Atmospheric
Chemistry
and Physics

Technical note: Deep learning for creating surrogate models of precipitation in Earth system models

Theodore Weber[1], Austin Corotan[1], Brian Hutchinson[1,2], Ben Kravitz[3,4], and Robert Link[5]

[1]Computer Science Department, Western Washington University, Bellingham, WA, USA
[2]Computing and Analytics Division, Pacific Northwest National Laboratory, Seattle, WA, USA
[3]Department of Earth and Atmospheric Sciences, Indiana University, Bloomington, IN, USA
[4]Atmospheric Sciences and Global Change Division, Pacific Northwest National Laboratory, Richland, WA, USA
[5]Joint Global Change Research Institute, Pacific Northwest National Laboratory, College Park, MD, USA

Correspondence: Brian Hutchinson (brian.hutchinson@wwu.edu)

Abstract. We investigate techniques for using deep neural networks to produce surrogate models for short-term climate forecasts. A convolutional neural network is trained on 97 years of monthly precipitation output from the 1pctCO2 run (the CO_2 concentration increases by 1 % per year) simulated by the second-generation Canadian Earth System Model (CanESM2). The neural network clearly outperforms a persistence forecast and does not show substantially degraded performance even when the forecast length is extended to 120 months. The model is prone to underpredicting precipitation in areas characterized by intense precipitation events. Scheduled sampling (forcing the model to gradually use its own past predictions rather than ground truth) is essential for avoiding amplification of early forecasting errors. However, the use of scheduled sampling also necessitates preforecasting (generating forecasts prior to the first forecast date) to obtain adequate performance for the first few prediction time steps. We document the training procedures and hyperparameter optimization process for researchers who wish to extend the use of neural networks in developing surrogate models.

1 Introduction

Climate prediction is a cornerstone in numerous scientific investigations and decision-making processes (e.g., Stocker et al., 2013; Jay et al., 2018). In the long term (decades to centuries), different possible climate outcomes pose very different hazards, risks, and societal challenges, such as build-ing and maintaining infrastructure (e.g., Moss et al., 2017). On decadal timescales, predictability of major modes of variability (like the El Niño–Southern Oscillation) is an important driver of extreme events, such as flooding and drought (e.g., Yeh et al., 2018). On similar timescales, disappearing Arctic sea ice has been implicated in changes in midlatitude winter storm patterns (Cohen et al., 2014). On shorter timescales (weeks to months), also sometimes called the subseasonal-to-seasonal (S2S) regime, climate forecasts can be critical for agriculture, water resource management, flooding and drought mitigation, and military force mobilization (Robertson et al., 2015).

Improvements in climate predictability in some of these regimes are slow to be realized. Decadal predictability studies have found that predictability skill is greatly influenced by proper initialization of hindcasts to ensure that any modeled changes due to internal variability are in phase with observations (Bellucci et al., 2015). This highlights the importance of climate memory in predictive skill in that the response to processes can be lagged and the responses themselves can depend upon the model state. Yuan et al. (2019) found the existence of processes with a relatively high portion of response that can be explained by memory on all timescales, from monthly through multidecadal lengths. As an example, Guemas et al. (2013) found that properly initialized hindcasts were able to predict the global warming slowdown of the early 2000s (Fyfe et al., 2016) up to 5 years ahead.

The best technique for climate prediction is to run an Earth system model (ESM), as these models capture the state of

the art in our knowledge of climate dynamics. These models, however, are difficult and costly to run, and for many researchers access to ESM output is limited to a handful of runs deposited into public archives. Therefore, there has been a great deal of interest in *surrogate models* that can produce results similar to ESMs but are more accessible to the broader research community due to ease of use and lower computational expense. These surrogates are advantageous for numerous applications, such as exploring wide ranges of scenarios; such efforts are quite costly in ESMs.

Building surrogates of ESMs can take numerous forms. The most basic is pattern scaling (Santer et al., 1990), involving scaling a time-invariant pattern of change by global mean temperature (e.g., Mitchell, 2003; Lynch et al., 2017). Other methods include statistical emulation based on a set of precomputed ESM runs (Castruccio et al., 2014), linear time-invariant approaches (MacMartin and Kravitz, 2016), or dimension reduction via empirical orthogonal functions (e.g., Herger et al., 2015). While all of these methods have shown some degree of success, they inherently neither incorporate information about the internal model state (Goddard et al., 2013) nor capture highly nonlinear behavior.

Various methods of incorporating state information into surrogate models have been studied. Lean and Rind (2009) explored using a linear combination of autoregressive processes with different lag timescales in explaining global mean temperature change. However, such reduced-order modeling approaches, which explicitly capture certain physical processes, will invariably have limited structure; this can result in inaccurate predictions when evaluating variables like precipitation, which have fine temporal and spatial features, important nonlinear components in their responses to forcing, and decidedly non-normal distributions of intensity and frequency. Many studies have focused on initializing the internal model state of Earth system models (or models of similar complexity) to capture low frequency variability; this has been found to add additional skill beyond external forcing alone (Goddard et al., 2013). However, the required computational time to create a decadal prediction ensemble is rather expensive.

Machine learning methods offer the possibility of overcoming these limitations without imposing an insurmountable computational burden. The field of machine learning studies algorithms that learn patterns (e.g., regression or classification) from data. The subfield of deep learning focuses on models, typically neural-network variants, that involve multiple layers of nonlinear transformation of the input to generate predictions. Although many of the core deep-learning techniques were developed decades ago, work in deep learning has recently exploded, achieving state-of-the-art results on a wide range of prediction tasks. This progress has been fueled by increases in both computing power and available training data. While training deep-learning models is computationally intensive, trained models can make accurate predictions quickly (often in a fraction of a second).

The use of deep learning in climate science is relatively new. Most of the applications have used convolutional neural networks (see Sect. 3 for further definitions and details) to detect and track features, including extreme events (tropical cyclones, atmospheric rivers, and weather fronts; e.g., Liu et al., 2016; Pradhan et al., 2018; Deo et al., 2017; Hong et al., 2017) or cloud types (Miller et al., 2018). Other promising applications of deep learning are to build new parameterizations of multiscale processes in models (Jiang et al., 2018; Rasp et al., 2018) or to build surrogates of entire model components (Lu and Ricciuto, 2019). More generally, McDermott and Wikle (2018) have explored using deep neural networks in nonlinear dynamical spatio-temporal environmental models, of which climate models are an example, and Ouyang and Lu (2018) applied *echo state networks* to monthly rainfall prediction. These studies have clearly demonstrated the power that deep learning can bring to climate science research and the new insights it can provide. However, to the best of our knowledge, there have been few attempts to assess the ability of deep learning to improve predictability, in particular the ability to incorporate short-term memory. Several deep-learning architectures (described below) are particularly well suited for this sort of application.

Here we explore techniques for using deep learning to produce accurate surrogate models for predicting the forced response of precipitation in Earth system models. Key to this approach is training the model on past precipitation data (as described later, a sliding 5-year window), allowing it to capture relevant information about the state space that may be important for predictability in the presence of important modes of variability. The surrogate models will be trained on climate model output of precipitation under a scenario of increasing CO_2 concentration and then used to project precipitation outcomes into a period beyond the training period. That forecast will then be compared to the actual climate model output for the same period to quantify performance. Several model designs will be compared to evaluate the effectiveness of various deep-learning techniques in this application. The performance of the deep-learning surrogates will be compared to other methods of forecasting, such as persistence and autoregressive processes (described later). A key area of investigation will be the prediction horizon (how far out is the predictive skill of the surrogate model better than naive extrapolation methods) for the chosen window size (how much past information is used to condition the surrogate's predictions).

2 Study description

The dataset used for this study was idealized precipitation output from the second-generation Canadian Earth System Model (CanESM2; Arora et al., 2011). The atmospheric model has a horizontal resolution of approximately 2.8° with 35 vertical layers, extending up to 1 hPa. The atmosphere

is fully coupled to the land surface model CLASS (Arora and Boer, 2011) and an ocean model with approximately 1° horizontal resolution. The model output used corresponds to the 1pctCO2 simulation, in which, starting from the preindustrial era, the carbon dioxide concentration increases by 1 % per year for 140 years to approximately quadruple the original concentration. This idealized simulation was chosen to reduce potential complicating factors resulting from precipitation responses to multiple forcings (carbon dioxide, methane, sulfate aerosols, black-carbon aerosols, dust, etc.) that might occur under more comprehensive scenarios, such as the Representative Concentration Pathways (van Vuuren et al., 2011). For this study, only monthly average precipitation output was used; results for daily average precipitation are the subject of future work.

