Enhancing geophysical flow machine learning performance via scale separation

Davide Faranda[1,2,3], Mathieu Vrac[1], Pascal Yiou[1], Flavio Maria Emanuele Pons[1], Adnane Hamid[1], Giulia Carella[1], Cedric Ngoungue Langue[1], Soulivanh Thao[1], and Valerie Gautard[4]

[1]Laboratoire des Sciences du Climat et de l'Environnement, CE Saclay l'Orme des Merisiers, UMR 8212 CEA-CNRS-UVSQ, Université Paris-Saclay & IPSL, 91191 Gif-sur-Yvette, France.
[2]London Mathematical Laboratory, 8 Margravine Gardens, London, W68RH, UK.
[3]LMD/IPSL, Ecole Normale Superieure, PSL research University, Paris, France
[4]DRF/IRFU/DEDIP//LILAS Departement d'Electronique des Detecteurs et d'Informatique pour la Physique, CE Saclay l'Orme des Merisiers, 91191 Gif-sur-Yvette, France.

Correspondence: Davide Faranda (davide.faranda@lsce.ipsl)

Abstract. Recent advances in statistical and machine learning have opened the possibility to forecast the behavior of chaotic systems using recurrent neural networks. In this article we investigate the applicability of such a framework to geophysical flows, known to involve multiple scales in length, time and energy and to feature intermittency. We show that both multiscale dynamics and intermittency introduce severe limitations on the applicability of recurrent neural networks, both for short-term forecasts, as well as for the reconstruction of the underlying attractor. We suggest that possible strategies to overcome such limitations should be based on separating the smooth large-scale dynamics from the intermittent/small-scale features. We test these ideas on global sea-level pressure data for the past 40 years, a proxy of the atmospheric circulation dynamics. Better short- and long-term forecasts of sea-level pressure data can be obtained with an optimal choice of spatial coarse-graining and time filtering.

1 Introduction

The advent of high-performance computing has paved the way for advanced analyses of high-dimensional datasets (Jordan and Mitchell, 2015; LeCun et al., 2015). Those successes have naturally raised the question of whether it is possible to learn the behavior of a dynamical system without resolving or even without knowing the underlying evolution equations. Such an interest is motivated on one side by the fact that many complex systems still miss a universally accepted state equation — e.g. brain dynamics (Bassett and Sporns, 2017), macro-economical and financial systems (Quinlan et al., 2019) — and, on the other, by the need of reducing the complexity of the dynamical evolution for the systems of which the underlying equations are known — e.g. on geophysical and turbulent flows (Wang et al., 2017). Evolution equations are difficult to solve for large systems such as the geophysical flows, so that approximations and parameterizations are needed for meteorological and climatological

applications (Buchanan, 2019). These difficulties are enhanced by those encountered in the modelling of phase transitions that lead to cloud formation and convection, which are major sources of uncertainty in climate modelling (Bony et al., 2015). Machine Learning techniques capable of learning geophysical flows dynamics would help improve those approximations and avoid running costly simulations resolving explicitly all spatial/temporal scales.

