337.1

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

Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural Networks*, *61*, 85–117. https://doi.org/10.1016/j.neunet.2014.09.003

Schneider, T., Teixeira, J., Bretherton, C. S., Brient, F., Pressel, K. G., Schär, C., & Siebesma, A. P. (2017). Climate goals and computing the future of clouds. *Nature Climate Change*, *7*(1), 3–5. https://doi.org/10.1038/nclimate3190

Sherwood, S. C., Bony, S., & Dufresne, J. L. (2014). Spread in model climate sensitivity traced to atmospheric convective mixing. *Nature*, *505*(7481), 37–42. https://doi.org/10.1038/nature12829

Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., van den Driessche, G., et al. (2016). Mastering the game of Go with deep neural networks and tree search. *Nature*, *529*(7587), 484–489. https://doi.org/10.1038/nature16961

Stevens, B., & Bony, S. (2013). What are climate models missing? *Science*, *340*(6136), 1053–1054. https://doi.org/10.1126/science.1237554

Subramanian, A. C., & Palmer, T. N. (2017). Ensemble superparameterization versus stochastic parameterization: A comparison of model uncertainty representation in tropical weather prediction. *Journal of Advances in Modeling Earth Systems*, *9*(2), 1231–1250. https://doi.org/10.1002/2016MS000857

Sun, J., & Pritchard, M. S. (2016). Effects of explicit convection on global land-atmosphere coupling in the superparameterized CAM. *Journal of Advances in Modeling Earth Systems*, *8*(3), 1248–1269. https://doi.org/10.1002/2016MS000689

Tan, J., Jakob, C., Rossow, W. B., & Tselioudis, G. (2015). Increases in tropical rainfall driven by changes in frequency of organized deep convection. *Nature*, *519*(7544), 451–454. https://doi.org/10.1038/nature14339

Tao, Y., Gao, X., Hsu, K., Sorooshian, S., & Ihler, A. (2016). A deep neural network modeling framework to reduce bias in satellite precipitation products. *Journal of Hydrometeorology*, *17*(3), 931–945. https://doi.org/10.1175/JHM-D-15-0075.1

Taylor, C. M., Harris, P. P., & Parker, D. J. (2009). Impact of soil moisture on the development of a Sahelian mesoscale convective system: A case-study from the AMMA special observing period. *Quarterly Journal of the Royal Meteorological Society*, *136*(S1), 456–470. https://doi.org/10.1002/qj.465

Tulich, S. N. (2015). A strategy for representing the effects of convective momentum transport in multiscale models: Evaluation using a new superparameterized version of the Weather Research and Forecast model (SP-WRF). *Journal of Advances in Modeling Earth Systems*, *7*(2), 938–962. https://doi.org/10.1002/2014MS000417

Wing, A. A., & Emanuel, K. A. (2014). Physical mechanisms controlling self-aggregation of convection in idealized numerical modeling simulations. *Journal of Advances in Modeling Earth Systems*, *6*(1), 59–74. https://doi.org/10.1002/2013MS000269

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/0.1002/2017MS001176

Zhou, Z. H., Jiang, Y., Yang, Y. B., & Chen, S. F. (2002). Lung cancer cell identification based on artificial neural network ensembles. *Artificial Intelligence in Medicine*, *24*(1), 25–36. https://doi.org/10.1016/S0933-3657(01)00094-X

AGU PUBLICATIONS

Geophysical Research Letters

Supporting Information for

Could machine learning break the convection parameterization deadlock?

P. Gentine[1], M. Pritchard[2], S. Rasp[3], G. Reinaudi[1] and G. Yacalis[2]

[1]Columbia University, New York, NY 10027.
[2]University of California - Irvine, Irvine, CA 92697.
[3]LMU Munich, Munich, Germany

Corresponding author: Pierre Gentine (pg2328@columbia.edu)

Contents of this file

The supporting information contains one supplemtary table and two figures.

Approximate number of parameters	30k	125k	500k	2M
Shallow	128	512	2048	8192
Medium	90 x 2	256 x 2	600 x 2	1300 x 2
Deep	50 x 8	115 x 8	256 x 8	512 x 8

Table S1: Neural network architectures. All networks have 124 input nodes and 120 output nodes. The numbers in the table represent the nodes in the fully connected hidden layers. Note that powers of two are commonly chosen to speed up computations on the GPU.

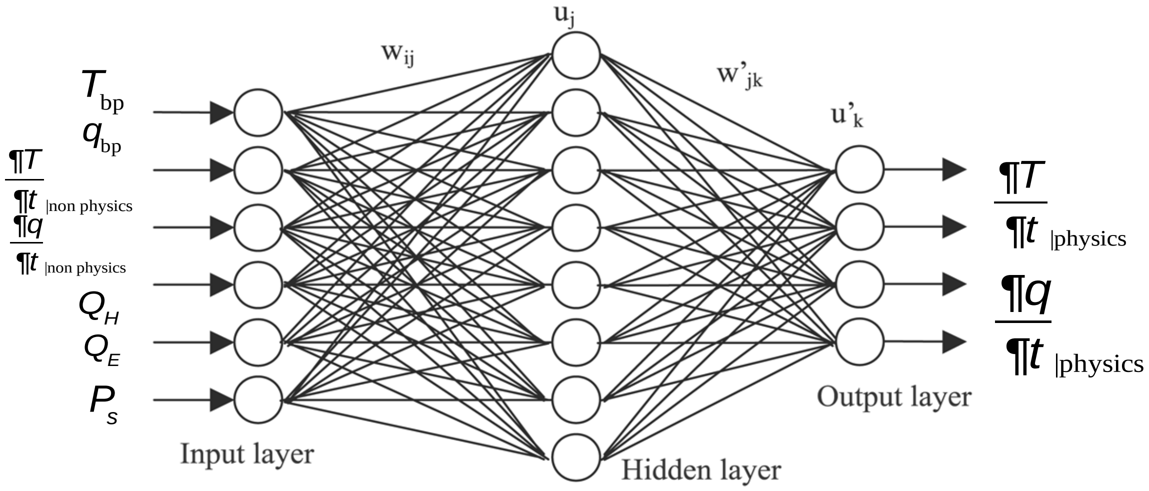


Figure S 1: Presentation of a feedforward neural network architecture and the inputs used as well as the predicted tendencies

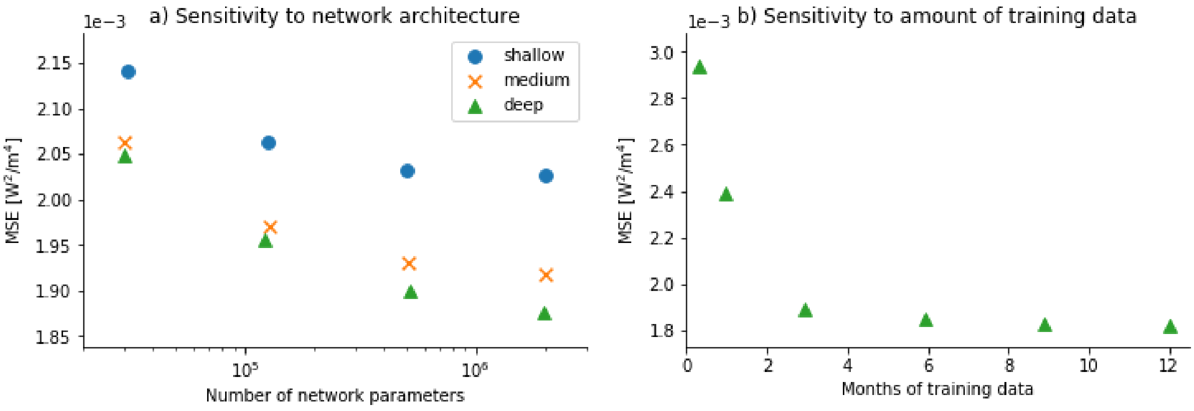


Figure S 2: Sensitivity tests to (a) network architecture and (b) amount of training data. The score is the mean squared error averaged over time, space and variables in energy units computed from the validation set.

2.3 P3: A stable prognostic climate simulation with a deep learning parameterization

DEEP LEARNING TO REPRESENT SUB-GRID PROCESSES IN CLIMATE MODELS

Stephan Rasp, Michael S. Pritchard and Pierre Gentine,
2018.
Proceedings of the National Academy of Sciences, 115(39),
9684–9689.

Context This paper follows on from P2 by implementing the neural network (with some changes) in the climate model code and running a multi-year prognostic simulation. We show that the deep learning model can reproduce the key features of the high-resolution simulation at much reduced computational cost. We also highlight the key remaining challenges for this data-driven approach, stability, physical constraints, generalizability and variability.

Author contribution All authors designed research. I ran the climate model simulations, trained the neural networks, performed the analysis with support from MP. I led the writing of the paper with input from PG and MP.

Deep learning to represent subgrid processes in climate models

Stephan Rasp[a,b,1], Michael S. Pritchard[b], and Pierre Gentine[c,d]

[a]Meteorological Institute, Ludwig-Maximilian-University, 80333 Munich, Germany; [b]Department of Earth System Science, University of California, Irvine, CA 92697; [c]Department of Earth and Environmental Engineering, Earth Institute, Columbia University, New York, NY 10027; and [d]Data Science Institute, Columbia University, New York, NY 10027

Edited by Isaac M. Held, Geophysical Fluid Dynamics Laboratory, National Oceanic and Atmospheric Administration, Princeton, NJ, and approved August 8, 2018 (received for review June 14, 2018)

The representation of nonlinear subgrid processes, especially clouds, has been a major source of uncertainty in climate models for decades. Cloud-resolving models better represent many of these processes and can now be run globally but only for short-term simulations of at most a few years because of computational limitations. Here we demonstrate that deep learning can be used to capture many advantages of cloud-resolving modeling at a fraction of the computational cost. We train a deep neural network to represent all atmospheric subgrid processes in a climate model by learning from a multiscale model in which convection is treated explicitly. The trained neural network then replaces the traditional subgrid parameterizations in a global general circulation model in which it freely interacts with the resolved dynamics and the surface-flux scheme. The prognostic multiyear simulations are stable and closely reproduce not only the mean climate of the cloud-resolving simulation but also key aspects of variability, including precipitation extremes and the equatorial wave spectrum. Furthermore, the neural network approximately conserves energy despite not being explicitly instructed to. Finally, we show that the neural network parameterization generalizes to new surface forcing patterns but struggles to cope with temperatures far outside its training manifold. Our results show the feasibility of using deep learning for climate model parameterization. In a broader context, we anticipate that data-driven Earth system model development could play a key role in reducing climate prediction uncertainty in the coming decade.

climate modeling | deep learning | subgrid parameterization | convection

Many of the atmosphere's most important processes occur on scales smaller than the grid resolution of current climate models, around 50–100 km horizontally. Clouds, for example, can be as small as a few hundred meters; yet they play a crucial role in determining the Earth's climate by transporting heat and moisture, reflecting and absorbing radiation, and producing rain. Climate change simulations at such fine resolutions are still many decades away (1). To represent the effects of such subgrid processes on the resolved scales, physical approximations—called parameterizations—have been heuristically developed and tuned to observations over the last decades (2). However, owing to the sheer complexity of the underlying physical system, significant inaccuracies persist in the parameterization of clouds and their interaction with other processes, such as boundary-layer turbulence and radiation (1, 3, 4). These inaccuracies manifest themselves in stubborn model biases (5–7) and large uncertainties about how much the Earth will warm as a response to increased greenhouse gas concentrations (1, 8, 9). To improve climate predictions, therefore, novel, objective, and computationally efficient approaches to subgrid parameterization development are urgently needed.

Cloud-resolving models (CRMs) alleviate many of the issues related to parameterized convection. At horizontal resolutions of at least 4 km deep convection can be explicitly treated (10), which substantially improves the representation of land–atmosphere coupling (11, 12), convective organization (13), and weather extremes. Further increasing the resolution to a few hundred meters allows for the direct representation of the most important boundary-layer eddies, which form shallow cumuli and stratocumuli. These low clouds are crucial for the Earth's energy balance and the cloud–radiation feedback (14). CRMs come with their own set of tuning and parameterization decisions but the advantages over coarser models are substantial. Unfortunately, global CRMs will be too computationally expensive for climate change simulations for many decades (1). Short-range simulations covering periods of months or even a few years, however, are beginning to be feasible and are in development at modeling centers around the world (15–18).

In this study, we explore whether deep learning can provide an objective, data-driven approach to using high-resolution modeling data for climate model parameterization. The paradigm shift from heuristic reasoning to machine learning has transformed computer vision and natural language processing over the last few years (19) and is starting to impact more traditional fields of science. The basic building blocks of deep learning are deep neural networks which consist of several interconnected layers of nonlinear nodes (20). They are capable of approximating arbitrary nonlinear functions (21) and can easily be adapted to novel problems. Furthermore, they can handle large datasets during training and provide fast predictions at inference time. All of

Significance

Current climate models are too coarse to resolve many of the atmosphere's most important processes. Traditionally, these subgrid processes are heuristically approximated in so-called parameterizations. However, imperfections in these parameterizations, especially for clouds, have impeded progress toward more accurate climate predictions for decades. Cloud-resolving models alleviate many of the gravest issues of their coarse counterparts but will remain too computationally demanding for climate change predictions for the foreseeable future. Here we use deep learning to leverage the power of short-term cloud-resolving simulations for climate modeling. Our data-driven model is fast and accurate, thereby showing the potential of machine-learning–based approaches to climate model development.

Author contributions: M.S.P. and P.G. designed research; S.R. and M.S.P. performed research; S.R. analyzed data; and S.R., M.S.P., and P.G. wrote the paper.

The authors declare no conflict of interest.

This article is a PNAS Direct Submission.

Data deposition: All code can be found in the following repositories: https://doi.org/10.5281/zenodo.1402384 and https://gitlab.com/mspritch/spcam3.0-neural-net/tree/nn_fbp_engy_ess.

[1]To whom correspondence should be addressed. Email: s.rasp@lmu.de.

This article contains supporting information online at www.pnas.org/lookup/suppl/doi:10.1073/pnas.1810286115/-/DCSupplemental.

Published online September 6, 2018.

these traits make deep learning an attractive approach for the problem of subgrid parameterization.

Extending on previous offline or single-column neural network cumulus parameterization studies (22–24), here we take the essential step of implementing the trained neural network in a global climate model and running a stable, prognostic multiyear simulation. To show the potential of this approach we compare key climate statistics between the deep learning-powered model and its training simulation. Furthermore, we tackle two crucial questions for a climate model implementation: First, does the neural network parameterization conserve energy? And second, to what degree can the network generalize outside of its training climate? We conclude by highlighting crucial challenges for future data-driven parameterization development.

Climate Model and Neural Network Setup

Our base model is the superparameterized Community Atmosphere Model v3.0 (SPCAM) (25) in an aquaplanet setup (see *SI Appendix* for details). The sea surface temperatures (SSTs) are fixed and zonally invariant with a realistic equator-to-pole gradient (26). The model has a full diurnal cycle but no seasonal variation. The horizontal grid spacing of the global circulation model (GCM) is approximately 2° with 30 vertical levels. The GCM time step is 30 min. In superparameterization, a 2D CRM is embedded in each GCM grid column (27). This CRM explicitly resolves deep convective clouds and includes parameterizations for small-scale turbulence and cloud microphysics. In our setup, we use 84-km–wide columns with a CRM time step of 20 s, as in ref. 28. For comparison, we also run a control simulation with the traditional parameterization suite (CTRLCAM) that is based on an undilute plume parameterization of moist convection. CTRLCAM exhibits many typical problems associated with traditional subgrid cloud parameterizations: a double intertropical convergence zone (ITCZ) (5), too much drizzle and missing precipitation extremes, and an unrealistic equatorial wave spectrum with a missing Madden–Julian oscillation (MJO). In contrast, SPCAM captures the key benefits of full 3D CRMs in improving the realism all of these issues with respect to observations (29–31). In this context, a key test for a neural network parameterization is whether it learns sufficiently from the explicitly resolved convection in SPCAM to remedy such problems while being computationally more affordable.

Analogous to a traditional parameterization, the task of the neural network is to predict the subgrid tendencies as a function of the atmospheric state at every time step and grid column (*SI Appendix*, Table S1). Specifically, we selected the following input variables: the temperature $T(z)$, specific humidity $Q(z)$ and wind profiles $V(z)$, surface pressure P_s, incoming solar radiation S_{in}, and the sensible H and latent heat fluxes E. These variables mirror the information received by the CRM and radiation scheme with a few omissions (*SI Appendix*). The output variables are the sum of the CRM and radiative heating rates ΔT_{phy}, the CRM moistening rate ΔQ_{phy}, the net radiative fluxes at the top of atmosphere and surface F_{rad}, and precipitation P. The input and output variables are stacked to vectors $\mathbf{x} = [T(z), Q(z), V(z), P_s, S_{\text{in}}, H, E]^T$ with length 94 and $\mathbf{y} = [\Delta T_{\text{phy}}(z), \Delta Q_{\text{phy}}(z), F_{\text{rad}}, P]^T$ with length 65 and normalized to have similar orders of magnitude (*SI Appendix*). We omit condensed water to reduce the complexity of the problem (*Discussion*). Furthermore, there is no momentum transport in our version of SPCAM. Informed by our previous sensitivity tests (24), we use 1 y of SPCAM simulation as training data for the neural network, amounting to around 140 million training samples.

The neural network itself $\hat{\mathbf{y}} = \mathcal{N}(\mathbf{x})$ is a nine-layer deep, fully connected network with 256 nodes in each layer. In total, the network has around 0.5 million parameters that are optimized to minimize the mean-squared error between the network's predictions $\hat{\mathbf{y}}$ and the training targets $\mathbf{y}$ (*SI Appendix*). This neural network architecture is informed by our previous sensitivity tests (24). Using deep rather than shallow networks has two main advantages: First, deeper, larger networks achieve lower training losses; and second, deep networks proved more stable in the prognostic simulations (for details see *SI Appendix* and *SI Appendix*, Fig. S1). Unstable modes and unrealistic artifacts have been the main issue in previous studies that used shallow architectures (22, 23).

Once trained, the neural network replaces the superparameterization's CRM as well as the radiation scheme in CAM. This neural network version of CAM is called NNCAM. In our prognostic global simulations, the neural network parameterization interacts freely with the resolved dynamics as well as with the surface flux scheme. The neural network parameterization speeds up the model significantly: NNCAM's physical parameterization is around 20 times faster than SPCAM's and even 8 times faster than NNCAM's, in which the radiation scheme is particularly expensive. The key fact to keep in mind is that the neural network does not become more expensive at prediction time even when trained with higher-resolution training data. The approach laid out here should, therefore, scale easily to neural networks trained with vastly more expensive 3D global CRM simulations.

The subsequent analyses are computed from 5-y prognostic simulations after a 1-y spin-up. All neural network, model, and analysis code is available in *SI Appendix*.

Results

Mean Climate. To assess NNCAM's ability to reproduce SPCAM's climate we start by comparing the mean subgrid tendencies and the resulting mean state. The mean subgrid heating (Fig. 1*A*) and moistening rates (*SI Appendix*, Fig. S2) of SPCAM and NNCAM are in close agreement with a single latent heating tower at the ITCZ and secondary free-tropospheric heating maxima at the midlatitude storm tracks. The ITCZ peak, which is colocated with the maximum SSTs at 5° N, is slightly sharper in NNCAM compared with SPCAM. In contrast, CTRLCAM exhibits a double ITCZ signal, a common issue of traditional convection parameterizations (5). The resulting mean state in temperature (Fig. 1*B*), humidity, and wind (*SI Appendix*, Fig. S2 *B* and *C*) of NNCAM also closely resembles that of SPCAM throughout the troposphere. The only larger deviations are temperature biases in the stratosphere. Since the mean heating rate bias there is small, the temperature anomalies most likely have a secondary cause—for instance, differences in circulation or internal variability. In any case, these deviations are not of obvious concern because the upper atmosphere is poorly resolved in our setup and highly sensitive to changes in the model setup (*SI Appendix*, Fig. S5 *C* and *D*). In fact, CTRLCAM has even larger differences compared with SPCAM in the stratosphere but also throughout the troposphere for all variables.

The radiative fluxes predicted by the neural network parameterization also closely match those of SPCAM for most of the globe, whereas CTRLCAM has large differences in the tropics and subtropics caused by its double-ITCZ bias (Fig. 1*C* and *SI Appendix*, Fig. S2*D*). Toward the poles NNCAM's fluxes diverge slightly, the reasons for which are yet unclear. The mean precipitation of NNCAM and SPCAM follows the latent heating maxima with a peak at the ITCZ, which again is slightly sharper for NNCAM.

In general, the neural network parameterization, freely interacting with the resolved dynamics, reproduces the most important aspects of its training model's mean climate to a remarkable degree, especially compared with the standard parameterization.

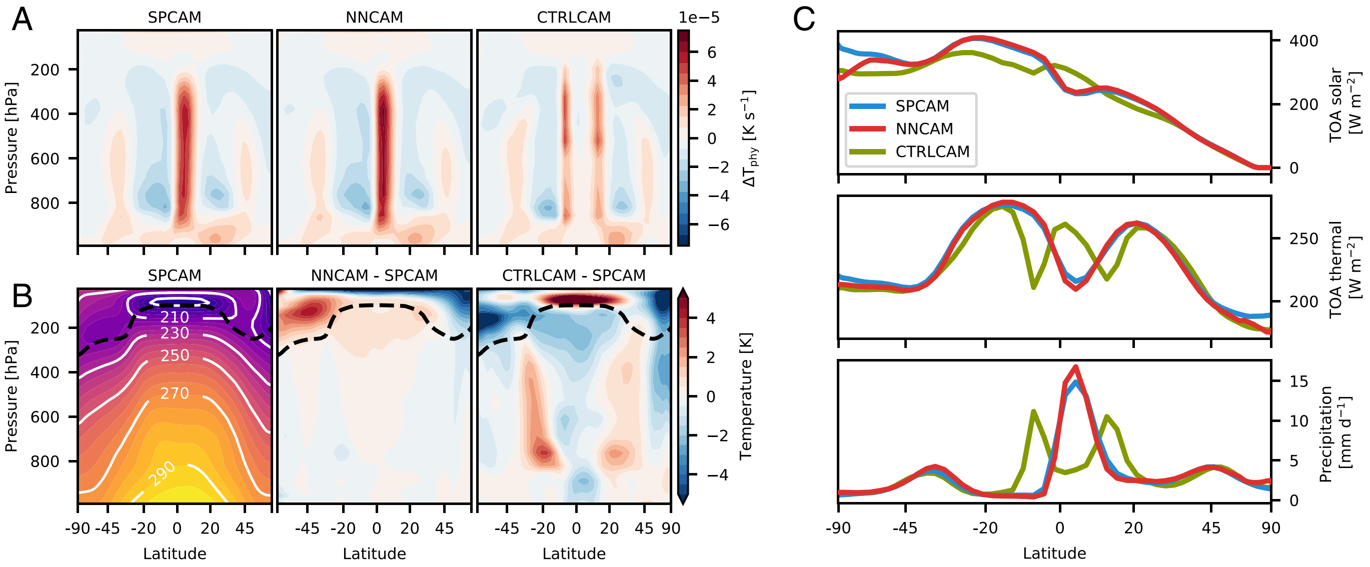


Fig. 1. (*A–C*) Longitudinal and 5-y temporal averages. (*A*) Mean convective and radiative subgrid heating rates ΔT_{phy}. (*B*) Mean temperature *T* of SPCAM and biases of NNCAM and CTRLCAM relative to SPCAM. The dashed black line denotes the approximate position of the tropopause, determined by a $\partial p\theta$ contour. (*C*) Mean shortwave (solar) and longwave (thermal) net fluxes at the top of the atmosphere and precipitation. Note that the latitude axis is area weighted.

Variability. Next, we investigate NNCAM's ability to capture SPCAM's higher-order statistics—a crucial test since climate modeling is as much concerned about variability as it is about the mean. One of the key statistics for end users is the precipitation distribution (Fig. 2*A*). CTRLCAM shows the typical deficiencies of traditional convection parameterizations—too much drizzle and a lack of extremes. SPCAM remedies these biases and has been shown to better fit to observations (31). The precipitation distribution in NNCAM closely matches that of SPCAM, including the tail. The rarest events are slightly more common in NNCAM than in SPCAM, which is consistent with the narrower and stronger ITCZ (Fig. 1 *A* and *C*).

We now focus on the variability of the heating and moistening rates (Fig. 2*B* and *SI Appendix*, Fig. S3*A*). Here, NNCAM shows reduced variance compared with SPCAM and even CTRLCAM, mostly located at the shallow cloud level around 900 hPa and in the boundary layer. Snapshots of instantaneous heating and moistening rates (*SI Appendix*, Fig. S3 *B* and *C*) confirm that the neural network's predictions are much smoother; i.e., they lack the vertical and horizontal variability of SPCAM and CTRLCAM. We hypothesize that this has two separate causes: First, low training skill in the boundary layer (24) suggests that much of SPCAM's variability in this region is chaotic and, therefore, has limited inherent predictability. Faced with such seemingly random targets during training, the deterministic neural network will opt to make predictions that are close to the mean to lower its cost function across samples. Second, the omission of condensed water in our network inputs and outputs limits NNCAM's ability to produce sharp radiative heating gradients at the shallow cloud tops. Because the circulation is mostly driven by midtropospheric heating in tropical deep convection and midlatitude storms, however, the lack of low-tropospheric variability does not seem to negatively impact the mean state and precipitation predictions. This result is also of interest for climate prediction in general.

The tropical wave spectrum (32) depends vitally on the interplay between convective heating and large-scale dynamics. This makes it a demanding, indirect test of the neural network parameterization's ability to interact with the dynamical core. Current-generation climate models are still plagued by issues in representing tropical variability: In CTRLCAM, for instance, moist kelvin waves are too active and propagate too fast while the MJO is largely missing (Fig. 3). SPCAM drastically improves the realism of the wave spectrum (29), including in our aquaplanet setup (26). NNCAM captures the key improvements of SPCAM relative to CTRLCAM: a damped kelvin wave spectrum, albeit

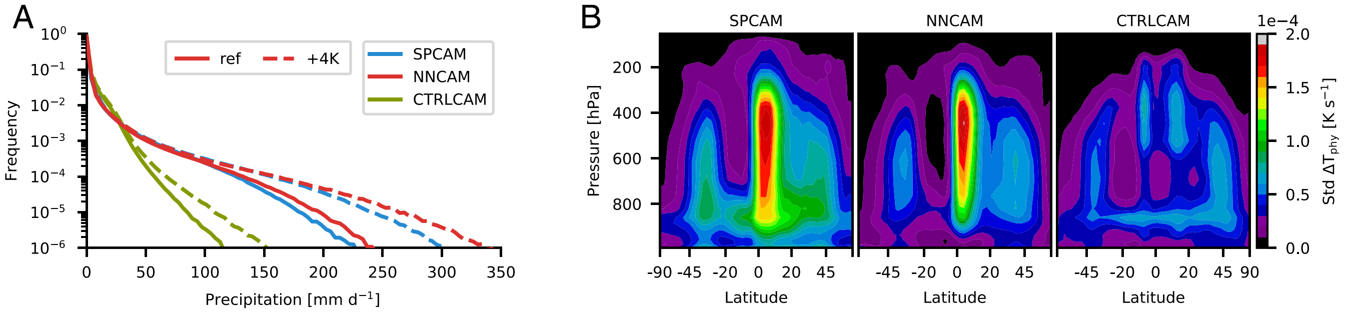


Fig. 2. (*A*) Precipitation histogram of time-step (30 min) accumulation. The bin width is 3.9 mm·d^{-1}. Solid lines denote simulations for reference SSTs. Dashed lines denote simulations for +4-K SSTs (explanation in *Generalization*). The neural network in the +4-K case is NNCAM-ref + 4 K. (*B*) Zonally averaged temporal SD of convective and radiative subgrid heating rates ΔT_{phy}.