We divided the model output into three time periods. The *training* set consists of the period 1850–1947 and is used to train the surrogate model. The *development* set (sometimes called the validation set) consists of the period 1948–1968 and is used to evaluate the performance of the trained surrogate model to guide further tuning of the model's hyperparameters (i.e., configurations external to the model that are not estimated during training). The *test* set consists of the period 1969–1989 and is used only in computing the end results, which are reported below in Sect. 4.

3 Deep learning methodologies for improving predictability

Deep learning is a subfield of machine learning that has achieved widespread success in the past decade in numerous science and technology tasks, including speech recognition (Hinton et al., 2012; Chan et al., 2016), image classification (Krizhevsky et al., 2012; Simonyan and Zisserman, 2014; He et al., 2016), and drug design (Gawehn et al., 2016). Its success is often attributed to a few key characteristics. First, rather than operating on predetermined (by humans) features, deep learning is typically applied to raw inputs (e.g., pixels), and all of the processing of those inputs is handled in several steps (*layers*) of nonlinear transformation; the addition of these layers increases the *depth* of the network. This allows the model to learn to extract discriminative, nonlinear features for the task at hand. Second, all stages of the deep-learning models, including the training objectives (defined by a loss function), are designed to ensure differentiability with respect to model parameters, allowing models to be trained efficiently with stochastic gradient descent techniques. With enough training data and computational power, deep-learning models can learn complex, highly nonlinear input–output mappings, such as those found in Earth system models.

3.1 Architectures

In this work we consider convolutional neural networks (CNNs; LeCun et al., 1998) to model the spatial precipitation patterns over time. CNNs are able to process data with a known grid-like topology and have been demonstrated as effective models for understanding image content (Krizhevsky et al., 2012; Karpathy et al., 2014; He et al., 2016). Unlike standard fully connected neural networks, CNNs employ a convolution operation in place of a general matrix multiplication in at least one of their layers. In a convolutional layer, an $m \times n \times p$ input tensor is convolved with a set of k $i \times j \times p$ kernels to output k feature maps that serve as inputs for the next layer. In this setting, m and n correspond to the width and height of the input, and p corresponds to the depth (i.e., number of channels). Similarly, i and j correspond to the width and height of the kernel, and p corresponds to the depth, which is equal to that of the input tensor. In practice we choose a kernel where $i \ll m$ and $j \ll n$ (e.g., $i = j = 3$). By using a small kernel, we limit the number of parameters required by the model while maintaining the ability to detect small, meaningful features in a large and complex input space. This both reduces memory requirements of the model and improves statistical efficiency (Krizhevsky et al., 2012; Goodfellow et al., 2016). The learning capacity of the model can also be adjusted by varying the width (i.e., number of kernels) and the depth (i.e., number of layers) of the network.

The spatial structure of each time step of the precipitation field bears a resemblance to the structure of image data, making CNNs a promising candidate model. However, to produce accurate forecasts, the model must also incorporate temporal evolution of the precipitation field. To address long-term and short-term trends we implement a sliding-window approach where our input tensor is built using precipitation outcomes from the most recent K time steps as input channels. Our model predicts the global precipitation outcome at next time step. Then this procedure is iterated, using output from the previous model forecast as input into the next one, allowing for arbitrarily long prediction horizons.

We consider adding depth to our network because many studies in image classification have achieved leading results using very deep models (Simonyan and Zisserman, 2014; Szegedy et al., 2015; He et al., 2015). Many of the fields of interest in climate science are effectively images, making image classification an appropriate analogue for our aims. In deep residual networks (He et al., 2016), rather than train each i layer to directly produce a new hidden representation $h_{(i)}$ given the hidden representation $h_{(i-1)}$ it was provided, we instead train the layer to produce a residual map $f_{(i)} = h_{(i)} - h_{(i-1)}$, which is then summed with the $h_{(i-1)}$ to give the output for the layer. This way, each layer explicitly refines the previous one. The residual modeling approach is motivated by the idea that it is easier to optimize residual mapping than to optimize the original, unreferenced mapping. Architecturally, outputting $f_{(i)} + h_{(i-1)}$ at each layer is

accomplished by introducing *shortcut* connections (e.g., He et al., 2016) that skip one or more layers in the network. In our CNN, a shortcut connection spans every few consecutive layers. This identity mapping is then summed with the residual mapping $f_{(i)}$ produced by the series of stacked layers encompassed by the shortcut connection. While developing our model, we also explored using plain CNNs (without shortcut connections). The best-performing model is described in the following section. All of the neural-network architectures considered in this study are summarized in Table 1.

3.2 Implementation

Our residual network implementation follows the work in He et al. (2016). We construct a 18-layer residual network with shortcut connections encompassing every two consecutive convolutional layers (see Fig. 1). Each convolutional layer uses a 3×3 kernel with a stride of 1, and zero padding is applied to preserve the spatial dimensionality of the input throughout the network. Directly after the convolution we apply batch normalization following practices in Ioffe and Szegedy (2015) and then the rectified linear unit (ReLU) (Nair and Hinton, 2010) as the nonlinear activation function. Every four layers we double the number of kernels to increase the learning capacity of our model. When this occurs we zero pad our identity mapping in the shortcut connection to match dimensions.

The total number of parameters in the 18-layer deep neural network is 34 578, relatively small compared to the size of each input: $60 \cdot 128 \cdot 64 = 491\,520$. The length of the training set is 1116 time steps. For a fixed set of hyperparameters, training each model takes 1–3 h (depending upon the number of training epochs) on a single NVIDIA 980 Ti GPU (graphics processing unit). The computational cost is effectively all in the training; predictions take a matter of seconds, even on CPU (central processing unit) architectures. For comparison, a 5-year simulation in an ESM can take roughly 1 d even on hundreds of CPUs.

We chose to initialize our network parameters by drawing from a Gaussian distribution as suggested in He et al. (2015). We use stochastic gradient descent to optimize our loss function $L(\theta)$, which is defined as the mean squared area-weighted difference, calculated as

$$L(\theta) = \frac{1}{Z} \sum_x \left[\left(B(x) - \hat{B}(x; \theta) \right) \cdot A(x) \right]^2, \qquad (1)$$

where x iterates over the Z distinct spatial positions, $B(x)$ is the ground truth outcome, $\hat{B}(x; \theta)$ is the CNN reconstruction (CNN with parameters θ), and $A(x)$ is the cosine of latitude (for area weighting).

Our plain CNN baselines follow a similar structure as the 18-layer residual network, but all shortcut connections are removed (see Fig. 1). By evaluating the performance of the 18-layer plain network, we investigate the benefits of using shortcut connections. We also experiment with a 5-layer
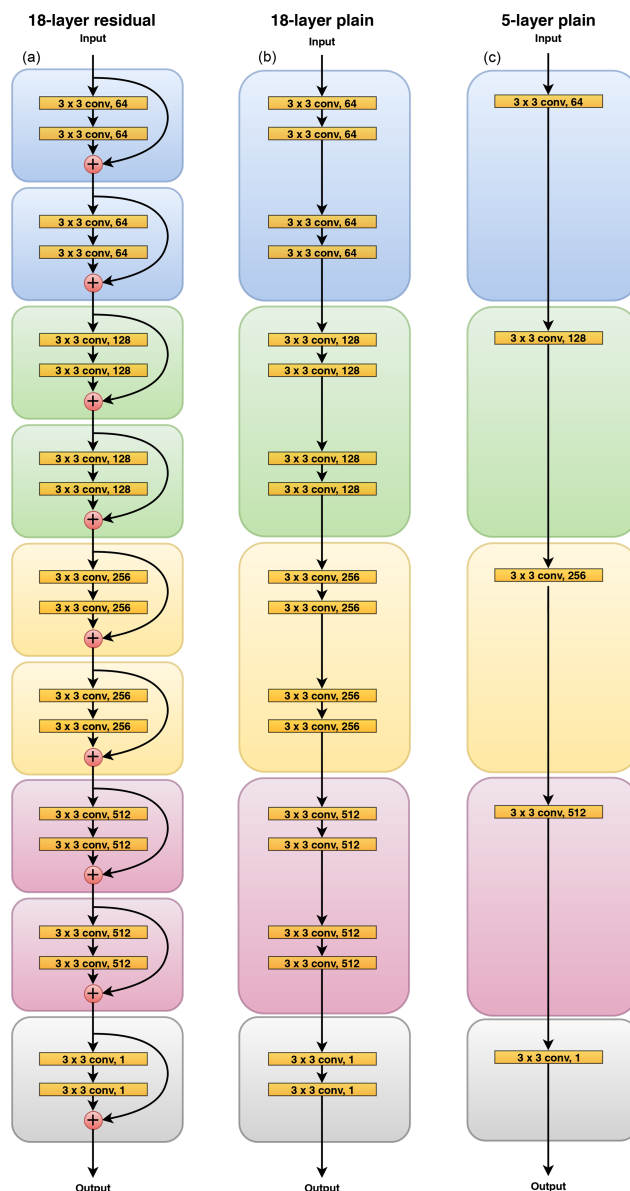
Figure 1. Deep architectures for precipitation forecasting: **(a)** 18-layer residual network, **(b)** 18-layer plain network (no shortcut connections), and **(c)** 5-layer plain network (no shortcut connections).

plain network to understand how depth affects the performance and trainability of our models. Finally, we train the 18-layer residual network without scheduled sampling (described below) to determine if this training approach actually improves forecasting ability.

