Using neural networks to improve simulations in the gray zone

Raphael Kriegmair[1], Yvonne Ruckstuhl[1], Stephan Rasp[2], and George Craig[1]

[1]Meteorological Institute Munich, Ludwig-Maximilians-Universität München, Germany
[2]ClimateAi, Inc.

Correspondence: Yvonne Ruckstuhl (yvonne.ruckstuhl@lmu.de)

Abstract. Machine learning represents a potential method to cope with the gray zone problem of representing motions in dynamical systems on scales comparable to the model resolution. Here we explore the possibility of using a neural network to directly learn the error caused by unresolved scales. We use a modified shallow water model which includes highly nonlinear processes mimicking atmospheric convection. To create the training dataset we run the model in a high and a low-resolution setup and compare the difference after one low resolution time step starting from the same initial conditions, thereby obtaining an exact target. The neural network is able to learn a large portion of the difference when evaluated ~~"offline"~~ on single time step predictions on a validation set. When coupled to the low-resolution model, we find large forecast improvements up to one day on average. After this, the accumulated error due to the mass conservation violation of the neural network starts to dominate and deteriorates the forecast. This deterioration can effectively be delayed by adding a penalty term to the loss function used to train the ANN to conserve mass in a weak sense. This study reinforces the need to include physical constraints in neural network parameterizations.

1 Introduction

Current limitations on computational power force weather and climate prediction to use relatively low resolution simulations. Subgrid scale ~~(SGS)~~ processes, i.e. processes that are not resolved by the model grid, are typically represented using physical parameterizations (Stensrud, 2009). Inaccuracies in these parameterizations are known to cause errors in weather forecasts and biases in climate projections. While parameterizations are becoming more sophisticated over time, there is evidence that key structural uncertainties remain (Randall et al., 2003; Randall, 2013; Jones and Randall, 2011).

A particularly difficult problem in the representation of unresolved processes is the so-called gray zone (Chow et al., 2019; Honnert et al., 2020), where a certain physical ~~phenomena~~ phenomenon such as a cumulus cloud is similar in size to the model resolution and hence partially resolved. In the development of many classical parameterizations, features are assumed to be small in comparison to the model resolution. This scale separation provides a conceptual basis for specifying the average effects of the unresolved flow features on the resolved flow. In contrast, there is no theoretical basis for determining such a

relationship in the gray zone. Instead, the truncation error of the numerical model is a significant factor. While we might still expect there to be some relationship between the resolved and unresolved parts of the flow, we have no way to define it.

~~The gray zone is of great importance in practice since the~~ Viewing the atmosphere as a turbulent flow, with up- and downscale cascades, phenomena like synoptic cyclones and cumulus clouds emerge where geometric or physical constraints impose length scales on the flow (Lovejoy and Schertzer, 2010; Marino et al., 2013; Faranda et al., 2018). If a numerical model is truncated near one of these scales, the corresponding phenomenon will be only partially resolved and the simulation will be inaccurate. In particular, the properties of the phenomenon may be determined by the truncation length, rather than by the physical scale. A thorough review of the gray zone problem from a turbulence perspective is provided by Honnert et al. (2020).

An important example of the gray zone in practice is the simulation of deep convective clouds in kilometer-scale models used operationally for regional weather prediction~~are in the gray zone for cumulus convection. These models are typically run without a parameterization for deep convection, but the~~ . The models typically have a horizontal resolution of 2-4 ~~kmdoes not give accurate results for typical convective cloud structures are often less than~~ km, which is not sufficient to fully resolve the cumulus clouds with sizes in the range from 1 to 10 ~~kmin size (Bryan et al., 2003; Wagner et al., 2018). The flow on these scales is also influenced by partially-resolved orography and other surface properties, which also belong to the gray zone. With no obvious methodology for developing a parameterization suitable for these scales, the most that can be hoped for current schemes is that their influence diminishes with increasing resolution (Jeworrek et al., 2019)~~km. In these models, the simulated cumulus clouds collapse to a scale proportional to the model grid length, unrealistically becoming smaller and more intense as resolution is increased (Bryan et al., 2003; Wagner et al., 2018). In models with grid lengths over 10 km, the convective clouds are completely subgrid and should be parameterized, while models with resolution under 100 m will accurately reproduce the dynamics of cumulus clouds provided that the turbulent mixing processes are well represented. In the gray zone in between, the performance of the models depends sensitively on resolution and details of the parameterizations that are used (Jeworrek et al., 2019).

Using machine learning methods such as artificial neural networks (ANNs) for alleviating the problems described above has received increasing attention over the past years. One approach is to avoid the need of parameterizations all together by emulating the entire model using observations (Brunton et al., 2016; Pathak et al., 2018; Faranda et al., 2020; Fablet et al., 2018; Scher, 2018; Dueben and Bauer, 2018). In these studies a dense and noise free observation network is often assumed. Brajard et al. (2020a) and Bocquet et al. (2020) circumvent the requirement of this assumption by using data assimilation to form targets for ANNs from sparse and noisy observations.

Though studies have shown that surrogate models produced by machine learning can be accurate for small dynamical systems, replacing an entire numerical weather prediction model for operational use is not yet within our reach. Therefore, a more practical approach is to use ANNs as replacement for uncertain parameterizations. This has been done either by learning from physics based expensive parametrization schemes (O'Gorman and Dwyer, 2018; Rasp et al., 2018) or high resolution simulations (Krasnopolsky et al., 2013; Brenowitz and Bretherton, 2019; Bolton and Zanna, 2019; Rasp, 2020; Yuval and O'Gorman, 2020), which is the approach we take here. Such data driven techniques could be a way to reduce the structural uncertainty of traditional parameterizations, even at gray zone resolutions where the physical basis of the parameterization is

no longer valid. The first challenge is to create the training data, i.e. to separate the resolved and unresolved scales from the high-resolution simulation. Brenowitz and Bretherton (2019) use a coarse-graining approach based on subtracting the coarse grained advection term from the local tendencies. This approach can be used for any model and resolution but is sensitive to the choice of grid and time step. Further, the resulting subgrid tendencies are only an approximation and may not represent the real difference between the low and high-resolution model. Yuval and O'Gorman (2020) use the same model version for low and high-resolution simulations and compute exact differences after a single low-resolution time step by starting both model versions from the same initial conditions. They manage to obtain stable long-term simulations using the low-resolution model with a machine learning correction that come close the the high-resolution ground truth.

Here, we use the modified rotating shallow water (modRSW) model to explore the use of a machine learning subgrid representation in a highly non-linear dynamical system. The modRSW is an idealized fluid model of convective-scale numerical weather prediction, in which convection is triggered by orography. As such, the model mimics the gray zone problem of operational kilometer-scale models. Using a simplified model allows us to focus on some key conceptual questions surrounding machine learning parameterizations, such as how choices in neural network training affect long-term physical consistency. In particular, we include weak physical constraints in the training procedure.

The contents of this work are outlined in the following. Section 2 introduces the experiment setup used to obtain and analyze results. The modRSW model is briefly explained in Section 2.1, followed by a description of the training data generation in Section 2.2. The architecture and training process of the ANN used in this research are given in 2.3. ~~Results~~ Our verification metrics are defined in Section 3. The results are presented in Section 4~~. followed by a conclusion~~, followed by concluding remarks in Section 5.

2 Experiment setup

2.1 The modRSW Model

The modRSW model (Kent et al., 2017) used in this research represents an extended version of the 1D shallow water equations, i.e. 1D fluid flow over orography. Its prognostic variables are fluid height h, wind speed u and a rain mass fraction r. Based on the model by Würsch and Craig (Würsch and Craig, 2014) it implements two threshold heights, $H_c < H_r$, initiating convection and rain production, respectively. Convection is stimulated by modifying the pressure term to remain constant where h rises above H_c. In contrast to Würsch and Craig (2014), the modRSW model does not apply diffusion or stochastic forcing. The model is mass conserving, meaning that the domain mean of h is constant over time. In this study, a small but significant model-intrinsic drift in the domain mean of u is ~~accounted for~~ removed by adding a relaxation term. This term is defined using a corresponding time scale t_{relax}, as $(\bar{u}_0 - \bar{u}_t) \cdot t_{relax}$, where the overbar denotes the domain mean. Depending on the orography used, this model yields a ~~continuous~~ range of dynamical organization between regular and chaotic behaviour. ~~We pick one simulation from each extreme and compare results to identify general and flow dependent aspects.~~ Orography is defined as a superposition of cosines with wavenumbers $k = 1/L, ..., k_{max}/L$ (L domain length). Amplitudes are given as $A(k) = 1/k$, while phase shifts for each term are randomly chosen from $[0, L]$. In this work, two realizations of the orography

are selected to represent regular and more chaotic dynamical behavior. Figure 1 displays a 24 hour segment of the simulation corresponding to each orography.

Schematic of training data generation process. A HR run is coarse grained to LR to generate model truth. Each model truth state is integrated forward for one time step using LR dynamics. The difference between the obtained states and corresponding model truth defines the desired network output (red arrows), while the preceding model truth defines network input.