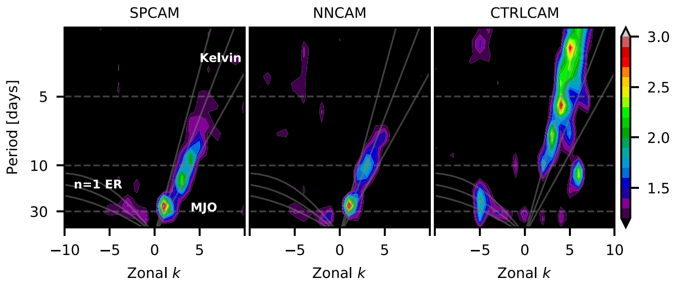


Fig. 3. Space–time spectrum of the equatorially symmetric component of 15S–15N daily precipitation anomalies divided by background spectrum as in figure 3b in ref. 32. Negative (positive) values denote westward (eastward) traveling waves.

slightly weaker and faster in NNCAM, and an MJO-like intraseasonal, eastward traveling disturbance. The background spectra also agree well with these results (*SI Appendix*, Fig. S6*A*).

Overall, NNCAM's ability to capture key advantages of the cloud-resolving training model—representing precipitation extremes and producing realistic tropical waves—is to some extent unexpected and represents a major advantage compared with traditional parameterizations.

Energy Conservation. A necessary property of any climate model parameterization is that it conserves energy. In our setup, energy conservation is not prescribed during network training. Despite this, NNCAM conserves column moist static energy to a remarkable degree (Fig. 4*A*). Note that because of our omission of condensed water, the balance shown is only approximately true and exhibits some scatter even for SPCAM. The spread is slightly larger for NNCAM, but all points lie within a reasonable range, which shows that NNCAM never severely violates energy conservation. These results suggest that the neural network has approximately learned the physical relation between the input and output variables without being instructed to. This permits a simple postprocessing of the neural network's raw predictions to enforce exact energy conservation. We tested this correction without noticeable changes to the main results. Conservation of total moisture is equally as important but the lack of condensed water makes even an approximate version impossible.

The globally integrated total energy and moisture are also stable without noticeable drift or unreasonable scatter for multiyear simulations (Fig. 4*B*). This is still true for a 50-y NNCAM simulation that we ran as a test. The energy conservation properties of the neural network parameterization are promising and show that, to a certain degree, neural networks can learn higher-level concepts and physical laws from the underlying dataset.

Generalization. A key question for the prediction of future climates is whether such a neural network parameterization can generalize outside of its training manifold. To investigate this we run a set of sensitivity tests with perturbed SSTs. We begin by breaking the zonal symmetry of our reference state by adding a wavenumber one SST perturbation with 3-K amplitude (Fig. 5*A* and *SI Appendix*). Under such a perturbation SPCAM develops a thermally direct Walker circulation within the tropics with convective activity concentrated at the downwind sector of the warm pool. The neural network trained with the zonally invariant reference SSTs only (NNCAM) is able to generate a similar heating pattern even though the heating maximum is slightly weaker and more spread out. The resulting mean temperature state in the troposphere is also in close agreement, with biases of less than 1 K (*SI Appendix*, Fig. S4). Moreover, NNCAM runs stably despite the fact that the introduced SST perturbations exceed the training climate by as much as 3 K. CTRLCAM, for comparison, has a drastically damped heating maximum and a double ITCZ to the west of the warm pool.

Our next out-of-sample test is a global SST warming of up to 4 K in 1-K increments. We use the mass-weighted absolute temperature differences relative to the SPCAM reference solution at each SST increment as a proxy for the mean climate state difference (Fig. 5*B*). The neural network trained with the reference climate only (NNCAM) is unable to generalize to much warmer climates. A look at the mean heating rates for the +4-K SST simulation reveals that the ITCZ signal is washed out and unrealistic patterns develop in and above the boundary layer (*SI Appendix*, Fig. S5*B*). As a result the temperature bias is significant, particularly in the stratosphere (*SI Appendix*, Fig. S5*D*). This suggests that the neural network cannot handle temperatures that exceed the ones seen during training. To test the opposite case, we also trained a neural network with data from the +4-K SST SPCAM simulation only (NNCAM + 4 K). The respective prognostic simulation for the reference climate has a realistic heating rate and temperature structure at the equator but fails at the poles, where temperatures are lower than in the +4-K training dataset (*SI Appendix*, Fig. S5 *A* and *C*).

Finally, we train a neural network using 0.5 y of data from the reference and the +4-K simulations each, but not the intermediate increments (NNCAM-ref + 4 K). This version performs well for the extreme climates and also in between (Fig. 5*B* and *SI Appendix*, Fig. S5). Reassuringly, NNCAM-ref + 4 K is also able to capture important aspects of global warming: an increase in the precipitation extremes (Fig. 2*A*) and an amplification and acceleration of the MJO and kelvin waves (*SI Appendix*, Fig. S6*B*). These sensitivity tests suggest that the neural network is unable to extrapolate much beyond its training climate but can interpolate in between extremes.

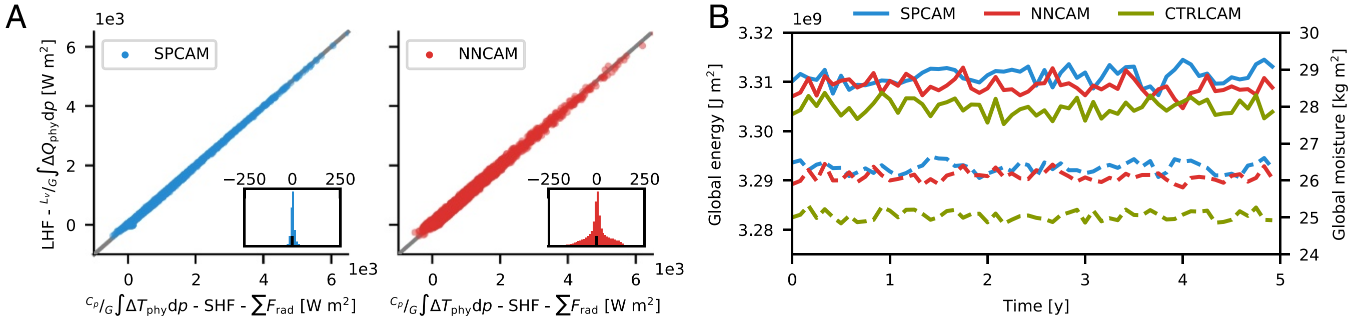


Fig. 4. (*A*) Scatter plots of vertically integrated column heating ${}^{c_p}/_{G} \int \Delta T_{\mathrm{phy}} dp$ minus the sensible heat flux *H* and the sum of the radiative fluxes at the boundaries $\sum F_{\mathrm{rad}}$ against the vertically integrated column moistening ${}^{L_v}/_{G} \int \Delta T_{\mathrm{phy}} dp$ minus the latent heat flux *H*. Each solid circle represents a single prediction at a single column. A total of 10 time steps are shown. *Inset* shows distribution of differences. (*B*) Globally integrated total energy (static, potential, and kinetic; solid lines) and moisture (dashed lines) for the 5-y simulations after 1 y of spin-up.

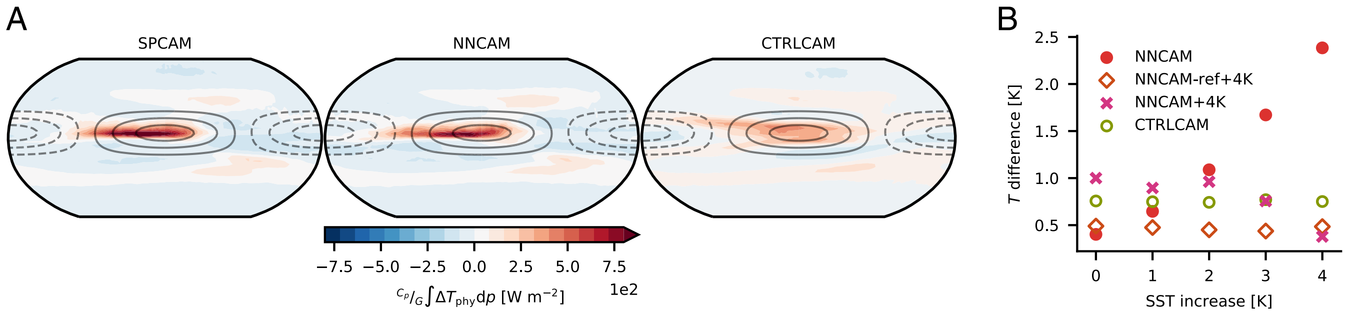


Fig. 5. (*A*) Vertically integrated mean heating rate $^{c_p}/_G \int \Delta T_{\text{phy}} dp$ for zonally perturbed SSTs. Contour lines show SST perturbation in 1-K intervals starting at 0.5 K. Dashed contours represent negative values. (*B*) Global mean mass-weighted absolute temperature difference relative to SPCAM reference at each SST increment. The different NNCAM experiments are explained in the key.

Discussion

In this study we have demonstrated that a deep neural network can learn to represent subgrid processes in climate models from cloud-resolving model data at a fraction of the computational cost. Freely interacting with the resolved dynamics globally, our deep learning-powered model produces a stable mean climate that is close to its training climate, including precipitation extremes and tropical waves. Moreover, the neural network learned to approximately conserve energy without being told so explicitly. It manages to adapt to new surface forcing patterns but struggles with out-of-sample climates. The ability to interpolate between extremes suggests that short-term, high-resolution simulations which target the edges of the climate space can be used to build a comprehensive training dataset. Our study shows a potential way for data-driven development of climate and weather models. Opportunities but also challenges abound.

An immediate follow-up task is to extend this methodology to a less idealized model setup and incorporate more complexity in the neural network parameterization. This requires ensuring positive cloud water concentrations and stability which we found challenging in first tests. Predicting the condensation rate, which is not readily available in SPCAM, could provide a convenient way to ensure conservation properties. Another intriguing approach would be to predict subgrid fluxes instead of absolute tendencies. However, computing the flux divergence to obtain the tendencies amplifies any noise produced by the neural network. Additional complexities like topography, aerosols, and chemistry will present further challenges but none of those seem insurmountable from our current vantage point.

Limitations of our method when confronted with out-of-sample temperatures are related to the traditional problem of overfitting in machine learning—the inability to make accurate predictions for data unseen during training. Convolutional neural networks and regularization techniques are commonly used to fight overfitting. It may well be possible that a combination of these and novel techniques improves the out-of-sample predictions of a neural network parameterization. Note also that our idealized training climate is much more homogeneous than the real world climate, for instance a lack of the El Niño-Southern Oscillation, which probably exacerbated the generalization issues.

Convolutional and recurrent neural networks could be used to capture spatial and temporal dependencies, such as propagating mesoscale convective systems or convective memory across time steps. Furthermore, generative adversarial networks (20) could be one promising avenue toward creating a stochastic machine-learning parameterization that captures the variability of the training data. Random forests (33) have also recently been applied to learn and model subgrid convection in a global climate model (34). Compared with neural networks, they have the advantage that conservation properties are automatically obeyed but suffer from computational limitations.

Recently, it has been argued (35) that machine learning should be used to learn the parameters or parametric functions within a traditional parameterization framework rather than the full parameterization as we have done. Because the known physics are hard coded, this could lead to better generalization capabilities, a reduction of the required data amount, and the ability to isolate individual components of the climate system for process studies. On the flip side, it still leaves the burden of heuristically finding the framework equations, which requires splitting a coherent physical system into subprocesses. In this regard, our method of using a single network naturally unifies all subgrid processes without the need to prescribe interactions.

Regardless of the exact type of learned algorithm, once implemented in the prognostic model some biases will be unavoidable. In our current methodology there is no way of tuning after the training stage. We argue, therefore, that an online learning approach, where the machine-learning algorithm runs and learns in parallel with a CRM, is required for further development. Superparameterization presents a natural fit for such a technique. For full global CRMs this likely is more technically challenging.

A grand challenge is how to learn directly from observations—our closest knowledge of the truth—rather than high-resolution simulations which come with their own baggage of tuning and parameterization (turbulence and microphysics) (35). Complications arise because observations are sparse in time and space and often only of indirect quantities, for example satellite observations. Until data assimilation algorithms for parameter estimation advance, learning from high-resolution simulations seems the more promising route toward tangible progress in subgrid parameterization.

Our study presents a paradigm shift from the manual design of subgrid parameterizations to a data-driven approach that leverages the advantages of high-resolution modeling. This general methodology is not limited to the atmosphere but can equally as well be applied to other components of the Earth system and beyond. Challenges must still be overcome, but advances in computing capabilities and deep learning in recent years present novel opportunities that are just beginning to be investigated. We believe that machine-learning approaches offer great potential that should be explored in concert with traditional model development.

Materials and Methods

Detailed explanations of the model and neural network setup can be found in *SI Appendix*. *SI Appendix* also contains links to the online code repositories. The raw model output data amount to several TB and are available from the authors upon request.

ACKNOWLEDGMENTS. We thank Gaël Reinaudi, David Randall, Galen Yacalis, Jeremy McGibbon, Chris Bretherton, Phil Rasch, Tapio Schneider, Padhraic Smyth, and Eric Nalisnick for helpful conversations during this work. S.R. acknowledges funding from the German Research Foundation Project SFB/TRR 165 "Waves to Weather." M.S.P. acknowledges funding from the Department of Energy (DOE) Scientific Discovery Through Advanced Computing (SciDAC) and Early Career Programs DE-SC0012152 and DE-SC0012548 and the NSF Programs AGS-1419518 and AGS-1734164. P.G. acknowledges funding from the NSF Programs AGS-1734156 and AGS-1649770, the NASA Program NNX14AI36G, and the DOE Early Career Program DE-SC0014203. Computational resources were provided through the NSF Extreme Science and Engineering Discovery Environment (XSEDE) allocations TG-ATM120034 and TG-ATM170029.

1. Schneider T, et al. (2017) Climate goals and computing the future of clouds. *Nat Clim Change* 7:3–5.
2. Hourdin F, et al. (2017) The art and science of climate model tuning. *Bull Am Meteorol Soc* 98:589–602.
3. Stevens B, Bony S, Ginoux P, Ming Y, Horowitz LW (2013) What are climate models missing? *Science* 340:1053–1054.
4. Bony S, et al. (2015) Clouds, circulation and climate sensitivity. *Nat Geosci* 8:261–268.
5. Oueslati B, Bellon G (2015) The double ITCZ bias in CMIP5 models: Interaction between SST, large-scale circulation and precipitation. *Clim Dyn* 44:585–607.
6. Arnold NP, Randall DA (2015) Global-scale convective aggregation: Implications for the Madden-Julian oscillation. *J Adv Model Earth Syst* 7:1499–1518.
7. Gentine P, et al. (2013) A probabilistic bulk model of coupled mixed layer and convection. Part I: Clear-sky case. *J Atmos Sci* 70:1543–1556.
8. Bony S, Dufresne J (2005) Marine boundary layer clouds at the heart of tropical cloud feedback uncertainties in climate models. *Geophys Res Lett* 32:L20806.
9. Sherwood SC, Bony S, Dufresne JL (2014) Spread in model climate sensitivity traced to atmospheric convective mixing. *Nature* 505:37–42.
10. Weisman ML, Skamarock WC, Klemp JB (1997) The resolution dependence of explicitly modeled convective systems. *Mon Weather Rev* 125:527–548.
11. Sun J, Pritchard MS (2016) Effects of explicit convection on global land-atmosphere coupling in the superparameterized CAM. *J Adv Model Earth Syst* 8:1248–1269.
12. Leutwyler D, Lüthi D, Ban N, Fuhrer O, Schär C (2017) Evaluation of the convection-resolving climate modeling approach on continental scales. *J Geophys Res Atmos* 122:5237–5258.
13. Muller C, Bony S (2015) What favors convective aggregation and why? *Geophys Res Lett* 42:5626–5634.
14. Soden BJ, Vecchi GA (2011) The vertical distribution of cloud feedback in coupled ocean-atmosphere models. *Geophys Res Lett* 38:L12704.
15. Miyamoto Y, et al. (2013) Deep moist atmospheric convection in a subkilometer global simulation. *Geophys Res Lett* 40:4922–4926.
16. Bretherton CS, Khairoutdinov MF (2015) Convective self-aggregation feedbacks in near-global cloud-resolving simulations of an aquaplanet. *J Adv Model Earth Syst* 7:1765–1787.
17. Yashiro H, et al. (2016) Resolution dependence of the diurnal cycle of precipitation simulated by a global cloud-system resolving model. *SOLA* 12:272–276.
18. Klocke D, Brueck M, Hohenegger C, Stevens B (2017) Rediscovery of the doldrums in storm-resolving simulations over the tropical Atlantic. *Nat Geosci* 10:891–896.
19. LeCun Y, Bengio Y, Hinton G (2015) Deep learning. *Nature* 521:436–444.
20. Goodfellow I, Bengio Y, Courville A (2016) *Deep Learning* (MIT Press, Cambridge, MA).
21. Nielsen MA (2015) *Neural Networks and Deep Learning*. Available at neuralnetworksanddeeplearning.com/. Accessed August 23, 2018.
22. Krasnopolsky VM, Fox-Rabinovitz MS, Belochitski AA (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. *Adv Artif Neural Syst* 2013:1–13.
23. Brenowitz ND, Bretherton CS (2018) Prognostic validation of a neural network unified physics parameterization. *Geophys Res Lett* 45:6289–6298.
24. Gentine P, Pritchard M, Rasp S, Reinaudi G, Yacalis G (2018) Could machine learning break the convection parameterization deadlock? *Geophys Res Lett* 45:5742–5751.
25. Collins WD, et al. (2006) The formulation and atmospheric simulation of the community atmosphere model version 3 (CAM3). *J Clim* 19:2144–2161.
26. Andersen JA, Kuang Z (2012) Moist static energy budget of MJO-like disturbances in the atmosphere of a zonally symmetric aquaplanet. *J Clim* 25:2782–2804.
27. Khairoutdinov MF, Randall DA (2001) A cloud resolving model as a cloud parameterization in the NCAR Community Climate System Model: Preliminary results. *Geophys Res Lett* 28:3617–3620.
28. Pritchard MS, Bretherton CS, DeMott CA (2014) Restricting 32-128 km horizontal scales hardly affects the MJO in the Superparameterized Community Atmosphere Model v.3.0 but the number of cloud-resolving grid columns constrains vertical mixing. *J Adv Model Earth Syst* 6:723–739.
29. Benedict JJ, Randall DA (2009) Structure of the Madden–Julian oscillation in the superparameterized CAM. *J Atmos Sci* 66:3277–3296.
30. Arnold NP, Randall DA (2015) Global-scale convective aggregation: Implications for the Madden-Julian oscillation. *J Adv Model Earth Syst* 7:1499–1518.
31. Kooperman GJ, Pritchard MS, O'Brien TA, Timmermans BW (2018) Rainfall from resolved rather than parameterized processes better represents the present-day and climate change response of moderate rates in the community atmosphere model. *J Adv Model Earth Syst* 10:971–988.
32. Wheeler M, Kiladis GN (1999) Convectively coupled equatorial waves: Analysis of clouds and temperature in the wavenumber–frequency domain. *J Atmos Sci* 56:374–399.
33. Breiman L (2001) Random forests. *Machine Learn* 45:5–32.
34. O'Gorman PA, Dwyer JG (2018) Using machine learning to parameterize moist convection: Potential for modeling of climate, climate change and extreme events. arXiv:1806.11037.
35. 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. *Geophys Res Lett* 44:12,396–12,417.

PNAS
www.pnas.org

Supplementary Information for

Deep learning to represent sub-grid processes in climate models

Stephan Rasp, Michael S. Pritchard and Pierre Gentine

Stephan Rasp.
E-mail: s.rasp@lmu.de

This PDF file includes:

Supplementary text
Figs. S1 to S6
Table S1
References for SI reference citations

www.pnas.org/cgi/doi/10.1073/pnas.1810286115

Supporting Information Text

SPCAM Setup. The SPCAM model source code along with our modifications, including the neural network implementation, is available at https://gitlab.com/mspritch/spcam3.0-neural-net (branch: `nn_fbp_engy_ess`).

We use the Community Atmosphere Model 3.0 (1) with super-parameterization (2) as our training and reference model. The model has an approximately two-degree horizontal resolution with 30 vertical levels and a 30 minute time step. The embedded two-dimensional cloud resolving models consist of eight 4 km-wide columns oriented meriodinally, as in Ref. (3). The CRM time step is 20 seconds. Sub-grid turbulence in the CRM is parameterized with a local 1.5-order closure. Each GCM time step the CRM tendencies are applied to the resolved grid. Note that our SPCAM setup does not feed back momentum tendencies from the CRM to the global grid. While these might be important (4), our neural network also cannot capture momentum fluxes. Using global CRM data or augmented SP that includes 3D CRM domains with interactive momentum (or 2D SP equipped with a downgradient momentum parameterization after Ref. (5)) would prove beneficial for this purpose, especially towards ocean-coupled simulations in which cumulus friction is known to be important to the equatorial cold tongue/ITCZ nexus (6). After the SP update, the radiation scheme is called which uses sub-grid cloud information from the CRM. This is followed by a computation of the surface fluxes with a simple bulk scheme and the dynamical core. CTRLCAM uses the default parameterizations which includes the Zhang-McFarlane convection scheme (7) and a simple vertical turbulent diffusion scheme.

The physical parameterization of NNCAM is 20 times faster than SPCAM and 8 times faster than CTRLCAM. This results in a total model speed-up of factor 10 compared to SPCAM and factor 4 compared to CTRLCAM. To generate the best possible training data for the neural network we run the radiation scheme every GCM time step for SPCAM and CTRLCAM. In CTRLCAM, therefore, the radiation scheme is much more computationally expensive than in the standard setup where the radiation scheme is only called every few GCM time steps.

The sea surface temperatures (SSTs) are prescribed in our aquaplanet setup that follows Ref. (8). The reference state is zonally symmetric with a maximum shifted five degrees to the North of the equator to avoid unstable behaviors observed for equatorially symmetric aquaplanet setups:

$$\mathrm{SST}(\phi) = 2 + \frac{27}{2}(2 - \zeta - \zeta^2), \qquad [1]$$

where the SST is given in Celcius, ϕ is the latitude in degrees and

$$\zeta = \begin{cases} \sin^2\left(\pi \frac{\phi-5}{110}\right) & 5 < \phi \leq 60 \\ \sin^2\left(\pi \frac{\phi-5}{130}\right) & -60 \leq \phi < 5 \\ 1 & \text{if} |\phi| < 60 \end{cases} \qquad [2]$$

Additionally, we run simulations with a globally increased SSTs up to 4K in increments of 1K and a zonally asymmetric run with a wavenumber one perturbation added to the reference SSTs:

$$\mathrm{SST}'(\lambda, \phi) = 3\cos\left(\frac{\lambda\pi}{180}\right)\cos\left(0.5\pi\frac{(\frac{\phi\pi}{180} - 5)}{30}\right)^2 \quad \text{if} \quad -25 \leq \phi \leq 35, \qquad [3]$$

where λ is longitude in degrees. The sun is in perpetual equinox with a full diurnal cycle. All experiments were started with the same initial conditions and allowed to spin up for a year. The subsequent five years were used for analysis. Training data for the neural network was taken from the second year of the SPCAM simulations.

Neural network. All neural network code is available at https://github.com/raspstephan/CBRAIN-CAM

We use the Python library Keras (9) with the Tensorflow (10) backend for all neural network experiments. Our neural network architecture consists of nine fully-connected layers with 256 nodes each. This adds up to a total of 567,361 learnable parameters. The LeakyReLU activation function $\max(0.3x, x)$ resulted in the lowest training losses. The neural network was trained for 18 epochs with a batch size of 1024. The optimizer used was Adam (11) with a mean squared error loss function. We started with a learning rate of 1×10^{-3} which was divided by five every three epochs. The total training time was on the order of 8 hours on a single Nvidia GTX 1080 graphics processing unit (GPU).

The input variables for the neural network were chosen to mirror the information received by the CRM and radiation scheme but lack the condensed water species and the dynamical tendencies. The latter are applied as a constant forcing during the CRM integration. We found, however, that they did not improve the neural network performance and trimmed the input variables for the sake of simplicity. Another option would be to include the surface flux computation in the network as well. In this option the fluxes are removed from the input and the surface temperature is added. This option yielded similar results but did not allow us to investigate column energy conservation.

The input values are normalized by subtracting each element of the stacked input vector (Table S1) by its mean across samples and then dividing it by the maximum of its range and the standard deviation computed across all levels of the respective physical variable. This is done to avoid dividing by very small values, e.g. for humidity in the upper levels, which can cause the input values to become very large if the neural network predicts noisy tendencies. For the outputs, the heating and moistening rates are brought to the same order of magnitude by converting them to W kg^{-1} . The radiative fluxes and precipitation were normalized to be on the same order of magnitude as the heating and moistening rates (see Table S1 for multiplication factors). The magnitude of the output values determines their importance in the loss function. In our quadratic loss function differences are highlighted even further. Making sure that no single value dominates the loss is important to get a consistent prediction quality. For a reasonable range (factor five) around our normalization values the results are largely unaffected, however.

Deep neural networks appear to be essential to achieve a stable and realistic prognostic implementation. Similar to other studies which used shallow neural networks (12, 13) we encountered unstable modes and unrealistic artifacts for networks with two or one hidden layers (Fig. S1). A four layer network was the minimal complexity to provide good results for our configuration. Adding further layers shows little correlation between training skill and prognostic performance. We chose our network design to lie well withing the range of stable network configurations.

Table S1. Table showing input and output variables and their number of vertical levels N_z. For the output variables the normalization factors are also listed. C_p is the specific heat of air. L_v is the latent heat of vaporization.

Input variables	Unit	$\mathbf{N_z}$	Output variables	Unit	$\mathbf{N_z}$	Normalization
Temperature	K	30	Heating rate ΔT_{phy}	K s^{-1}	30	C_p
Humidity	kg kg^{-1}	30	Moistening rate ΔQ_{phy}	kg kg^{-1} s^{-1}	30	L_v
Meridional wind	m s^{-1}	30	Shortwave flux at TOA	W m^{-2}	1	10^{-3}
Surface pressure	Pa	1	Shortwave flux at surface	W m^{-2}	1	10^{-3}
Incoming solar radiation	W m^{-2}	1	Longwave flux at TOA	W m^{-2}	1	10^{-3}
Sensible heat flux	W m^{-2}	1	Longwave flux at surface	W m^{-2}	1	10^{-3}
Latent heat flux	W m^{-2}	1	Precipitation	kg m^{-2} d^{-1}	1	2×10^{-2}
Size of stacked vectors		94			65	

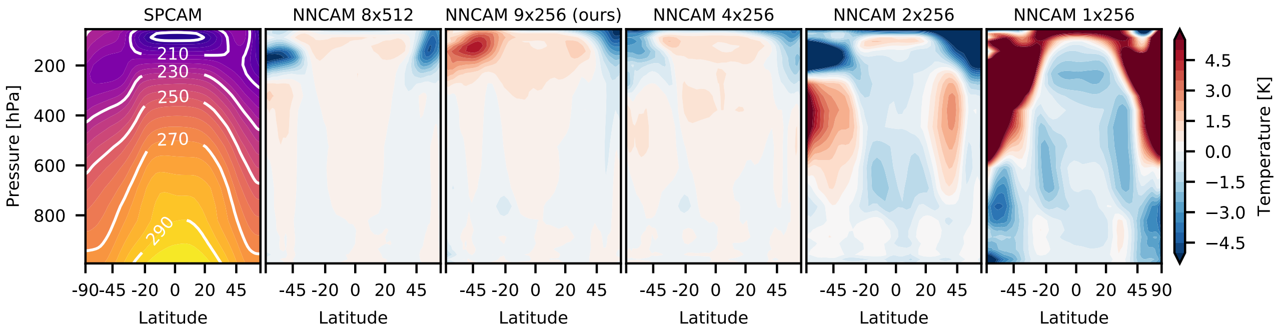


Fig. S1. All figures show longitudinal and five year-temporal averages as in Fig. 1. Zonally and temporally averaged temperature relative to SPCAM for different network configurations (Number of hidden layers x Nodes per hidden layer). 8x512 corresponds to the network in Ref. (14).