3.3 Training

The distribution of our training data (aggregated over the entire $m \cdot n \cdot p$ space) was heavy-tailed and positively skewed (Fisher kurtosis ≈ 11.3; Fischer–Pearson coefficient of skewness ≈ 2.72). Performance of deep architectures tends to im-

Table 1. Description of all of the neural-network architectures used in this study, including the number of layers, whether the network is a residual or plain network (plain networks do not have shortcut connections), whether scheduled sampling was used, whether preforecasting was used, and the figures of the paper in which each network is used. In addition, these results are compared with persistence forecasting and an autoregressive model. Table A1 provides information about the hyperparameters used for each configuration, and Table A2 describes the hyperparameter space used in model training.

Name	No. layers	Residual?	Sched. samp.?	Preforecasting?	Figures
18-layer residual	18	Residual	Yes	Yes	1, 2, 3, 4, 5, 6
18-layer residual (no scheduled sampling)	18	Residual	No	Yes	4, 5
18-layer residual (no preforecasting)	18	Residual	Yes	No	6
18-layer plain	18	Plain	Yes	Yes	1, 4, 5
5-layer plain	5	Plain	Yes	Yes	1, 4, 5

prove when training with Gaussian-like input features (Bengio, 2012), so we apply a log transformation on our dataset to reduce skewness. In addition, we scale our input values between -1 and 1, bringing the mean over the training set closer to 0. Scaling has been shown to balance out the rate at which parameters connected to the inputs nodes learn, and having a mean closer to zero tends to speed up learning by reducing the bias to update parameters in a particular direction (LeCun et al., 2012).

Our model makes fixed-window forecasts: it requires K previous (ground truth or predicted) precipitation outcomes to generate a forecast for the subsequent time step. During training, since ground truth is available for all time steps, one typically feeds in ground truth data for all K input time steps. After training, when used in the inference mode on new data without ground truth, some or all of the K inputs will need to be previous model predictions. Without care, this mismatch can lead to poor extrapolation: the model is not used to consuming its own predictions as input, and any mistakes made in early forecasts will be amplified in later forecasts. Scheduled sampling (Bengio et al., 2015) alleviates this issue by gradually forcing the model to use its own outputs at training time (despite the availability of ground truth). This is realized by a sampling mechanism that randomly chooses to use the ground truth outcome with a probability ϵ, or the model-generated outcome with probability $1-\epsilon$, when constructing its input. In other words, if $\epsilon = 1$ the model always uses the ground truth outcome from the previous time step, and when $\epsilon = 0$ the model is trained in the same setting as inference (prediction). As we continue to train the model we gradually decrease ϵ from 1 to 0 according to a linear decay function. Practically speaking, this has the effect of explicitly degrading our initial states at training time; we discuss the implications for our results below. To improve the forecasting ability of our models, we employed scheduled sampling during training. Scheduled sampling requires a predetermined number of epochs (to decay properly). For our results that do not use scheduled sampling, we use early stopping (Yao et al., 2007) as a regularization technique.

Each model has its own set of hyperparameters with the exception of window size, as modifying window size would result in each model being conditioned on different numbers of priors, making them difficult to compare fairly. The most significant hyperparameters in our models are the learning rate, input depth, and number of training epochs. For each model, we tuned these hyperparameters using a random search (Bergstra and Bengio, 2012) for 60 iterations each. Our best residual network used a learning rate of ~ 0.07, has a window size of 60, and was trained for 90 epochs with scheduled sampling. (See Appendix A for a discussion of window size.) We trained our baseline CNNs using the same window size and similar ranges for the learning rate and number of training epochs in an attempt to generate comparable models. Each model was trained on a single GPU.
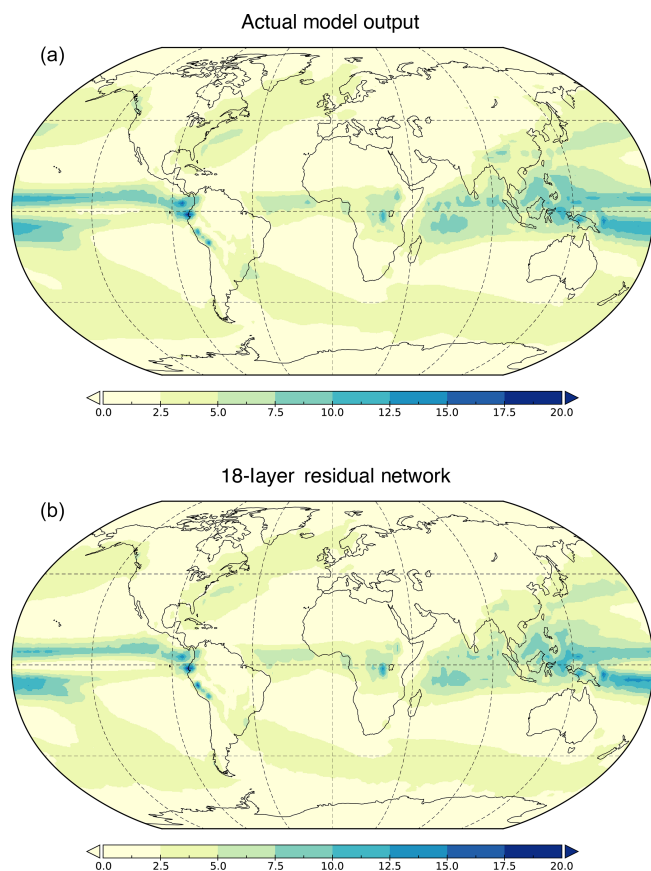
4 Predictability and performance

We evaluate the forecasting ability of our models using the CanESM2 output over the period 1969–1989 as ground truth. We also compare the performance of our best models to two naive methods of forecasting. The lowest bar, persistence forecasting, extrapolates the most recent precipitation outcome (i.e., the last outcome in the development set), assuming that the climatology remains unchanged over time. In perhaps a closer comparison, we also use a first-order autoregressive model in each grid box, abbreviated AR(1), in which the most recent precipitation outcome depends upon a linear combination of the previous value and a stochastic (white noise) term:

$$\hat{B}(x;\theta,i) = c + \phi\hat{B}(x;\theta,i-1) + \epsilon, \tag{2}$$

where $\hat{B}(x;\theta,i)$ is the surrogate prediction at time i, c is a constant, ϵ represents white noise, and ϕ is a parameter that is tuned or trained such that the model has accurate predictions over the development set.

To quantify our generalization error, we compute the root-mean square over the area-weighted difference $\hat{B} - B$ for six different spatial regions – polar, midlatitude, and tropics over both land and ocean. This is calculated as

$$\text{RMSE} = \sqrt{\frac{\sum_x [(\hat{B}(x;\theta) - B(x)) \cdot A(x) \cdot M_k(x)]^2}{\sum_x [A(x) \cdot M_k(x)]^2}}, \tag{3}$$

Actual model output

(a)

18-layer residual network

(b)

Figure 2. Precipitation outcomes (mm d^{-1}) for the period 1969–1989. Top shows the average output of the CanESM2 over that period. Bottom shows the average output of a 252-month forecast over the same time period using the 18-layer residual network with a window size of 60. Both models show qualitatively similar features.

where x iterates over the spatial positions, $\hat{B}(x;\theta)$ is the surrogate prediction, $B(x)$ is the ground truth outcome, $A(x)$ is the cosine of latitude weights, and $M_k(x)$ is the region mask (for $k \in \{1, 2, \ldots, 6\}$).