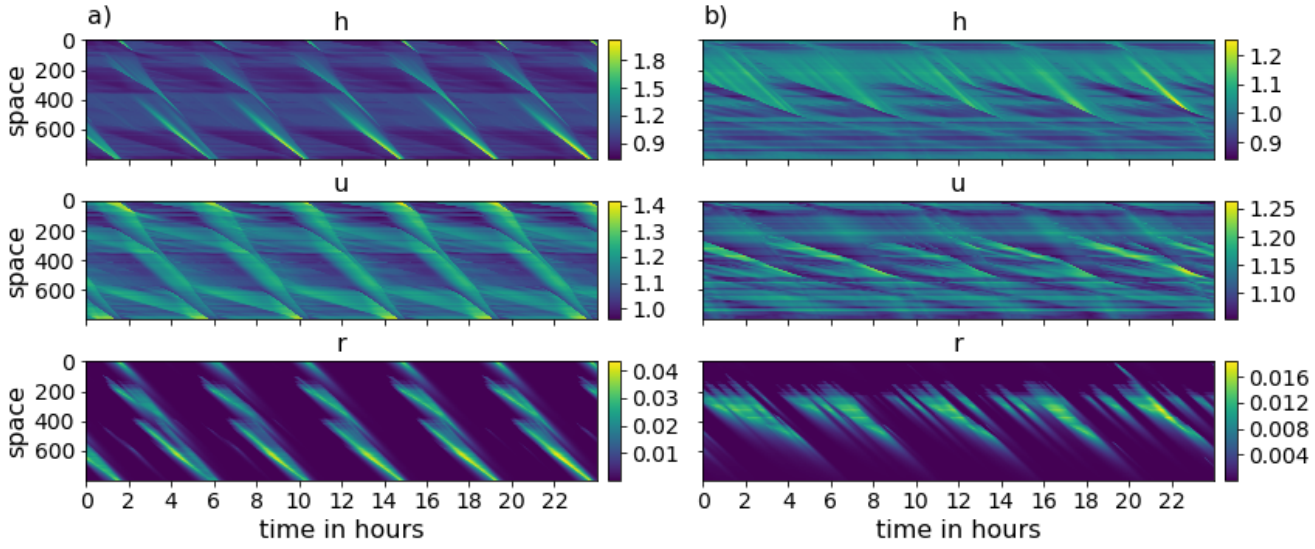

Figure 1. 24 hour segment of the HR simulation for the three model variables h, u, r (from top to bottom) corresponding to the regular case (left panel) and the chaotic case (right panel).

2.2 Training Data Generation

Conceptually, the ANN's task is to correct a low resolution (LR) model forecast towards the model truth, which is a coarse grained high resolution (HR) model simulation. The coarse graining factor in this study is set to ~~4.~~ 4, which is analogous to the range of scales found in the gray zone where deep cumulus convection is partially resolved (e.g. 2.5-10 km). Faranda et al. (2020) show that the choice of coarse graining factor can substantially affect the performance of ML methods. In our case, however, choosing a larger factor would correspond to a coarse model grid length that is larger than the typical cloud size, changing the nature of the problem from learning to improve poorly resolved existing features in the coarse simulation to parameterizing features that might not be seen at all. The dynamical time step of the model is determined at each iteration based on the Courant–Friedrichs–Lewy (CFL) criterion. To achieve temporally equidistant output states for both resolutions, the time step is truncated accordingly when necessary.

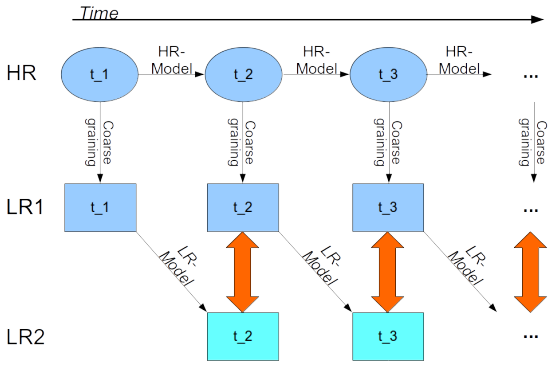

Figure 2. Schematic of training data generation process. A HR run is coarse grained to LR to generate model truth. Each model truth state is integrated forward for one time step using LR dynamics. The difference between the obtained states and corresponding model truth defines the desired network output (red arrows), while the preceding model truth defines the network input.

A training sample (input-target pair) is defined by the model truth at some time ~~t_n~~ t_i and the difference between the model truth and the corresponding LR forecast at ~~$t_{n+1} = t_n + dt$~~$t_{i+1} = t_i + dt$, respectively (see Figure 2). To generate the model truth, HR data are obtained by integrating the modRSW model forward using the parameters shown in Table 1. All states and the orography are subsequently coarse grained to LR, resulting in model truth (LR1). Each LR1 state is integrated forward for a single timestep using the modRSW model on LR with the coarse grained orography, resulting in a single step prediction (LR2). The synchronized differences $LR1(t_i) - LR2(t_i)$ then define the training targets corresponding to the input $LR1(t_{i-1})$, which includes the orography. A time series of $T = 200000$ time steps, which is equivalent to approximately 57 days in real time, is generated for both orographies. The first day of the simulation is discarded as spin up, the subsequent 30 days are used for training and the remaining 26 days are used for validation purposes. The decorrelation length scale of the model is roughly 4 hours.

~~Orography is defined as a superposition of cosines with wavenumbers $k = 1/L, ..., k_{max}/L$ (L domain length). Amplitudes are given as $A(k) = 1/k$, while phase shifts for each term are randomly chosen from $[0, L]$. All states and the orography are subsequently coarse grained to LR, resulting in model truth (LR1). Each LR1 state is integrated forward for a single timestep using the modRSW model on LR with the coarse grained orography, resulting in a single step prediction (LR2). The synchronized differences $LR1(t_i) - LR2(t_i)$ then define the training targets corresponding to the input $LR1(t_{i-1})$, which includes the orography. A time series of $T = 200000$ time steps is generated for both orographies, of which the odd time steps are used for training and the even time steps for validation.~~

2.3 Convolutional ANN

A characteristic property of convolutional ANNs is that they reflect spatial invariance and localization. These two properties also apply to the dynamics of many physical systems, such as the one investigated here. They differ from e.g. dense networks

Model Parameter	Symbol	Value	Notes
HR gridpoint number	~~HR~~ N_{HR}	800	-
LR gridpoint number	~~LR~~ N_{LR}	200	-
Time step	dt	0.001	-
Domain size (non dim.)	L	1.0	-
CFL	-	0.5	-
Convection threshold	H_c	1.02	-
Rain threshold	H_r	1.05	-
Initial total height	H_0	1.0	-
Rossby Number	Ro	∞	-
Froude Number	Fr	1.1	-
Effective gravity	g	Fr^{-2}	-
Beta	β	0.2	-
Alpha	α^2	10	-
Rain Conversion Factor	c^2	$0.1 \times g \times H_r$	-
Wind Relaxation time scale	t_{relax}	dt	-
Orography Generation			
Maximum wave number	k_{max}	100	-
Maximum Amplitude	B_{max}	0.1	-

Table 1. Model setting parameters

by the use of a so called kernel. This vector is "moved" step by step across the domain grid, covering k grid points at each position. At each position, the dot product of kernel and current grid values is computed, determining (along with an activation function) the corresponding output value. For more details on convolutional ANNs we refer to Goodfellow et al. (2016).

The ANN structure used in this research is described in the following. 5 hidden layers are applied, each using the *ReLU* activation function. The input layer uses *ReLU* as well, while the output layer uses a linear activation function. All hidden layers have 32 filters. The input and output layer shapes are defined by input and target data. The kernel size is set uniformly to 3 grid points. Biases are applied throughout the ANN.

The loss is determined during training by comparing the ANN output to the corresponding target. A standard measure for loss is the mean squared error (MSE). However, any loss function can be used to ~~to~~ tailor to the application. For example, additional terms can be added to impose weak constraints on the training process, as for example done in Ruckstuhl et al. (2021). This possibility is exploited here to impose mass conservation in a weak sense. The constraint is implemented by penalizing the deviation of the square of domain mean *h* corrections from zero. The loss function is defined as

$$MSE\text{MSE}(y_{out}, y_{target}) + w_{mass} \cdot \left(\overline{y^h_{out}}\right)^2 \quad (1)$$

where the second term represents a weighted mass conservation constraint. In this expression, ~~MSE~~ y_{out} and y_{target} are the output and corresponding target of the ANN respectively, MSE denotes the mean ~~square error,~~ squared error, the scalar w_{mass} is the mass conservation constraint weighting, ~~Δh are ANN corrections~~ y^h_{out} is the ANN output for h and the overbar denotes the domain mean.

The Adam algorithm with a learning rate of 10^{-3} is used to minimize the loss function over the ANN weights in batches of 256 samples. Since the loss function is typically not convex, the ANN likely converges to a local minimum. To sample this error, we repeat the training of each ANN presented in this paper with different initial weights 5 times. The initial weights are drawn randomly. For all ANNs a total of 1000 epochs is performed. The ANN architecture and hyperparameters were selected based on a loose tuning procedure, where no strong sensitivities were detected. The training is done using python library Keras (Chollet et al., 2015).

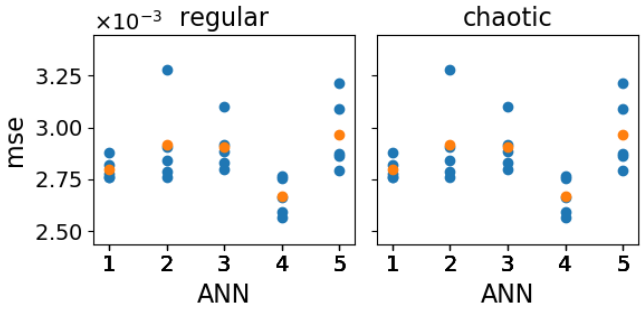

Figure 3. Loss function value for $w_{mass} = 0$ (mse) of the validation data corresponding to the last 5 epochs of the training process (~~y-axis~~blue) for each trained ~~ANNs~~ ANN (x-axis). For each ANN the mean loss function value over the last 5 epochs is depicted in orange.