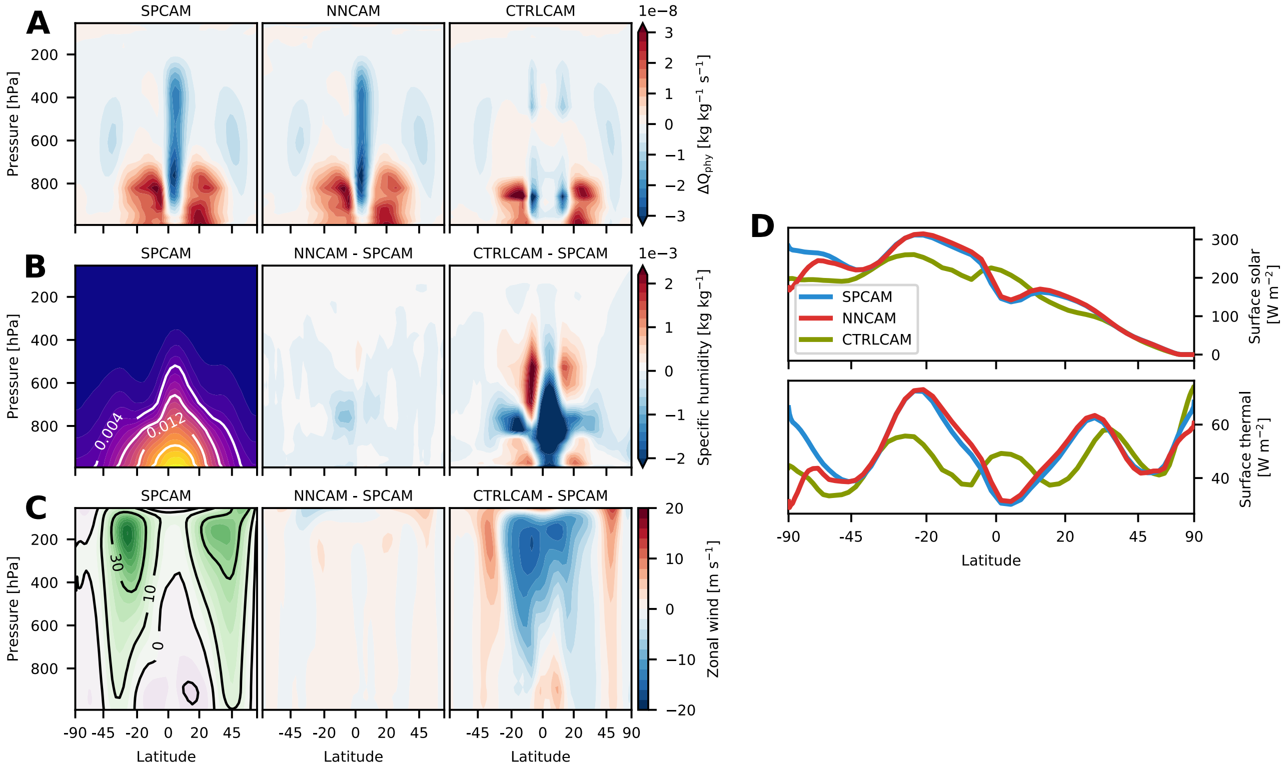


Fig. S2. (A) Mean convective sub-grid moistening rates ΔQ_{phy}. (B) Mean specific humidity Q and (C) zonal wind V of SPCAM and biases of NNCAM and CTRLCAM relative to SPCAM. (D) Mean shortwave (solar) and longwave (thermal) net fluxes at the surface. The latitude axis is area-weighted.

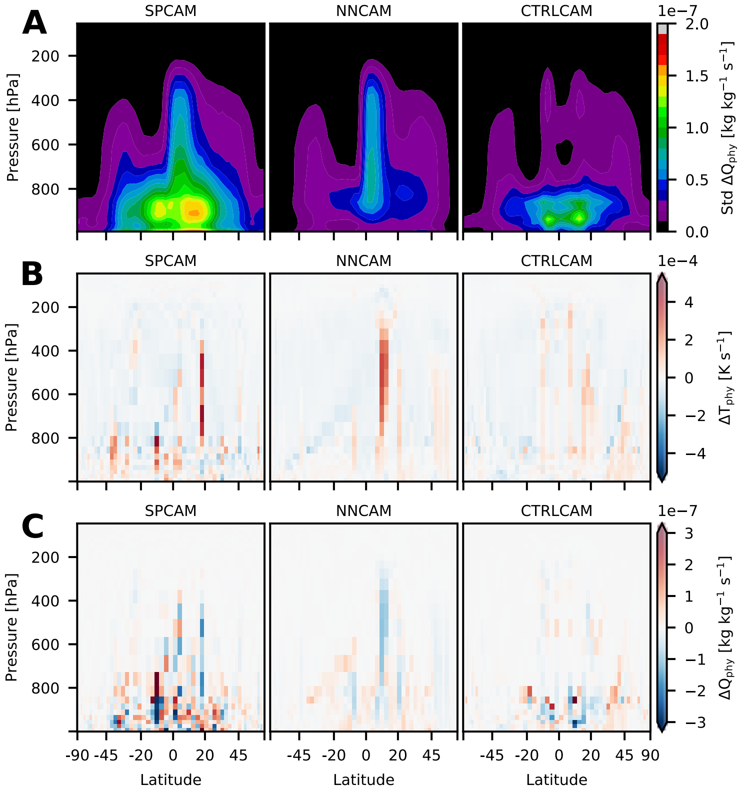


Fig. S3. (A) Zonally averaged temporal standard deviation of the convective sub-grid moistening rate ΔQ_{phy}. (B, C) Snapshots of heating ΔT_{phy} and moistening rate ΔQ_{phy}. Note that these are taken from the free model simulations and should, therefore, not correspond one-to-one between the experiments.

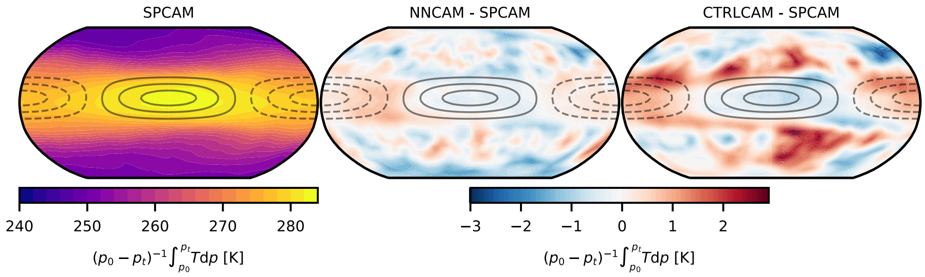

SPCAM
NNCAM - SPCAM
CTRLCAM - SPCAM
240
250
260
270
280
$(p_0 - p_t)^{-1}\int_{p_0}^{p_t} T dp$ [K]
−3
−2
−1
0
1
2
$(p_0 - p_t)^{-1}\int_{p_0}^{p_t} T dp$ [K]

Fig. S4. Mass-weighted temperature integrated over the troposphere from $p_0 = 1000$ hPa to $p_t = 380$ hPa for SPCAM reference and differences of NNCAM and CTRLCAM with respect to reference for zonally perturbed simulations.

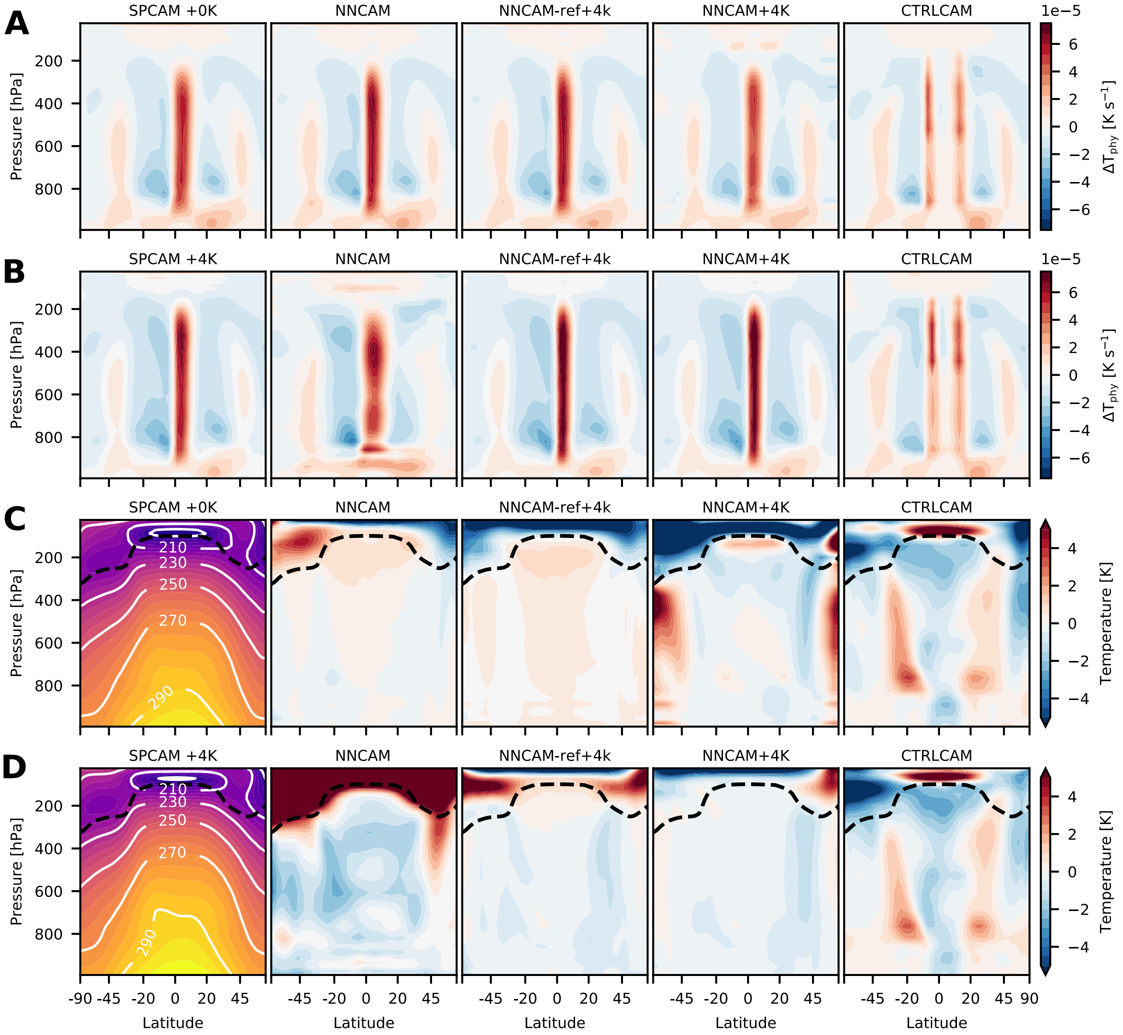


Fig. S5. Zonally and temporally averaged (A, B) heating rate and (C, D) temperature relative to SPCAM. Panels A and C show reference SSTs while panels B and D show global 4 K perturbation. Temperature panels show SPCAM reference and differences to reference for several experiments described in the text.

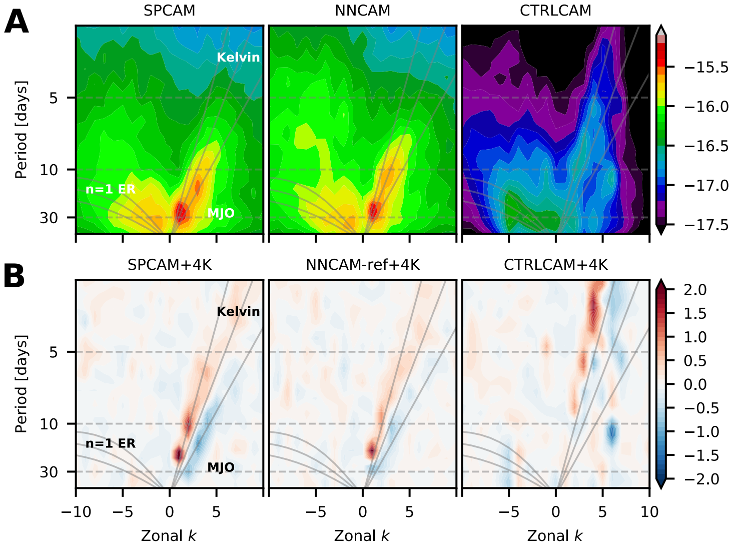


Fig. S6. (A) Space-time spectrum of the equatorially symmetric component of 15S-15N daily precipitation anomalies. As in Fig. 1b of Ref. (15). (B) Space-time spectrum of the equatorially symmetric component of 15S-15N daily precipitation anomalies divided by background spectrum. As in Fig. 3b of Ref. (15). Figure shows +4K SST minus reference SST. Negative (positive) values denote westward (eastward) traveling waves.

References

1. Collins WD, et al. (2006) The Formulation and Atmospheric Simulation of the Community Atmosphere Model Version 3 (CAM3). *Journal of Climate* 19(11):2144–2161.
2. Khairoutdinov MF, Randall DA (2001) A cloud resolving model as a cloud parameterization in the NCAR Community Climate System Model: Preliminary results. *Geophysical Research Letters* 28(18):3617–3620.
3. Pritchard MS, Bretherton CS, DeMott CA (2014) Restricting 32-128 km horizontal scales hardly affects the MJO in the Superparameterized Community Atmosphere Model v.3.0 but the number of cloud-resolving grid columns constrains vertical mixing. *Journal of Advances in Modeling Earth Systems* 6(3):723–739.
4. Moncrieff MW, Liu C, Bogenschutz P (2017) Simulation, Modeling and Dynamically Based Parameterization of Organized Tropical Convection for Global Climate Models. *Journal of the Atmospheric Sciences* pp. 16–0166.
5. Tulich SN (2015) A strategy for representing the effects of convective momentum transport in multiscale models: Evaluation using a new superparameterized version of the Weather Research and Forecast model (SP-WRF). *J. Adv. Model. Earth Syst.* 7(2):938–962.
6. Woelfle MD, Yu S, Bretherton CS, Pritchard MS (2018) Sensitivity of Coupled Tropical Pacific Model Biases to Convective Parameterization in CESM1. *J. Adv. Model. Earth Syst.* 10(1):126–144.
7. Zhang G, McFarlane NA (1995) Sensitivity of climate simulations to the parameterization of cumulus convection in the Canadian climate centre general circulation model. *Atmosphere-Ocean* 33(3):407–446.
8. Andersen JA, Kuang Z (2012) Moist Static Energy Budget of MJO-like Disturbances in the Atmosphere of a Zonally Symmetric Aquaplanet. *Journal of Climate* 25(8):2782–2804.
9. Chollet F, Others (2015) Keras. https://keras.io/.
10. Abadi M, et al. (2016) TensorFlow: A system for large-scale machine learning in *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16).* pp. 265–283.
11. Kingma DP, Ba J (2014) Adam: A Method for Stochastic Optimization. arXiv: 1412.6980.
12. Krasnopolsky VM, Fox-Rabinovitz MS, Belochitski AA (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* 2013:1–13.
13. Brenowitz ND, Bretherton CS (2018) Prognostic Validation of a Neural Network Unified Physics Parameterization. *Geophys. Res. Lett.* 45(12):6289–6298.
14. Gentine P, Pritchard M, Rasp S, Reinaudi G, Yacalis G (2018) Could Machine Learning Break the Convection Parameterization Deadlock? *Geophys. Res. Lett.* 45(11):5742–5751.
15. Wheeler M, Kiladis GN (1999) Convectively Coupled Equatorial Waves: Analysis of Clouds and Temperature in the Wavenumber–Frequency Domain. *Journal of the Atmospheric Sciences* 56(3):374–399.

2.4 P4: Neural networks for probabilistic calibration of temperature forecasts

NEURAL NETWORKS FOR POST-PROCESSING ENSEMBLE
WEATHER FORECASTS.

Stephan Rasp and Sebastian Lerch, 2018.
Monthly Weather Review, 146(11), 3885–3900.

Context Here we focus on the task of calibrating an ECMWF ensemble temperature forecast to produce a sharp and reliable forecast distribution. We use a neural network approach which outperforms previous state-of-the-art techniques while being computationally more affordable. We also show that machine learning methods can be used to gain insight into the underlying system.

Author contribution SL and I designed research. SL ran the benchmark postprocessing experiments. I ran the neural network experiments. SL and I wrote the paper.

SPECIAL
Waves to Weather (W2W)
COLLECTION

Neural Networks for Postprocessing Ensemble Weather Forecasts

STEPHAN RASP

Meteorological Institute, Ludwig-Maximilians-Universität, Munich, Germany

SEBASTIAN LERCH

Institute for Stochastics, Karlsruhe Institute of Technology, Heidelberg Institute for Theoretical Studies, Karlsruhe, Germany

(Manuscript received 23 May 2018, in final form 14 August 2018)

ABSTRACT

Ensemble weather predictions require statistical postprocessing of systematic errors to obtain reliable and accurate probabilistic forecasts. Traditionally, this is accomplished with distributional regression models in which the parameters of a predictive distribution are estimated from a training period. We propose a flexible alternative based on neural networks that can incorporate nonlinear relationships between arbitrary predictor variables and forecast distribution parameters that are automatically learned in a data-driven way rather than requiring prespecified link functions. In a case study of 2-m temperature forecasts at surface stations in Germany, the neural network approach significantly outperforms benchmark postprocessing methods while being computationally more affordable. Key components to this improvement are the use of auxiliary predictor variables and station-specific information with the help of embeddings. Furthermore, the trained neural network can be used to gain insight into the importance of meteorological variables, thereby challenging the notion of neural networks as uninterpretable black boxes. Our approach can easily be extended to other statistical postprocessing and forecasting problems. We anticipate that recent advances in deep learning combined with the ever-increasing amounts of model and observation data will transform the postprocessing of numerical weather forecasts in the coming decade.

1. Introduction

Numerical weather prediction based on physical models of the atmosphere has improved continuously since its inception more than four decades ago (Bauer et al. 2015). In particular, the emergence of ensemble forecasts—simulations with varying initial conditions and/or model physics—added another dimension by quantifying the flow-dependent uncertainty. Yet despite these advances the raw forecasts continue to exhibit systematic errors that need to be corrected using statistical postprocessing methods (Hemri et al. 2014). Considering the ever-increasing social and economical value of numerical weather prediction—for example, in the renewable energy industry—producing accurate and calibrated probabilistic forecasts is an urgent challenge.

Most postprocessing methods correct systematic errors in the raw ensemble forecast by learning a function that relates the response variable of interest to predictors. From a machine learning perspective, postprocessing can be viewed as a supervised learning task. For the purpose of this study we will consider postprocessing in a narrower distributional regression framework where the aim is to model the conditional distribution of the weather variable of interest given a set of predictors. The two most prominent approaches for probabilistic forecasts, Bayesian model averaging (BMA; Raftery et al. 2005) and nonhomogeneous regression, also referred to as ensemble model output statistics (EMOS; Gneiting et al. 2005), rely on parametric forecast distributions. This means one has to specify a predictive distribution and estimate its parameters, for example, the mean and the standard deviation in the case of a Gaussian distribution. Within the

Denotes content that is immediately available upon publication as open access.

Supplemental information related to this paper is available at the Journals Online website: https://doi.org/10.1175/MWR-D-18-0187.s1.

Corresponding author: Stephan Rasp, s.rasp@lmu.de

DOI: 10.1175/MWR-D-18-0187.1

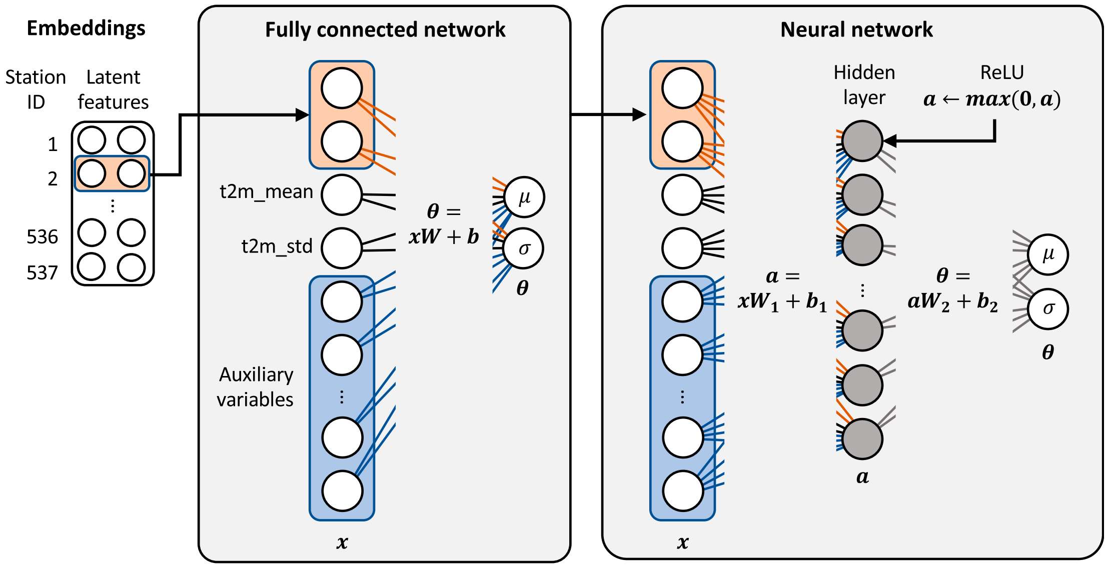


FIG. 1. Schematic of (left) an FCN and (right) an NN with one hidden layer. In both cases, data flow from left to right. Orange nodes and connections illustrate station embeddings, and blue nodes are for auxiliary input variables. Mathematical operations are to be understood as elementwise operations for vector objects.

EMOS framework the distribution parameters are connected to summary statistics of the ensemble predictions through suitable link functions that are estimated by minimizing a probabilistic loss function over a training dataset. Including additional predictors, such as forecasts of cloud cover or humidity, is not straightforward within this framework and requires elaborate approaches to avoid overfitting (Messner et al. 2017), a term that describes the inability of a model to generalize to data outside the training dataset. We propose an alternative approach based on modern machine learning methods, which is capable of including arbitrary predictors and learns nonlinear dependencies in a data-driven way.

Much work over the past years has been spent on flexible machine learning techniques for statistical modeling and forecasting (McGovern et al. 2017). Random forests (Breiman 2001), for instance, can model nonlinear relationships including arbitrary predictors while being robust to overfitting. They have been used for the classification and prediction of precipitation (Gagne et al. 2014), severe wind (Lagerquist et al. 2017), and hail (Gagne et al. 2017). Within a postprocessing context, quantile regression forest models have been proposed by Taillardat et al. (2016).

Neural networks are a flexible and user-friendly machine learning algorithm that can model arbitrary nonlinear functions (Nielsen 2015). They consist of several layers of interconnected nodes that are modulated with simple nonlinearities (Fig. 1; section 4). Over the past decade many fields, most notably computer vision and natural language processing (LeCun et al. 2015), but also biology, physics, and chemistry (Angermueller et al. 2016; Goh et al. 2017), have been transformed by neural networks. In the atmospheric sciences, neural networks have been used to detect extreme weather in climate datasets (Liu et al. 2016) and parameterize subgrid processes in general circulation models (Gentine et al. 2018; Rasp et al. 2018). Neural networks have also been used for forecasting solar irradiances (Wang et al. 2012; Chu et al. 2013) and damaging winds (Lagerquist et al. 2017). However, the complexity of the neural networks used in these studies was limited.

Here, we demonstrate how neural networks can be used for probabilistic postprocessing of ensemble forecasts within the distributional regression framework. The presented model architecture allows for the incorporation of various features that are relevant for correcting systematic deficiencies of ensemble predictions, and to estimate the network parameters by optimizing the continuous ranked probability score—a mathematically principled loss function for probabilistic forecasts. Specifically, we explore a case study of 2-m temperature forecasts at surface stations in Germany with data from 2007 to 2016. We compare different neural network configurations to benchmark postprocessing methods for varying training period lengths. We further use the trained neural networks to gain meteorological insight into the problem at hand. Our ultimate goal is to present an efficient, multipurpose approach to statistical postprocessing and probabilistic forecasting. To the best of our knowledge, this study is the first to tackle ensemble postprocessing using neural networks.

The remainder of the paper is structured as follows. Section 2 describes the forecast and observation data as well as the notation used throughout the study. In section 3 we describe the benchmark postprocessing models, followed by a description of the neural network techniques in section 4. The main results are presented in section 5. In section 6 we explore the relative importance of the predictor variables. A discussion of possible extensions follows in section 7 before our conclusions are presented in section 8.

Python (Python Software Foundation 2017) and R (R Core Team 2017) code for reproducing the results is available online (https://github.com/slerch/ppnn).

2. Data and notation

a. *Forecast data*

For this study, we focus on 2-m temperature forecasts at surface stations in Germany at a forecast lead time of 48 h. The forecasts are taken from the THORPEX Interactive Grand Global Ensemble (TIGGE) dataset[1] (Bougeault et al. 2010). In particular, we use the global European Centre for Medium-Range Weather Forecasts (ECMWF) 50-member ensemble forecasts initialized at 0000 UTC every day. The data in the TIGGE archive are upscaled onto a 0.5° × 0.5° grid, which corresponds to a horizontal grid spacing of around 35/55 km (zonal/meridional). For comparison with the station observations, the gridded data were bilinearly interpolated to the observation locations. In addition to the target variable, we retrieved several auxiliary predictor variables (Table 1[2]). These were chosen broadly based on meteorological intuition.[3] For each variable, we reduced the 50-member ensemble to its mean and standard deviation.

Ensemble predictions are available from 3 January 2007 to 31 December 2016 every day. For model estimation we use two training periods, 2007–15 and 2015 only, to assess the importance of training sample size. To validate the performance of the different models correctly, it is important to mimic operational conditions as closely as possible. For this reason we chose future dates only, in our case the entire year 2016, rather than a random subsample of the entire dataset. Note also that the ECMWF forecasting system has undergone major changes during this 10-yr period. This might counteract the usefulness of using longer training periods.

TABLE 1. Abbreviations and descriptions of all features.

Feature	Description
Ensemble predictions (mean and std dev)	
t2m	2-m temperature
cape	Convective available potential energy
sp	Surface pressure
tcc	Total cloud cover
sshf	Sensible heat flux
slhf	Latent heat flux
u10	10-m *U* wind
v10	10-m *V* wind
d2m	2-m dewpoint temperature
ssr	Shortwave radiation flux
str	Longwave radiation flux
sm	Soil moisture
u_pl500	*U* wind at 500 hPa
v_pl500	*V* wind at 500 hPa
u_pl850	*U* wind at 850 hPa
v_pl850	*V* wind at 850 hPa
gh_pl500	Geopotential at 500 hPa
q_pl850	Specific humidity at 850 hPa
Station-specific information	
station_alt	Altitude of station
orog	Altitude of model grid point
station_lat	Lat of station
station_lon	Lon of station

b. *Observation data*

The forecasts are evaluated at 537 weather stations in Germany (see Fig. 2[4]). The 2-m temperature data are available from the Climate Data Center of the German Weather Service [Deutscher Wetterdienst (DWD)[5]]. Several stations have periods of missing data, which are omitted from the analysis. During the evaluation period in calendar year 2016, observations are available at 499 stations.