In addition to the root-mean-square error (RMSE), we also compute the anomaly correlation coefficient (ACC), a commonly used metric for quantifying differences in spatial fields in forecasting (Joliffe and Stephenson, 2003). The ACC is defined as (JMA, 2019)

$$ \mathrm{ACC} = \frac{\sum_{i=1}^{n}(f_i - \bar{f})(a_i - \bar{a})}{\sqrt{\sum_{i=1}^{n}(f_i - \bar{f})^2 \sum_{i=1}^{n}(a_i - \bar{a})^2}}, \qquad (4) $$

where n is the number of samples, f_i is the difference between forecast and reference, and a_i is the difference between some verifying value and the reference. We use the average precipitation over the period 1938–1968 as our reference (the 30 years preceding the test set period). $\bar{f}$ and $\bar{a}$ indicate area-weighted averages over the number of samples. ACC can take values in $[-1, 1]$, where an ACC of 1 indicates that the anomalies of the forecast match the anomalies of the

Precipitation difference

(a)

Percent error

(b)
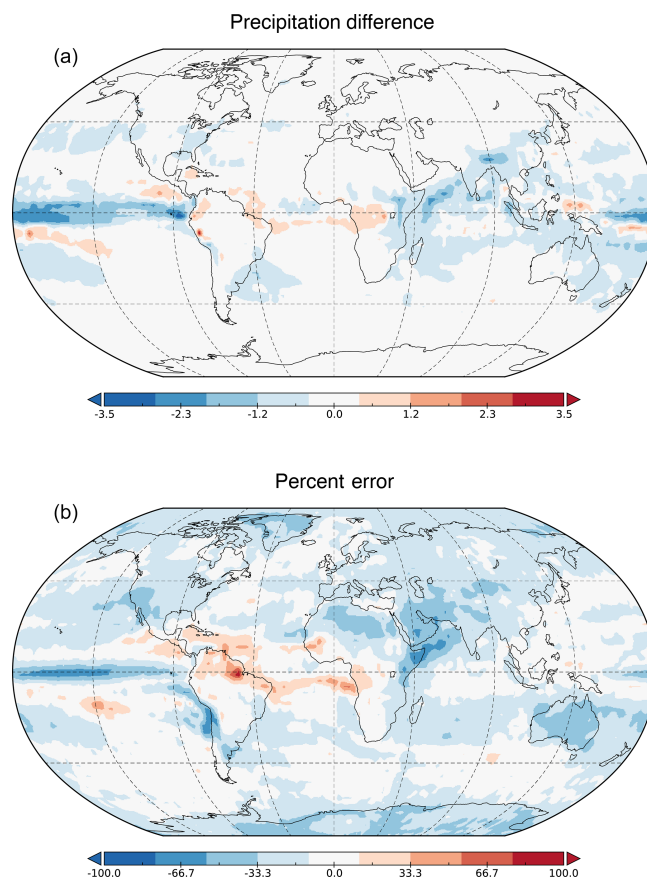
Figure 3. Comparison of the average precipitation outputs (mm d^{-1}) of the CanESM2 (B) and the 18-layer residual network ($\hat{B}$) using a window size of 60 over the years 1969–1989. Top shows $\hat{B} - B$. Bottom shows the percent error between $\hat{B}$ and B. The residual network tends to underpredict near the Equator, midlatitude storm tracks, and areas associated with monsoon precipitation.

verifying value, and an ACC of -1 indicates a reversal of the variation pattern. Figure 5 shows ACC values for the different models considered in this study. The message is similar to that of Fig. 4, with the 18-layer residual network showing the greatest skill (ACC exceeding 0.5 in all six regions), the other neural networks showing little skill, and the persistence forecast showing variable skill, depending upon the forecast length. Although it is difficult to make exact quantitative comparisons, the 18-layer residual network has higher values of ACC than the Community Earth System Model Decadal Prediction Large Ensemble (CESM-DPLE) in all six regions (Yeager et al., 2018). Performance is similar over the Sahel, indicating some ability of the residual network to capture relevant precipitation dynamics.

Figure 2 shows the average precipitation for a 252-month forecast over the period 1969–1989 in the 18-layer residual network (CNN with lowest forecasting error) and the average precipitation over the same period in the CanESM2 under a 1pctCO2 simulation. Both models show qualitatively simi-
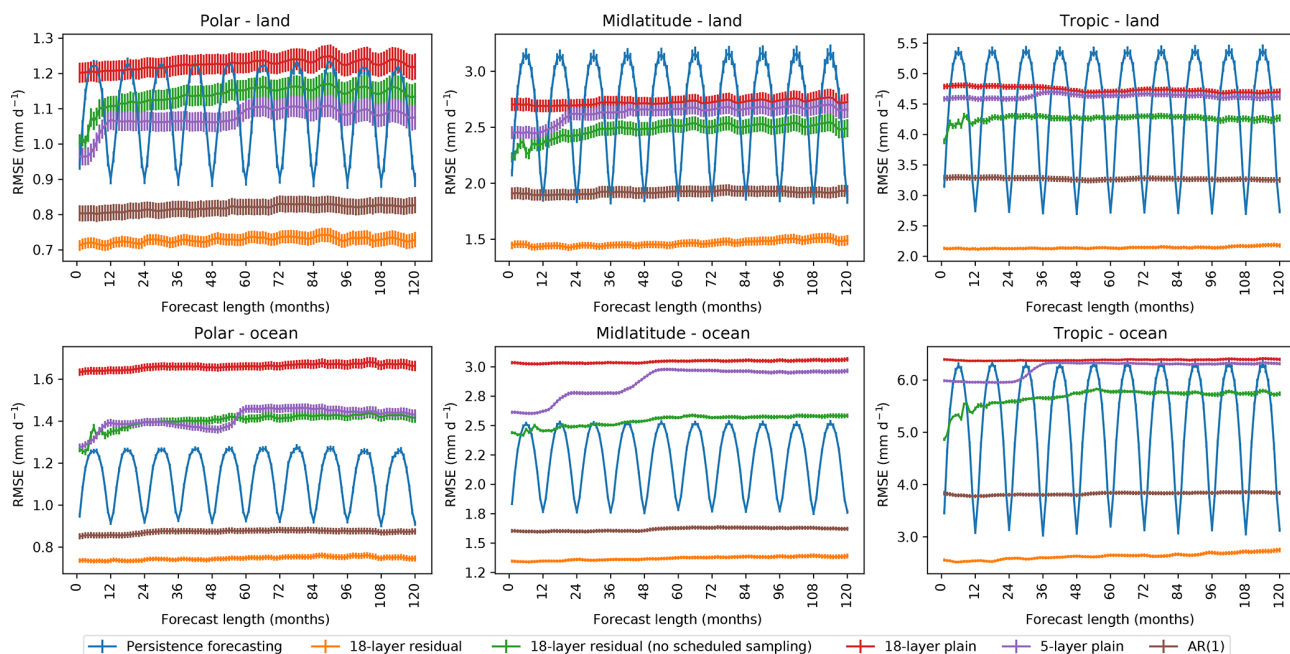
Figure 4. RMSE for decadal precipitation forecasts for six regions of interest. Both the plain and residual CNNs used a window size of 60. Vertical bars denote the standard error over all possible starting dates in the test set. The deep residual network with scheduled sampling outperforms all models in all regions and achieves consistent error over time. Removing any of these features from the network results in substantially lower performance.
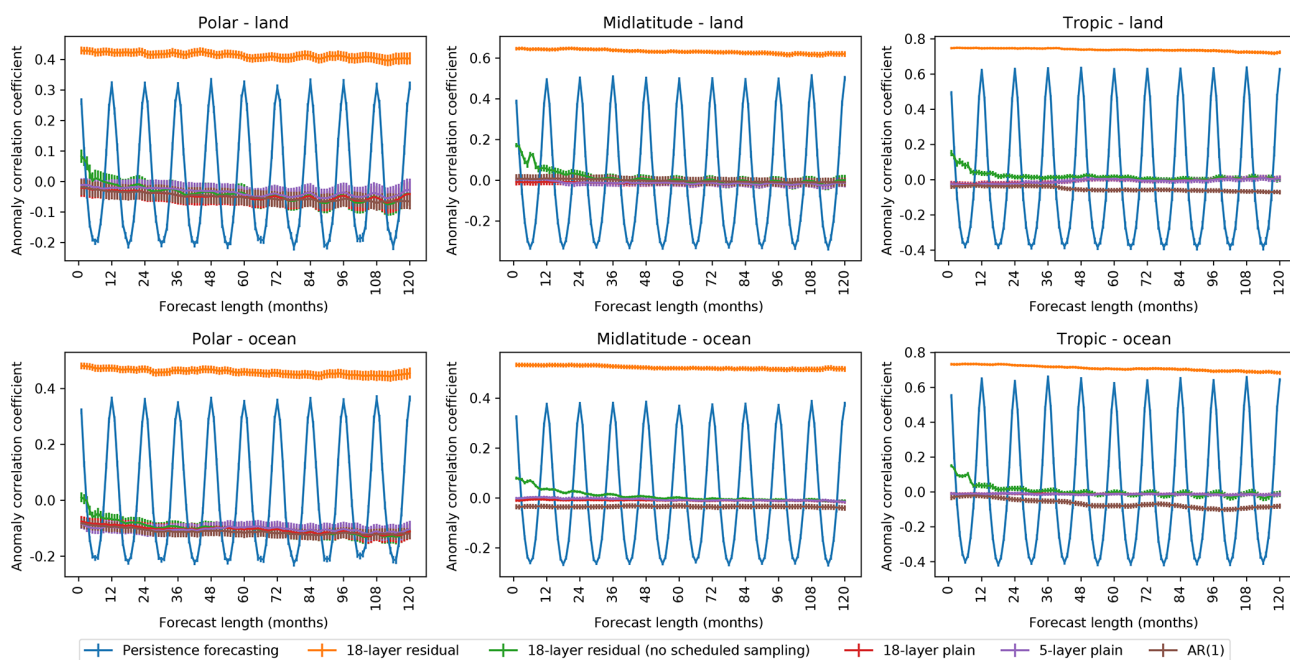
Figure 5. Anomaly correlation coefficient for decadal precipitation forecasts for six regions of interest. Both the plain and residual CNNs used a window size of 60. Vertical bars denote the standard error over all possible starting dates in the test set. As in Fig. 4, the deep residual network with scheduled sampling outperforms all models in all regions, consistently exhibiting a positive correlation with the ground truth outcomes.