Recently, several efforts have been made to apply machine learning to the prediction of geophysical data (Wu et al., 2018), to learn parameterizations of subgrid processes in climate models (Krasnopolsky et al., 2005; Krasnopolsky and Fox-Rabinovitz, 2006; Rasp et al., 2018; Gentine et al., 2018; Brenowitz and Bretherton, 2018, 2019; Yuval and O'Gorman, 2020; Gettelman et al., 2020; Krasnopolsky et al., 2013), to the forecasting (Liu et al., 2015; Grover et al., 2015; Haupt et al., 2018; Weyn et al., 2019) and nowcasting (i.e. extremely short-term forecasting) of weather variables (Xingjian et al., 2015; Shi et al., 2017; Sprenger et al., 2017), and to quantify the uncertainty of deterministic weather prediction (Scher and Messori, 2018). One of the greatest challenges is to replace equations of climate models with neural networks capable to produce reliable long- and short-term forecasts of meteorological variables. A first great step in this direction was the use of Echo State Networks (ESN, (Jaeger, 2001)), a particular case of Recurrent Neural Networks (RNN), to forecast the behavior of chaotic systems, such as the Lorenz (1963) and the Kuramoto-Sivashinsky dynamics (Hyman and Nicolaenko, 1986). It was shown that ESN predictions of both systems attain performances comparable to those obtained with the exact equations (Pathak et al., 2017, 2018). Good performances were obtained adopting regularized ESN in the short-term prediction of multidimensional chaotic time series, both from simulated and real data (Xu et al., 2018). This success motivated several follow-up studies with a focus on meteorological and climate data. These are based on the idea of feeding various statistical learning algorithms with data issued from dynamical systems of different complexity, in order to study short-term predictability and capability of machine learning to reproduce long-term features of the input data dynamics. Recent examples include equation-informed moment-matching for the Lorenz96 model (Lorenz, 1996; Schneider et al., 2017), multi-layer perceptrons to reanalysis data (Scher, 2018), or convolutional neural networks to simplified climate simulation models (Dueben and Bauer, 2018; Scher and Messori, 2019). All these learning algorithms were capable to provide some short-term predictability, but failed at obtaining a long-term behavior coherent with the input data.

The motivation for this study came from the evidence that a straightforward application of ESN to high dimensional geophysical data does not yield to the same result quality obtained by Pathak et al. (2018) for the Lorenz 1963 and the Kuramoto-Sivashinsky models. Here we will investigate the causes for this behavior. Indeed, previous results (Scher, 2018; Dueben and Bauer, 2018; Scher and Messori, 2019) suggest that simulations of large-scale climate fields through deep learning algorithms are not as straightforward as those of the chaotic systems considered by Pathak et al. (2018). We identify two main mechanisms responsible for these limitations: (i) the non-trivial interactions with small-scale motions carrying energy at large scale and (ii) the intermittent nature of the dynamics. Intermittency triggers large fluctuations of observables of the motion in time and space (Schertzer et al., 1997) and can result in non-smooth trajectories within the flow, leading to local unpredictability and increasing the number of degrees of freedom needed to describe the dynamics (Paladin and Vulpiani, 1987).

By applying ESN to multiscale and intermittent systems, we investigate how scale separation improves ESN predictions. Our goal is to reproduce a surrogate of the large-scale dynamics of global sea-level pressure fields, a proxy of the atmospheric circu-

lation. We begin by analysing three different dynamical systems: we simulate the effects of small scales by artificially introducing small-scale dynamics in the Lorenz 1963 equations (Lorenz, 1963) via additive noise, in the spirit of recent deep-learning studies with add-on stochastic component (Mukhin et al., 2015; Seleznev et al., 2019). We investigate the Pomeau-Manneville equations (Manneville, 1980) stochastically perturbed with additive noise to have an example of intermittent behavior. We then analyse the performance of ESN in the Lorenz 1996 system (Lorenz, 1996). The dynamics of this system is meant to mimic that of the atmospheric circulation, featuring both large-scale and small-scale variables with an intermittent behavior. For all of those systems, as well as for the sea-level pressure data, we show how the performance of ESN in predicting the behavior of the system deteriorates rapidly when small-scale dynamics feedback to large scale is important. The idea of using moving average for scale separation is already established for meteorological variables (Eskridge et al., 1997). We choose the ESN framework following the results of Pathak et al. (2017, 2018), and an established literature about its ability to forecast chaotic time series and its stability to noise. For example, Shi and Han (2007); Li et al. (2012) analyse and compare the predictive performance of simple and improved ESN on simulated and observed one-dimensional chaotic time series. We aim at understanding this sensitivity in a deeper way, while assessing the possibility to reduce its impact on prediction through simple noise reduction methods.