After removing missing observations, the 2016 validation set contains 182 218 samples, the 2007–15 training set contains 1 626 724 samples, and the 2015 training set contains 180 849 samples.

c. *Notation*

We now introduce the notation that is used throughout the rest of the paper. An observation of 2-m temperature

[1] Available at http://apps.ecmwf.int/datasets/data/tigge/, see https://github.com/slerch/ppnn/tree/master/data_retrieval.

[2] Detailed definitions are available at https://software.ecmwf.int/wiki/display/TIGGE/Parameters.

[3] Similar sets of predictors have been used, for example, in Messner et al. (2017), Schlosser et al. (2018), and Taillardat et al. (2016, 2017).

[4] All maps in this article were produced using the R package ggmap (Kahle and Wickham 2013).

[5] Available at https://www.dwd.de/DE/klimaumwelt/cdc/cdc_node.html.

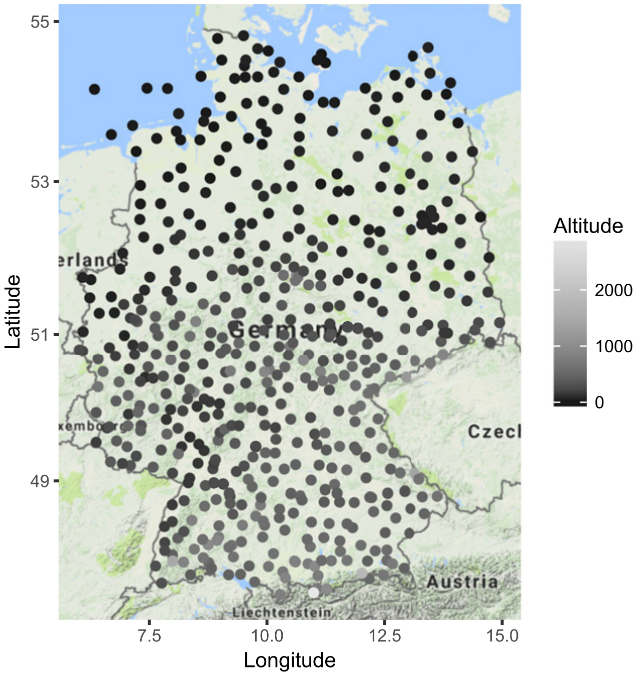


FIG. 2. Locations of DWD surface observation stations. The grayscale values of the points indicate the altitude (m).

at station $s \in \{1, \ldots, S\}$ and time $t \in \{1, \ldots, T\}$ will be denoted by $y_{s,t}$. For each s and t, the 50-member ECMWF ensemble forecast of variable v is given by $x_{s,t}^{v,1}, \ldots, x_{s,t}^{v,50}$, with mean value $x_{s,t}^{v,\mathrm{mean}}$ and standard deviation $x_{s,t}^{v,\mathrm{sd}}$. The mean values and standard deviations of all variables in the top part of Table 1 are combined with station-specific features in the bottom part, and aggregated into a vector of predictors $\mathbf{X}_{s,t} \in \mathbb{R}^p$, $p = 42$. Further, we write $\mathbf{X}_{s,t}^{\mathrm{t2m}}$ to denote the vector of predictors that only contains the mean value and standard deviation of the 2-m temperature forecasts.

3. Benchmark postprocessing techniques

a. Ensemble model output statistics

Within the general EMOS framework proposed by Gneiting et al. (2005), the conditional distribution of the weather variable of interest, $y_{s,t}$, given ensemble predictions $\mathbf{X}_{s,t}$, is modeled by a single parametric forecast distribution $F_{\boldsymbol{\theta}_{s,t}}$ with parameters $\boldsymbol{\theta}_{s,t} \in \mathbb{R}^d$:

$$y_{s,t}|\mathbf{X}_{s,t} \sim F_{\boldsymbol{\theta}_{s,t}}. \tag{1}$$

The parameters vary over space and time, and depend on the ensemble predictions $\mathbf{X}_{s,t}$ through suitable link functions $g: \mathbb{R}^p \rightarrow \mathbb{R}^d$:

$$\boldsymbol{\theta}_{s,t} = g(\mathbf{X}_{s,t}). \tag{2}$$

Here, we are interested in modeling the conditional distribution of temperature and follow Gneiting et al. (2005), who introduced a model based on ensemble predictions of temperature, $\mathbf{X}_{s,t}^{\mathrm{t2m}}$, only, where the forecast distribution is Gaussian with parameters $\boldsymbol{\theta}_{s,t} \in \mathbb{R}^2$ given by mean $\mu_{s,t}$ and standard deviation $\sigma_{s,t}$, that is,

$$y_{s,t}\Big|\mathbf{X}_{s,t}^{\mathrm{t2m}} \sim \mathcal{N}_{(\mu_{s,t},\sigma_{s,t})},$$

and where the link functions for the mean and standard deviation are affine functions of the ensemble mean and standard deviation, respectively:

$$(\mu_{s,t}, \sigma_{s,t}) = g(\mathbf{X}_{s,t}^{\mathrm{t2m}}) = (a_{s,t} + b_{s,t}x_{s,t}^{\mathrm{t2m,mean}}, c_{s,t} + d_{s,t}x_{s,t}^{\mathrm{t2m,sd}}). \tag{3}$$

Over the past decade, the EMOS framework has been extended from temperature to other weather variables including wind speed (Thorarinsdottir and Gneiting 2010; Lerch and Thorarinsdottir 2013; Baran and Lerch 2015; Scheuerer and Möller 2015) and precipitation (Messner et al. 2014; Scheuerer 2014; Scheuerer and Hamill 2015).

The model parameters (or EMOS coefficients) $\kappa_{s,t} = (a_{s,t}, b_{s,t}, c_{s,t}, d_{s,t})$ are estimated by minimizing the mean continuous ranked probability score (CRPS) as a function of the parameters over a training set. The CRPS is an example of a proper scoring rule (i.e., a mathematically principled loss function for distribution forecasts) and is a standard choice in meteorological applications. Details on the mathematical background of proper scoring rules and their use for model estimation are provided in the appendix.

Training sets are often considered to be composed of the most recent days only. However, as we did not find substantial differences in predictive performance, we estimate the coefficients over a fixed training set, they thus do not vary over time and we denote them by κ_s. Estimation is usually either performed locally (i.e., considering only forecast cases from the station of interest) or globally by pooling together forecasts and observations from all stations. We refer to the corresponding EMOS models as EMOS-loc and EMOS-gl, respectively. The parameters κ of the global model do not depend on the station s and are, thus, unable to correct location-specific deficiencies of the ensemble forecasts. Alternative approaches where training sets are selected based on similarities of weather situations or observation station characteristics were proposed by Junk et al. (2015) and Lerch and Baran (2017). Both EMOS-gl and EMOS-loc are implemented in R with the help of the scoringRules package (Jordan et al. 2018).

b. *Boosting for predictor selection in EMOS models*

Extending the EMOS framework to allow for including additional predictor variables is nontrivial as the increased number of parameters can result in overfitting. Messner et al. (2017) proposed a boosting algorithm for this purpose. In this approach components of the link function g in (2) are chosen to be an affine function for the mean $\mu_{s,t}$ and an exponential transformation of an affine function for the standard deviation $\sigma_{s,t}$:

$$(\mu_{s,t}, \sigma_{s,t}) = g(\mathbf{X}_{s,t}) = \{(1, \mathbf{X}_{s,t})^{\mathrm{T}}\boldsymbol{\beta}_{s,t}, \exp[(1, \mathbf{X}_{s,t})^{\mathrm{T}}\boldsymbol{\gamma}_{s,t}]\}\,. \tag{4}$$

Here, $\boldsymbol{\beta}_{s,t} \in \mathbb{R}^{p+1}$ and $\boldsymbol{\gamma}_{s,t} \in \mathbb{R}^{p+1}$ denote coefficient vectors corresponding to the vector of predictors $\mathbf{X}_{s,t}$ extended by a constant. As for the standard EMOS models, the coefficient vectors are estimated over fixed training periods and thus do not depend on t; we suppress the index in the following.

The boosting algorithm proceeds iteratively by updating the coefficient of the predictor that improves the current model fit most. As the coefficient vectors are initialized as $\boldsymbol{\beta}_s = \boldsymbol{\gamma}_s = 0$, only the most important variables will have nonzero coefficients if the algorithm is stopped before convergence. The contributions of the different predictors are assessed by computing average correlations to partial derivatives of the loss function with respect to $\mu_{s,t}$ and $\sigma_{s,t}$ over the training set. If the current model fit is improved, the coefficient vectors are updated by a predefined step size into the direction of steepest descent of linear approximations of the gradients.

We denote local EMOS models with an additional boosting step by EMOS-loc-bst. The tuning parameters of the algorithm were chosen by fitting models for a variety of choices and picking the configuration with the best out-of-sample predictions (see the online supplemental material) based on implementations in the R package crch (Messner et al. 2016). Note, however, that the results are not very sensitive to the exact choice of tuning parameters. For the local model considered here, the station-specific features in the bottom part of Table 1 are not relevant and are excluded from $\mathbf{X}_{s,t}$. Boosting-based variants of global EMOS models have also been tested, but result in worse forecasts.

The boosting-based EMOS-loc-bst model differs from the standard EMOS models (EMOS-gl and EMOS-loc) in several aspects. First, the boosting step allows us to include covariate information from predictor variables other than temperature forecasts. Second, the parameters are estimated by maximum likelihood estimation (i.e., by minimizing the mean logarithmic score by contrast to minimum CRPS estimation; see the appendix for details).[6] Further, the affine link function for the standard deviation in (3) is replaced by an affine function for the logarithm of the standard deviation in (4). By construction the boosting-based EMOS approach is unable to model interactions of the predictors. In principle, including nonlinear combinations (e.g., products) of predictors as additional input allows us to introduce such effects; however, initial tests indicated no substantial improvements.

c. *Quantile regression forests*

Parametric distributional regression models such as the EMOS methods described above require the choice of a suitable parametric family $F_{\boldsymbol{\theta}}$. While the conditional distribution of temperature can be well approximated by a Gaussian distribution, this poses a limitation for other weather variables such as wind speed or precipitation where the choice is less obvious (see, e.g., Baran and Lerch 2018).

Nonparametric distributional regression approaches provide alternatives that circumvent the choice of the parametric family. For example, quantile regression approaches approximate the conditional distribution by a set of quantiles. Within the context of postprocessing ensemble forecasts, Taillardat et al. (2016) proposed a quantile regression forest (QRF) model based on the work of Meinshausen (2006) that allows us to include additional predictor variables.

The QRF model is based on the idea of generating random forests from classification and regression trees (Breiman et al. 1984). These are binary decision trees obtained by iteratively splitting the training data into two groups according to some threshold for one of the predictors, chosen such that every split minimizes the sum of the variance of the response variable in each of the resulting groups. The splitting procedure is iterated until a stopping criterion is reached. The final groups (or terminal leaves) thus contain subsets of the training observations based on the predictor values, and out-of-sample forecasts at station s and time t can be obtained by proceeding through the decision tree according to the corresponding predictor values $\mathbf{X}_{s,t}$. Random forest models (Breiman 2001) increase the stability of the predictions by averaging over many random decision trees generated by selecting a random subset of the

[6] A recent development version of the R package crch provides implementations of CRPS-based model estimation and boosting. However, initial tests indicated slightly worse predictive performance; we thus focus on maximum likelihood-based methods instead.

predictors at each candidate split in conjunction with bagging (i.e., bootstrap aggregation of random subsamples of training sets). In the quantile regression forest approach, each tree provides an approximation of the distribution of the variable of interest given by the empirical cumulative distribution function (CDF) of the observation values in the terminal leaf associated with the current predictor values $\mathbf{X}_{s,t}$. Quantile forecasts can then be computed from the combined forecast distribution, which is obtained by averaging over all tree-based empirical CDFs.

We implement a local version of the QRF model where separate models are estimated for each station based on training sets that only contain past forecasts and observations from that specific station. As discussed by Taillardat et al. (2016), the predicted quantiles are necessarily restricted to the range of observed values in the training period by construction, which may be disadvantageous in cases of shorter training periods. However, global variants of the QRF model did not result in improved forecast performance even with only one year of training data; we will thus restrict attention to the local QRF model. The models are implemented using the quantregForest package (Meinshausen 2017) for R. Tuning parameters are chosen as for the EMOS-loc-bst model (see the supplemental material).

The QRF approach has recently been extended in several directions. Athey et al. (2016) propose a generalized version of random forest-based quantile regression based on theoretical considerations (GRF), which has been tested but did not result in improved forecast performance. Taillardat et al. (2017) combine QRF (and GRF) models and parametric distributional regression by fitting a parametric CDF to the observations in the terminal leaves instead of using the empirical CDF. Schlosser et al. (2018) combine parametric distributional regression and random forests for parameter estimation within the framework of a generalized additive model for location, scale, and shape.

4. Neural networks

In this section we will give a brief introduction to neural networks. For a more detailed treatment the interested reader is referred to more comprehensive resources (e.g., Nielsen 2015; Goodfellow et al. 2016). The network techniques are implemented using the Python libraries Keras (Chollet et al. 2015) and TensorFlow (Abadi et al. 2016).

Neural networks consist of several layers of nodes (Fig. 1), each of which is a weighted sum of all nodes j from the previous layer plus a bias term:

$$\sum_j w_j x_j + b\,. \tag{5}$$

The first layer contains the input values, or features, while the last layer represents the output values, or targets. In the layers in between, called hidden layers, each node value is passed through a nonlinear activation function. For this study, we use a rectified linear unit (ReLU):

$$\mathrm{ReLU}(x) = \max(0, x)\,.$$

This activation function allows the neural network to represent nonlinear functions. We tried other common nonlinear activation functions, such as sigmoid or hyperbolic tangent, but obtained the best results with ReLUs, which are the first choice for most applications these days. The weights and biases are optimized to reduce a loss function using stochastic gradient descent (SGD). Here, we employ an SGD version called Adam (Kingma and Ba 2014).

In this study we use networks without a hidden layer and with a single hidden layer (Fig. 1). The former, which we will call fully connected networks (FCNs), model the outputs as a linear combination of the inputs. The latter, called neural networks (NNs) here, are capable of representing nonlinear relationships and interactions. Introducing additional hidden layers to neural networks did not improve the predictions as additional model complexity increases the potential of overfitting. For more details on network hyperparameters, see the supplemental material.

a. *Neural networks for ensemble postprocessing*

Neural networks can be applied to a range of problems, such as regression and classification. The main difference between those options is in the contents and activation function of the output layer, as well as the loss function. Here, we use the neural network for the distributional regression task of postprocessing ensemble forecasts. Our output layer represents the distribution parameters $\mu_{s,t}$ and $\sigma_{s,t}$ of the Gaussian predictive distribution. No activation function is applied. The corresponding probabilistic forecast describes the conditional distribution of the observation $y_{s,t}$ given the predictors $\mathbf{X}_{s,t}$ as input features. As a loss function for determining the network parameters, we use the closed form expression of the CRPS for a Gaussian distribution; see (A2). This is a nonstandard choice in the neural network literature [D'Isanto and Polsterer (2018) is the only previous study to our knowledge] but provides a mathematically principled choice for the distributional regression problem at hand (see the appendix for the mathematical background). Other probabilistic neural network approaches include quantile regression (Taylor 2000) and distribution-to-distribution regression (Kou et al. 2018).

The simplest network model is a fully connected model based on predictors $\mathbf{X}_{s,t}^{\text{t2m}}$ [i.e., mean and standard deviation of ensemble predictions of temperature only (denoted by FCN)]. Apart from additional connections for the mean and standard deviation to the ensemble standard deviation and mean, respectively, the FCN model is conceptually equivalent to EMOS-gl, but differs in the parameter estimation approaches. A neural network with a hidden layer for the $\mathbf{X}_{s,t}^{\text{t2m}}$ input did not show any improvements over the simple linear model, suggesting that there are no nonlinear relationships to exploit. Additional information from auxiliary variables can be taken into account by considering the entire vector $\mathbf{X}_{s,t}$ of predictors as input features. The corresponding fully connected and neural network models are referred to as FCN-aux and NN-aux.

b. *Station embeddings*

To enable the networks to learn station-specific information, we use embeddings, a common technique in natural language processing and recommender systems. An embedding e is a mapping from a discrete object, in our case the station ID s, to a vector of real numbers $\mathbf{X}_s^{\text{emb}}$ (Guo and Berkhahn 2016):

$$e : s \mapsto \mathbf{X}_s^{\text{emb}} \,,$$

where $\mathbf{X}_s^{\text{emb}} \in \mathbb{R}^{n_{\text{emb}}}$; n_{emb} is the number of elements in the embedding vector which are also referred to as latent features. These latent features encode information about each station s but do not correspond to any real variable. In total then, the embedding matrix has dimension $S \times n_{\text{emb}}$, where S is the number of stations. The latent features $\mathbf{X}_s^{\text{emb}}$ are concatenated with the predictors, $\mathbf{X}_{s,t}^{\text{t2m}}$ or $\mathbf{X}_{s,t}$, and are updated along with the weights and biases during training. This allows the algorithm to learn a specific set of numbers for each station. Here, we use $n_{\text{emb}} = 2$ because larger values did not improve the predictions.

The fully connected network with input features $\mathbf{X}_{s,t}^{\text{t2m}}$ and embeddings is abbreviated by FCN-emb. As with FCN, adding a hidden layer did not improve the results. Fully connected and neural networks with both, station embeddings and auxiliary inputs $\mathbf{X}_{s,t}$, are denoted by FCN-aux-emb and NN-aux-emb.

c. *Further network details*

Neural networks with a large number of parameters (i.e., weights and biases) can suffer from overfitting. One way to reduce overfitting is to stop training early. When to stop can be guessed by taking out a subset (20%) from the training set (2007–15 or 2015) and checking when the score on this separate dataset stops improving. This gives a good approximation of when to stop training on the full training set without using the actual 2016 validation set during training. Other common regularization techniques to prevent overfitting, such as dropout or weight decay (L2 regularization), were not successful in our case for reasons unclear to us. Further investigation in follow-on studies may be helpful.

Finally, we train ensembles of 10 neural networks with different random initial parameters for each configuration and average over the forecast distribution parameter estimates to obtain $\boldsymbol{\theta}_{s,t}$. For the more complex network models this helps to stabilize the parameter estimates by reducing the variability due to random variations between model runs and slightly improves the forecasts.

5. Results

Tuning parameters for all benchmark and network models are listed in the supplemental material (Tables S1 and S2). Details on the employed evaluation methods are provided in the appendix.

a. *General results*

The CRPS values averaged over all stations and the entire 2016 validation period are summarized in Table 2.[7] For the 2015 training period, EMOS-gl gives a 13% relative improvement compared to the raw ECMWF ensemble forecasts in terms of mean CRPS. As expected, FCN, which mimics the design of EMOS-gl, achieves a very similar score. Adding local station information in EMOS-loc and FCN-emb improves the global score by another 10%. While EMOS-loc estimates a separate model for each station, FCN-emb can be seen as a global network–based implementation of EMOS-loc. Adding covariate information through auxiliary variables results in an improvement for the fully connected models similar to that of adding station information. Combining auxiliary variables and station embeddings in FCN-emb-aux improves the mean CRPS further to 0.88 but the effects do not stack linearly. Adding covariate information in EMOS models using boosting (EMOS-loc-bst) outperforms FCN-emb-aux by 3%. Allowing for nonlinear interactions of station information and auxiliary variables using a neural

[7] To account for the intertwined choice of scoring rules for model estimation and evaluation (Gebetsberger et al. 2017), we have also evaluated the models using LogS. However, as the results are very similar to those reported here and computation of LogS for the raw ensemble and QRF forecasts is problematic (Krüger et al. 2016), we focus on CRPS-based evaluation.

TABLE 2. Mean CRPSs for raw and postprocessed ECMWF ensemble forecasts, averaged over all available observations during calendar year 2016. The lowest (i.e., best) values are marked in boldface.

Model	Description	Mean CRPS for training period	
		2015	2007–15
Raw ensemble		1.16	1.16
Benchmark postprocessing methods			
EMOS-gl	Global EMOS	1.01	1.00
EMOS-loc	Local EMOS	0.90	0.90
EMOS-loc-bst	Local EMOS with boosting	0.85	0.80
QRF	Local quantile regression forest	0.95	0.81
Neural network models			
FCN	Fully connected network	1.01	1.01
FCN-aux	. . .with auxiliary predictors	0.92	0.91
FCN-emb	. . .with station embeddings	0.91	0.91
FCN-aux-emb	. . .with both of the above	0.88	0.87
NN-aux	One-hidden-layer NN with auxiliary predictors	0.90	0.86
NN-aux-emb	. . .and station embeddings	**0.82**	**0.78**

network (NN-aux-emb) achieves the best results, improving the best benchmark technique (EMOS-loc-bst) by 3% for a total improvement compared to the raw ensemble of 29%. The QRF model is unable to compete with most of the postprocessing models for the 2015 training period.

The relative scores and model rankings for the 2007–15 training period closely match those of the 2015 period. For the linear models (EMOS-gl, EMOS-loc, and all FCN) more data does not improve the score by much. For EMOS-loc-bst and the neural network models, however, the skill is increased by 4%–5%. This suggests that longer training periods are most efficiently exploited by more complex, nonlinear models. QRF improves the most, now being among the best models, which indicates a minimum data amount required for this method to work. This is likely due to the limitation of predicted quantiles to the range of observed values in the training data; see section 3c.

To assess calibration, verification rank and probability integral transform (PIT) histograms of raw and postprocessed forecasts are shown in the supplemental material. The raw ensemble forecasts are underdispersed, as indicated by the U-shaped verification rank histogram; that is, observations tend to fall outside the range of the ensemble too frequently. By contrast, all postprocessed forecast distributions are substantially better calibrated and the corresponding PIT histograms show much smaller deviations from uniformity. All models show a slight overprediction of high temperatures and, with the exception of QRF, an underprediction of low values. This might be due to residual skewness (Gebetsberger et al. 2018). The linear EMOS and FCN models as well as QRF are further slightly overdispersive, as indicated by the inverse U-shaped top parts of the histogram.

b. Station-by-station results

Figure 3 shows the station-wise distribution of the continuous ranked probability skill score (CRPSS), which measures the probabilistic skill relative to a reference model. Positive values indicate an improvement over the reference. Compared to the raw ensemble, forecasts at most stations are improved by all postprocessing methods with only a few negative outliers. Compared to EMOS-loc, only FCN-aux-emb, the neural network models, and EMOS-loc-bst show improvements at the majority of the stations. Corresponding plots with the three best-performing models as reference experiments are provided in the supplemental material. It is interesting to note that the network models, with the exception of FCN and FCN-emb, have more outliers, particularly for negative values compared to the EMOS methods and QRF, which have very few negative outliers. This might be due to a few stations with strongly location-specific error characteristics that the locally estimated benchmark models are better able to capture. Training with data from 2007 to 2015 alleviates this somewhat.

Figure 4 shows maps with the best-performing models in terms of mean CRPS for each station. For the majority of stations NN-aux-emb provides the best predictions. The variability of station-specific best models is greater for the 2015 training period compared to 2007–15. The top three models for the 2015 period are NN-aux-emb (best at 65.9% of stations), EMOS-loc-bst (16.0%), and NN-aux (7.2%), and for 2007–15 they are

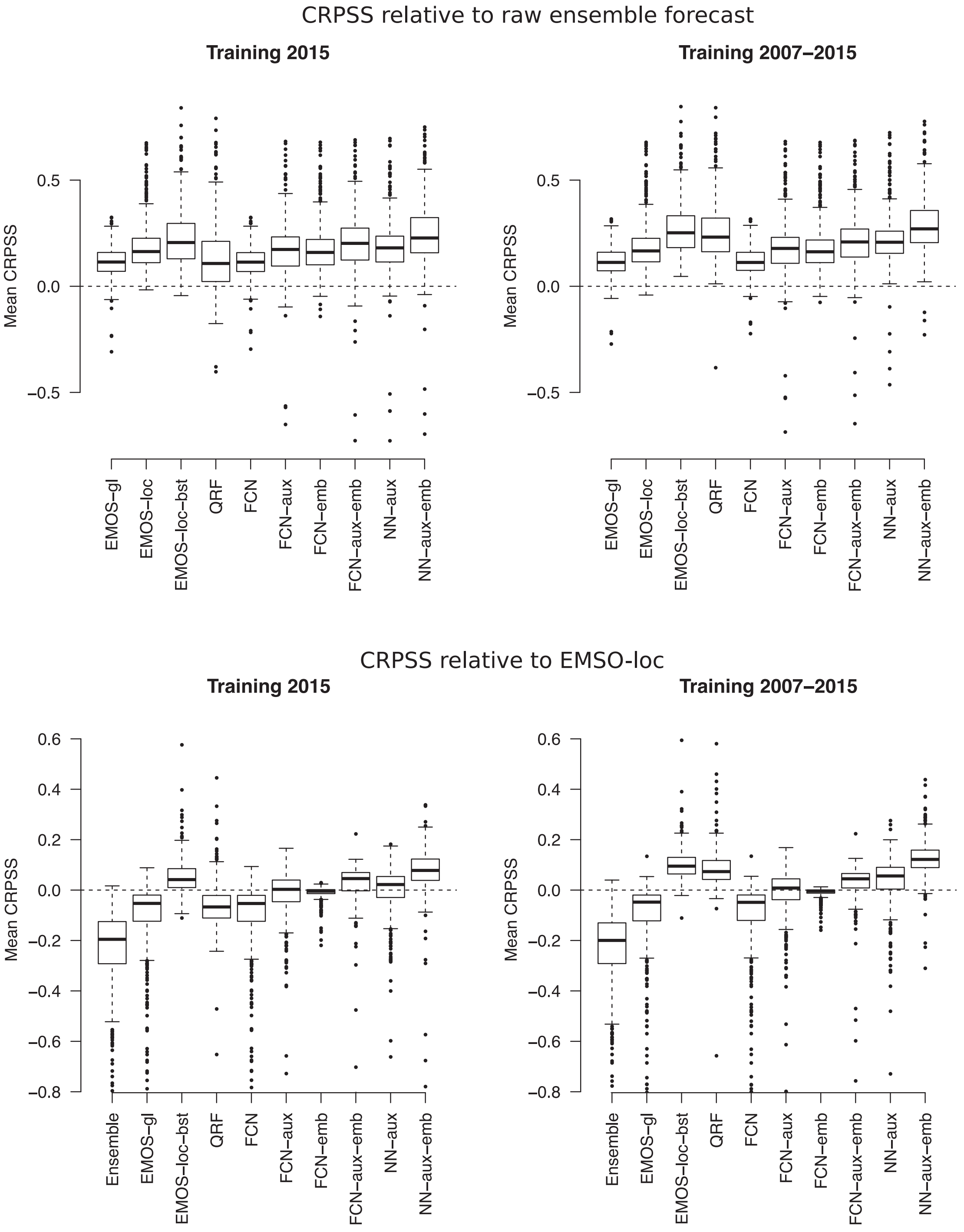


FIG. 3. Boxplots of stationwise mean CRPSS of all postprocessing models using the (top) raw ensemble and (bottom) EMOS-loc as the reference forecast. A dot within each box represents the mean CRPSS at one of the observation stations. The CRPSS is computed so that positive values indicate an improvement of the model specified on the horizontal axis over the reference. Similar plots with different reference models are provided in the supplemental material.