3 ~~Results~~Verification methods

As the initial training weights of the ANNs and the exact number of epochs performed is to some extent arbitrary, it is desirable to measure the sensitivity of our results to the realization of these quantities. Figure 3 shows the MSE of the validation data set of the last 5 epochs (y-axis) for 5 ANNs with different realizations of initial training weights (x-axis) for both orographies. Since the MSE appears sensitive to both the initial weights and the epoch number, we use both to sample the total ANN variability, resulting in 5×5=25 samples for each ANN training setup that is presented in the remainder of this paper. , spatial mean error (SME) and bias:

$$\mathrm{RMSE}(\mathbf{y}) = \sqrt{\frac{1}{N_{LR}} \sum_{j=1}^{N_{LR}} \left(y_j - y_j^{true}\right)^2} \tag{2}$$

$$\mathrm{SME}(\mathbf{y}) = \left| \frac{1}{N_{LR}} \sum_{j=1}^{N_{LR}} \left(y_j - y_j^{true}\right) \right| \tag{3}$$

$$\mathrm{bias}(\mathbf{y}) = \frac{1}{N_{LR}} \sum_{j=1}^{N_{LR}} \left(y_j - y_j^{true}\right) \tag{4}$$

where $N_{LR} = 200$ is the number of grid points and $\mathbf{y}^{true} \in \mathbb{R}^{N_{LR}}$ is a snapshot of the model truth. Multiple samples of these scores are obtained using both the 25 realizations of the ANN, and a sequence of initial conditions provided by the time dimension. The final verification metrics are then the mean and standard deviation (SD) of the respective scores $X \in \{\mathrm{RMSE}, \mathrm{SME}, \mathrm{bias}\}$:

$$\overline{X}(\mathbf{y}) = \frac{1}{L_{ANN} T_{veri}} \sum_{l=1}^{L_{ANN}} \sum_{t=0}^{T_{veri}} X(\mathbf{y}_t^l) \tag{5}$$

$$\mathrm{SD}_{\mathrm{total}}\left(X(\mathbf{y})\right) = \sqrt{\frac{1}{L_{ANN} T_{veri}} \sum_{l=1}^{L_{ANN}} \sum_{t=0}^{T_{veri}} \left(X(\mathbf{y}_t^l) - \overline{X}(\mathbf{y})\right)^2} \tag{6}$$

$$\mathrm{SD}_{\mathrm{time}}\left(X(\mathbf{y})\right) = \frac{1}{L_{ANN}} \sum_{l=1}^{L_{ANN}} \sqrt{\frac{1}{T_{veri}} \sum_{t=0}^{T_{veri}} \left(X(\mathbf{y}_t^l) - \overline{X}(\mathbf{y}^l)\right)^2} \tag{7}$$

$$\mathrm{SD}_{\mathrm{ANN}}\left(X(\mathbf{y})\right) = \frac{1}{T_{veri}} \sum_{l=0}^{T_{veri}} \sqrt{\frac{1}{L_{ANN}} \sum_{l=1}^{L_{ANN}} \left(X(\mathbf{y}_t^l) - \overline{X}(\mathbf{y}_t)\right)^2} \tag{8}$$

where t and l index time and ANN realizations respectively, $\overline{X}(\mathbf{y}_t)$ indicates the mean over time steps and $\overline{X}(\mathbf{y}^l)$ the mean over ANN realizations. Note that equations (7) and (8) are meant to isolate the variability inherited from initial conditions and ANN realizations respectively. We apply these verification metrics to both single time step predictions and 48-hour forecasts.

- **Single time step predictions:**
  For each time step corresponding to the validation data set ($T_{veri} = 92863$), the model truth is used as initial condition for a single time step prediction of the LR model, creating a LR prediction (LR). This LR prediction is subsequently corrected by the ANN, creating the corresponding ANN-corrected prediction (LR_{ANN}).

– **48-hour forecasts:**
The 48-hour forecasts are generated from a set of 50 initial conditions ($T_{veri} = 50$) taken from the validation data set. To ensure independence, the initial conditions are set 4 hours apart, which is roughly the decorrelation length scale of the model. After each low resolution single time step prediction, the ANN is applied to create initial conditions for the next LR single time step prediction, creating a 48-hour LR ANN-corrected forecast (LR_{ANN}). As reference, a LR simulation without ANN corrections (LR) is run in parallel.

For the single time step predictions and the 48-hour forecasts our verification metrics are applied to both LR and LR_{ANN} and compared in section 4. Note that $L_{ANN} = 1$ for LR, yielding $\overline{X}(LR) = \mathrm{SD}_{\mathrm{ANN}}(X(LR))$ and $\mathrm{SD}_{\mathrm{total}}(X(LR)) = \mathrm{SD}_{\mathrm{time}}(X(LR))$.

4 Results

We performed a series of experiments designed to investigate the feasibility of using an ANN to correct for model error due to unresolved scales. In section 4.1 we first explore the performance of the ANNs trained with the standard MSE as loss function ($w_{mass} = 0$ in equation (1)). Next, the weak constraint is added to the loss function as in equation (1) and the benefits are examined in section 4.2.

4.1 ANN with standard loss function

Figure 4 shows the results for the single time step predictions. The improvements ~~are large for both the chaotic and regular case~~achieved by the ANN with respect to LR in terms of RMSE are substantial, for h,u and r amounting to 97%, 89% and 90% for the regular case and 96%, 84% 92% for the chaotic case. Also, the negative biases present in u and r for LR virtually disappear when the ANN is applied. However, as the ANN is not explicitly instructed to conserve mass, a small SME is introduced in variable h. Its significance will become apparent when analyzing the 48-hour forecasts. It is interesting to note that the ANN's architecture and learning process is unbiased (small $\overline{\mathrm{bias}}$), although single ANN realizations may be biased (large $\mathrm{SD}_{\mathrm{ANN}}$). Also, in contrast to the SME and bias, for the RMSE the initial conditions are the main source of variability ($\mathrm{SD}_{\mathrm{time}} \gg \mathrm{SD}_{\mathrm{ANN}}$). This is better visible in Figures 7 and 8, where the results for $w_{mass} = 0$ are plotted again.

Next we examine the effect of the ANN on a 48-hour forecast. Here we compare ~~a~~ the LR simulation with (LR_{ANN}) and without (LR) the use of the ANN. Both simulations start from the same initial conditions as the model truth. ~~This is repeated for 50 initial conditions, each 2 hours apart. The results in terms of RMSE with respect to the model truth are~~ The evolution of $\overline{\mathrm{RMSE}}$ is presented in Figure 5. ~~Note that the shaded region for LR_{ANN} includes the variability due to the ANN and the initial conditions.~~ The RMSEs corresponding to the regular case are higher than for the chaotic case. This is because the regular case exhibits a repeating pattern of long-lived, high amplitude convective events~~(not shown)~~. In comparison, the chaotic case produces short-lived perturbations with very small amplitude~~(not shown)~~, leading to smaller climatological variability.

For both orographies the ANN has a clear positive effect on the forecast until the error of LR saturates, after which the error of LR_{ANN} continues to grow. For the chaotic case this leads to a detrimental impact of the ANN after about ~~30 hours.~~ 1 day.

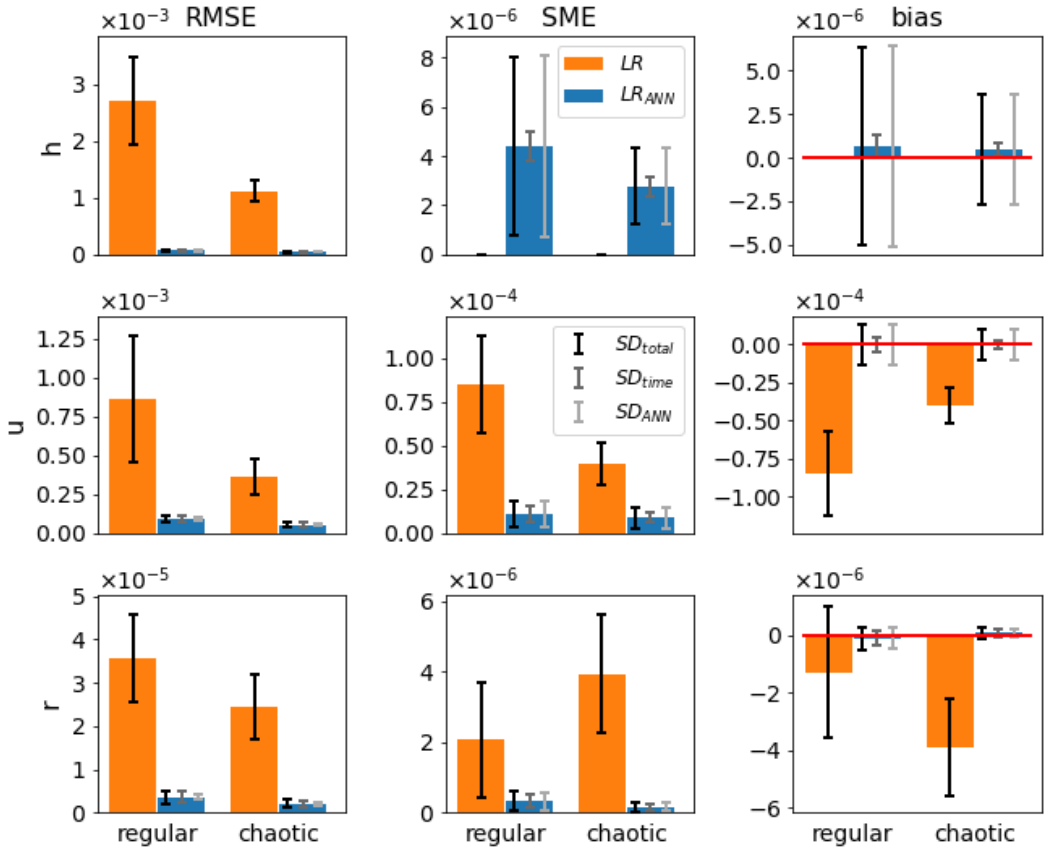

Figure 4. Mean (bars) and standard deviations SD_{total}, SD_{time}, SD_{ANN} (error bars, from dark to light respectively) of the RMSE (left panel), SME (middle panel) and bias (right panel) of ANN corrected (blue) and uncorrected (orange) single time step predictions of the validation data with respect to the model truth for the regular and chaotic case (y-axis) and for variables h, u, r (from top to bottom).