The remaining of this article is organised as follows: in section 2, we give an overview of the ESN method (2.1), then we introduce the metrics used to evaluate ESN performance (2.2) and introduce the moving average filter used to improve ESN performance (2.3). Section 3 presents the results for each analysed system. First we show the results for the perturbed Lorenz 1963 equations, then for the Pomeau-Manneville intermittent map, and for the Lorenz 1996 equations. Finally, we discuss the improvement in short-term prediction and the long-term attractor reconstruction obtained with the moving average filter. We conclude by testing these ideas on atmospheric circulation data.

2 Methods

Reservoir computing is a variant of recurrent neural networks (RNN) in which the input signal is connected to a fixed, randomly assigned dynamical system called reservoir (Hinaut, 2013). The principle of Reservoir computing first consists in projecting the input signal to a high-dimensional space in order to obtain a non-linear representation of the signal; and then in performing a new projection between the high-dimensional space and the output units, usually via linear regression or ridge regression. In our study, we use ESN, a particular case of RNN where the output and the input have the same dynamical form. In an ESN, neuron layers are replaced by a sparsely connected network (the reservoir), with randomly assigned fixed weights. We harvest reservoir states via a nonlinear transform of the driving input and compute the output weights to create reservoir-to-output connections. The code is given in appendix, and it shows the parameters used for the computations.

We now briefly describe the ESN implementation. Vectors will be denoted in bold and matrices in upper case. Let $\boldsymbol{x}(t)$ be the K-dimensional observable consisting of $t = 1, 2, \ldots, T$ time iterations, originating from a dynamical system, and $\boldsymbol{r}(t)$ be the N-dimensional reservoir state, then:

$$\boldsymbol{r}(t + dt) = \tanh(W\boldsymbol{r}(t) + W_{in}\boldsymbol{x}(t)), \tag{1}$$

where W is the adjacency matrix of the reservoir: its dimensions are $N \times N$, and N is the number of neurons in the reservoir. In ESN, the neuron layers of classic deep neural networks are replaced by a single layer consisting of a sparsely connected random network, with coefficients uniformly distributed in $[-0.5; 0.5]$. The $N \times K$-dimensional matrix W_{in} is the weight matrix of the connections between the input layer and the reservoir, and the coefficients are randomly sampled, as for W. The output of the network at time step $t + dt$ is

$$W_{out}\boldsymbol{r}(t + dt) = \boldsymbol{y}(t + dt) \tag{2}$$

where $\boldsymbol{y}(t + dt)$ is the ESN prediction, W_{out} with dimensions $K \times N$, is the weight matrix of the connections between the reservoir neurons and the output layer. We estimate W_{out} via a ridge regression (Hastie et al., 2015):

$$W_{out} = \boldsymbol{y}_{train}\boldsymbol{r}^T[\boldsymbol{r}\boldsymbol{r}^T - \lambda I]^{-1} \tag{3}$$

with $\lambda = 10^{-8}$ and $\boldsymbol{y}_{train} \equiv \{\boldsymbol{y}(t) : (0 < t < T_{train})\}$ as training dataset . Note that we have investigated different values of λ spanning $10^{-8} < \lambda < 10^{-2}$ on the Lorenz 1963 example and found little improvement only when the network size was large, with λ partially preventing overfitting. Values of $\lambda < 10^{-8}$ have not been investigated because too close or below the numerical precision. In the prediction phase we have a recurrent relationship:

$$\boldsymbol{r}(t + dt) = \tanh(W\boldsymbol{r}(t) + W_{in}W_{out}\boldsymbol{r}(t)). \tag{4}$$

2.1 ESN performance indicators

In this paper, we use three different indicators of performance of the ESN: a statistical distributional test to measure how the distributions of observables derived from ESN match those of the target data, a predictability horizon test and the initial forecast error. They are described below.