NN-aux-emb (73.5%), EMOS-loc-bst (12.4%), and QRF (7.4%). At coastal and offshore locations, particularly for the shorter training period, the benchmark methods tend to outperform the network methods. Ensemble forecast errors at these locations likely have a strong location-specific component that might be easier to capture for the locally estimated EMOS and QRF methods.

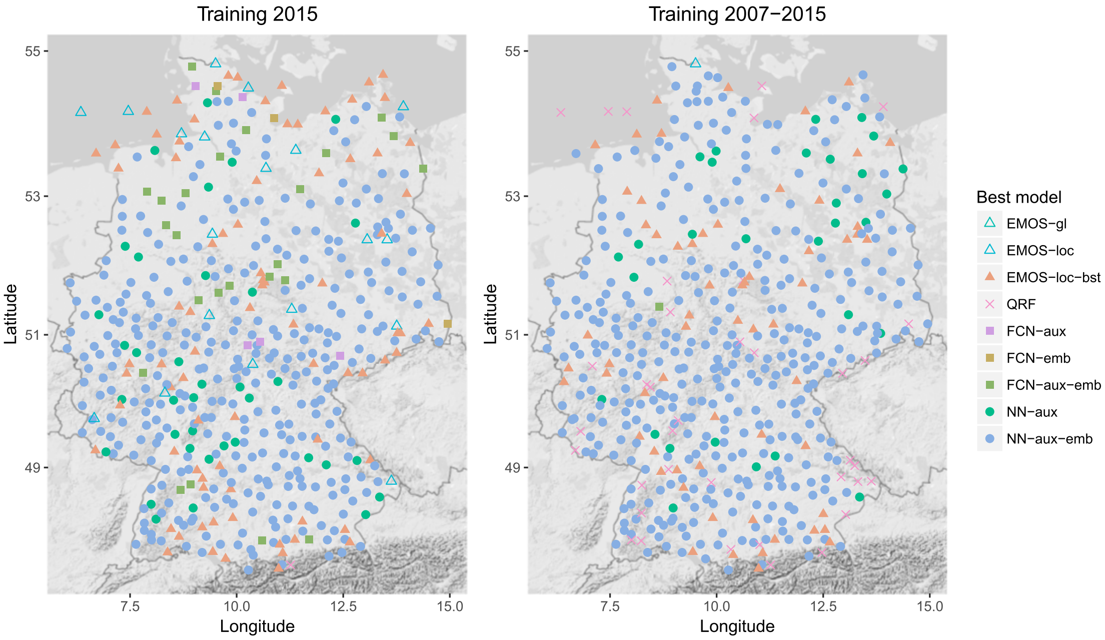


FIG. 4. Observation station locations color coded by the best performing model (in terms of mean CRPS over calendar year 2016) for models trained on data from (left) 2015 and (right) 2007 to 2015. Point shapes indicate the type of model.

Additionally, we evaluated the statistical significance of the differences between the competing postprocessing methods using a combination of Diebold–Mariano tests (Diebold and Mariano 1995) and a Benjamini and Hochberg (1995) procedure to account for temporal and spatial dependencies of forecast errors. We thereby follow the suggestions of Wilks (2016); the mathematical details are deferred to the appendix. The results (provided in the supplemental material) generally indicate high ratios of stations with significant score differences in favor of the neural network models. Even when compared to the second-best-performing model, EMOS-loc-bst, NN-aux-emb is significantly better at 30% of the stations and worse at only 2% or less for both training periods.

c. *Computational aspects*

While a direct comparison of computation times for the different methods is difficult, even the most complex network methods are a factor of 2 or more faster than EMOS-loc-bst. This includes creating an ensemble of 10 different model realizations. QRF is by far the slowest method, being roughly 10 times slower than EMOS-loc-bst. Complex neural networks benefit substantially from running on a graphics processing unit (GPU) compared to running on the core processing unit (CPU; roughly 6 times slower for NN-aux-emb). Neural network–ready GPUs are now widely available in many scientific computing environments or via cloud computing.[8] For more details on the computational methods and results see the supplemental material.

6. Feature importance

To assess the relative importance of all features, we use a technique called permutation importance that was first described within the context of random forests (Breiman 2001). We randomly shuffle each predictor/feature in the validation set one at a time and observe the increase in mean CRPS compared to the unpermuted features. While unable to capture colinearities between features, this method does not require reestimating the model with each individual feature omitted.

Consider a random permutation of station and time indices $\pi(s, t)$ and let $\mathbf{X}_{s,t}^{\mathrm{perm}_v}$ denote the vector of predictors where variable v is permuted according to π (i.e., a vector with jth entry):

$$\mathbf{X}_{s,t}^{\mathrm{perm}_v(j)} = \begin{cases} \mathbf{X}_{s,t}^{(j)}, & j \neq v \\ \mathbf{X}_{\pi(s,t)}^{(v)}, & j = v \end{cases} \quad \text{for} \quad j = 1, \ldots, p\,.$$

The importance of input feature v is computed as the mean CRPS difference:

[8] For example, see https://colab.research.google.com/.

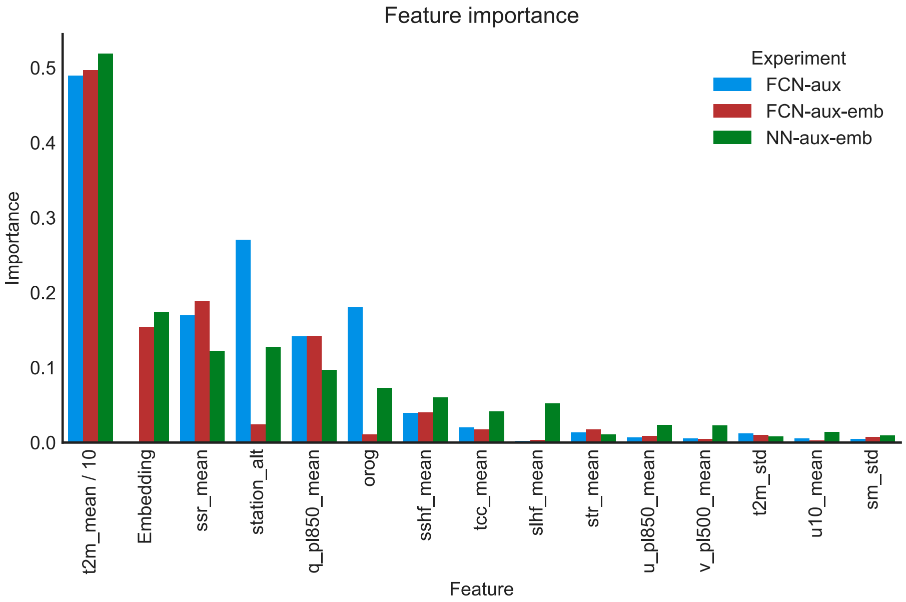


FIG. 5. Feature importance for the 15 most important predictors. Note that the values for t2m_mean are divided by 10. See Table 1 for variable abbreviations and descriptions.

$$\text{Importance}(v) = \frac{1}{ST}\sum_{s=1}^{S}\sum_{t=1}^{T}[\text{CRPS}(F|\mathbf{X}_{s,t}^{\text{perm}_v}, y_{s,t}) - \text{CRPS}(F|\mathbf{X}_{s,t}, y_{s,t})] \, ,$$

where we average over the entire evaluation set and $F|\mathbf{X}$ denotes the conditional forecast distribution given a vector of predictors.

We picked three network setups to investigate how feature importance changes by adding station embeddings and a nonlinear layer (Fig. 5). For the linear model without station embeddings (FCN-aux), the station altitude and orography, the altitude of the model grid cell, are the most important predictors after the mean temperature forecast. This makes sense since our interpolation from the forecast model grid to the station does not adjust for the height of the surface station. The only other features with significant importance are the mean shortwave radiation flux and the 850-hPa specific humidity. Adding station embeddings (FCN-aux-emb) reduces the significance of the station altitude information, which now seems to be encoded in the latent embedding features. The nonlinearity added by the hidden layer in NN-aux-emb increases the sensitivity to permuting input features overall and distributes the feature importance more evenly. In particular, we note an increase in the importance of the station altitude and orography but also the sensible and latent heat flux and total cloud cover.

The most important features, apart from the obvious mean forecast temperature and station altitude, seem to be indicative of insolation, either directly like the shortwave flux or indirectly like the 850-hPa humidity. It is interesting that the latter seems to be picked by the algorithms as a proxy for cloud cover rather than the direct cloud cover feature, potentially due to a lack of forecast skill of the total cloud cover predictions (e.g., Hemri et al. 2016). Curiously, the temperature standard deviation is not an important feature for the postprocessing models. We suspect that this is a consequence of the low correlation between the raw ensemble standard deviation and the forecast error ($r = 0.15$ on the test set) and the general underdispersion (mean spread–error ratio of 0.51). The postprocessing algorithms almost double the spread to achieve a spread–error ratio of 0.95. The correlation of the raw and postprocessed ensemble spreads is 0.39. suggesting that the postprocessing is mostly an additive correction to the ensemble spread.

Note that this method of assessing feature importance is in principle possible for boosting- and QRF-based models. However, for the local implementations of the algorithm the importance changes from station to station, making interpretation more difficult.

7. Discussion

Here, we discuss some approaches we attempted that failed to improve our results, as well as directions for future research.

Having to describe the distribution of the target variable in parametric techniques is a nontrivial task. For temperature, a Gaussian distribution is a good approximation but for other variables, such as wind speed or precipitation, finding a distribution that fits the data is a substantial challenge (e.g., Taillardat et al. 2016;

Baran and Lerch 2018). Ideally, a machine learning algorithm would learn to predict the full probability distribution rather than distribution parameters only. One way to achieve this is to approximate the forecast distribution by a combination of uniform distributions and predicting the probability of the temperature being within prespecified bins. Initial experiments indicate that the neural network is able to produce a good approximation of a Gaussian distribution but the skill was comparable only to the raw ensemble. This suggests that for target variables that are well approximated by a parametric distribution, utilizing these distributions is advantageous. One direction for future research is to apply this approach to more complex variables.

Standard EMOS models are often estimated based on so-called rolling training windows with data from previous days only in order to incorporate temporal dependencies of ensemble forecast errors. For neural networks, one way to incorporate temporal dependencies is to use convolutional or recurrent neural networks (Schmidhuber 2015) which can proces sequences as an input. In our tests, this leads to more overfitting without an improvement in the validation score. For other datasets, however, we believe that these approaches are worth revisiting. Temporal dependencies of forecast errors might further include seasonal effects. For standard EMOS models, it is possible to account for seasonality by estimating the model based on a centered window $[d_0 - m, d_0 + m]$ around the current day d_0. For the local EMOS model this resulted in negligible improvements only. For postprocessing models with additional predictors seasonal effects can, for example, be included by considering the month of d_0 as an input feature.

One popular way to combat overfitting in machine learning algorithms is through data augmentation. In the example of image recognition models, the training images are randomly rotated, flipped, zoomed, etc. to artificially increase the sample size (e.g., Krizhevsky et al. 2012). We tried a similar approach by adding random noise of a reasonable scale to the input features, but found no improvement in the validation score. A potential alternative to adding random noise might be augmenting the forecasts for a station with data from neighboring stations or grid points.

Similarly to rolling training windows for the traditional EMOS models, we tried updating the neural network each day during the validation period with the data from the previous time step, but found no improvements. This supports our observation that rolling training windows only bring marginal improvements for the benchmark EMOS models. Such an online learning approach could be more relevant in an operational setting, however, where model versions might change frequently or it is too expensive to reestimate the entire postprocessing model every time new data become available.

We have restricted the set of predictors to observation station characteristics and summary statistics (mean and standard deviation) of ensemble predictions of several weather variables. Recently, flexible distribution-to-distribution regression network models have been proposed in the machine learning literature (e.g., Oliva et al. 2013; Kou et al. 2018). Adaptations of such approaches might enable the use of the entire ensemble forecast of each predictor variable as an input feature. However, training of these substantially more complex models likely requires longer training periods than were possible in our study.

Another possible extension would be to postprocess forecasts on the entire two-dimensional grid, rather than individual stations locations, for example, by using convolutional neural networks. This adds computational complexity and probably requires more training data but could provide information about the large-scale weather patterns and help to produce spatially consistent predictions.

We have considered probabilistic forecasts of a single weather variable at a single location and look-ahead time only. However, many applications require accurate models of cross-variable, spatial, and temporal dependence structures, and much recent work has been focused on multivariate postprocessing methods (e.g., Schefzik et al. 2013). Extending the neural network–based approaches to multivariate forecast distributions accounting for such dependencies presents a promising starting point for future research.

8. Conclusions

In this study we demonstrated how neural networks can be used for distributional regression postprocessing of ensemble weather forecasts. Our neural network models significantly outperform state-of-the-art postprocessing techniques while being computationally more efficient. The main advantages of using neural networks are the ability to capture nonlinear relations between arbitrary predictors and distribution parameters without having to specify appropriate link functions, and the ease of adding station information into a global model by using embeddings. The network model parameters are estimated by optimizing the CRPS, a nonstandard choice in the machine learning literature tailored to probabilistic forecasting. Furthermore, the rapid pace of development in the deep learning community provides flexible and efficient modeling techniques and software libraries. The presented approach can therefore be easily applied to other problems.

The building blocks of our network model architecture provide general insight into the relative importance of model properties for postprocessing ensemble forecasts. Specifically, the results indicate that encoding local information is very important for providing skillful probabilistic temperature forecasts. Further, including covariate information via auxiliary variables improves the results considerably, particularly when allowing for nonlinear relations of predictors and forecast distribution parameters. Ideally, any postprocessing model should thus strive to incorporate all of these aspects.

We also showed that a trained machine learning model can be used to gain meteorological insight. In our case, it allowed us to identify the variables that are most important for correcting systematic temperature forecast errors of the ensemble. Within this context, neural networks are somewhat interpretable and give us more information than we originally asked for. While a direct interpretation of the individual parameters of the model is intractable, this challenges the common notion of neural networks as pure black boxes.

Because of their flexibility, neural networks are ideally suited to handle the increasing amounts of model and observation data as well as the diverse requirements for correcting multifaceted aspects of systematic ensemble forecast errors. We anticipate, therefore, that they will provide a valuable addition to the modeler's toolkit for many areas of statistical postprocessing and forecasting.

Acknowledgments. The research leading to these results has been done within the subprojects A6 "Representing forecast uncertainty using stochastic physical parameterizations" and C7 "Statistical postprocessing and stochastic physics for ensemble predictions" of the Transregional Collaborative Research Center SFB/TRR 165 "Waves to Weather" funded by the German Research Foundation (DFG). SL is grateful for infrastructural support by the Klaus Tschira Foundation. The authors thank Tilmann Gneiting, Alexander Jordan, and Maxime Taillardat for helpful discussions and for providing code. The initial impetus for this work stems from a meeting with Kai Polsterer, who presented a probabilistic neural network–based approach to astrophysical image data analysis. We are grateful to Jakob Messner and two anonymous referees for constructive comments on an earlier version of the manuscript.

APPENDIX

Forecast Evaluation

For the purpose of this appendix, we denote a generic probabilistic forecast for 2-m temperature $y_{s,t}$ at station s and time t by $F_{s,t}$. Note that $F_{s,t}$ may be a parametric forecast distribution represented by CDF or a probability density function (PDF), an ensemble forecast $x_{s,t}^{\mathrm{t2m},1}, \ldots, x_{s,t}^{\mathrm{t2m},50}$, or a set of quantiles. We may choose to suppress the index s, t at times for ease of notation.

a. Calibration and sharpness

As argued by Gneiting et al. (2007), probabilistic forecasts should generally aim to maximize sharpness subject to calibration. In a nutshell, a forecast is called calibrated if the realizing observation cannot be distinguished from a random draw from the forecast distribution. Calibration thus refers to the statistical consistency between forecast distribution and observation. By contrast, sharpness is a property of the forecast only and refers to the concentration of the predictive distribution. The calibration of ensemble forecasts can be assessed via verification rank (VR) histograms summarizing the distribution of ranks of the observation $y_{s,t}$ when it is pooled with the ensemble forecast (Hamill 2001; Gneiting et al. 2007; Wilks 2011). For continuous forecast distributions, histograms of the PIT $F_{s,t}(y_{s,t})$ provide analogs of verification rank histograms. Calibrated forecasts result in uniform VR and PIT histograms, and deviations from uniformity indicate specific systematic errors such as biases or an underrepresentation of the forecast uncertainty.

b. Proper scoring rules

For comparative model assessment, proper scoring rules allow simultaneous evaluation of calibration and sharpness (Gneiting and Raftery 2007). A scoring rule $S(F, y)$ assigns a numerical score to a pair of probabilistic forecasts F and corresponding realizing observations y, and is called proper relative to a class of forecast distributions $\mathscr{F}$ if

$$\mathbb{E}_{Y\sim G}S(G,Y) \le \mathbb{E}_{Y\sim G}S(F,Y) \quad \text{for all} \quad F,G \in \mathscr{F},$$

that is, if the expected score is optimized if the true distribution of the observation is issued as forecast. Here, scoring rules are considered to be negatively oriented, with smaller scores indicating better forecasts

Popular examples of proper scoring rules include the logarithmic score (LogS; Good 1952):

$$\mathrm{LogS}(F,y) = -\log[f(y)],$$

where y denotes the observations and f denotes the PDF of the forecast distribution and the continuous ranked probability score (CRPS; Matheson and Winkler 1976):

$$\mathrm{CRPS}(F,y) = \int_{-\infty}^{\infty} [F(z) - 1(y \le z)]^2\, dz, \quad \text{(A1)}$$

where F denotes the CDF of the forecast distribution with finite first moment and $1(y \le z)$ is an indicator function that is 1 if $y \le z$ and 0 otherwise. The integral in (A1) can be computed analytically for ensemble forecasts and a variety of continuous forecast distributions (see, e.g., Jordan et al. 2018). Specifically, the CRPS of a Gaussian distribution with mean value μ and standard deviation σ can be computed as

$$\mathrm{CRPS}(F_{\mu,\sigma}, y) = \sigma\left\{\frac{y-\mu}{\sigma}\left[2\Phi\left(\frac{y-\mu}{\sigma}\right) - 1\right] + 2\varphi\left(\frac{y-\mu}{\sigma}\right) - \frac{1}{\sqrt{\pi}}\right\}, \tag{A2}$$

where Φ and φ denote the CDF and PDF of a standard Gaussian distribution, respectively (Gneiting et al. 2005).

Apart from forecast evaluation, proper scoring rules can also be used for parameter estimation. Following the generic optimum score estimation framework of Gneiting and Raftery (2007, section 9.1), the parameters of a forecast distribution are determined by optimizing the value of a proper scoring rule, on average over a training sample. Optimum score estimation based on the LogS then corresponds to classical maximum likelihood estimation, whereas optimum score estimation based on the CRPS is often employed as a more robust alternative in meteorological applications. Analytical closed-form solutions of the CRPS, for example for a Gaussian distribution in (A2), allow for computing analytical gradient functions that can be leveraged in numerical optimization; see Jordan et al. (2018) for details.

In practical applications, scoring rules are usually computed as averages over stations and/or time periods. To assess the relative improvement over a reference forecast F_{ref}, we further introduce the continuous ranked probability skill score:

$$\mathrm{CRPSS}(F, y) = 1 - \frac{\mathrm{CRPS}(F, y)}{\mathrm{CRPS}(F_{\mathrm{ref}}, y)},$$

which is positively oriented and can be interpreted as a relative improvement over the reference. The CRPSS is usually computed as the skill score of the CRPS averages.

c. Statistical tests of equal predictive performance

Formal statistical tests of equal forecast performance for assessing statistical significance of score differences have been widely used in the economic literature. Consider two forecasts, F^1 and F^2, with corresponding mean scores $\overline{S}(F^i) = 1/n\sum_{j=1}^{n} S(F_j^i, y_j)$ for $i = 1, 2$ over a test $j = 1, \ldots, n$, where we assume that the forecast F_j^i was issued k time steps before the observation y_j was recorded. Diebold and Mariano (1995) propose the test statistic

$$t_n = \sqrt{n}\,\frac{\overline{S}(F^1) - \overline{S}(F^2)}{\hat{\sigma}_n},$$

where $\hat{\sigma}_n$ is an estimator of the asymptotic standard deviation of the score difference between F^1 and F^2. Under standard regularity conditions, t_n asymptotically follows a standard normal distribution under the null hypothesis of equal predictive performance of F^1 and F^2. Thereby, negative values of t_n indicate superior predictive performance of F^1, whereas positive values indicate superior performance of F^2. To account for temporal dependencies in the score differences, we use the square root of the sample autocovariance up to lag $k-1$ as estimator $\hat{\sigma}_n$ following Diebold and Mariano (1995). We employ Diebold–Mariano tests on an observation station level; that is, the mean CRPS values are determined by averaging over all scores at the specific station $s_0 \in \{1, \ldots, S\}$ of interest:

$$\overline{\mathrm{CRPS}}(F_{s_0}^i) = \frac{1}{T}\sum_{t=1}^{T} F_{s_0,t}^i,$$

where $t = 1, \ldots, T$ denotes days in the evaluation period.

Compared to previous uses of Diebold–Mariano tests in postprocessing applications (e.g., Baran and Lerch 2016), we further account for spatial dependencies of score differences at the different stations. Following the suggestions of Wilks (2016), we apply a Benjamini and Hochberg (1995) procedure to control the false discovery rate at level $\alpha = 0.05$. In a nutshell, the algorithm requires a higher standard in order to reject a local null hypothesis of equal predictive performance by selecting a threshold p value (p^*) based on the set of ordered local p values: $p_{(1)}, \ldots, p_{(S)}$. Particularly, p^* is the largest $p_{(i)}$ that is not larger than $i/S \times \alpha$, where S is the number of tests (i.e., the number of stations in the evaluation set).

REFERENCES

Abadi, M., and Coauthors, 2016: Tensorflow: A system for large-scale machine learning. *Proc. USENIX 12th Symp. on Operating Systems Design and Implementation*, Savannah, GA, Advanced Computing Systems Association, 265–283, https://www.usenix.org/system/files/conference/osdi16/osdi16-abadi.pdf.

Angermueller, C., T. Pärnamaa, L. Parts, and O. Stegle, 2016: Deep learning for computational biology. *Mol. Syst. Biol.*, **12**, 878, https://doi.org/10.15252/msb.20156651.

Athey, S., J. Tibshirani, and S. Wager, 2016: Generalized random forests. arXiv.org, https://arxiv.org/abs/1610.01271.

Baran, S., and S. Lerch, 2015: Log-normal distribution based ensemble model output statistics models for probabilistic wind-speed forecasting. *Quart. J. Roy. Meteor. Soc.*, **141**, 2289–2299, https://doi.org/10.1002/qj.2521.

——, and ——, 2016: Mixture EMOS model for calibrating ensemble forecasts of wind speed. *Environmetrics*, **27**, 116–130, https://doi.org/10.1002/env.2380.

——, and ——, 2018: Combining predictive distributions for the statistical post-processing of ensemble forecasts. *Int. J. Forecasting*, **34**, 477–496, https://doi.org/10.1016/j.ijforecast.2018.01.005.

Bauer, P., A. Thorpe, and G. Brunet, 2015: The quiet revolution of numerical weather prediction. *Nature*, **525**, 47–55, https://doi.org/10.1038/nature14956.

Benjamini, Y., and Y. Hochberg, 1995: Controlling the false discovery rate: A practical and powerful approach to multiple testing. *J. Roy. Stat. Soc.*, **57B**, 289–300.

Bougeault, P., and Coauthors, 2010: The THORPEX Interactive Grand Global Ensemble. *Bull. Amer. Meteor. Soc.*, **91**, 1059–1072, https://doi.org/10.1175/2010BAMS2853.1.

Breiman, L., 2001: Random forests. *Mach. Learn.*, **45**, 5–32, https://doi.org/10.1023/A:1010933404324.

——, J. H. Friedman, R. A. Olshen, and C. J. Stone, 1984: *Classification and Regression Trees*. Wadsworth, 368 pp.

Chollet, F., and Coauthors, 2015: Keras: The Python Deep Learning library. https://keras.io.

Chu, Y., H. T. C. Pedro, and C. F. M. Coimbra, 2013: Hybrid intra-hour DNI forecasts with sky image processing enhanced by stochastic learning. *Sol. Energy*, **98**, 592–603, https://doi.org/10.1016/j.solener.2013.10.020.

Diebold, F. X., and R. S. Mariano, 1995: Comparing predictive accuracy. *J. Bus. Econ. Stat.*, **13**, 253–263, https://doi.org/10.1080/07350015.1995.10524599.

D'Isanto, A., and K. L. Polsterer, 2018: Photometric redshift estimation via deep learning-generalized and pre-classification-less, image based, fully probabilistic redshifts. *Astron. Astrophys.*, **609**, A111, https://doi.org/10.1051/0004-6361/201731326.

Gagne, D. J., A. McGovern, and M. Xue, 2014: Machine learning enhancement of storm-scale ensemble probabilistic quantitative precipitation forecasts. *Wea. Forecasting*, **29**, 1024–1043, https://doi.org/10.1175/WAF-D-13-00108.1.

——, ——, S. E. Haupt, R. A. Sobash, J. K. Williams, and M. Xue, 2017: Storm-based probabilistic hail forecasting with machine learning applied to convection-allowing ensembles. *Wea. Forecasting*, **32**, 1819–1840, https://doi.org/10.1175/WAF-D-17-0010.1.

Gebetsberger, M., J. W. Messner, G. J. Mayr, and A. Zeileis, 2017: Estimation methods for non-homogeneous regression models: Minimum continuous ranked probability score vs. maximum likelihood. Faculty of Economics and Statistics Working Paper 2017-23, University of Innsbruck, 21 pp., https://ideas.repec.org/p/inn/wpaper/2017-23.html.

——, R. Stauffer, G. J. Mayr, and A. Zeileis, 2018: Skewed logistic distribution for statistical temperature post-processing in mountainous areas. Faculty of Economics and Statistics Working Paper 2018-06, University of Innsbruck, 16 pp., https://ideas.repec.org/p/inn/wpaper/2018-06.html.

Gentine, P., M. S. Pritchard, S. Rasp, G. Reinaudi, and G. Yacalis, 2018: Could machine learning break the convection parameterization deadlock? *Geophys. Res. Lett.*, **45**, 5742–5751, https://doi.org/10.1029/2018GL078202.

Gneiting, T., and A. E. Raftery, 2007: Strictly proper scoring rules, prediction, and estimation. *J. Amer. Stat. Assoc.*, **102**, 359–378, https://doi.org/10.1198/016214506000001437.

——, ——, A. H. Westveld, and T. Goldman, 2005: Calibrated probabilistic forecasting using ensemble model output statistics and minimum CRPS estimation. *Mon. Wea. Rev.*, **133**, 1098–1118, https://doi.org/10.1175/MWR2904.1.

——, F. Balabdaoui, and A. E. Raftery, 2007: Probabilistic forecasts, calibration and sharpness. *J. Roy. Stat. Soc.*, **69B**, 243–268, https://doi.org/10.1111/j.1467-9868.2007.00587.x.

Goh, G. B., N. O. Hodas, and A. Vishnu, 2017: Deep learning for computational chemistry. *J. Comput. Chem.*, **38**, 1291–1307, https://doi.org/10.1002/jcc.24764.