lar features, indicating that the residual network was capable of reproducing Earth system model outputs reasonably well. Figure 3 shows the area-weighted difference $\hat{B} - B$ as well as the area-weighted percent error given by

$$\text{pct\_err} = \frac{(\hat{B}(x) - B(x)) \cdot A(x)}{B(x) \cdot A(x)}. \tag{5}$$

The residual model is prone to underpredict near the Equator, in the midlatitude storm tracks, and in areas associated with monsoon precipitation. All of these regions experience intense precipitation events (storms), which constitute the right tail of the precipitation distribution. The mean-square error loss function is less robust to outliers (Friedman et al., 2001, chap. 10), which are far more common in these regions than others, potentially explaining why the residual network tends to be biased low in these regions. On average, our model achieves reasonably low error on the test set, with a mean precipitation difference of -0.237 mm d^{-1} and mean percent error of $-13.22\,\%$.

Figure 4 shows the forecasting performance of each model on a decadal timescale. The 18-layer residual network outperforms all models in all regions and exhibits relatively consistent error over time. The AR(1) model generally performs second best in all cases except some seasons in the tropics when persistence tends to perform better. Our plain CNNs have less consistent error over time and performed worse than our residual networks overall. These networks proved more difficult to optimize and would often learn to predict similar values at each pixel regardless of spatial location. The 5-layer network showed lower generalization error than the 18-layer network, which was expected behavior, as plain networks become more difficult to train with increased depth (He et al., 2016). This challenge is well addressed in our 18-layer residual network, however, as it achieves good accuracy with significant depth.

To assess the benefits of scheduled sampling, we evaluated the performance of an identical residual network architecture trained without scheduled sampling (see "18-layer residual (no scheduled sampling)" in Fig. 4). For this model we observe the RMSE quickly increasing during the first few forecasts, indicating that it is not accustomed to making forecasts conditioned on past predictions. Surprisingly, this model also had significantly higher RMSE for the 1-month forecast, which is entirely conditioned on ground truth outcomes. We would expect a model trained without scheduled sampling to perform well in this case, as the input contains no model-generated data. However, as there are sufficient differences in the training setting (i.e., the use of early stopping), it is likely that these models converged to disparate minima. We hypothesize that additional hyperparameter tuning could decrease the RMSE for 1-month forecasts in these models.

Importantly, for the model using scheduled sampling, the skill of the forecast does not change appreciably with lead time, whereas one might expect the model to have some

initial-value predictability (e.g., Branstator et al., 2012) and thus more skillful predictability in the near term. In early forecasts, the input tensors primarily consist of ground truth precipitation outcomes. A model trained using scheduled sampling performs worse in this setting because it is accustomed to seeing inputs that contain model-generated data – that is, the initial states are explicitly degraded. This was a choice in terms of the problem we attempted to solve (reducing initial-value predictability in favor of longer forecasts), and scheduled sampling may not be an appropriate choice for other applications. To address this poor early forecast skill, we explored *preforecasting* – generating forecasts preceding the first forecast date to prime the model for inference (prediction). By taking this approach we ensure that the input tensor for the first forecast will contain at least some portion of model-generated data. To employ preforecasting, the number of preceding outcomes generated must be in the range $[1, \ldots, \text{window\_size}]$ and should be chosen relative to the sampling decay function used during training. We suggest generating window_size/2 preforecasts for a model trained using a linear decay function. We take this approach in Fig. 6 and find that it adequately reduces the RMSE for early forecasts while still maintaining low error for longer forecasts.

5 Discussion and conclusions

This study explored the application of deep-learning techniques to create surrogate models for precipitation fields in Earth system models under CO_2 forcing. From our experiments we found that a CNN architecture was effective for modeling spatio-temporal precipitation patterns. We also observed increased accuracy with deeper networks, which could be adequately trained using a residual learning approach. Finally, we found that scheduled sampling (supplemented with preforecasting) significantly improved long-term forecasting ability, improving upon the commonly used autoregressive model (although we admit that we could have more thoroughly explored the span of different linear methods, such as higher-order AR or ARIMA models).

It might be expected that the forecast model skill asymptotically approaches persistence as the predictions move farther from the initial state. We argue three reasons for why our neural network continues to have good skill and low error:

1. Scheduled sampling helps the model extrapolate from its own predictions, reducing errors in later forecasts.

2. Because the model is trained on numerous time periods in the 1pctCO2 experiment, it learns some inherent properties of precipitation response to CO_2 forcing.

3. We condition each prediction on 5 years worth of data, so it is likely easier for our model to retain signals coming from the initial conditions.
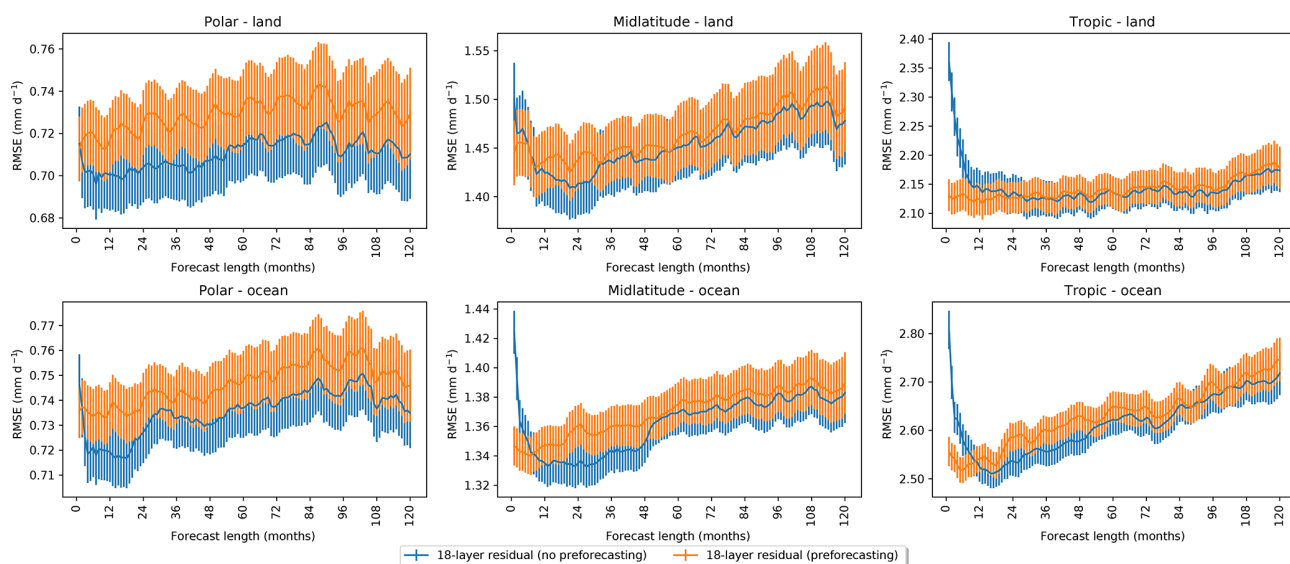
Figure 6. Comparing the RMSE for decadal precipitation forecasts with and without preforecasting. The model was trained using a window size of 60, and 30 preforecasts were generated. Vertical bars denote the standard error over all possible starting dates in the test set. Preforecasting significantly reduces the RMSE for short-range forecasts without degrading long-range forecasts.

Appendix A provides a comparison between using window sizes of 6 and 60 months, with the former showing steadily decreasing predictive skill due to its inability to learn the forced response. This is another point of verification for a conclusion well known in the decadal predictability community: although the initial state is important for decadal predictions, in forced runs, a great deal of skill is due to the underlying climate signals (Boer et al., 2019).