Statistical distributional test

As a first diagnostic of the performance of ESN, we aim at assessing whether the marginal distribution of the forecast values for a given dynamical system is significantly different from the invariant distribution of the system itself. To this purpose, we conduct a χ^2 test (Cochran, 1952), designed as follows. Let U be a system observable, linked to the orginal variables of the systems via a function ζ, a function mapping between two spaces, such that $u(t) = \zeta(\boldsymbol{x}(t))$ with support R_U and probability density function $f_U(u)$, and let $u(t)$ be a sample trajectory from U. Note that $u(t)$ does not correspond to $x(t)$, it is constructed using the observable output of the dynamical system. Let now $\hat{f}_U(u)$ be an approximation of $f_U(u)$, namely the histogram of u over $i = 1, \ldots, M$ bins. Note that, if u spans the entire phase space, $\hat{f}_U(u)$ is the numerical approximation of the Sinai-Ruelle-Bowen measure of the dynamical system (Eckmann and Ruelle, 1985; Young, 2002). Let now V be the variable generated by the ESN forecasting, with support $R_V = R_U$, $v(t)$ the forecast sample, $g_V(v)$ its probability density function and $\hat{g}_V(v)$ the histogram of the forecast sample. Formally, R_U and R_V are Banach spaces, whose dimension depends on the choice of

the function ζ. For example, we will use as ζs one of the variable of the system or the sum of all variables. We test the null hypothesis that the marginal distribution of the forecast sample is the same as the invariant distribution of the system, against the alternative hypothesis that the two distributions are significantly different:

$$H_0: \quad f_U(u) = g_V(v) \quad \text{for every } u \in R_U$$
$$H_1: \quad f_U(u) \neq g_V(v) \quad \text{for any } u \in R_U$$

Under H_0, $\hat{f}_U(u)$ is the expected value for $\hat{g}_V(v)$, which implies that observed differences $(\hat{g}_V(v) - \hat{f}_U(u))$ are due to random errors, and are then independent and identically distributed Gaussian random variables. Statistical theory shows that, given H_0 true, the test statistics

$$\Sigma = \sum_{i=1}^{M} \frac{(\hat{g}_V^i(v) - \hat{f}_U^i(u))^2}{\hat{f}_U^i(u)} \tag{5}$$

is distributed as a chi-squared random variable with M degrees of freedom, $\chi^2(M)$. Then, to test the null hypothesis at the level α, the observed value of the test statistics Σ is compared to the critical value corresponding to the $1-\alpha$ quantile of the chi-square distribution, $\Sigma_c = \chi^2_{1-\alpha}(M)$: if $\Sigma > \Sigma_c$, the null hypothesis must be rejected in favour of the specified alternative. Since we are evaluating the proximity between distributions, the Kullback-Leibler (KL) divergence could be considered a more natural measure. However, we decide to rely on the χ^2 test because of its feasibility while maintaining some equivalence to KL. In fact, both the KL and the χ^2 are non symmetric statistical divergences, an ideal property when measuring a proximity to a reference probability distribution (Basu et al., 2019). It has been known for a long time (Berkson et al., 1980) that statistical estimation based on minimum χ^2 is equivalent to most other objective functions, including the maximum likelihood and, thus, the minimum KL. In fact, the KL divergence is linked to maximum likelihood both in an estimation and in a testing setting. In point estimation, maximum likelihood parameter estimates minimize the KL divergence from the chosen statistical model; in hypothesis testing, the KL divergence can be used to quantify the loss of power of likelihood ratio tests in case of model misspecification (Eguchi and Copas, 2006). On the other hand, contrary to the KL, the χ^2 divergence has the advantage of converging to a known distribution under the null hypothesis, independently on the parametric form of the reference distribution.