Good, I. J., 1952: Rational decisions. *J. Roy. Stat. Soc.*, **14B**, 107–114.

Goodfellow, I., Y. Bengio, and A. Courville, 2016: *Deep Learning*. MIT Press, 775 pp.

Guo, C., and F. Berkhahn, 2016: Entity embeddings of categorical variables. arXiv.org, https://arxiv.org/abs/1604.06737.

Hamill, T. M., 2001: Interpretation of rank histograms for verifying ensemble forecasts. *Mon. Wea. Rev.*, **129**, 550–560, https://doi.org/10.1175/1520-0493(2001)129<0550:IORHFV>2.0.CO;2.

Hemri, S., M. Scheuerer, F. Pappenberger, K. Bogner, and T. Haiden, 2014: Trends in the predictive performance of raw ensemble weather forecasts. *Geophys. Res. Lett.*, **41**, 9197–9205, https://doi.org/10.1002/2014GL062472.

——, T. Haiden, and F. Pappenberger, 2016: Discrete postprocessing of total cloud cover ensemble forecasts. *Mon. Wea. Rev.*, **144**, 2565–2577, https://doi.org/10.1175/MWR-D-15-0426.1.

Jordan, A., F. Krüger, and S. Lerch, 2018: Evaluating probabilistic forecasts with scoringRules. arXiv.org, https://arxiv.org/abs/1709.04743.

Junk, C., L. Delle Monache, and S. Alessandrini, 2015: Analog-based ensemble model output statistics. *Mon. Wea. Rev.*, **143**, 2909–2917, https://doi.org/10.1175/MWR-D-15-0095.1.

Kahle, D., and H. Wickham, 2013: Ggmap: Spatial visualization with ggplot2. *R J.*, **5**, 144–161.

Kingma, D. P., and J. Ba, 2014: Adam: A method for stochastic optimization. arXiv.org, https://arxiv.org/abs/1412.6980.

Kou, C., H. K. Lee, and T. K. Ng, 2018: Distribution regression network. arXiv.org, https://arxiv.org/abs/1804.04775.

Krizhevsky, A., I. Sutskever, and G. E. Hinton, 2012: Imagenet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems 25*, F. Pereira et al., Eds., Curran Associates, 1097–1105.

Krüger, F., S. Lerch, T. L. Thorarinsdottir, and T. Gneiting, 2016: Probabilistic forecasting and comparative model assessment based on Markov chain Monte Carlo output. arXiv.org, https://arxiv.org/abs/1608.06802.

Lagerquist, R., A. McGovern, and T. Smith, 2017: Machine learning for real-time prediction of damaging straight-line convective wind. *Wea. Forecasting*, **32**, 2175–2193, https://doi.org/10.1175/WAF-D-17-0038.1.

LeCun, Y., Y. Bengio, and G. Hinton, 2015: Deep learning. *Nature*, **521**, 436–444, https://doi.org/10.1038/nature14539.

Lerch, S., and T. L. Thorarinsdottir, 2013: Comparison of non-homogeneous regression models for probabilistic wind speed forecasting. *Tellus*, **65A**, 21206, https://doi.org/10.3402/tellusa.v65i0.21206.

——, and S. Baran, 2017: Similarity-based semilocal estimation of post-processing models. *J. Roy. Stat. Soc.*, **66C**, 29–51, https://doi.org/10.1111/rssc.12153.

Liu, Y., E. Racah, J. Correa, A. Khosrowshahi, D. Lavers, K. Kunkel, M. Wehner, and W. Collins, 2016: Application of deep convolutional neural networks for detecting extreme weather in climate datasets. arXiv.org, https://arxiv.org/abs/1605.01156.

Matheson, J. E., and R. L. Winkler, 1976: Scoring rules for continuous probability distributions. *Manage. Sci.*, **22**, 1087–1096, https://doi.org/10.1287/mnsc.22.10.1087.

McGovern, A., K. L. Elmore, D. J. Gagne, S. E. Haupt, C. D. Karstens, R. Lagerquist, T. Smith, and J. K. Williams, 2017: Using artificial intelligence to improve real-time decision-making for high-impact weather. *Bull. Amer. Meteor. Soc.*, **98**, 2073–2090, https://doi.org/10.1175/BAMS-D-16-0123.1.

Meinshausen, N., 2006: Quantile regression forests. *J. Mach. Learn. Res.*, **7**, 983–999.

——, 2017: QuantregForest: Quantile regression forests. R Package version 1.3-7, https://CRAN.R-project.org/package=quantregForest.

Messner, J. W., G. J. Mayr, D. S. Wilks, and A. Zeileis, 2014: Extending extended logistic regression: Extended versus separate versus ordered versus censored. *Mon. Wea. Rev.*, **142**, 3003–3014, https://doi.org/10.1175/MWR-D-13-00355.1.

——, ——, and A. Zeileis, 2016: Heteroscedastic censored and truncated regression with crch. *R J.*, **8**, 173–181.

——, ——, and ——, 2017: Nonhomogeneous boosting for predictor selection in ensemble postprocessing. *Mon. Wea. Rev.*, **145**, 137–147, https://doi.org/10.1175/MWR-D-16-0088.1.

Nielsen, M. A., 2015: *Neural Networks and Deep Learning*. Determination Press, http://neuralnetworksanddeeplearning.com/.

Oliva, J., B. Póczos, and J. Schneider, 2013: Distribution to distribution regression. *Proc. 30th Int. Conf. on Machine Learning*, Atlanta, GA, Association for Computing Machinery, 1049–1057.

Python Software Foundation, 2017: Python software, version 3.6.4. Python Software Foundation, https://www.python.org/.

Raftery, A. E., T. Gneiting, F. Balabdaoui, and M. Polakowski, 2005: Using Bayesian model averaging to calibrate forecast ensembles. *Mon. Wea. Rev.*, **133**, 1155–1174, https://doi.org/10.1175/MWR2906.1.

Rasp, S., M. S. Pritchard, and P. Gentine, 2018: Deep learning to represent subgrid processes in climate models. *Proc. Natl. Acad. Sci. USA*, **115**, 9684–9689, https://doi.org/10.1073/pnas.1810286115.

R Core Team, 2017: R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria, https://www.R-project.org/.

Schefzik, R., T. L. Thorarinsdottir, and T. Gneiting, 2013: Uncertainty quantification in complex simulation models using ensemble copula coupling. *Stat. Sci.*, **28**, 616–640, https://doi.org/10.1214/13-STS443.

Scheuerer, M., 2014: Probabilistic quantitative precipitation forecasting using ensemble model output statistics. *Quart. J. Roy. Meteor. Soc.*, **140**, 1086–1096, https://doi.org/10.1002/qj.2183.

——, and T. M. Hamill, 2015: Statistical postprocessing of ensemble precipitation forecasts by fitting censored, shifted gamma distributions. *Mon. Wea. Rev.*, **143**, 4578–4596, https://doi.org/10.1175/MWR-D-15-0061.1.

——, and D. Möller, 2015: Probabilistic wind speed forecasting on a grid based on ensemble model output statistics. *Ann. Appl. Stat.*, **9**, 1328–1349, https://doi.org/10.1214/15-AOAS843.

Schlosser, L., T. Hothorn, R. Stauffer, and A. Zeileis, 2018: Distributional regression forests for probabilistic precipitation forecasting in complex terrain. arXiv.org, https://arxiv.org/abs/1804.02921.

Schmidhuber, J., 2015: Deep learning in neural networks: An overview. *Neural Networks*, **61**, 85–117, https://doi.org/10.1016/j.neunet.2014.09.003.

Taillardat, M., O. Mestre, M. Zamo, and P. Naveau, 2016: Calibrated ensemble forecasts using quantile regression forests and ensemble model output statistics. *Mon. Wea. Rev.*, **144**, 2375–2393, https://doi.org/10.1175/MWR-D-15-0260.1.

——, A.-L. Fougères, P. Naveau, and O. Mestre, 2017: Forest-based methods and ensemble model output statistics for rainfall ensemble forecasting. arXiv.org, https://arxiv.org/abs/1711.10937.

Taylor, J. W., 2000: A quantile regression neural network approach to estimating the conditional density of multiperiod returns. *J. Forecasting*, **19**, 299–311, https://doi.org/10.1002/1099-131X(200007)19:4<299::AID-FOR775>3.0.CO;2-V.

Thorarinsdottir, T. L., and T. Gneiting, 2010: Probabilistic forecasts of wind speed: Ensemble model output statistics by using heteroscedastic censored regression. *J. Roy. Stat. Soc.*, **173A**, 371–388, https://doi.org/10.1111/j.1467-985X.2009.00616.x.

Wang, F., Z. Mi, S. Su, and H. Zhao, 2012: Short-term solar irradiance forecasting model based on artificial neural network using statistical feature parameters. *Energies*, **5**, 1355–1370, https://doi.org/10.3390/en5051355.

Wilks, D. S., 2011: *Statistical Methods in the Atmospheric Sciences*. 3rd ed. Elsevier, 676 pp.

——, 2016: "The stippling shows statistically significant grid points": How research results are routinely overstated and overinterpreted, and what to do about it. *Bull. Amer. Meteor. Soc.*, **97**, 2263–2273, https://doi.org/10.1175/BAMS-D-15-00267.1.

Supplement for Neural networks for post-processing ensemble weather forecasts

Stephan Rasp[1] and Sebastian Lerch[2,3]

[1]Meteorological Institute, Ludwig-Maximilians-Universität, Munich
[2]Institute for Stochastics, Karlsruhe Institute of Technology
[3]Heidelberg Institute for Theoretical Studies

August 9, 2018

1 Calibration assessment

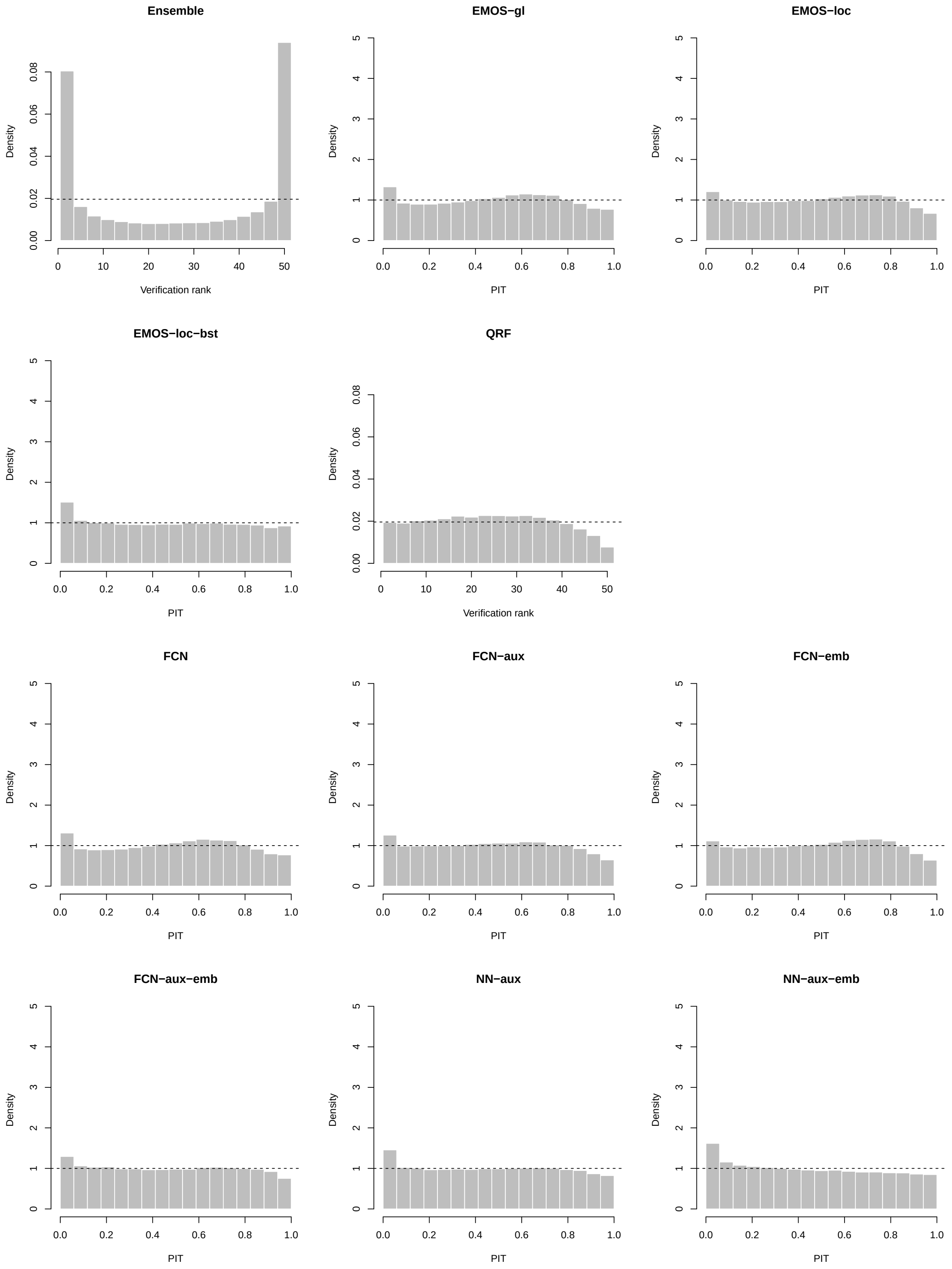


Figure 1: Verification rank and PIT histograms for raw and post-processed ensemble forecasts based on models estimated using data from 2015, aggregated over all forecast cases during the evaluation period in calendar year 2016.

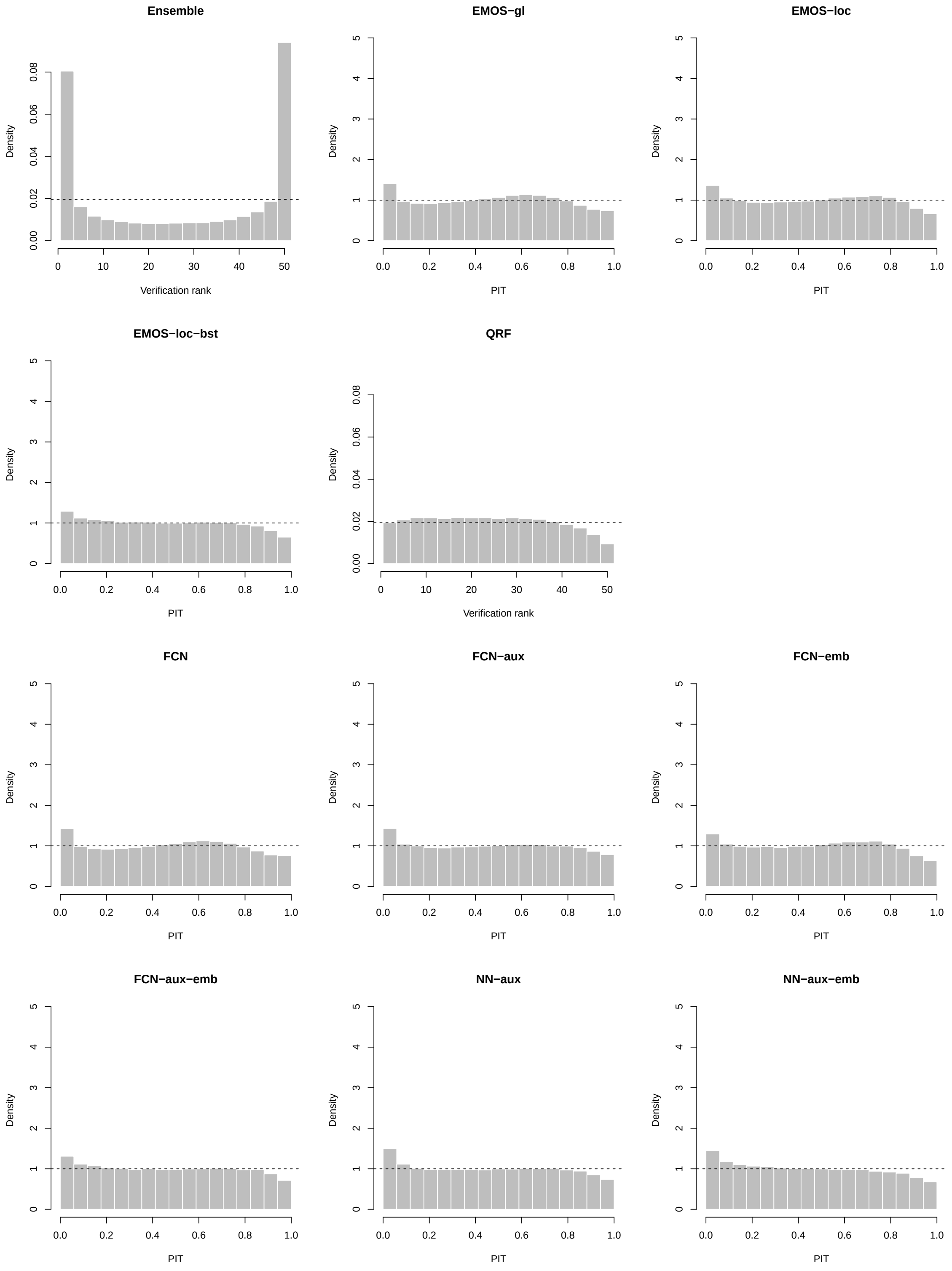
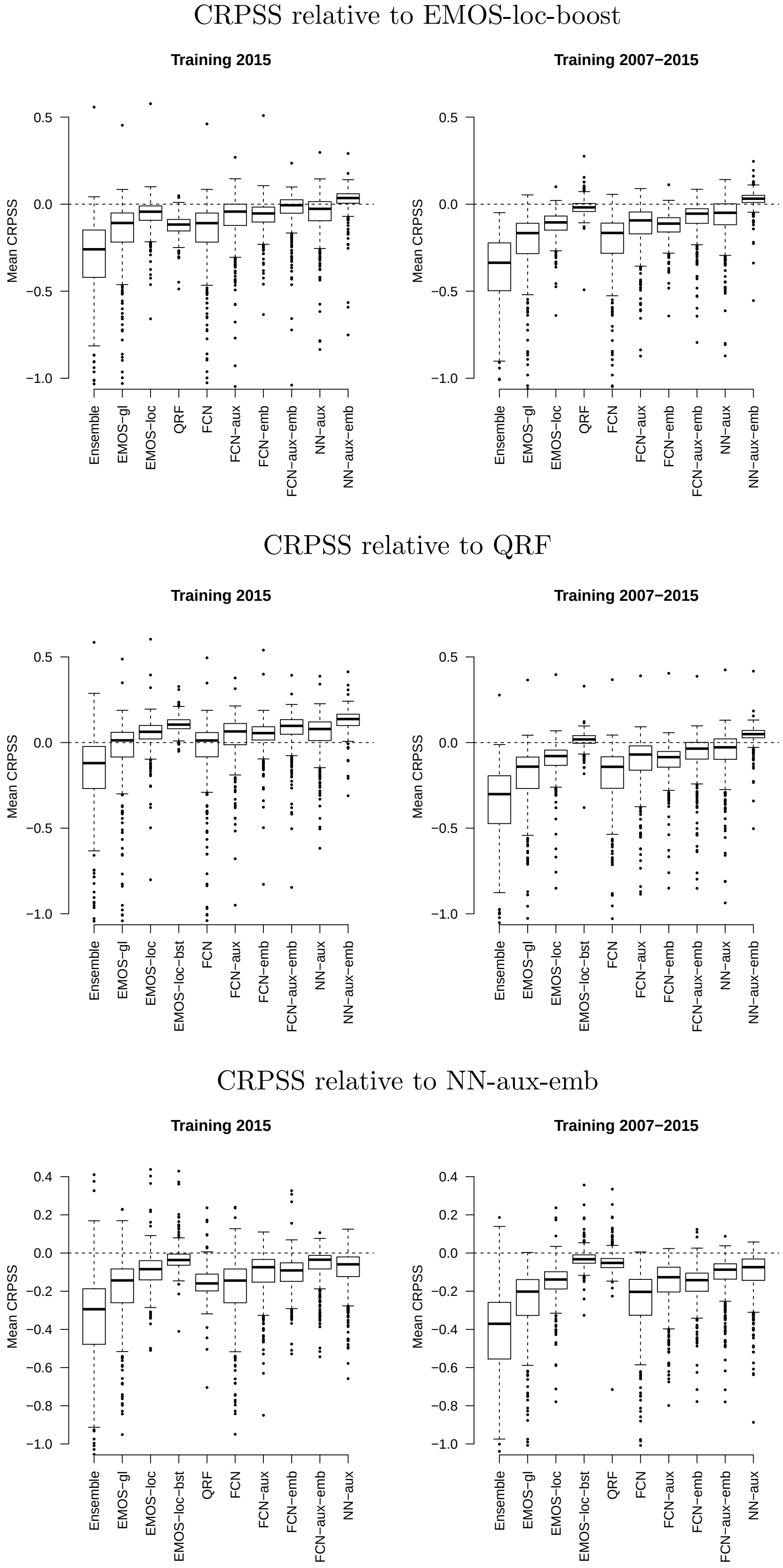


Figure 2: Verification rank and PIT histograms for raw and post-processed ensemble forecasts based on models estimated using data from 2007–2015, aggregated over all forecast cases during the evaluation period in calendar year 2016..

2 CRPSS results for alternative benchmark models



CRPSS relative to EMOS-loc-boost

CRPSS relative to QRF

CRPSS relative to NN-aux-emb

Figure 3: As Figure 3, but with different reference models.

3 Details on computational aspects

Table 1 shows computation times required for training the different post-processing models for both training sets. As noted before, the computation times are not directly comparable due to implementations in different programming languages and hardware environments. The computation times for the benchmark models, implemented in `R` using the `crch` (Messner et al., 2016), `quantregForest` (Meinshausen, 2017) and `scoringRules` (Jordan et al., 2018) packages, were obtained on a standard laptop computer, whereas the network models were implemented with the `Python` libraries Keras (Chollet et al., 2015) and TensorFlow (Abadi et al., 2016), and run on a single GPU (Nvidia Tesla K20). Computation times on a regular CPU are roughly 6 times longer for the most complex networks. For the simple networks the difference is negligible. Note that the inference time, i.e., the time to make a prediction after the model has been trained, is on the order of a few seconds for all models. Further, note that all computation times reported here are substantially lower compared to the computational costs of generating the raw ensemble forecast.

Tables 2 and 3 list hyperparameters for the benchmark and network models.

Table 1: Computation times (in minutes) for estimating post-processing models with the two training sets and computing out-of-sample forecasts for the evaluation period.

Model	Computation time (min) with training data from	
	2015	2007–2015
Benchmark models		
EMOS-gl	< 1	< 1
EMOS-loc	< 1	1
EMOS-loc-bst	14	48
QRF	8	430
Network models		
FCN	< 1	1
FCN-aux	< 1	2
FCN-emb	< 1	3
FCN-aux-emb	< 1	3
NN-aux	4	25
NN-aux-emb	9	16

Table 2: Hyperparameters for benchmark models. AIC denotes the Akaike information criterion.

Model	Parameter	Value
EMOS-gl	none	
EMOS-loc	none	
EMOS-loc-bst	maximum number of iterations	1 000
	step size	0.05
	stopping criterion for boosting algorithm	AIC
QRF	number of trees	1 000
	minimum size of terminal leaves	10
	number of variables randomly sampled as candidates at each split	25

Table 3: Hyperparameters for network models. Values in parentheses indicate settings for the longer training period from 2007–2015. Parameters refers to all learnable values: weights, biases and latent embedding features. An epoch refers to one pass through all training samples. Batch size refers to the number of random training samples considered per gradient update in the SGD optimization.

Model	Number of parameters	Epochs	Learning rate	Batch size	Hidden nodes	Embedding size
FCN	6	30 (15)	0.1 (0.1)	4 096 (4 096)		
FCN-aux	82	30 (10)	0.02 (0.02)	1024 (1 024)		
FCN-emb	1 084	30 (10)	0.02 (0.02)	1 024 (1 024)		2 (2)
FCN-aux-emb	1 160	30 (10)	0.02 (0.02)	1 024 (1 024)		2 (2)
NN-aux	3 326	(10)	(0.02)	(1 024)	(64)	(2)
NN-aux-emb	24 116	30 (10)	0.01 (0.002)	1 024 (4 096)	50 (512)	2 (2)

4 Statistical significance of score differences

Pair-wise one-sided Diebold-Mariano tests are applied to all possible comparisons of forecast models at each of the 499 stations individually. To account for multiple hypothesis testing and spatial correlations of score differences, we apply a Benjamini-Hochberg procedure to the corresponding p-values when aggregating the results by determining the ratio of stations with significant score differences, see Appendix **??** for details.

Table 4 summarizes pair-wise Diebold-Mariano tests by showing the ratio of stations with statistically significant CRPS differences after applying a Benjamini-Hochberg procedure for a nominal level of $\alpha = 0.05$. Generally, the results indicate large numbers of stations with significant differences of the network models when compared to standard EMOS approaches. NN-aux-emb shows the highest ratios of significant score differences over any competitor, and is significantly outperformed at very few station and only by the best-performing alternatives.

Table 4: Ratio of stations (in %) where pair-wise Diebold-Mariano tests indicate statistically significant CRPS differences after applying a Benjamini-Hochberg procedure to account for multiple testing for a nominal level of $\alpha = 0.05$ of the corresponding one-sided tests. The (i, j)-entry in the i-th row and j-th column indicates the ratio of stations where the null hypothesis of equal predictive performance of the corresponding one-sided Diebold-Mariano test is rejected in favor of the model in the i-th row when compared to the model in the j-th column. The remainder of the sum of (i, j)- and (j, i)-entry to 100% is the ratio of stations where the score differences are not significant.

Training with 2015 data

	Ens.	EMOS -gl	EMOS -loc	EMOS -loc-bst	QRF	FCN	FCN -aux	FCN -emb	FCN -aux-emb	NN -aux	NN -aux-emb
Ens.		0.6	0.0	0.0	0.6	0.6	0.8	0.0	0.8	0.8	0.6
EMOS-gl	83.2		0.2	0.0	10.4	10.2	3.0	0.2	0.6	2.0	0.2
EMOS-loc	96.2	71.3		0.0	50.5	71.9	17.4	24.8	5.2	9.6	1.4
EMOS-loc-bst	93.8	72.7	40.5		89.8	74.3	41.7	49.1	21.0	30.5	2.0
QRF	54.7	22.0	3.6	0.0		22.4	8.0	3.6	3.4	5.2	0.2
FCN	83.0	7.4	0.2	0.0	10.4		3.0	0.2	0.6	2.0	0.2
FCN-aux	83.2	60.3	17.2	1.8	47.5	62.3		19.0	1.0	0.4	0.2
FCN-emb	89.4	67.1	1.0	0.0	44.1	68.1	11.4		0.8	6.4	0.6
FCN-aux-emb	86.6	78.8	53.1	7.6	69.1	79.6	55.1	58.5		27.1	0.2
NN-aux	87.2	69.5	25.9	2.0	57.5	70.7	22.8	30.9	8.0		0.4
NN-aux-emb	93.6	89.4	67.1	30.3	92.2	90.2	67.3	72.7	43.5	64.9	

Training with 2007-2015 data

	Ens.	EMOS -gl	EMOS -loc	EMOS -loc-bst	QRF	FCN	FCN -aux	FCN -emb	FCN -aux-emb	NN -aux	NN -aux-emb
Ens.		0.6	0.0	0.0	0.0	0.6	0.8	0.0	0.8	0.8	0.0
EMOS-gl	86.8		0.2	0.0	0.2	2.6	3.0	0.2	0.6	0.2	0.0
EMOS-loc	98.8	72.7		0.0	0.2	71.7	17.2	17.4	3.6	6.8	0.6
EMOS-loc-bst	99.4	98.0	91.4		21.0	97.8	82.0	94.2	70.3	49.7	1.4
QRF	98.6	94.2	79.2	1.4		94.2	57.7	84.4	38.1	33.5	1.2
FCN	87.8	11.0	0.2	0.0	0.2		3.2	0.2	0.6	0.2	0.0
FCN-aux	87.6	65.5	24.2	0.0	0.4	65.5		26.7	0.8	1.4	0.0
FCN-emb	93.4	71.3	0.0	0.0	0.2	70.5	12.0		1.2	4.6	0.0
FCN-aux-emb	91.2	82.8	60.3	0.0	0.6	81.8	58.1	64.1		16.4	0.0
NN-aux	95.6	84.8	54.5	1.4	9.8	84.8	72.9	58.5	34.5		0.0
NN-aux-emb	98.8	97.8	95.2	29.9	52.9	97.6	92.0	96.0	91.0	74.5	

References

Abadi, M., and Coauthors, 2016: Tensorflow: A system for large-scale machine learning. *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)*, 265–283.