Also, the SD_{total} of LR_{ANN} is rapidly exceeding that of LR. This is because, in contrast to LR, the shaded region for LR_{ANN} includes the variability due to the ANN realizations which significantly contributes to the total variability. This is seen in Figure 12 and discussed further in the next section.

It is not surprising that LR_{ANN} deteriorates as the forecast lead time increases, since the ANNs are not perfect (as opposed to the data they were trained on) and the resulting errors accumulate over time, leading to biases. This is clearly visible in Figure 6, where it is seen that the ~~domain mean error~~ SME of h and r ~~diverges~~diverge, in contrast to LR. ~~It is worth noting that the domain mean error~~ The SME of r for LR is the result of a negative bias in the amount of rain produced (see Figure 4), caused by the coarse graining of the orography~~(not shown)~~. This bias is ~~partially (sometimes over) corrected~~ significantly reduced by the ANNs. The divergence of the ~~domain mean error~~ SME of h is the result of applying ANNs that, in contrast to the model, do not conserve mass. This leads to accumulated mass errors, causing biases in the wind field due to momentum

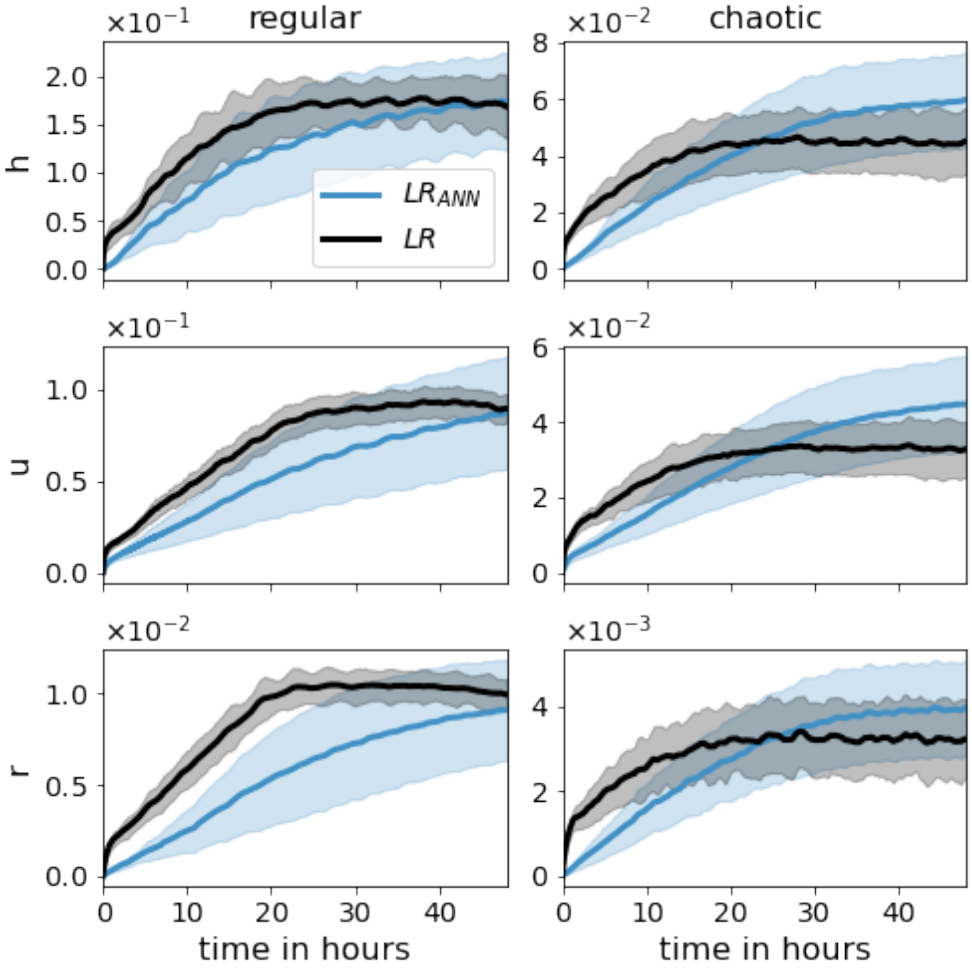

Figure 5. ~~RMSE~~ RMSE evolution of 48-hour forecasts for ~~the respective~~ model variables h, u and r (from top to bottom) of LR (black) and LR_{ANN} (blue)~~, averaged over 50 initial conditions and in the case of LR_{ANN} 25 ANNs~~. The shaded region ~~depicts the standard deviation of the RMSE~~corresponds to SD_{total}.

conservation and a change in probability for the fluid to rise above H_c and H_r. We therefore investigate if reducing the mass error, by adding a penalty term to the loss function of the ANN, can increase the forecast skill further.

4.2 ANN with mass conservation in a weak sense

~~We have~~ Instead of including mass conservation in the training process of the ANN, it is natural to first try to correct the mass violation by post processing the ANN corrections. We tested two approaches: homogeneously subtracting the spatial mean of the h corrections, and multiplying the vector of positive (negative) h corrections with the appropriate scalar when

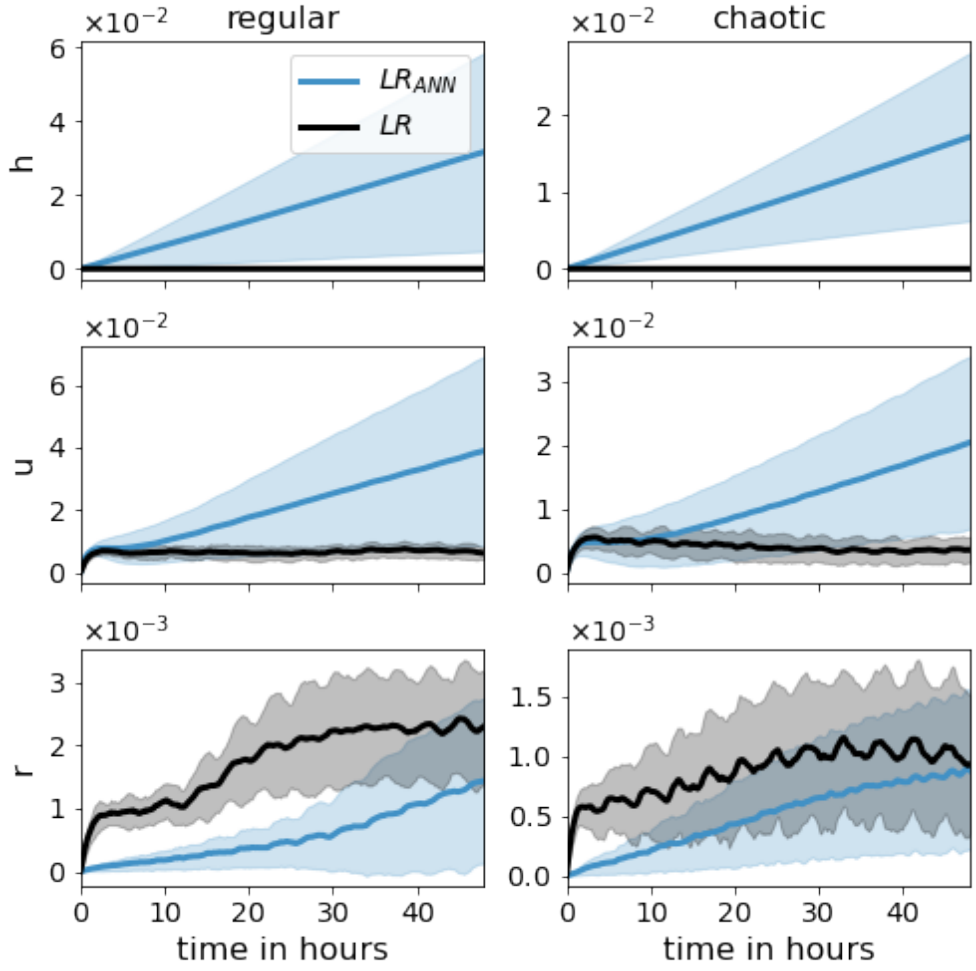

Figure 6. Same as Figure 5, but for the ~~spatial mean absolute error of the variables~~MSE.

the mass violation is positive (negative). Neither of these simple approaches led to improvements. We therefore included mass conservation in a weak sense in the training process of the ANN, as described equation (1). We trained ANNs with mass conservation weightings of $w_{mass} = 1, 10, 100, 1000$. These weightings result in a contribution to the loss function of roughly $0.2\%, 0.7\%, 2\%$ and 5% throughout the training process respectively (not shown). Note that the ANNs presented in the previous section correspond to $w_{mass} = 0$.