In our setup, we encounter two limitations in using the standard χ^2 test. First, problems may arise when $\hat{f}_U(u)$, i.e. if the sample distribution does not span the entire support of the invariant distribution of the system. We observe this in a relatively small number of cases; since aggregating the bins would introduce unwanted complications, we decide to discard the pathological cases, controlling the effect empirically as described below. Moreover, even producing relatively large samples, we are not able to actually observe the invariant distribution of the considered system, which would require much longer simulations. As a consequence, we would observe excessive rejection rates when testing samples generated under H_0.

We decide to control these two effects by using a Monte Carlo approach. To this purpose, we generate 10^5 samples $u(t) = \zeta(\boldsymbol{x}(t))$ under the null hypothesis, and we compute the test statistic for each one according to Eq. (5). Then, we use the $(1-\alpha)$ quantile of the empirical distribution of Σ — instead of the theoretical $\chi^2(M)$ — to determine the critical threshold Σ_c. As

a last remark, we notice that we are making inference in repeated tests setting, as the performance of the ESN is tested 10^5 times. Performing a high number of independent tests at a chosen level α increases the observed rejection rate: in fact, even if the samples are drawn under H_0, extreme events become more likely, resulting in an increased probability to erroneously reject the null hypothesis. To avoid this problem, we apply the Bonferroni correction (Bonferroni, 1936), testing each one of the $m = 10^5$ available samples at the level $\alpha' = \frac{\alpha}{m}$, with $\alpha = 0.05$.

Averaging the test results over several sample pairs $u(t)$, $v(t)$ we obtain a rejection rate $0 < \phi < 1$ that we use to measure the adherence of a ESN trajectory $v(t)$ to trajectories obtained via the equations. If $\phi = 0$, almost all the ESN trajectories can shadow original trajectories, if $\phi = 1$ none of the ESN trajectories resemble those of the systems of equations.

Predictability Horizon

As a measure of the predictability horizon of the ESN forecast compared to the equations, we use the absolute prediction error (APE):

$$APE(t) = |u(t) - v(t)| \tag{6}$$

and we define the predictability horizon τ_s as the first time that APE exceeds a certain threshold s:

$$\tau_s = \inf\{t, APE(t) > s\} \tag{7}$$

Indeed APE can equivalently be written as $\Delta t \dot{u}$. We link s to the average separation of observations in the observable u and we fix

$$s = \frac{1}{T-1} \sum_{t=2}^{T-1} [u(t) - u(t-1)].$$

We have tested the sensitivity of results against the exact definition of s.

We interpret τ_s as a natural measure of the Lyapunov time ϑ, namely the time it takes for an ensemble of trajectories of a dynamical system to diverge (Faranda et al., 2012; Panichi and Turchetti, 2018).

Initial Forecast Error

The initial error is given by $\eta = APE(t = 1)$, for the first time step after the initial condition at $t = 0$. We expect η to reduce as the training time increases.

2.2 Moving average filter

Equipped with these indicators, we analyze two sets of simulations performed with and without smoothing, which was implemented using a moving average filter. The moving average operation is the integral of $u(t)$ between t and $t - w$, where w is the window size of the moving average. The simple moving average filter can be seen as a nonparametric time series smoother (see e.g. Brockwell and Davis, 2016, chapter 1.5). It can be applied to smooth out (relatively) high frequencies in a time series,

both to de-noise the observations of a process or to estimate trend-cycle components, if present. Moving averaging consists, in practice, in replacing the trajectory $\boldsymbol{x}(t)$ by a value $\boldsymbol{x}^{(f)}(t)$, obtained by averaging the previous w observations. If the time dimension is discrete (like in the Pomeau-Manneville system) it is defined as:

$$\boldsymbol{x}^{(f)}(t) = \frac{1}{w}\sum_{i=0}^{w-1}\boldsymbol{x}(t-i), \tag{8}$$

while for continuous time systems (like the Lorenz 1963 system), the sum is formally replaced by an integral:

$$\boldsymbol{x}^{(f)}(t) = \frac{1}{w}\int_{t}^{t+w}\boldsymbol{x}(\varsigma)d\varsigma. \tag{9}$$