Chollet, F., and Coauthors, 2015: Keras. https://keras.io.

Jordan, A., F. Krüger, and S. Lerch, 2018: Evaluating probabilistic forecasts with scoringRules. *Journal of Statistical Software*, in press, preprint available at https://arxiv.org/abs/1709.04743.

Meinshausen, N., 2017: *quantregForest: Quantile Regression Forests*. URL https://CRAN.R-project.org/package=quantregForest, r package version 1.3-7.

Messner, J. W., G. J. Mayr, and A. Zeileis, 2016: Heteroscedastic censored and truncated regression with crch. *The R Journal*, **8**, 173–181.

3 Conclusion

The papers in this thesis explore novel techniques to improve weather and climate prediction. In the context of the forecast chain, P1–3 focus specifically on the representation of subgrid processes while P4 deals with the statistical calibration of numerical weather forecasts. In the paragraphs below, I will briefly summarize the key results of each paper, put it into the wider context of current research and present an outlook for ongoing work. I will conclude with a global picture of data-driven model development.

P1: Stochastic representation of convection

Summary P1 deals with two gray areas in representing convection. The first comes when grid spacings in models with parameterized convection become too small to host a large number of individual clouds. This happens roughly below 100 km. The second gray area is located around the km-scale, when convection is treated explicitly but the triggering processes, primarily boundary layer turbulence, cannot yet be resolved. As a response to the second issue *Kober and Craig* (2016) proposed introducing stochastic perturbations to temperature, humidity and vertical wind in the boundary layer, in order to re-introduce the missing variability. In their first tests, they showed an improved coupling between the subgrid turbulence and the resolved convection. In P1, the PSP scheme was used as a tool to create different convective realizations in the same large scale flow. This technique enabled, for the first time, to test the assumptions of the *Craig and Cohen* (2006) theory in non-idealized settings.

The results in P1 show the general applicability of the *Craig and Cohen* (2006) theory even in situations that is was not originally designed for. This is important because the theory increasingly finds application in global modeling, where it has to deal with a wide range of convective regimes. Systematic errors do exist, however, which are primarily related to the organization of clouds.

Context The importance of cloud-cloud interactions has also been found by a number of other recent studies. *Moseley et al.* (2016) found that the clustering of clouds is crucial to produce precipitation extremes, and that this clustering responds particularly strongly to climate change. One of the key mechanisms leading to cloud organization are cold pools. They can trigger new convection in two ways: due to mechanical lifting (*Rotunno et al.*, 1988) and due to thermodynamically decreasing CIN and increasing CAPE (*Tompkins*, 2001). There is still debate in the literature which effect dominates when but the importance of cold pools on cloud development is clear (*Schlemmer and Hohenegger*, 2014). First attempts to incorporate cold pool effects in convection parameterizations have been made (*Grandpeix and Lafore*, 2010) but most researchers are still searching for good ways of describing subgrid cold pools.

Outlook Given the strong theoretical argument for physically based stochastic parameterizations, one might be surprised to see them used relatively rarely in operational weather and climate models. One issue is the "curse of model tuning". Operational models have been highly tuned over many years. Conceptually, tuning means finding the best model score in the space spanned by all the tuning parameters. Introducing a new parameterization, even one that is physically more consistent, changes the entire surface of the score, so that most likely the previous tuning choices will result in a worse performance. A re-evaluation of the tuning in all dimensions would be necessary to find a potential new global minimum. Because tuning is done manually and is computationally expensive this exercise is unfeasible for most academic researchers.

A related problem is defining how good a model even is. How does one, for instance, weigh an improvement in a precipitation score against a increase in the temperature error? This problem is particularly hard for climate models, for which there are a large number of potential statistics to look at. For a new parameterization to be implemented in a new model version, usually operational centers require that only very few of their metrics become worse with a significant improvement in others. Typically, this is only feasible for incremental model changes rather than complete redesigns, such as stochastic parameterizations.

While the issues above are mostly related to common practices in atmospheric modeling, a more fundamental issue may be that most current physically-based stochastic parameterization are based on a single sub-component of the subgrid system. The *Craig and Cohen* (2006) theory, for example, presents a nice theory for the fluctuation of an ensemble of convective clouds but does not include the triggering of convection or the horizontal organization and propagation of clouds. Designing a "holistic" stochastic parameterization is a task that might just exceed our current understanding.

Despite these problems, there is an increasing push towards stochastic parameterization development (*Berner et al.*, 2017). It is, therefore, likely that atmospheric model will gradually incorporate more and more stochastic elements, particularly as evidence of their advantages will mount. Naturally, stochasticity is a topic also for data-driven parameterizations which we will come to now.

P2–3: Deep learning to represent subgrid processes

Summary In P2 and, subsequently, P3 we examined a completely different approach to subgrid parameterization: using algorithms to learn efficient representations of subgrid physics from high-resolution data. In our work, we used a super-parameterized global model as our reference model. We then trained a neural network to replace the super-parameterization and the radiation scheme. In P2, we showed that a neural network can indeed capture the complex physics. In P3, we took the essential step of re-implementing the trained neural network into the climate model and running it prognostically. The results show that most of the features of the reference simulation are reproduced at significantly reduced computational cost.

Context Over the last couple of years (2017–2018), machine learning parameterizations have been proposed by several research groups, mostly independently, with remarkably similar approaches. In particular, *Brenowitz and Bretherton* (2018) and *O'Gorman and Dwyer* (2018) deserve a comparison to P2–3. All studies aimed at predicting the subgrid tendencies using a machine learning algorithm. Surprisingly or not, all studies used an aquaplanet setup and ignored condensed water. *Brenowitz and Bretherton* (2018) ran a near global convection-permitting simulation that they coarse-grained to extract the subgrid tendencies, unified for turbulence, convection and radiation, as in P3. They also used a neural network to do the prediction. However, they did not succeed in producing a stable prognostic simulation. Speculating on the reasons is difficult but the most probable culprit is their shallow network architecture. In the Supplement of P3, we tested several network architectures and found that three or more layers are necessary to achieve a stable, realistic forward integration. Other reasons could be their long time step (3 h). To combat their stability problems, they came up with a clever way or averaging the loss function over several time steps. Yet, despite this, they were only able to run a single column model with their neural network parameterization.

O'Gorman and Dwyer (2018) aimed to predict the tendencies of a traditional parameterization with a random forest. Random forests are a machine learning technique that is based on an ensemble of binary decision trees (*Breiman*, 2001). *O'Gorman and Dwyer* (2018) managed to produce a stable prognostic simulation with good agreement to the reference model. In this way, their results are similar to ours. In comparison to neural networks, which learn a function to map an input to an output, random forests make predictions that are linear combinations of the training targets. The advantage of the random forest approach is that it ensures physical consistency, for example energy conservation, and stability, i.e. the predictions cannot blow up. On the other hand, neural networks are able to capture more complexity and, perhaps most important for actual application, they are able to cope with very large training data amounts and are exceedingly fast when implemented in a climate model. Random forests require all data to fit into memory during training and saving the full tree structure can be memory-consuming. With their ups and downs, both techniques should probably be developed and compared further.

Finally, *Schneider et al.* (2017a) initiated yet another approach at building a data-driven parameterization. In contrast to the approaches above, however, they aim to take existing parameterization frameworks and learn the free parameters in them. This has several conceptual advantages over a "full" machine learning parameterization. First, by hard-coding the known equations one can ensure physical constraints and stability. Another advantage is the interpretability of the resulting parameterization. Yet, this also requires picking a suitable framework, which might lack features encountered in nature.

Outlook There are three key challenges for machine learning parameterizations:

1. **Stability**: Above all, a machine learning parameterization has to be stable. So far, this has only been achieved by simplifying the problem (idealized planet and omission of "secondary" variables like cloud water).
2. **Physical constraints**: Energy and mass have to be conserved. In P3, we have shown that the deep neural network has learned to approximately conserve energy, but the conservation should really be exact. Additionally, positivity has to be ensured for concentrations.
3. **Generalizability**: How far outside of their training regime are machine learning parameterization applicable? P3 suggests that after a certain point, they are not reliable. This is an issue for climate change simulations where new extremes are reached in many dimensions of the phase space.

The task for future development of machine learning parameterization thus must be to create the most accurate and efficient model possible given the conditions above are fulfilled.

Stability is a difficult problem because the neural network's predictions feed back, modified by the dynamical core of the model, to its input in the next time step. This allows feedback loops to develop which can cause the model to gradually blow up. Similar problems have been encountered in machine learning research when making time series prediction. The solution here is to use advanced recurrent neural networks where the gradient is computed backwards over several time steps. Unfortunately, this only works if the system is fully differentiable and implemented in a deep learning framework. In the case of atmospheric modeling, the dynamical core is typically mathematically complex and written in Fortran while the deep learning component is done in Python. This makes a recurrent approach difficult to achieve technically. A potentially simpler solution would be to train the neural network online, i.e. the gradient descent update is computed after every time step by running a neural network parameterization alongside the high-resolution truth. For such a method, the super-parameterization approach is a perfect fit.

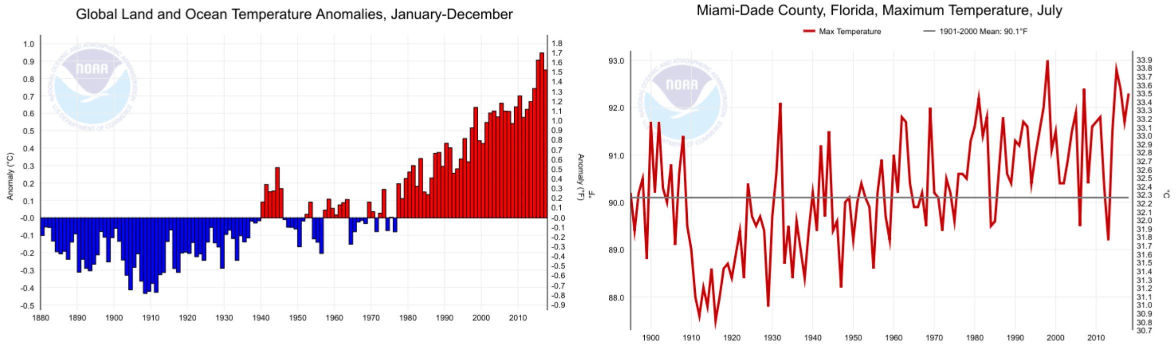


Fig. 3.1: (left) Annual mean global temperatures. A sharp increase can be seen for the last 50 years or so. (right) The maximum temperature in July for Miami-Dade County in Florida. On smaller spatial and time scales, the variability is much larger. [1]

The problem of obeying physical constraints could be easier to solve. Two solutions jump to mind: first, one could try to reformulate the problem, so that conservation laws would be guaranteed. For example, instead of predicting absolute tendencies of temperature and humidity one could predict the vertical subgrid fluxes, e.g. $\overline{w'\phi'}$ and then predict $\frac{\partial\bar{\phi}}{\partial t} = -w'\frac{\partial\overline{w'\phi'}}{\partial z}$ using a standard finite differencing scheme (*Bar-Sinai et al.*, 2018). This would ensure the conservation of the tracer ϕ. Similarly, it could be possible to predict the conversion rates between the microphysical quantities to appropriately link temperature and humidity tendencies and enforce positive tracer concentrations. The second solution would be to add a loss term that penalizes the neural network for violating physical constraints during training. The results in P3 suggest that the neural network already learned an approximate version of energy conservation. With a physical loss function this approximation could be made more exact and the residual be removed by postprocessing.

Another hard problem is generalization to unseen climate states. The neural network parameterization in P3 shows problems once the global mean temperatures are increase by more than 1 K. It has to be said, however, that these results may not be representative for the real climate. In the real atmosphere, there is a much larger year-to-year variability caused by effects such as ENSO (Fig. 3.1). While an increase in the global mean temperatures will certainly bring new, previously unexplored extremes, the effects are likely much less drastic than in out idealized experiments. Still, this is an important issue if data-driven parameterizations should ever be used in an operational setup that informs policy. If high-resolution simulations are used for training, a simple solution would be to explore a wide range of possible climate states during training. Furthermore, one could hope that making the parameterization more physical, as described in the previous paragraph, will also aid generalizability.

[1] NOAA National Centers for Environmental information, Climate at a Glance: National Time Series, published October 2018, retrieved on October 26, 2018 from `https://www.ncdc.noaa.gov/cag/`

Finally, neural network architectures could be tailored to make the algorithm more general.

Finally, what about a stochastic machine learning parameterization? While this is likely not a first order problem, given the fact that most operational parameterizations are deterministic, it would nevertheless be interesting to explore this route. In P2–3 we found that using a mean loss function on a chaotic problem suppresses variability. The natural conclusion from this is to use a different loss function. One particularly promising approach could be using generative adversarial networks (GANs; *Goodfellow et al.* (2014)). Here two network, called the generator and the discriminator, are pitted against each other. The generator tries to produce realistic samples, while the discriminator tries to distinguish the "fake" generator samples from real samples. The two networks are trained alternatively, leading to better and better predictions from the generator. In a way the discriminator can be seen as a learned loss function that judges the realism of the predictions. For our parameterization problem, a specific form of GANs, called Conditional GANs, could be applicable (*Mirza and Osindero*, 2014). Ideally then, the generator would predict one realistic subgrid realization rather than a mean of all possible states, as it does now. GANs are very finicky to train, however, and their behavior is often hard to predict. Nevertheless, this could be an exciting route to explore to produce a stochastic data-driven parameterization.

P4: Improving forecasts a posteriori with neural networks

Summary In P4 we tackled a sub-problem in statistical postprocessing, the calibration of probabilistic forecasts. Traditionally, this has been done with linear regression methods and, more recently, using random forests. We used a simple neural network architecture in combination with embeddings, a technique that originally comes from recommender systems (think Netflix or Amazon) and natural language processing. These embeddings allowed us to learn a set of parameters specific for each measurement station while still using a global postprocessing model. Our technique outperformed the previous state-of-the-art methods while being computationally more affordable.

Context Statistical postprocessing has always been framed as a supervised learning task and, therefore, presents an ideal target for modern machine learning techniques. Over the last years machine learning has been applied to aid decision making in numerical weather forecasting (*McGovern et al.*, 2017), mostly based on random forests. Furthermore, superresolution neural networks have been used for the task of statistical downscaling, i.e. statistically interpolating forecasts, e.g. of precipitation, from a coarse grid to a finer grid (*Rodrigues et al.*, 2018). Neural networks are also commonly used for more process-specific statistical forecasting tasks, such as predicting the power output of a solar power plant (*Yadav and Chandel*, 2014).

Outlook In traditional NWP and climate science, most postprocessing is still based on standard regression techniques. One problem for applying more advanced techniques is the limited amount of data. For postprocessing forecast-observation pairs are required for training. While forecasts can be generated at will, observations cannot. For daily forecasts at a single point in space even a 10 year dataset will only result in a few thousand samples, typically not enough to train a model with even intermediate complexity without overfitting. In P4, treating each station independently and using embeddings enabled us to "increase" our sample size but the amount of available data was still at the lower limit. One could think of many more advanced techniques, such as incorporating spatial structures, but the feasibility is always limited by the data amount. Nevertheless, there likely are still many low hanging fruit in the realm of postprocessing for the application of neural networks.

The big picture

These are exciting times in atmospheric science. Several initiatives are pressing for paradigm shifts in the way atmospheric modeling is done. The two key frontiers are the use of machine learning and the push for higher-resolution simulations. *Schneider et al.* (2017a) lament the slow progress in climate modeling over the last decades and call for a better incorporation of high-resolution modeling and observations in the parameterization design process. To achieve this, they are in the process of initiating a project to radically redesign climate modeling using data-driven methods (*Voosen*, 2018). In addition to learning from high-resolution modeling, they also aim to learn from observations, a much harder problem (*Schneider et al.*, 2017b) (Fig. 3.2). Observations are sparse in time and space and mostly indirect which means clever data assimilation methods have to be employed. For finding the initial conditions of a forecast, significant progress has been made over the last decades. For tuning the parameters inside the model, however, computationally feasible approaches still need to be found.

Another, even bigger initiative follows a more brute-force approach and aims to simply resolve all the difficult processes. A recently proposed EU flagship project [2] wants to build the infrastructure to run climate simulations at km or sub-km resolutions. This requires an increase in computational capabilities of several orders of magnitude. Even if such projects succeed in the near future, parameterizations will not go away. First, even cloud-resolving models still need to parameterize turbulence and microphysics. Second, running these costly simulations routinely, e.g. in academic research, or for paleo-climate simulation, will remain unfeasible for many decades. Yet multi-year cloud-resolving simulations could provide the perfect training data for data-driven parameterizations.

Atmospheric science is increasingly becoming a data science. In fact, one could view atmospheric modeling as one of the original big data sciences. Models are running on the latest

[2] `http://www.extremeearth.eu/`, accessed on 26 Oct 2018

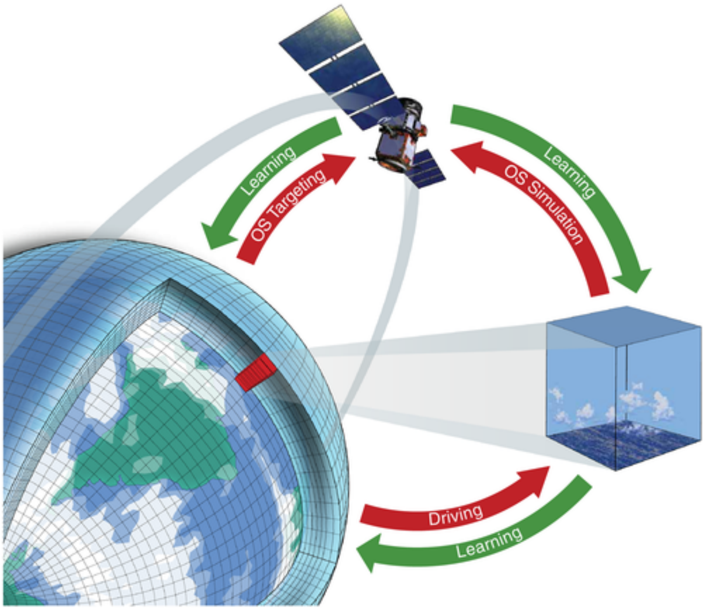


Fig. 3.2: Schematic representation of a data-driven climate model. The model itself has a coarse grid which requires the parameterization of subgrid processes such as clouds. These parameterizations are informed by targeted observations and high-resolution modeling. From *Schneider et al.* (2017b).

supercomputers and are producing astonishing amounts of data. Yet one could argue that the use and handling of the data is, at this stage, unsatisfactory. The manual iteration process of developing models seems less than ideal. That is not to understate the progress that has been made, particularly in weather forecasting. But the growing interest in data-driven methods suggests that more might be possible. In the end, it is unlikely that atmospheric models will be completely data-driven, not is is likely that things will remain as they are now. Rather, a hybrid approach with machine learning methods incorporated at many points in the forecast chain seems like the most likely and promising outcome.

Bibliography

Angermueller, C., T. Pärnamaa, L. Parts, and O. Stegle (2016), Deep learning for computational biology., *Molecular systems biology*, *12*(7), 878, doi:10.15252/MSB.20156651.

Arakawa, A., and W. H. Schubert (1974), Interaction of a Cumulus Cloud Ensemble with the Large-Scale Environment, Part I, *Journal of the Atmospheric Sciences*, *31*(3), 674–701, doi:10.1175/1520-0469(1974) 031<0674:IOACCE>2.0.CO;2.

Arnold, N. P., and D. A. Randall (2015), Global-scale convective aggregation: Implications for the Madden-Julian Oscillation, *Journal of Advances in Modeling Earth Systems*, *7*(4), 1499–1518, doi:10.1002/2015MS000498.

Baldauf, M., A. Seifert, J. Förstner, D. Majewski, M. Raschendorfer, and T. Reinhardt (2011), Operational Convective-Scale Numerical Weather Prediction with the COSMO Model: Description and Sensitivities, *Monthly Weather Review*, *139*(12), 3887–3905, doi:10.1175/MWR-D-10-05013.1.

Bar-Sinai, Y., S. Hoyer, J. Hickey, and M. P. Brenner (2018), Data-driven discretization: a method for systematic coarse graining of partial differential equations, *arXiv*, *1808.04930*.

Barthlott, C., and C. Hoose (2015), Spatial and temporal variability of clouds and precipitation over Germany: multiscale simulations across the "gray zone", *Atmospheric Chemistry and Physics*, *15*(21), 12,361–12,384, doi:10.5194/acp-15-12361-2015.

Bauer, P., A. Thorpe, and G. Brunet (2015), The quiet revolution of numerical weather prediction, *Nature*, *525*(7567), 47–55, doi:10.1038/nature14956.

Bechtold, P., N. Semane, P. Lopez, J.-P. Chaboureau, A. Beljaars, and N. Bormann (2014), Representing Equilibrium and Nonequilibrium Convection in Large-Scale Models, *Journal of the Atmospheric Sciences*, *71*(2), 734–753, doi:10.1175/JAS-D-13-0163.1.

Benedict, J. J., and D. A. Randall (2009), Structure of the Madden–Julian Oscillation in the Superparameterized CAM, *Journal of the Atmospheric Sciences*, *66*(11), 3277–3296, doi:10.1175/2009JAS3030.1.

Bengtsson, L., M. Steinheimer, P. Bechtold, and J.-F. Geleyn (2013), A stochastic parametrization for deep convection using cellular automata, *Quarterly Journal of the Royal Meteorological Society*, *139*(675), 1533–1543, doi:10.1002/qj.2108.

Berner, J., U. Achatz, L. Batté, L. Bengtsson, A. d. l. Cámara, H. M. Christensen, M. Colangeli, D. R. B. Coleman, D. Crommelin, S. I. Dolaptchiev, C. L. E. Franzke, P. Friederichs, P. Imkeller, H. Järvinen, S. Juricke, V. Kitsios, F. Lott, V. Lucarini, S. Mahajan, T. N. Palmer, C. Penland, M. Sakradzija, J.-S. von Storch, A. Weisheimer, M. Weniger, P. D. Williams, and J.-I. Yano (2017), Stochastic Parameterization: Toward a New View of Weather and Climate Models, *Bulletin of the American Meteorological Society*, *98*(3), 565–588, doi:10.1175/BAMS-D-15-00268.1.

Bony, S., and J. Dufresne (2005), Marine boundary layer clouds at the heart of tropical cloud feedback uncertainties in climate models, *Geophysical Research Letters*, *32*(20), L20,806, doi:10.1029/2005GL023851.

Bony, S., B. Stevens, D. M. W. Frierson, C. Jakob, M. Kageyama, R. Pincus, T. G. Shepherd, S. C. Sherwood, A. P. Siebesma, A. H. Sobel, M. Watanabe, and M. J. Webb (2015), Clouds, circulation and climate sensitivity, *Nature Geoscience*, *8*(4), 261–268, doi:10.1038/ngeo2398.

Breiman, L. (2001), Random Forests, *Machine Learning*, *45*, 5–32.

Brenowitz, N. D., and C. S. Bretherton (2018), Prognostic Validation of a Neural Network Unified Physics Parameterization, *Geophysical Research Letters*, *45*(12), 6289–6298, doi:10.1029/2018GL078510.

Bretherton, C. S., and P. K. Smolarkiewicz (1989), Gravity Waves, Compensating Subsidence and Detrainment around Cumulus Clouds, *Journal of the Atmospheric Sciences*, *46*(6), 740–759, doi:10.1175/1520-0469(1989)046<0740:GWCSAD>2.0.CO;2.

Buizza, R., M. Milleer, and T. N. Palmer (1999), Stochastic representation of model uncertainties in the ECMWF ensemble prediction system, *Quarterly Journal of the Royal Meteorological Society*, *125*(560), 2887–2908, doi:10.1002/qj.49712556006.

Carleo, G., and M. Troyer (2017), Solving the quantum many-body problem with artificial neural networks, *Science*, *355*(6325), 602–606, doi:10.1126/science.aag2302.

Charney, J. G., R. FjÖrtoft, and J. V. Neumann (1950), Numerical Integration of the Barotropic Vorticity Equation, *Tellus*, *2*(4), 237–254, doi:10.3402/tellusa.v2i4.8607.

Charney, J. G., A. Arakawa, D. J. Baker, B. Bolin, R. E. Dickinson, R. M. Goody, C. E. Leith, H. M. Stommel, and C. I. Wunsch (1979), *Carbon dioxide and climate: a scientific assessment*, National Academy of Sciences, Washington, DC.

Christensen, H. M., J. Berner, D. R. B. Coleman, T. N. Palmer, H. M. Christensen, J. Berner, D. R. B. Coleman, and T. N. Palmer (2017), Stochastic Parameterization and El Niño–Southern Oscillation, *Journal of Climate*, *30*(1), 17–38, doi:10.1175/JCLI-D-16-0122.1.

Cintineo, R. M., and D. J. Stensrud (2013), On the Predictability of Supercell Thunderstorm Evolution, *Journal of the Atmospheric Sciences*, *70*(7), 1993–2011, doi:10.1175/JAS-D-12-0166.1.

Courant, R., H. Lewy, and K. Friedrichs (1928), Über die partiellen Differenzengleichungen der mathematischen Physik, *Mathematische Annalen*, *100*, 32–74.

Couvreux, F., R. Roehrig, C. Rio, M.-P. Lefebvre, M. Caian, T. Komori, S. Derbyshire, F. Guichard, F. Favot, F. D'Andrea, P. Bechtold, and P. Gentine (2015), Representation of daytime moist convection over the semi-arid Tropics by parametrizations used in climate and meteorological models, *Quarterly Journal of the Royal Meteorological Society*, *141*(691), 2220–2236, doi:10.1002/qj.2517.

Craig, G. C., and B. G. Cohen (2006), Fluctuations in an Equilibrium Convective Ensemble. Part I: Theoretical Formulation, *Journal of the Atmospheric Sciences*, *63*(8), 1996–2004, doi:10.1175/JAS3709.1.

Craig, G. C., and A. Dörnbrack (2008), Entrainment in Cumulus Clouds: What Resolution is Cloud-Resolving?, *Journal of the Atmospheric Sciences*, *65*(12), 3978–3988, doi:10.1175/2008JAS2613.1.

Done, J. M., G. C. Craig, S. L. Gray, P. A. Clark, and M. E. B. Gray (2006), Mesoscale simulations of organized convection: Importance of convective equilibrium, *Quarterly Journal of the Royal Meteorological Society*, *132*(616), 737–756, doi:10.1256/qj.04.84.