Figures 7 and 8 ~~shows~~ show the single time step predictions ~~improvements for both orographies~~for the regular and chaotic case respectively. Clearly, the mass conservation penalty term in the loss function has the desired effect of reducing the mass error for both orographies~~, though for the regular case a weighting of $w_{mass} > 1$ is necessary. As hypothesised, reducing the mass error also has a positive effect on the domain mean error of~~ . Also, the ~~wind u. The standard deviation of the bias is~~

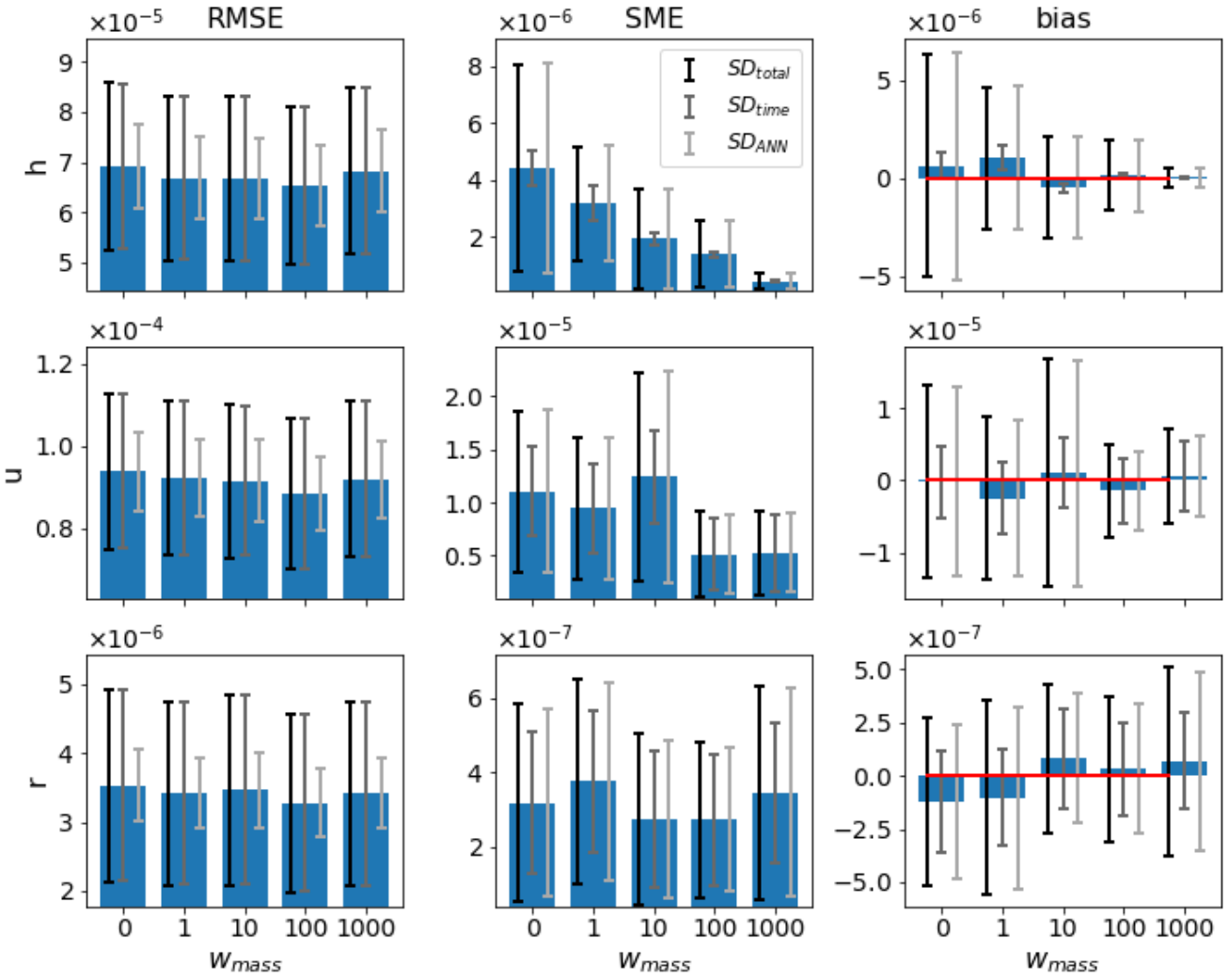

Figure 7. ~~Mean RMSE expressed in improvement as in Figure 4~~ RMSE (left), ~~absolute error of the spatial mean~~ MSE (middle), and bias (right) of the validation data for the different weightings (x-axis) and the respective model variables (rows) for the regular case. Error bars indicate ~~the standard deviation~~ (from dark to light) SD_{total}, SD_{time} and SD_{ANN} and the red lines in the right panel indicate the zero line.

~~much larger than the mean bias for all variables and both orographies. We believe that the mean bias would go to zero as the~~ ~~sample size (which is now only 25) increases. This indicates that single ANNs do not have a preference for either a positive~~ ~~or negative bias. The decrease of standard deviation of the bias as w_{mass} increases confirms the~~ error bars of the mass bias go down. A clear, convincing correlation between the ~~mass error in h and the wind bias . Contrary to what we expected, the~~ ~~RMSE does not generally go up when~~ reduction in SME and bias for h and any other field and/or metric is not detected, with possibly the exception of the SME for u in the chaotic case. A trade-off between increasing RMSE and decreasing MSE for

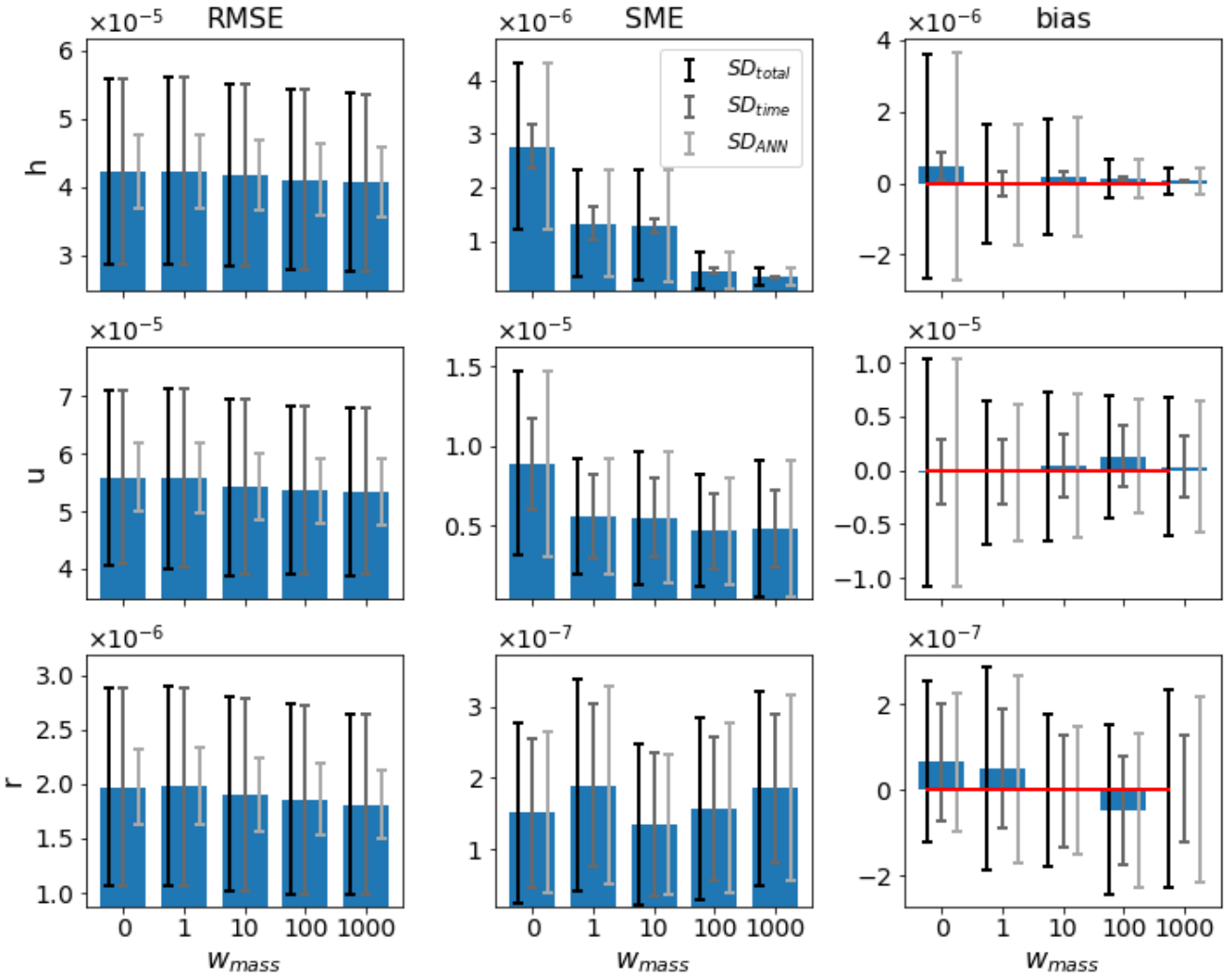

Figure 8. Same as Figure 7, but for the chaotic case.

increasing w_{mass} ~~is increased. It even seems to go down for all variables. We do see the RMSE for $w_{mass} = 1000$ going up for the regular case, which indicates that there is threshold after which the penalty terms becomes detrimental to the RMSE. For the regular case this threshold lies between $100 < W_{mass} < 1000$ and~~ was expected, but is not observed. The RMSE even tends to decrease a minimal amount for the chaotic case~~it lies (assuming it exists) above $w_{mass} > 1000$.~~.