Based on these results we can identify several ways to enhance our current surrogate models, as well as a few promising deep-learning architectures applicable to this work, with the overall goal of understanding which deep-learning techniques may work well for creating surrogates of climate models.

Bengio et al. (2015) proposed three scheduled sampling decay functions: linear, exponential, and inverse sigmoid. Determining the optimal decay schedule for our problem could have significant effects on model predictability. Weight initialization has also been proven to affect model convergence and gradient propagation (He et al., 2015; Glorot and Bengio, 2010); therefore this must be investigated further. Window size was a fixed hyperparameter during tuning, but we cannot rule out its potential impact on forecasting (see Appendix A). Tuning these existing hyperparameters would be the first step in improving results.

Incorporating additional features, such as data from Earth system models with different forcings, global mean temperature, and daily average precipitation, would provide more relevant information and likely improve predictability. These could be incorporated by modifying our input tensor to include these as additional channels. Such augmentations

would be an important step toward designing practical, effective surrogate models in the future.

Two architectural features that we may consider adding are dropout and replay learning. Srivastava et al. (2014) showed that adding dropout with convolutional layers may lead to a performance increase and prevent overfitting. Replay learning is widely used in deep reinforcement learning and was shown to be successful in challenging domains (Zhang and Sutton, 2017). We believe that we can apply a similar concept to our architecture, where we train our network on random past input–output pairs so it can "remember" what it learned previously. This technique could aid in alleviating any bias from more recent training data and therefore boost performance.

Convolutional long short-term memory networks (LSTMs) have had great success in video prediction (Finn et al., 2016) and even precipitation nowcasting (Shi et al., 2015). They offer an alternate strategy for modeling both the spatial and temporal aspects present in our dataset: whereas our model treats the time dimension as channels, and acts upon a fixed length input window, convolutional LSTMs use recurrent connections to consume the input one time step at a time with no fixed length restriction on the input. This increases the effective depth between information contained in the distant past and the prediction, which may prove beneficial or harmful for our task; we leave it to future work to evaluate their suitability. One could also draw inspiration from the spatial downsampling, upsampling, and specific skip connectivity techniques often used work in semantic segmentation (e.g., U-Nets, introduced for biomedical image segmentation; Ronneberger et al., 2015).

Generative adversarial networks (GANs; Goodfellow et al., 2014) have proven to offer impressive generative modeling of grid-structured data. GANs are commonly used with image data for tasks such as super-resolution, image-to-image translation, image generation, and representation learning. The effectiveness of GANs to generate realistic data and ability to be conditioned on other variables (Goodfellow, 2016) make them quite appealing for spatio-temporal forecasting.

The results presented here show significant potential for deep learning to improve climate predictability. Applications of this work extend beyond providing a CNN forecasting library, as there are several enhancements that could yield a more practical alternative to traditional Earth system models. The ability to emulate alternate climate scenarios, for example, is desirable. Regardless, using an approach that can incorporate the internal model state appears to have promise in increasing prediction accuracy.

Appendix A: Effects of window size

We briefly explored the effects of different window sizes on predictability for the 18-layer residual network (without scheduled sampling). Figure A1 shows a comparison between a window size of 6 months versus the 60-month window that we used in our best-performing model. With a smaller window size, the forecast model is unable to learn enough of the forced response to improve predictive skill, so the forecast's model skill approaches that of persistence.
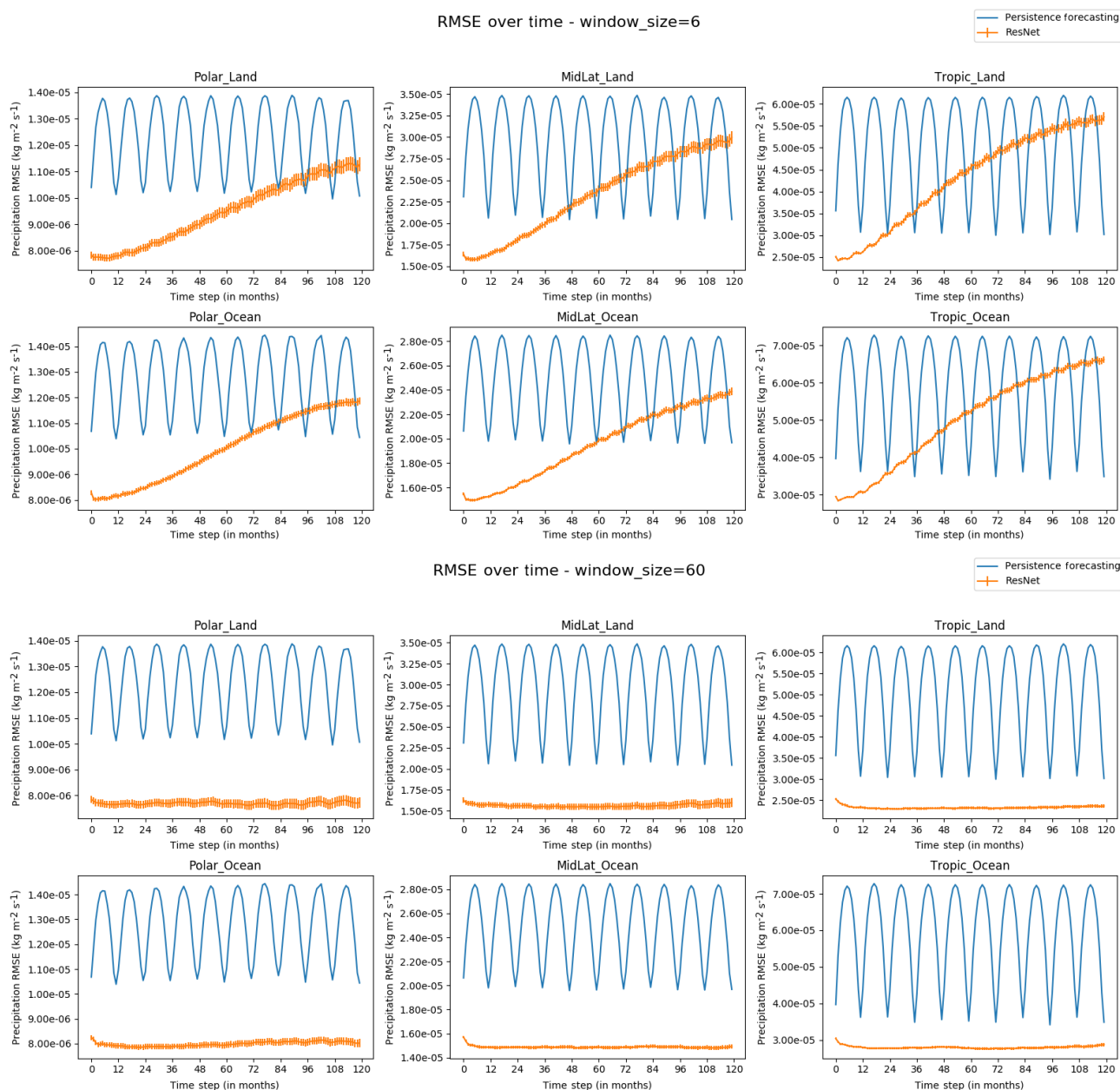
Figure A1. Comparing the RMSE for decadal precipitation forecasts in the 18-layer residual network with a 6-month window (top) and a 60-month window (bottom). Both networks were trained using scheduled sampling, and preforecasting was not employed when generating predictions.

Table A1. Hyperparameters for each network architecture used in this study (see Table 1). For each architecture, we tuned the learning rate, standard deviation used for weight initialization (sampling from a truncated normal distribution), and number of training epochs (unless training without scheduled sampling, in which case we used early stopping with a patience threshold of 10). For each architecture, the optimal hyperparameter configuration was selected after 60 iterations of random search.

Architecture	Decay function	Window size	Learning rate	Standard deviation (for Xavier initialization)	Epochs
18-layer residual	Linear	60	0.069	0.016	90
18-layer residual (no scheduled sampling)	N/A	60	0.095	0.021	82 (early stopping)
18-layer plain	Linear	60	0.013	0.01	100
5-layer plain	Linear	60	0.049	0.01	125

Table A2. Hyperparameter space used for training the models described in Table 1. For more information on how the hyperparameter space is defined, see https://github.com/hyperopt/hyperopt/wiki/FMin#21-parameter-expressions (last access: February 2020).

Hyperparameter	Min	Max	Step	Sampled from	Comment
Epochs	75	150	5	Quantized uniform	
Learning rate	0.001	0.01	n/a	Uniform	
Standard deviation	0.001	0.1	n/a	Uniform	
Window size	6	120	n/a	Choice	Only varied during initial experiments; fixed at 60 for the study

n/a means not applicable.