Durran, D. R. (2010), *Numerical methods for fluid dynamics : with applications to geophysics*, 516 pp., Springer.

Durran, D. R., and M. Gingrich (2014), Atmospheric Predictability: Why Butterflies Are Not of Practical Importance, *Journal of the Atmospheric Sciences*, *71*, 2476–2488, doi:10.1175/JAS-D-14-0007.1.

Gentine, P., A. K. Betts, B. R. Lintner, K. L. Findell, C. C. van Heerwaarden, A. Tzella, and F. D'Andrea (2013), A Probabilistic Bulk Model of Coupled Mixed Layer and Convection. Part I: Clear-Sky Case, *Journal of the Atmospheric Sciences*, *70*(6), 1543–1556, doi:10.1175/JAS-D-12-0145.1.

Gneiting, T., A. E. Raftery, A. H. Westveld, and T. Goldman (2005), Calibrated Probabilistic Forecasting Using Ensemble Model Output Statistics and Minimum CRPS Estimation, *Monthly Weather Review*, *133*(5), 1098–1118, doi:10.1175/MWR2904.1.

Goh, G. B., N. O. Hodas, and A. Vishnu (2017), Deep learning for computational chemistry, *Journal of Computational Chemistry*, *38*(16), 1291–1307, doi:10.1002/jcc.24764.

Goodfellow, I., J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio (2014), Generative adversarial nets, in *Advances in neural information processing systems*, pp. 2672–2680.

Goodfellow, I., Y. Bengio, and A. Courville (2016), *Deep Learning*, MIT Press.

Grams, C. M., S. C. Jones, C. A. Davis, P. A. Harr, and M. Weissmann (2013), The impact of Typhoon Jangmi (2008) on the midlatitude flow. Part I: Upper-level ridgebuilding and modification of the jet, *Quarterly Journal of the Royal Meteorological Society*, *139*(677), 2148–2164, doi:10.1002/qj.2091.

Grandpeix, J.-Y., and J.-P. Lafore (2010), A Density Current Parameterization Coupled with Emanuel's Convection Scheme. Part I: The Models, *Journal of the Atmospheric Sciences*, *67*(4), 881–897, doi:10.1175/2009JAS3044.1.

Hartman, C. M., and J. Y. Harrington (2005), Radiative Impacts on the Growth of Drops within Simulated Marine Stratocumulus. Part I: Maximum Solar Heating, *Journal of the Atmospheric Sciences*, *62*(7), 2323–2338, doi:10.1175/JAS3477.1.

Heinze, R., A. Dipankar, C. Carbajal Henken, C. Moseley, O. Sourdeval, S. Trömel, X. Xie, P. Adamidis, F. Ament, H. Baars, C. Barthlott, A. Behrendt, U. Blahak, S. Bley, S. Brdar, M. Brueck, S. Crewell, H. Deneke, P. Di Girolamo, R. Evaristo, J. Fischer, C. Frank, P. Friederichs, T. Göcke, K. Gorges, L. Hande, M. Hanke, A. Hansen, H.-C. Hege, C. Hoose, T. Jahns, N. Kalthoff, D. Klocke, S. Kneifel, P. Knippertz, A. Kuhn, T. van Laar, A. Macke, V. Maurer, B. Mayer, C. I. Meyer, S. K. Muppa, R. A. J. Neggers, E. Orlandi, F. Pantillon, B. Pospichal, N. Röber, L. Scheck, A. Seifert, P. Seifert, F. Senf, P. Siligam, C. Simmer, S. Steinke, B. Stevens, K. Wapler, M. Weniger, V. Wulfmeyer, G. Zängl, D. Zhang, and J. Quaas (2016), Large-eddy simulations over Germany using ICON: A comprehensive evaluation, *Quarterly Journal of the Royal Meteorological Society*, doi:10.1002/qj.2947.

Hemri, S., M. Scheuerer, F. Pappenberger, K. Bogner, and T. Haiden (2014), Trends in the predictive performance of raw ensemble weather forecasts, *Geophysical Research Letters*, *41*(24), 9197–9205, doi: 10.1002/2014GL062472.

Hohenegger, C., L. Schlemmer, and L. Silvers (2015), Coupling of convection and circulation at various resolutions, *Tellus A: Dynamic Meteorology and Oceanography*, *67*(1), 26,678, doi:10.3402/tellusa.v67.26678.

Holton, J. R. (1973), *An Introduction to Dynamic Meteorology*, vol. 41, 752 pp., doi:10.1119/1.1987371.

Hourdin, F., T. Mauritsen, A. Gettelman, J.-C. Golaz, V. Balaji, Q. Duan, D. Folini, D. Ji, D. Klocke, Y. Qian, F. Rauser, C. Rio, L. Tomassini, M. Watanabe, D. Williamson, F. Hourdin, T. Mauritsen, A. Gettelman, J.-C. Golaz, V. Balaji, Q. Duan, D. Folini, D. Ji, D. Klocke, Y. Qian, F. Rauser, C. Rio, L. Tomassini, M. Watanabe, and D. Williamson (2017), The Art and Science of Climate Model Tuning, *Bulletin of the American Meteorological Society*, *98*(3), 589–602, doi:10.1175/BAMS-D-15-00135.1.

Houze, R. A. (2004), Mesoscale convective systems, *Reviews of Geophysics*, *42*(4), RG4003, doi:10.1029/2004RG000150.

Iten, R., T. Metger, H. Wilming, L. del Rio, and R. Renner (2018), Discovering physical concepts with neural networks, *arXiv*, *1807.10300.*

Jakub, F., and B. Mayer (2017), The role of 1-D and 3-D radiative heating in the organization of shallow cumulus convection and the formation of cloud streets, *Atmospheric Chemistry and Physics*, *17*(21), 13,317–13,327, doi:10.5194/acp-17-13317-2017.

Jones, T. R., and D. A. Randall (2011), Quantifying the limits of convective parameterizations, *Journal of Geophysical Research*, *116*(D8), D08,210, doi:10.1029/2010JD014913.

Kalnay, E. (2003), *Atmospheric modeling, data assimilation, and predictability*, vol. 54, 341 pp.

Khairoutdinov, M., D. Randall, C. DeMott, M. Khairoutdinov, D. Randall, and C. DeMott (2005), Simulations of the Atmospheric General Circulation Using a Cloud-Resolving Model as a Superparameterization of Physical Processes, *Journal of the Atmospheric Sciences*, *62*(7), 2136–2154, doi:10.1175/JAS3453.1.

Khouider, B., J. Biello, A. J. Majda, and others (2010), A stochastic multicloud model for tropical convection, *Communications in Mathematical Sciences*, *8*(1), 187–216.

Kim, B., V. C. Azevedo, N. Thuerey, T. Kim, M. Gross, and B. Solenthaler (2018), Deep Fluids: A Generative Network for Parameterized Fluid Simulations, *arXiv*, *1806.02071*.

Kingma, D. P., and J. Ba (2014), Adam: A Method for Stochastic Optimization, *arXiv*, *1412.6980*.

Kober, K., and G. C. Craig (2016), Physically Based Stochastic Perturbations (PSP) in the Boundary Layer to Represent Uncertainty in Convective Initiation, *Journal of the Atmospheric Sciences*, *73*(7), 2893–2911, doi:10.1175/JAS-D-15-0144.1.

Kober, K., A. M. Foerster, and G. C. Craig (2015), Examination of a stochastic and deterministic convection parameterization in the COSMO model, *Monthly Weather Review*, p. 150701152623009, doi:10.1175/MWR-D-15-0012.1.

Kooperman, G. J., M. S. Pritchard, T. A. O'Brien, and B. W. Timmermans (2018), Rainfall From Resolved Rather Than Parameterized Processes Better Represents the Present-Day and Climate Change Response of Moderate Rates in the Community Atmosphere Model, *Journal of Advances in Modeling Earth Systems*, *10*(4), 971–988, doi:10.1002/2017MS001188.

LeCun, Y., Y. Bengio, and G. Hinton (2015), Deep learning, *Nature*, *521*(7553), 436–444, doi:10.1038/nature14539.

Leutbecher, M., and T. Palmer (2008), Ensemble forecasting, *Journal of Computational Physics*, *227*(7), 3515–3539, doi:10.1016/j.jcp.2007.02.014.

Leutwyler, D., D. Lüthi, N. Ban, O. Fuhrer, and C. Schär (2017), Evaluation of the convection-resolving climate modeling approach on continental scales, *Journal of Geophysical Research: Atmospheres*, *122*(10), 5237–5258, doi:10.1002/2016JD026013.

Lewis, J. M. (2005), Roots of Ensemble Forecasting, *Monthly Weather Review*, *133*(7), 1865–1885, doi:10.1175/MWR2949.1.

Lin, Y.-L. (2007), *Mesoscale Dynamics*, Cambridge University Press, Cambridge, doi:10.1017/CBO9780511619649.

Lorenz, E. N. (1963), Deterministic Nonperiodic Flow, *Journal of the Atmospheric Sciences*, *20*, 130–141, doi:10.1175/1520-0469(1963)020<0130:DNF>2.0.CO;2.

Lorenz, E. N. (1969), The predictability of a flow which possesses many scales of motion, *Tellus*, *21*(3), 289–307.

Madden, R. A., and P. R. Julian (1971), Detection of a 40–50 Day Oscillation in the Zonal Wind in the Tropical Pacific, *Journal of the Atmospheric Sciences*, *28*(5), 702–708, doi:10.1175/1520-0469(1971)028<0702:DOADOI>2.0.CO;2.

Manabe, S., J. Smagorinsky, and R. F. Strickler (1965), Simulated climatology of a general circulation model with a hydrological cycle, *Monthly Weather Review*, *93*(12), 769–798, doi:10.1175/1520-0493(1965) 093<0769:SCOAGC>2.3.CO;2.

Mauritsen, T., B. Stevens, E. Roeckner, T. Crueger, M. Esch, M. Giorgetta, H. Haak, J. Jungclaus, D. Klocke, D. Matei, U. Mikolajewicz, D. Notz, R. Pincus, H. Schmidt, and L. Tomassini (2012), Tuning the climate of a global model, *Journal of Advances in Modeling Earth Systems*, *4*(3), n/a–n/a, doi:10.1029/2012MS000154.

McGovern, A., K. L. Elmore, D. J. Gagne, S. E. Haupt, C. D. Karstens, R. Lagerquist, T. Smith, J. K. Williams, A. McGovern, K. L. Elmore, D. J. G. II, S. E. Haupt, C. D. Karstens, R. Lagerquist, T. Smith, and J. K. Williams (2017), Using Artificial Intelligence to Improve Real-Time Decision-Making for High-Impact Weather, *Bulletin of the American Meteorological Society*, *98*(10), 2073–2090, doi:10.1175/BAMS-D-16-0123.1.

Mellor, G. L., and T. Yamada (1974), A Hierarchy of Turbulence Closure Models for Planetary Boundary Layers, *Journal of the Atmospheric Sciences*, *31*(7), 1791–1806, doi:10.1175/1520-0469(1974)031<1791:AHOTCM>2.0.CO;2.

Melnikov, A. A., H. Poulsen Nautrup, M. Krenn, V. Dunjko, M. Tiersch, A. Zeilinger, and H. J. Briegel (2018), Active learning machine learns to create new quantum experiments., *Proceedings of the National Academy of Sciences of the United States of America*, *115*(6), 1221–1226, doi:10.1073/pnas.1714936115.

Mirza, M., and S. Osindero (2014), Conditional Generative Adversarial Nets, *arXiv*, *1411.1784*.

Moseley, C., C. Hohenegger, P. Berg, and J. O. Haerter (2016), Intensification of convective extremes driven by cloud–cloud interaction, *Nature Geoscience*, *9*(10), 748–752, doi:10.1038/ngeo2789.

Muller, C., and S. Bony (2015), What favors convective aggregation and why?, *Geophysical Research Letters*, *42*(13), 5626–5634, doi:10.1002/2015GL064260.

Neggers, R. A. J., M. Köhler, A. C. M. Beljaars, R. A. J. Neggers, M. Köhler, and A. C. M. Beljaars (2009), A Dual Mass Flux Framework for Boundary Layer Convection. Part I: Transport, *Journal of the Atmospheric Sciences*, *66*(6), 1465–1487, doi:10.1175/2008JAS2635.1.

Nielsen, M. A. (2015), *Neural Networks and Deep Learning*, Determination Press.

O'Gorman, P. A., and J. G. Dwyer (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*(10), 2548–2563, doi:10.1029/2018MS001351.

Oueslati, B., and G. Bellon (2015), The double ITCZ bias in CMIP5 models: interaction between SST, large-scale circulation and precipitation, *Climate Dynamics*, *44*(3-4), 585–607, doi:10.1007/s00382-015-2468-6.

Palmer, T. N. (2000), Predicting uncertainty in forecasts of weather and climate, *Reports on Progress in Physics*, *63*(2), 71–116, doi:10.1088/0034-4885/63/2/201.

Pathak, J., B. Hunt, M. Girvan, Z. Lu, and E. Ott (2018), Model-Free Prediction of Large Spatiotemporally Chaotic Systems from Data: A Reservoir Computing Approach, *Physical Review Letters*, *120*(2), 24,102, doi:10.1103/PhysRevLett.120.024102.

Peters, K., T. Crueger, C. Jakob, and B. Möbis (2017), Improved MJO-simulation in ECHAM6.3 by coupling a Stochastic Multicloud Model to the convection scheme, *Journal of Advances in Modeling Earth Systems*, *9*(1), 193–219, doi:10.1002/2016MS000809.

Plant, R. S., and G. C. Craig (2008), A Stochastic Parameterization for Deep Convection Based on Equilibrium Statistics, *Journal of the Atmospheric Sciences*, *65*(1), 87–105, doi:10.1175/2007JAS2263.1.

Radovic, A., M. Williams, D. Rousseau, M. Kagan, D. Bonacorsi, A. Himmel, A. Aurisano, K. Terao, and T. Wongjirad (2018), Machine learning at the energy and intensity frontiers of particle physics, *Nature*, *560*(7716), 41–48, doi:10.1038/s41586-018-0361-2.

Raissi, M. (2018), Deep Hidden Physics Models: Deep Learning of Nonlinear Partial Differential Equations.

Randall, D., M. Khairoutdinov, A. Arakawa, W. Grabowski, D. Randall, M. Khairoutdinov, A. Arakawa, and W. Grabowski (2003), Breaking the Cloud Parameterization Deadlock, *Bulletin of the American Meteorological Society*, *84*(11), 1547–1564, doi:10.1175/BAMS-84-11-1547.

Riemer, M., and S. C. Jones (2014), Interaction of a tropical cyclone with a high-amplitude, midlatitude wave pattern: Waviness analysis, trough deformation and track bifurcation, *Quarterly Journal of the Royal Meteorological Society Q. J. R. Meteorol. Soc*, *140*, 1362–1376, doi:10.1002/qj.2221.

Rodrigues, E. R., I. Oliveira, R. L. F. Cunha, and M. A. S. Netto (2018), DeepDownscale: a Deep Learning Strategy for High-Resolution Weather Forecast, *arXiv*, *1808.05264*.

Rodwell, M. J., L. Magnusson, P. Bauer, P. Bechtold, M. Bonavita, C. Cardinali, M. Diamantakis, P. Earnshaw, A. Garcia-Mendez, L. Isaksen, E. Källén, D. Klocke, P. Lopez, T. McNally, A. Persson, F. Prates, and N. Wedi (2013), Characteristics of Occasional Poor Medium-Range Weather Forecasts for Europe, *Bulletin of the American Meteorological Society*, *94*(9), 1393–1405, doi:10.1175/BAMS-D-12-00099.1.

Rotunno, R., J. B. Klemp, and M. L. Weisman (1988), A Theory for Strong, Long-Lived Squall Lines, *Journal of the Atmospheric Sciences*, *45*(3), 463–485, doi:10.1175/1520-0469(1988)045<0463:ATFSLL>2.0.CO;2.

Schlemmer, L., and C. Hohenegger (2014), The Formation of Wider and Deeper Clouds as a Result of Cold-Pool Dynamics, *Journal of the Atmospheric Sciences*, *71*(8), 2842–2858, doi:10.1175/JAS-D-13-0170.1.

Schneider, T., J. Teixeira, C. S. Bretherton, F. Brient, K. G. Pressel, C. Schär, and A. P. Siebesma (2017a), Climate goals and computing the future of clouds, *Nature Climate Change*, *7*(1), 3–5, doi:10.1038/nclimate3190.

Schneider, T., S. Lan, A. Stuart, and J. Teixeira (2017b), Earth System Modeling 2.0: A Blueprint for Models That Learn From Observations and Targeted High-Resolution Simulations, *Geophysical Research Letters*, *44*(24), 396–12, doi:10.1002/2017GL076101.

Selz, T., and G. C. Craig (2015a), Upscale Error Growth in a High-Resolution Simulation of a Summertime Weather Event over Europe*, *Monthly Weather Review*, *143*(3), 813–827, doi:10.1175/MWR-D-14-00140.1.

Selz, T., and G. C. Craig (2015b), Simulation of upscale error growth with a stochastic convection scheme, *Geophysical Research Letters*, *42*(8), 3056–3062, doi:10.1002/2015GL063525.

Sherwood, S. C., S. Bony, and J.-L. Dufresne (2014), Spread in model climate sensitivity traced to atmospheric convective mixing, *Nature*, *505*(7481), 37–42, doi:10.1038/nature12829.

Shutts, G. J., and T. N. Palmer (2007), Convective Forcing Fluctuations in a Cloud-Resolving Model: Relevance to the Stochastic Parameterization Problem, *Journal of Climate*, *20*(2), 187–202, doi:10.1175/JCLI3954.1.

Silver, D., A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. van den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, S. Dieleman, D. Grewe, J. Nham, N. Kalchbrenner, I. Sutskever, T. Lillicrap, M. Leach, K. Kavukcuoglu, T. Graepel, and D. Hassabis (2016), Mastering the game of Go with deep neural networks and tree search, *Nature*, *529*(7587), 484–489, doi:10.1038/nature16961.

Silver, D., J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker, M. Lai, A. Bolton, Y. Chen, T. Lillicrap, F. Hui, L. Sifre, G. van den Driessche, T. Graepel, and D. Hassabis (2017), Mastering the game of Go without human knowledge, *Nature*, *550*(7676), 354–359, doi:10.1038/nature24270.

Stensrud, D. J. (2007), *Parameterization Schemes*, Cambridge University Press, Cambridge, doi:10.1017/CBO9780511812590.

Stevens, B. (2005), Atmospheric Moist Convection, *Annual Review of Earth and Planetary Sciences*, *33*, 605–643, doi:10.1146/annurev.earth.33.092203.122658.

Stevens, B., M. Giorgetta, M. Esch, T. Mauritsen, T. Crueger, S. Rast, M. Salzmann, H. Schmidt, J. Bader, K. Block, R. Brokopf, I. Fast, S. Kinne, L. Kornblueh, U. Lohmann, R. Pincus, T. Reichler, and E. Roeckner (2013a), Atmospheric component of the MPI-M Earth System Model: ECHAM6, *Journal of Advances in Modeling Earth Systems*, *5*(2), 146–172, doi:10.1002/jame.20015.

Stevens, B., S. Bony, P. Ginoux, Y. Ming, and L. W. Horowitz (2013b), What Are Climate Models Missing?, *Science*, *340*(6136), 1053–1054, doi:10.1126/science.1237554.

Stocker, T. F., D. Qin, G.-K. Plattner, M. M. Tignor, S. K. Allen, J. Boschung, A. Nauels, Y. Xia, V. Bex, P. M. Midgley, L. V. Alexander, S. K. Allen, N. L. Bindoff, F.-M. Breon, J. A. Church, U. Cubasch, S. Emori, P. Forster, P. Friedlingstein, N. Gillett, J. M. Gregory, D. L. Hartmann, E. Jansen, B. Kirtman, R. Knutti, K. Kumar Kanikicharla, P. Lemke, J. Marotzke, V. Masson-Delmotte, G. A. Meehl, I. I. Mokhov, S. Piao, G.-K. Plattner, Q. Dahe, V. Ramaswamy, D. Randall, M. Rhein, M. Rojas, C. Sabine, D. Shindell, T. F. Stocker, L. D. Talley, D. G. Vaughan, S.-P. Xie, M. R. Allen, O. Boucher, D. Chambers, J. Hesselbjerg Christensen, P. Ciais, P. U. Clark, M. Collins, J. C. Comiso, V. Vasconcellos de Menezes, R. A. Feely, T. Fichefet, A. M. Fiore, G. Flato, J. Fuglestvedt, G. Hegerl, P. J. Hezel, G. C. Johnson, G. Kaser, V. Kattsov, J. Kennedy,

A. M. Klein Tank, C. Le Quere, G. Myhre, T. Osborn, A. J. Payne, J. Perlwitz, S. Power, M. Prather, S. R. Rintoul, J. Rogelj, M. Rusticucci, M. Schulz, J. Sedlacek, P. A. Stott, R. Sutton, P. W. Thorne, and D. Wuebbles (2013), *Climate Change 2013. The Physical Science Basis. Working Group I Contribution to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change - Abstract for decision-makers*.

Stull, R. B. (1988), *An introduction to boundary layer meteorology*, 666 pp., Kluwer Academic Publishers.

Sun, J., and M. S. Pritchard (2016), Effects of explicit convection on global land-atmosphere coupling in the superparameterized CAM, *Journal of Advances in Modeling Earth Systems*, *8*(3), 1248–1269, doi:10.1002/2016MS000689.

Tan, J., C. Jakob, W. B. Rossow, and G. Tselioudis (2015), Increases in tropical rainfall driven by changes in frequency of organized deep convection, *Nature*, *519*(7544), 451–454, doi:10.1038/nature14339.

Tan, Z., C. M. Kaul, K. G. Pressel, Y. Cohen, T. Schneider, and J. Teixeira (2018), An Extended Eddy-Diffusivity Mass-Flux Scheme for Unified Representation of Subgrid-Scale Turbulence and Convection, *Journal of Advances in Modeling Earth Systems*, *10*(3), 770–800, doi:10.1002/2017MS001162.

Tiedtke, M. (1989), A Comprehensive Mass Flux Scheme for Cumulus Parameterization in Large-Scale Models, *Monthly Weather Review*, *117*(8), 1779–1800, doi:10.1175/1520-0493(1989)117<1779:ACMFSF>2.0.CO;2.

Tompkins, A. M. (2001), Organization of Tropical Convection in Low Vertical Wind Shears: The Role of Cold Pools, *Journal of the Atmospheric Sciences*, *58*(13), 1650–1672, doi:10.1175/1520-0469(2001)058<1650:OOTCIL>2.0.CO;2.

Tulich, S. N. (2015), A strategy for representing the effects of convective momentum transport in multiscale models: Evaluation using a new superparameterized version of the Weather Research and Forecast model (SP-WRF), *Journal of Advances in Modeling Earth Systems*, *7*(2), 938–962, doi:10.1002/2014MS000417.

UNFCCC (2015), *Paris Agreement*.

Vitart, F., and A. W. Robertson (2018), The sub-seasonal to seasonal prediction project (S2S) and the prediction of extreme events, *npj Climate and Atmospheric Science*, *1*(1), 3, doi:10.1038/s41612-018-0013-0.

Voosen, P. (2016), Climate scientists open up their black boxes to scrutiny., *Science*, *354*(6311), 401–402, doi:10.1126/science.354.6311.401.

Voosen, P. (2018), The Earth Machine, *Science*, *361*(6400), 344–347, doi:10.1126/science.361.6400.

Wallace, J. M., and P. V. Hobbs (2006), *Atmospheric Science An introductory survey*, 505 pp., Academic Press, doi:10.1021/jp112019s.

Wang, Y., and G. J. Zhang (2016), Global climate impacts of stochastic deep convection parameterization in the NCARCAM5, *Journal of Advances in Modeling Earth Systems*, *8*(4), 1641–1656, doi:10.1002/2016MS000756.

Wang, Y., G. J. Zhang, and G. C. Craig (2016), Stochastic convective parameterization improving the simulation of tropical precipitation variability in the NCAR CAM5, *Geophysical Research Letters*, *43*(12), 6612–6619, doi:10.1002/2016GL069818.

Weisheimer, A., S. Corti, T. Palmer, and F. Vitart (2014), Addressing model error through atmospheric stochastic physical parametrizations: impact on the coupled ECMWF seasonal forecasting system., *Philosophical transactions. Series A, Mathematical, physical, and engineering sciences*, *372*(2018), 20130,290, doi:10.1098/rsta.2013.0290.

Weisman, M. L., W. C. Skamarock, and J. B. Klemp (1997), The Resolution Dependence of Explicitly Modeled Convective Systems, *Monthly Weather Review*, *125*(4), 527–548, doi:10.1175/1520-0493(1997)125<0527:TRDOEM>2.0.CO;2.

Wheeler, M., and G. N. Kiladis (1999), Convectively Coupled Equatorial Waves: Analysis of Clouds and Temperature in the Wavenumber–Frequency Domain, *Journal of the Atmospheric Sciences*, *56*(3), 374–399, doi:10.1175/1520-0469(1999)056<0374:CCEWAO>2.0.CO;2.

Wilks, D. S. (2006), *Statistical Methods in the Atmospheric Sciences*, Elsevier.

Woelfle, M. D., S. Yu, C. S. Bretherton, and M. S. Pritchard (2018), Sensitivity of Coupled Tropical Pacific Model Biases to Convective Parameterization in CESM1, *Journal of Advances in Modeling Earth Systems*, *10*(1), 126–144, doi:10.1002/2017MS001176.

Yadav, A. K., and S. Chandel (2014), Solar radiation prediction using Artificial Neural Network techniques: A review, *Renewable and Sustainable Energy Reviews*, *33*, 772–781, doi:10.1016/J.RSER.2013.08.055.

Zhang, C. (2005), Madden-Julian Oscillation, *Reviews of Geophysics*, *43*(2), RG2003, doi:10.1029/2004RG000158.

Zhang, F., N. Bei, R. Rotunno, C. Snyder, and C. C. Epifanio (2007), Mesoscale predictability of moist baroclinic waves: Convection-permitting experiments and multistage error growth dynamics, *Journal of the Atmospheric Sciences*, *64*(10), 3579–3594.

Zhang, T., L. Li, Y. Lin, W. Xue, F. Xie, H. Xu, and X. Huang (2015), An automatic and effective parameter optimization method for model tuning, *Geoscientific Model Development*, *8*(11), 3579–3591, doi:10.5194/gmd-8-3579-2015.

Acknowledgements

First, I would like to thank my supervisor George Craig, who has supported me for more than five years. His enthusiasm for science was a constant motivation. I am especially grateful for the free rein he gave me when I took off on my own endeavours.

Then I would like to thank all other people who have supported my work and collaborated with me over the last few years: Tobias Selz who always took the time to think thoroughly about any results I showed him. Sebastian Lerch who enabled the most fruitful collaboration I have experienced yet in my career. Mike Pritchard and Pierre Gentine who enthusiastically opened their doors when I came knocking as an unknown PhD student from Germany.

Special thanks go to Jeremy Howard and Rachel Thomas and their phenomenal online course fast.ai. This course has transformed my life, quite literally.

Next, I would like to thank my work colleagues and friends who have made my time at LMU as enjoyable as it was. In no particular order: Julia Windmiller, Mirjam Hirt, Tobias Necker, Kevin Bachmann, Florian Baur, Yvonne Ruckstuhl, and many others!

Finally, thanks go to my wife and my parents for their constant support!

This work was funded within the DFG Transregional Collaborative Research Center "Waves to Weather" (W2W; SFB/TRR165).

"The first principle is that you must not fool yourself and you are the easiest person to fool."
Richard Feynman