Figure 9 presents the mean RMSE of the 48-hour forecasts for all weightings. The weak mass conservation constraint has the desired effect on the forecast skill. For the chaotic case, ~~about 10~~ more than 15 hours in forecast quality ~~are~~ is gained. For the regular case the ~~exact~~ number is unclear since the RMSE is still lower than LR and has not yet saturated after 48 hours. However we can say that it is at least 30 hours. As hypothesized, Figure 10 indicates that the divergence of the domain mean

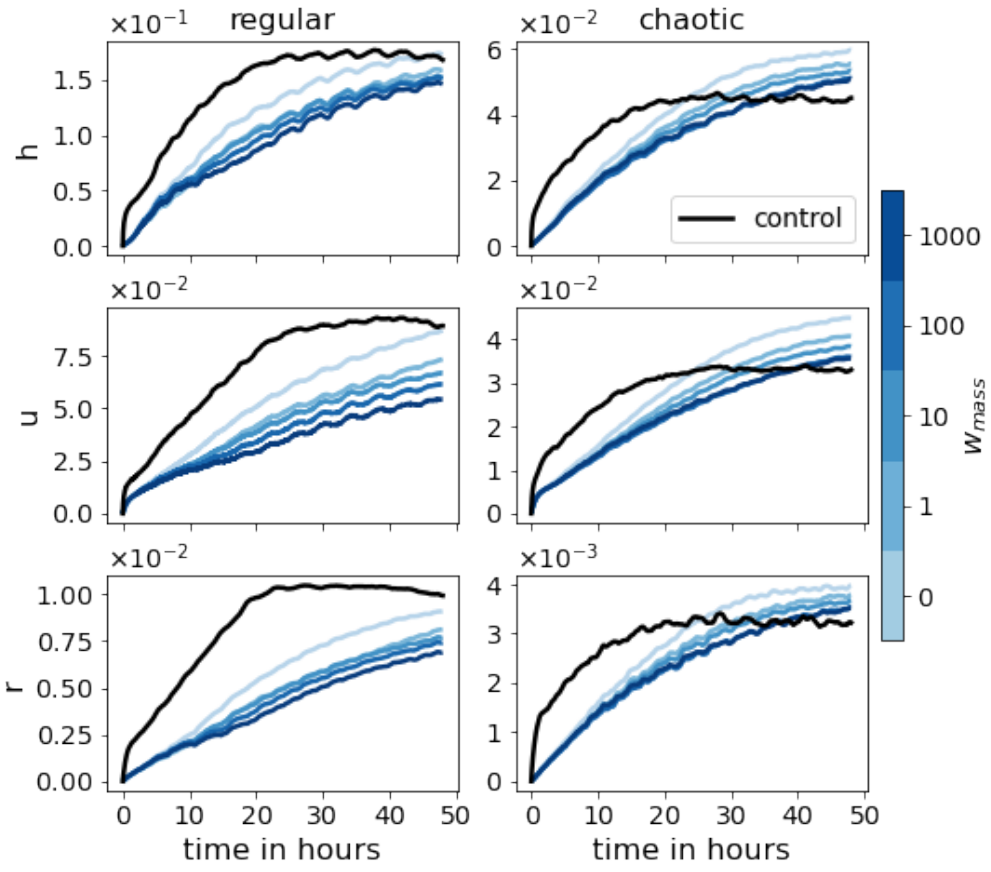

Figure 9. ~~RMSE~~ RMSE evolution of 48-hour forecasts for ~~the respective~~ model variables h, u and r (from top to bottom) of LR (black) and LR_{ANN} for the different weightings (blues)~~, averaged over 50 initial conditions and in the case of LR_{ANN} 25 ANNs~~.

error of the wind u is delayed as the weighting w_{mass} is increased. This in turn positively affects the domain mean of the rain r. To support these claims we look at Figure 11, which shows the correlation between the bias in h and the bias in wind u and rain r respectively. ~~The~~ In the single step predictions these correlations were not conclusively detected. However, as the forecast evolves, the wind bias becomes almost completely anticorrelated to the mass bias~~as the forecast lead time increases. In the single step predictions we did not detect a~~. A strong correlation between the mass bias and the rain bias ~~. Figure 11 however demonstrates that this correlation is~~ is also established after a few time steps, likely when the change in probability of crossing the rain threshold resulting from the mass bias has taken effect. We also note that the ~~smaller~~ larger w_{mass}, the ~~stronger the correlation~~weaker the correlations. We hypothesize that as the mass bias weakens, other causes for introducing domain mean biases in the wind and rain field become more significant. Such other causes may for example depend on the orography, or the state of u and r.

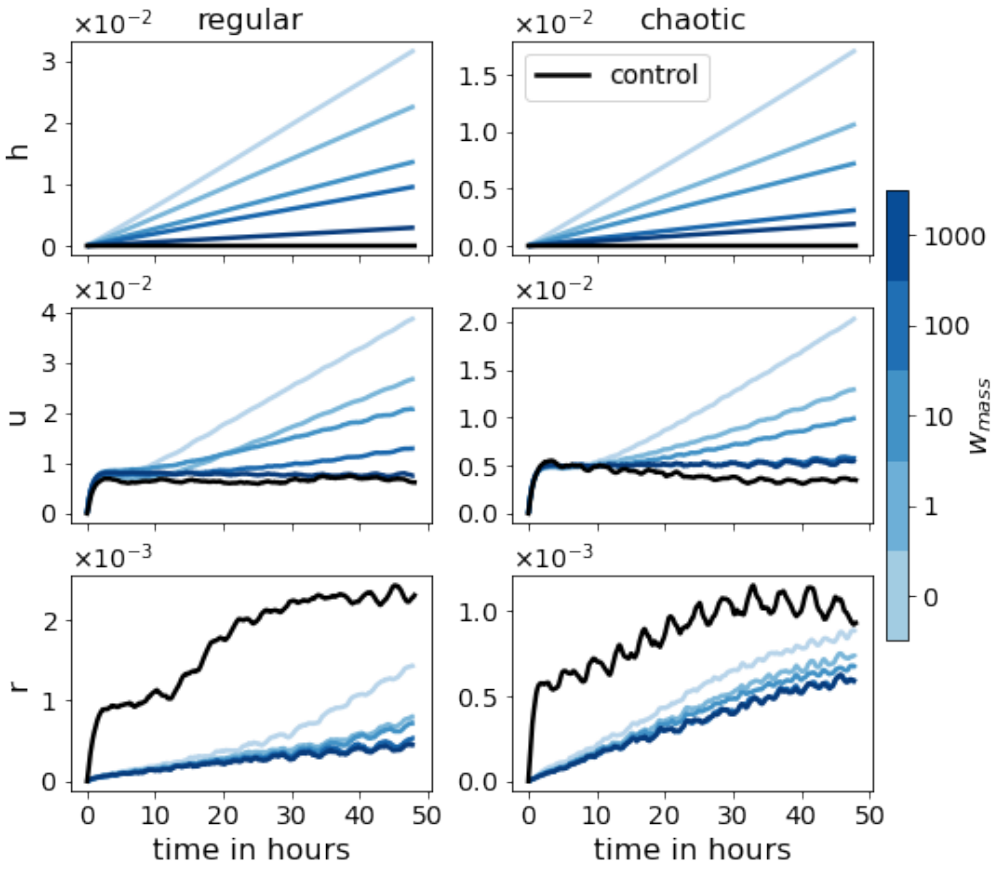

Figure 10. As Figure 9, but for the ~~spatial mean absolute error~~MSE.

Next we look at the variability of the forecast errors in ~~Figure 12. Here we look at the variability due to the initial conditions, the ANN, and the combination of both~~terms of SD_{total}, SD_{time} and SD_{ANN} in Figure 12. For small weightings the variability ~~due to the ANN seems to dominate~~ caused by ANN realizations dominates the total variability. However, as the weighting increases, the variability due to the initial conditions takes over. This again confirms the benefits of adding the mass penalty term to the loss function, as it demonstrates decrease in sensitivity of the forecast to the training process of the ANN.

~~Based on the subjective interpretation of the human brain of a hand full~~ A visual examination of animations of the forecast evolution ~~, it appears~~ suggests that convective events produced in the LR run are wider and shallower than in the coarse grained HR run~~, likely due to the smoothening of the orography. As discussed before, this results in~~ . This behavior mimics the collapse of convective clouds towards the grid length that is typical of km-scale numerical weather prediction models, as noted in the introduction. This then leads to a lack of rain mass, but also, via conservation of momentum, a drift in the wind field. The convective events in the LR simulations are therefore also increasingly misplaced as the forecast lead time increases. The ANNs are capable of sharpening the gradients of the convective events, leading to highly accurate forecasts of convective events up to

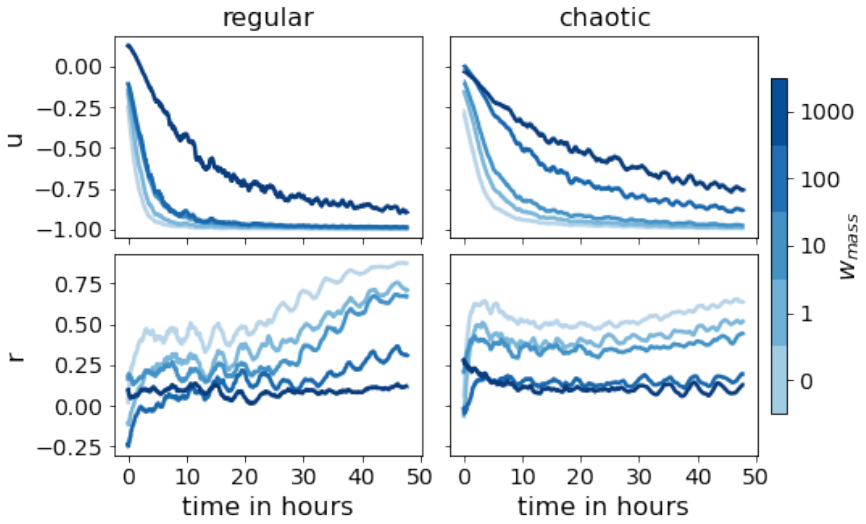

Figure 11. Correlation of the different weighting (blues) between the bias of h and the bias of u (top row) and r (bottom row) for the regular (left panel) and the chaotic ~~case~~ (right panel) ~~. See Figure 9 for the legend~~case.