References

Arora, V. K. and Boer, G. J.: Uncertainties in the 20th century carbon budget associated with land use change, Glob. Change Biol., 16, 3327–3348, https://doi.org/10.1111/j.1365-2486.2010.02202.x, 2011.

Arora, V. K., Scinocca, J. F., Boer, G. J., Christian, J. R., Denman, K. L., Flato, G. M., Kharin, V. V., Lee, W. G., and Merryfield, W. J.: Carbon emission limits required to satisfy future representative concentration pathways of greenhouse gases, Geophys. Res. Lett., 38, L05805, https://doi.org/10.1029/2010GL046270, 2011.

Bellucci, A., Haarsma, R., Bellouin, N., Booth, B., Cagnazzo, C., van den Hurk, B., Keenlyside, N., Koenigk, T., Massonnet, F.,

Materia, S., and Weiss, M.: Advancements in decadal climate predictability: The role of nonoceanic drivers, Rev. Geophys., 53, 165–202, https://doi.org/10.1002/2014RG000473, 2015.

Bengio, S., Vinyals, O., Jaitly, N., and Shazeer, N.: Scheduled sampling for sequence prediction with recurrent neural networks, in: Advances in Neural Information Processing Systems, NIPS Proceedings, 1171–1179, 2015.

Bengio, Y.: Practical recommendations for gradient-based training of deep architectures, in: Neural networks: Tricks of the trade, 437–478, Springer, 2012.

Bergstra, J. and Bengio, Y.: Random search for hyper-parameter optimization, J. Mach. Learn. Res., 13, 281–305, 2012.

Boer, G. J., Kharin, V. V., and Merryfield, W. J.: Differences in potential and actual skill in a decadal prediction experiment, Clim. Dynam., 52, 6619–6631, https://doi.org/10.1007/s00382-018-4533-4, 2019.

Branstator, G., Teng, H., and Meehl, G. A.: Systematic Estimates of Initial-Value Decadal Predictability for Six AOGCMs, J. Climate, 25, 1827–1846, https://doi.org/10.1175/JCLI-D-11-00227.1, 2012.

Castruccio, S., McInerney, D. J., Stein, M. L., Crouch, F. L., Jacob, R. L., and Moyer, E. J.: Statistical Emulation of Climate Model Projections Based on Precomputed GCM Runs, J. Climate, 27, 1829–1844, https://doi.org/10.1175/JCLI-D-13-00099.1, 2014.

Chan, W., Jaitly, N., Le, Q., and Vinyals, O.: Listen, attend and spell: A neural network for large vocabulary conversational speech recognition, in: Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on, 4960–4964, IEEE, 2016.

Cohen, J., Screen, J. A., Furtado, J. C., Barlow, M., Whittleston, D., Coumou, D., Francis, J., Dethloff, K., Entekhabi, D., Overland, J., and Jones, J.: Recent Arctic amplification and extreme mid-latitude weather, Nature Geosci., 7, 627–637, https://doi.org/10.1038/ngeo2234, 2014.

Deo, R. V., Chandra, R., and Sharma, A.: Stacked transfer learning for tropical cyclone intensity prediction, ArXiv e-prints, http://arxiv.org/abs/1708.06539, 2017.

Finn, C., Goodfellow, I., and Levine, S.: Unsupervised learning for physical interaction through video prediction, in: Advances in neural information processing systems, NIPS Proceedings, 64–72, 2016.

Friedman, J., Hastie, T., and Tibshirani, R.: The elements of statistical learning, vol. 1, Springer series in statistics New York, NY, USA, 2001.

Fyfe, J. C., Meehl, G. A., England, M. H., Mann, M. E., Santer, B. D., Flato, G. M., Hawkins, E., Gillett, N. P., Xie, S.-P., Kosaka, Y., and Swart, N. C.: Making sense of the early-2000s warming slowdown, Nat. Clim. Change, 6, 224–228, https://doi.org/10.1038/nclimate2938, 2016.

Gawehn, E., Hiss, J. A., and Schneider, G.: Deep learning in drug discovery, Mol. Inform., 35, 3–14, https://doi.org/10.1002/minf.201501008, 2016.

Glorot, X. and Bengio, Y.: Understanding the difficulty of training deep feedforward neural networks, in: Proceedings of the thirteenth international conference on artificial intelligence and statistics, J. Mach. Learn. Res., 9, 249–256, 2010.

Goddard, L., Kumar, A., Solomon, A., Smith, D., Boer, G., Gonzalez, P., Kharin, V., Merryfield, W., Deser, C., Mason, S. J., Kirtman, B. P., Msadek, R., Sutton, R., Hawkins,

E., Fricker, T., Hegerl, G., Ferro, C. A. T., Stephenson, D. B., Meehl, G. A., Stockdale, T., Burgman, R., Greene, A. M., Kushnir, Y., Newman, M., Carton, J., Fukumori, I., and Delworth, T.: A verification framework for interannual-to-decadal predictions experiments, Clim. Dynam., 40, 245–272, https://doi.org/10.1007/s00382-012-1481-2, 2013.

Goodfellow, I.: NIPS 2016 Tutorial: Generative Adversarial Networks, available at: http://arxiv.org/abs/1701.00160 (last access: 24 February 2020), 2016.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y.: Generative Adversarial Nets, in: Advances in Neural Information Processing Systems 27, edited by: Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N. D., and Weinberger, K. Q., 2672–2680, Curran Associates, Inc., available at: http://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf (last access: 24 February 2020), 2014.

Goodfellow, I., Bengio, Y., and Courville, A.: Deep Learning, MIT Press, available at: http://www.deeplearningbook.org (last access: 24 February 2020), 2016.

Guemas, V., Doblas-Reyes, F. J., Andreu-Burillo, I., and Asif, M.: Retrospective prediction of the global warming slowdown in the past decade, Nat. Clim. Change, 3, 649–653, https://doi.org/10.1038/nclimate1863, 2013.

He, K., Zhang, X., Ren, S., and Sun, J.: Delving deep into rectifiers: Surpassing human-level performance on imagenet classification, in: Proceedings of the IEEE international conference on computer vision, 1026–1034, 2015.

He, K., Zhang, X., Ren, S., and Sun, J.: Deep residual learning for image recognition, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 770–778, Computer Vision Foundation, 2016.

Herger, N., Sanderson, B. M., and Knutti, R.: Improved pattern scaling approaches for the use in climate impact studies, Geophys. Res. Lett., 42, 3486–3494, https://doi.org/10.1002/2015GL063569, 2015.

Hinton, G., Deng, L., Yu, D., Dahl, G. E., Mohamed, A.-R., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T. N., and Kingsbury, B.: Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups, IEEE Signal Processing Magazine, 29, 82–97, 2012.

Hong, S., Kim, S., Joh, M., and Song, S.-K.: GlobeNet: Convolutional Neural Networks for Typhoon Eye Tracking from Remote Sensing Imagery, ArXiv e-prints, http://arxiv.org/abs/1708.03417, 2017.

Ioffe, S. and Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift, arXiv preprint arXiv:1502.03167, 2015.

Jay, A., Reidmiller, D., Avery, C., Barrie, D., DeAngelo, B., Dave, A., Dzaugis, M., Kolian, M., Lewis, K., Reeves, K., and Winner, D.: Overview, in: Impacts, Risks, and Adaptation in the United States: Fourth National Climate Assessment, Volume II, edited by: Reidmiller, D., Avery, C., Easterling, D., Kunkel, K., Lewis, K., Maycock, T., and Stewart, B., 33–71, U.S. Global Change Research Program, Washington, DC, USA, https://doi.org/10.7930/NCA4.2018.CH1, 2018.

Jiang, G.-Q., Xu, J., and Wei, J.: A Deep Learning Algorithm of Neural Network for the Parameterization of Typhoon-Ocean

Feedback in Typhoon Forecast Models, Geophys. Res. Lett., 45, 3706–3716, https://doi.org/10.1002/2018GL077004, 2018.

JMA: Verification Indices, available at: https://www.jma.go.jp/jma/jma-eng/jma-center/nwp/outline2013-nwp/pdf/outline2013_Appendix_A.pdf (last access: February 2020), 2019.

Joliffe, I. and Stephenson, D.: Forecast verification, John Wiley and Sons, 2003.

Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., and Fei-Fei, L.: Large-scale video classification with convolutional neural networks, in: Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, Computer Vision Foundation, 1725–1732, 2014.

Krizhevsky, A., Sutskever, I., and Hinton, G. E.: Imagenet classification with deep convolutional neural networks, in: Advances in neural information processing systems, NIPS Proceedings, 1097–1105, 2012.