280 ~~5-10~~ 6-12 hours. After this, spurious, missed and misplaced events start to occur, although the forecast skill remains significant ~~up to at least 24 hours~~a while longer, in contrast to the LR simulations, where the forecast skill dissolves after just a few hours. A snapshot of the state for the chaotic case is presented in Figure 13. ~~Here, the correlations between the 3 variables is nicely visible. Between grid points 100 and 125, the fluid height h of LR is exceeding both the convective initiation and rain threshold, in contrast to the other simulations. As a result,~~ The main rain event is misplace for LR ~~has produced spurious~~
285 ~~rain mass at this location and the wind is underestimated. Directly right of this spurious convective event in LR, the roles are exchanged: The fluid height of~~ due to the bias is the wind field. Also, LR ~~does not reach the thresholds, and therefore lacks rain and overestimates the wind speed. The same correlations are discernible between grid points 50 and 75, where the ANN with~~ misses the small neighbouring events, which the LR_{ANN}s do catch. Further, it is also clear that LR_{ANN} for $w_{mass} = 100$ is closer to the truth for all variables then for $w_{mass} = 0$~~slightly overestimates the fluid height.~~ .

290 ## 5 Conclusions

In this paper we evaluated the feasibility of using an ANN to correct for model error in the gray zone, where important features occur on scales comparable to the model resolution. The model that was used in our idealized setup mimics key aspects of convection such as conditional instability triggered by orography and resulting convective events including rain. As such, this model is representative for fully complex convective scale numerical weather prediction models and in particular the
295 corresponding errors due to unresolved scales in the gray zone. We considered two cases, each with a different realization of the orography, leading to two ~~fairly~~ different regimes. One where the convective events are large and long-lived, and one where the

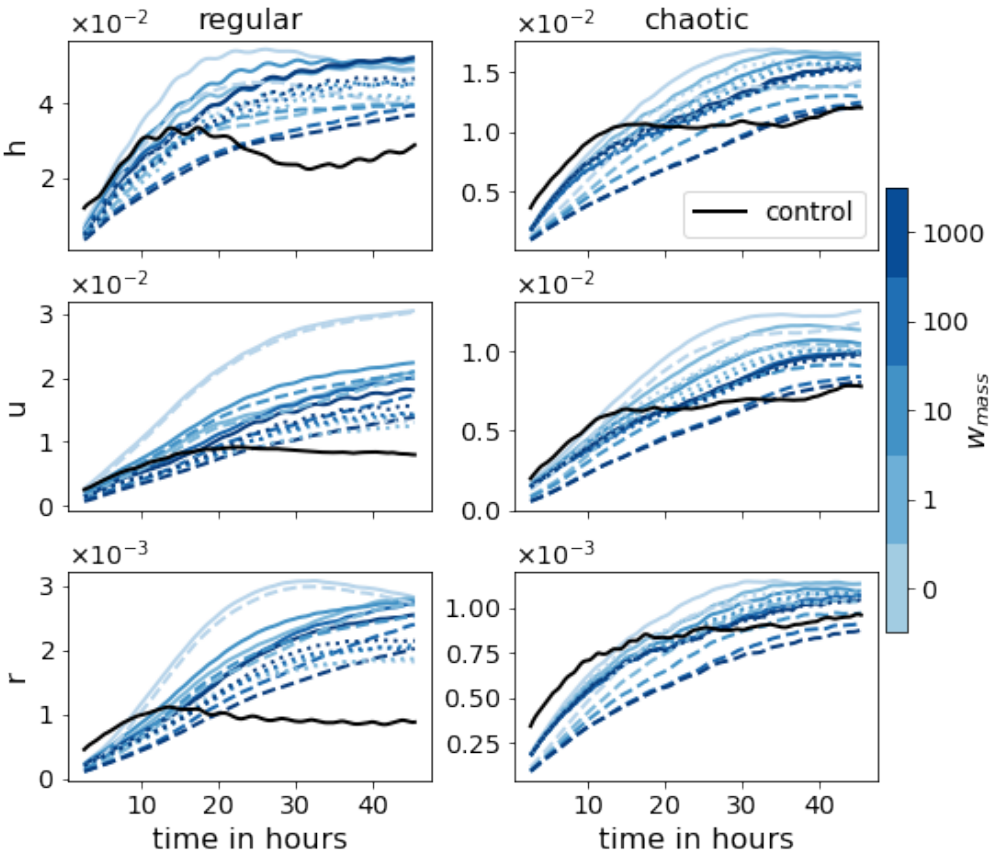

Figure 12. Evolution of ~~the standard deviation for the different weightings (blues) of the RMSE over both ANNs~~ SD_{total}, SD_{time} and ~~initial conditions~~ SD_{ANN} (solid~~)~~, ~~over ANNs only and averaged over initial conditions (~~dotted, dashed~~)~~, ~~and over initial conditions only and averaged over ANNS~~ respectively) for the different weightings (~~dotted~~blues) ~~,~~ for the regular (left panel) and the chaotic (right panel) case.~~See Figure 9 for the legend.~~

convective events are small and short-lived. We refer to the former case as regular and the latter case as chaotic. We showed that the ANNs are capable of accurately sharpening gradients where necessary in both cases to prevent the missing and flattening of convective events that is caused by the low resolution model's inability to resolve fine scales. For the regular case, the RMSE is still significantly lower than the low resolution simulation (LR) after 48 hours. For the chaotic case, the RMSE surpasses LR after about ~~30 hours~~1 day. Since the ANNs are not perfect, their errors accumulate over time, deteriorating the forecast skill. In particular, the accumulated mass error causes biases which are not present in LR, because the model conserves mass exactly. We therefore investigated the effects of adding a term to the loss function of the ANN's training process to penalize mass conservation violation. We found that reducing the mass error, reduces the biases in the wind and rain field, ~~yielding better forecasts up to 10 hours for the chaotic and~~ leading to further forecasts improvements. For the chaotic case, an additional 15 hours in forecast lead time is gained before the RMSE exceeds the LR control simulation and at least 30 hours for regular

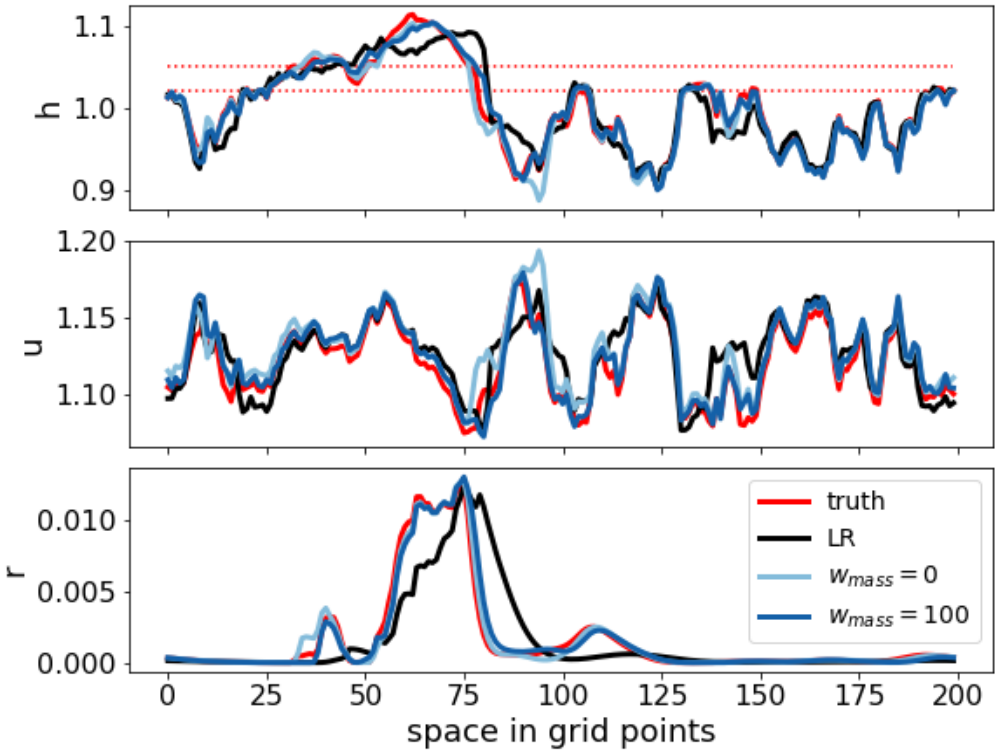

Figure 13. Snapshot of the state variables for the chaotic case of a ~~5-hour~~ 6-hour forecast starting from initial conditions of the validation data set for the truth (red), LR (black), and LR_{ANN} corresponding to weightings $w_{mass} = 0$ (light blue) and ~~$w_{mass} = 1000$~~ $w_{mass} = 100$ (dark blue). The dotted red line are the convection threshold H_c and rain threshold H_r.

case~~in terms of RMSE~~. Such positive effect of mass conservation was also found in for example Zeng et al. (2017); Ruckstuhl and Janjić (2018); Ruckstuhl et al. (2021). Furthermore, we showed that including the penalty term in the loss function reduces the sensitivity of the model forecasts to the learning process of the ANN.

While these results are encouraging, there are some issues to consider when applying this method to operational configurations. On a technical level, the generation of the training data and the training of the ANN can be costly and time consuming due to the requirement of sufficient HR data and the cumbersome exercise of tuning the ANN. The latter is a known problem that can be minimized through clever iteration of tested ANN settings, but cannot be fully avoided. Depending on the costs of generating HR data, it could be considered to use observations instead, as done by Brajard et al. (2021). They use data assimilation to generate HR data from available sparse and noisy observations. Aside from saving computational costs by replacing HR simulations with data assimilation, it might offer an advantage on a different issue as well: the effect of other model error. In contrast to what was assumed in this paper, in reality not all model error stems from unresolved scales. By using observations of the true state of the atmosphere, all model error is accounted for by the trained ANN. On the other hand, the training data contains the errors inherited from data assimilation. It is not clear which error source is more important and therefore both

approaches are worthwhile investigating. Not only to improve model forecasts, but also to gain more insight in the model error itself and its comparison to errors stemming from data assimilation.

Code availability. The provided source code (https://doi.org/10.5281/zenodo.4740252, Kriegmair et al., 2020) includes the necessary scripts to produce the data.

6

5.1

Author contributions. RK produced the source code. RK and YR ran experiments and visualized results. SR provided expertise on neural networks. GC provided expertise on convective scale dynamics. All authors contributed to the scientific design of the study, the analysis of the numerical results and the writing of the manuscript.

Competing interests. The authors declare that they have no conflict of interest.

Acknowledgements. The research leading to these results has been done within the Transregional Collaborative Research Center SFB/TRR 165 “Waves to Weather” (www.wavestoweather.de) funded by the German Research Foundation (DFG).

References

Bocquet, M., Brajard, J., Carrassi, A., and Bertino, L.: Bayesian inference of chaotic dynamics by merging data assimilation, machine learning and expectation-maximization, Foundations of Data Science, 2, 55–80, https://doi.org/10.3934/fods.2020004, 2020.

Bolton, T. and Zanna, L.: Applications of Deep Learning to Ocean Data Inference and Subgrid Parameterization, Journal of Advances in Modeling Earth Systems, 11, 376–399, https://doi.org/10.1029/2018MS001472, 2019.

Brajard, J., Carrassi, A., Bocquet, M., and Bertino, L.: Combining data assimilation and machine learning to emulate a dynamical model from sparse and noisy observations: A case study with the Lorenz 96 model, Journal of Computational Science, 44, 101 171, https://doi.org/https://doi.org/10.1016/j.jocs.2020.101171, 2020a.

Brajard, J., Carrassi, A., Bocquet, M., and Bertino, L.: Combining data assimilation and machine learning to infer unresolved scale parametrization, Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 379, 20200 086, https://doi.org/10.1098/rsta.2020.0086, 2021.

Brenowitz, N. D. and Bretherton, C. S.: Spatially Extended Tests of a Neural Network Parametrization Trained by Coarse-Graining, Journal of Advances in Modeling Earth Systems, 11, 2728–2744, https://doi.org/10.1029/2019MS001711, 2019.

Brunton, S. L., Proctor, J. L., and Kutz, J. N.: Discovering governing equations from data by sparse identification of nonlinear dynamical systems, Proceedings of the National Academy of Sciences, 113, 3932–3937, https://doi.org/10.1073/pnas.1517384113, 2016.

Bryan, G. H., Wyngaard, J. C., and Fritsch, J. M.: Resolution Requirements for the Simulation of Deep Moist Convection, Monthly Weather Review, 131, 2394 – 2416, https://doi.org/10.1175/1520-0493(2003)131<2394:RRFTSO>2.0.CO;2, 2003.

Chollet, F. et al.: Keras, https://keras.io, 2015.

Chow, F. K., Schär, C., Ban, N., Lundquist, K. A., Schlemmer, L., and Shi, X.: Crossing Multiple Gray Zones in the Transition from Mesoscale to Microscale Simulation over Complex Terrain, Atmosphere, 10, https://doi.org/10.3390/atmos10050274, 2019.

Dueben, P. D. and Bauer, P.: Challenges and design choices for global weather and climate models based on machine learning, Geoscientific Model Development, 11, 3999–4009, https://doi.org/10.5194/gmd-11-3999-2018, 2018.

Fablet, R., Ouala, S., and Herzet, C.: Bilinear Residual Neural Network for the Identification and Forecasting of Geophysical Dynamics, in: 2018 26th European Signal Processing Conference (EUSIPCO), pp. 1477–1481, https://doi.org/10.23919/EUSIPCO.2018.8553492, 2018.

Faranda, D., Lembo, V., Iyer, M., Kuzzay, D., Chibbaro, S., Daviaud, F., and Dubrulle, B.: Computation and Characterization of Local Subfilter-Scale Energy Transfers in Atmospheric Flows, Journal of the Atmospheric Sciences, 75, 2175 – 2186, https://doi.org/10.1175/JAS-D-17-0114.1, 2018.

Faranda, D., Vrac, M., Yiou, P., Pons, F. M. E., Hamid, A., Carella, G., Ngoungue Langue, C., Thao, S., and Gautard, V.: Boosting performance in machine learning of geophysical flows via scale separation, Nonlinear Processes in Geophysics Discussions, 2020, 1–30, https://doi.org/10.5194/npg-2020-39, 2020.

Goodfellow, I., Bengio, Y., and Courville, A.: Deep Learning, MIT Press, http://www.deeplearningbook.org, 2016.

Honnert, R., Efstathiou, G. A., Beare, R. J., Ito, J., Lock, A., Neggers, R., Plant, R. S., Shin, H. H., Tomassini, L., and Zhou, B.: The Atmospheric Boundary Layer and the "Gray Zone" of Turbulence: A Critical Review, Journal of Geophysical Research: Atmospheres, 125, e2019JD030 317, https://doi.org/https://doi.org/10.1029/2019JD030317, e2019JD030317 10.1029/2019JD030317, 2020.

Jeworrek, J., West, G., and Stull, R.: Evaluation of Cumulus and Microphysics Parameterizations in WRF across the Convective Gray Zone, Weather and Forecasting, 34, 1097 – 1115, https://doi.org/10.1175/WAF-D-18-0178.1, 2019.

Jones, T. R. and Randall, D. A.: Quantifying the limits of convective parameterizations, Journal of Geophysical Research: Atmospheres, 116, https://doi.org/10.1029/2010JD014913, 2011.

Kent, T., Bokhove, O., and Tobias, S.: Dynamics of an idealized fluid model for investigating convective-scale data assimilation, Tellus A: Dynamic Meteorology and Oceanography, 69, 1369 332, https://doi.org/10.1080/16000870.2017.1369332, 2017.

Krasnopolsky, V. M., Fox-Rabinovitz, M. S., and Belochitski, A. A.: 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, http://dx.doi.org/10.1155/2013/485913, 2013.

Lovejoy, S. and Schertzer, D.: Towards a new synthesis for atmospheric dynamics: Space–time cascades, Atmospheric Research, 96, 1–52, https://doi.org/https://doi.org/10.1016/j.atmosres.2010.01.004, 2010.

Marino, R., Mininni, P. D., Rosenberg, D., and Pouquet, A.: Inverse cascades in rotating stratified turbulence: Fast growth of large scales, EPL (Europhysics Letters), 102, 44 006, https://doi.org/10.1209/0295-5075/102/44006, 2013.

O'Gorman, P. A. and Dwyer, J. G.: Using Machine Learning to Parameterize Moist Convection: Potential for Modeling of Climate, Climate Change, and Extreme Events, Journal of Advances in Modeling Earth Systems, 10, 2548–2563, https://doi.org/https://doi.org/10.1029/2018MS001351, 2018.

Pathak, J., Hunt, B., Girvan, M., Lu, Z., and Ott, E.: Model-Free Prediction of Large Spatiotemporally Chaotic Systems from Data: A Reservoir Computing Approach, Phys. Rev. Lett., 120, 024 102, https://doi.org/10.1103/PhysRevLett.120.024102, 2018.

Randall, D., Khairoutdinov, M., Arakawa, A., and Grabowski, W.: Breaking the Cloud Parameterization Deadlock, Bulletin of the American Meteorological Society, 84, 1547–1564, https://doi.org/10.1175/BAMS-84-11-1547, 2003.

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

Rasp, S.: Coupled online learning as a way to tackle instabilities and biases in neural network parameterizations: general algorithms and Lorenz 96 case study (v1.0), Geoscientific Model Development, 13, 2185–2196, https://doi.org/10.5194/gmd-13-2185-2020, 2020.

Rasp, S., Pritchard, M. S., and Gentine, P.: Deep learning to represent subgrid processes in climate models, Proceedings of the National Academy of Sciences, 115, 9684–9689, https://doi.org/10.1073/pnas.1810286115, 2018.

Ruckstuhl, Y. and Janjić, T.: Parameter and state estimation with ensemble Kalman filter based algorithms for convective-scale applications, Quart .J. Roy. Meteorol. Soc, 144, 826–841, https://doi.org/10.1002/qj.3257, 2018.

Ruckstuhl, Y., Janjić, T., and Rasp, S.: Training a convolutional neural network to conserve mass in data assimilation, Nonlinear Processes in Geophysics, 28, 111–119, https://doi.org/10.5194/npg-28-111-2021, 2021.

Scher, S.: Toward Data-Driven Weather and Climate Forecasting: Approximating a Simple General Circulation Model With Deep Learning, Geophysical Research Letters, 45, 12,616–12,622, https://doi.org/https://doi.org/10.1029/2018GL080704, 2018.

Stensrud, D.: Parameterization Schemes: Keys to Understanding Numerical Weather Prediction Models, Cambridge University Press, https://books.google.de/books?id=kkZ0AgAAQBAJ, 2009.

Wagner, A., Heinzeller, D., Wagner, S., Rummler, T., and Kunstmann, H.: Explicit Convection and Scale-Aware Cumulus Parameterizations: High-Resolution Simulations over Areas of Different Topography in Germany, Monthly Weather Review, 146, 1925 – 1944, https://doi.org/10.1175/MWR-D-17-0238.1, 2018.

Würsch, M. and Craig, G. C.: A simple dynamical model of cumulus convection for data assimilation research, Meteorologische Zeitschrift, 23, 483–490, https://doi.org/10.1127/0941-2948/2014/0492, 2014.

Yuval, J. and O'Gorman, P. A.: Stable machine-learning parameterization of subgrid processes for climate modeling at a range of resolutions, Nature communications, 11, 1–10, 2020.

Zeng, Y., Janjić, T., Ruckstuhl, Y., and Verlaan, M.: Ensemble-type Kalman filter algorithm conserving mass, total energy and enstrophy, Quarterly Journal of the Royal Meteorological Society, 143, 2902–2914, https://doi.org/10.1002/qj.3142, 2017.