Lean, J. L. and Rind, D. H.: How will Earth's surface temperature change in future decades?, Geophys. Res. Lett., 36, L15708, https://doi.org/10.1029/2009GL038923, 2009.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P.: Gradient-based learning applied to document recognition, Proceedings of the IEEE, 86, 2278–2324, 1998.

LeCun, Y. A., Bottou, L., Orr, G. B., and Müller, K.-R.: Efficient backprop, in: Neural networks: Tricks of the trade, 9–48, Springer, 2012.

Liu, Y., Racah, E., Prabhat, Correa, J., Khosrowshahi, A., Lavers, D., Kunkel, K., Wehner, M., and Collins, W.: Application of Deep Convolutional Neural Networks for Detecting Extreme Weather in Climate Datasets, ArXiv e-prints, http://arxiv.org/abs/1605.01156, 2016.

Lu, D. and Ricciuto, D.: Efficient surrogate modeling methods for large-scale Earth system models based on machine-learning techniques, Geosci. Model Dev., 12, 1791–1807, https://doi.org/10.5194/gmd-12-1791-2019, 2019.

Lynch, C., Hartin, C., Bond-Lamberty, B., and Kravitz, B.: An open-access CMIP5 pattern library for temperature and precipitation: description and methodology, Earth Syst. Sci. Data, 9, 281–292, https://doi.org/10.5194/essd-9-281-2017, 2017.

MacMartin, D. G. and Kravitz, B.: Dynamic climate emulators for solar geoengineering, Atmos. Chem. Phys., 16, 15789–15799, https://doi.org/10.5194/acp-16-15789-2016, 2016.

McDermott, P. L. and Wikle, C. K.: Deep echo state networks with uncertainty quantification for spatio-temporal forecasting, Environmetrics, 30, e2553, https://doi.org/10.1002/env.2553, 2018.

Miller, J., Nair, U., Ramachandran, R., and Maskey, M.: Detection of transverse cirrus bands in satellite imagery using deep learning, Comput. Geosci., 118, 79–85, https://doi.org/10.1016/j.cageo.2018.05.012, 2018.

Mitchell, T. D.: Pattern Scaling: An Examination of the Accuracy of the Technique for Describing Future Climates, Clim. Change, 60, 217–242, https://doi.org/10.1023/A:1026035305597, 2003.

Moss, R. H., Kravitz, B., Delgado, A., Asrar, G., Brandenberger, J., Wigmosta, M., Preston, K., Buzan, T., Gremillion, M., Shaw, P., Stocker, K., Higuchi, S., Sarma, A., Kosmal, A., Lawless, S., Marqusee, J., Lipschultz, F., O'Connell, R., Olsen, R., Walker, D., Weaver, C., Westley, M., and Wright, R.: Nonstationary Weather Patterns and Extreme Events: Informing Design and Planning for Long-Lived Infrastructure, Tech. rep., ESTCP, ESTCP Project RC-201591, 2017.

Nair, V. and Hinton, G. E.: Rectified linear units improve restricted boltzmann machines, in: Proceedings of the 27th international conference on machine learning (ICML-10), Association for Computing Machinery, 807–814, 2010.

Ouyang, Q. and Lu, W.: Monthly rainfall forecasting using echo state networks coupled with data preprocessing methods, Water Resour. Mange., 32, 659–674, https://doi.org/10.1007/s11269-017-1832-1, 2018.

Pradhan, R., Aygun, R. S., Maskey, M., Ramachandran, R., and Cecil, D. J.: Tropical Cyclone Intensity Estimation Using a Deep Convolutional Neural Network, IEEE Transactions on Image Processing, 27, 692–702, https://doi.org/10.1109/TIP.2017.2766358, 2018.

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

Robertson, A. W., Kumar, A., Peña, M., and Vitart, F.: Improving and Promoting Subseasonal to Seasonal Prediction, B. Am. Meteor. Soc., 96, ES49–ES53, https://doi.org/10.1175/BAMS-D-14-00139.1, 2015.

Ronneberger, O., Fischer, P., and Brox, T.: U-Net: Convolutional Networks for Biomedical Image Segmentation, in: Medical Image Computing and Computer-Assisted Intervention (MICCAI), vol. 9351 of *LNCS*, 234–241, Springer, available at: http://lmb.informatik.uni-freiburg.de/Publications/2015/RFB15a (last access: February 2020), (available on arXiv:1505.04597 [cs.CV]), 2015.

Santer, B., Wigley, T., Schlesinger, M., and Mitchell, J.: Developing Climate Scenarios from Equilibrium GCM Results, Tech. rep., Hamburg, Germany, 1990.

Shi, X., Chen, Z., Wang, H., Yeung, D.-Y., Wong, W.-k., and WOO, W.-c.: Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting, in: Advances in Neural Information Processing Systems 28, edited by: Cortes, C., Lawrence, N. D., Lee, D. D., Sugiyama, M., and Garnett, R., 802–810, Curran Associates, Inc., available at: http://papers.nips.cc/paper/5955-convolutional-lstm-network-a-machine-learning-approach-for-precipitation-nowcasting.pdf (last access: February 2020), 2015.

Simonyan, K. and Zisserman, A.: Very deep convolutional networks for large-scale image recognition, arXiv preprint arXiv:1409.1556, 2014.

Srivastava, N., Hinton, G., Krizhevkskey, A., Sutskever, I., and Salakhutdinov, R.: Dropout: A simple way to prevent neural networks for overfitting, J. Mach. Learn. Res., 15, 1929–1958, 2014.

Stocker, T. F., Qin, D., Plattner, G.-K., Alexander, L. V., Allen, S. K., Bindoff, N. L., Bréon, F.-M., Church, J. A., Cubasch, U., Emori, S., Forster, P., Friedlingstein, P., Gillett, N., Gregory, J. M., Hartmann, D. L., Jansen, E., Kirtman, B., Knutti, R., Krishna Kumar, K., Lemke, P., Marotzke, J., Masson-Delmotte, V., Meehl, G. A., Mokhov, I. I., Piao, S., Ramaswamy, V., Randall, D., Rhein, M., Rojas, M., Sabine, C., Shindell, D., Talley, L. D., Vaughan, D. G., and Xie, S.-P.: Technical Summary, in: Climate Change 2013: The Physical Science Basis. Contribution of Working Group I to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change, edited by: Stocker, T. F., Qin, D., Plattner, G.-K., Tignor, M., Allen, S. K., Boschung, J., Nauels, A., Xia, Y., Bex, V., and Midgley, P. M., Cambridge University Press, Cambridge, United Kingdom and New York, NY, USA, 2013.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A.: Going deeper with convolutions, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 8–10 June 2015, Boston, Massachusetts, 1–9, 2015.

van Vuuren, D. P., Edmonds, J., Kainuma, M., Riahi, K., Thomson, A., Hibbard, K., Hurtt, G. C., Kram, T., Krey, V., Lamarque, J.-F., Masui, T., Meinshausen, M., Nakicenovic, N., Smith, S. J., and Rose, S. K.: The representative concentration pathways: An overview, Clim. Change, 109, 5–31, https://doi.org/10.1007/s10584-011-0148-z, 2011.

Weber, T., Corotan, A., Hutchinson, B., Kravitz, B., and Link, R. P.: A Deep Neural Network approach for estimating precipitation fields in Earth System Models, available at: https://github.com/hutchresearch/deep_climate_emulator, last access: 24 February 2020.

Yao, Y., Rosasco, L., and Caponnetto, A.: On early stopping in gradient descent learning, Constructive Approximation, 26, 289–315, 2007.

Yeager, S., Danabasoglu, G., Rosenbloom, N., Strand, W., Bates, S., Meehl, G., Karspeck, A., Lindsay, K., Long, M., Teng, H., and Lovenduski, N.: Predicting Near-Term Changes in the Earth System: A Large Ensemble of Initialized Decadal Prediction Simulations Using the Community Earth System Model, B. Am. Meteor. Soc., 99, 1867–1886, https://doi.org/10.1175/BAMS-D-17-0098.1, 2018.

Yeh, S.-W., Cai, W., Min, S.-K., McPhaden, M. J., Dommenget, D., Dewitte, B., Collins, M., Ashok, K., An, S.-I., Yim, B.-Y., and Kug, J.-S.: ENSO Atmospheric Teleconnections and Their Response to Greenhouse Gas Forcing, Rev. Geophys., 56, 185–206, https://doi.org/10.1002/2017RG000568, 2018.

Yuan, N., Huang, Y., Duan, J., Zhu, C., Xoplaki, E., and Luterbacher, J.: On climate prediction: How much can we expect from climate memory?, Clim. Dynam., 52, 855–864, https://doi.org/10.1007/s00382-018-4168-5, 2019.

Zhang, S. and Sutton, R. S.: A Deeper Look at Experience Replay, CoRR, abs/1712.01275, 2017.