We can define the residuals as:

$$\delta\boldsymbol{x}(t) = \boldsymbol{x}^{(f)}(t) - \boldsymbol{x}(t). \tag{10}$$

In practice, the computation always refers to the discrete time case, as continuous time systems are also sampled at finite time steps. Since Echo State Networks are known to be sensitive to noise (see e.g. Shi and Han, 2007), we exploit the simple moving average filter to smooth out high-frequency noise and assess the results for different smoothing windows w. We find that the choice of the moving averaging window w must respect two conditions: it should be large enough to smooth out the noise but smaller than the characteristic time τ of the large-scale fluctuations of the system. For chaotic systems, τ can be derived knowing the rate of exponential divergence of the trajectories, a quantity linked to the Lyapunov exponents (Wolf et al., 1985), and τ is known as the Lyapunov time.

We also remark that we can express explicitly the original variables $\boldsymbol{x}(t)$ as a function of the filtered variables $\boldsymbol{x}^{(f)}(t)$ as:

$$\boldsymbol{x}(t) = w(\boldsymbol{x}^{(f)}(t) - \boldsymbol{x}^{(f)}(t-1)) + \boldsymbol{x}(t-w). \tag{11}$$

We will test this formula for stochastically perturbed systems to evaluate the error introduced by the use of residuals $\delta\boldsymbol{x}$.

2.3 Testing ESN on filtered dynamics

Here we describe the algorithm used to test ESN performance on filtered dynamics:

1. Simulate the reference trajectory $\boldsymbol{x}(t)$ using the equations of the dynamical systems, and standardize $\boldsymbol{x}(t)$ by subtracting the mean and dividing by its standard deviation.
2. Perform the moving average filter to obtain $\boldsymbol{x}^{(f)}(t)$.
3. Extract from $\boldsymbol{x}^{(f)}(t)$ a training set $\boldsymbol{x}^{(f)}_{train}(t) \equiv \{\boldsymbol{x}(t) : (0 < t < T_{train})\}$.
4. Train the ESN on $\boldsymbol{x}^{(f)}_{train}(t)$ dataset.

5. Obtain the ESN forecast $\boldsymbol{y}^{(f)}(t)$ for $T_{train} < t < T$. Note that the relation between $\boldsymbol{y}(t)$ and $\boldsymbol{x}(t)$ is given in Eqs.1-2

6. Add residuals (Eq. 10) to $\boldsymbol{y}^{(f)}(t)$ sample as $\boldsymbol{y}(t) = \boldsymbol{y}^{(f)}(t) + \delta\boldsymbol{x}$, where $\delta\boldsymbol{x}$ is randomly sampled from the $\delta\boldsymbol{x}^{(f)}_{train}(t)$.

7. Compute the observables $v(t) = \zeta(\boldsymbol{y}(t))$ and $u(t) = \zeta(\boldsymbol{x}(T_{train} < t < T)$. Note that $\boldsymbol{x}(T_{train} < t < T)$ is the ensemble of *true values* of the original dynamical system.

8. Using $u(t)$ and $v(t)$, compute the metrics ϕ, τ and η and evaluate the forecasts.

As an alternative to step 6, one can also use Eq. (11) and obtain:

$$v(t) \simeq w(v^{(f)}(t) - v^{(f)}(t-1)) + v(t-w), \tag{12}$$

that does not require the use of residuals $\delta\boldsymbol{x}(t)$. Here $v^{(f)}(t) = \zeta(\boldsymbol{y}^{(f)}(t)))$. The latter equation is only approximate because of the moving average filter.

3 Results

The systems we analyze are the Lorenz 1963 attractor (Lorenz, 1963) with the classical parameters, discretized with a Euler scheme and a $dt = 0.001$, the Pomeau-Manneville intermittent map (Manneville, 1980), the Lorenz 1996 equations (Lorenz, 1996) and the NCEP sea-level pressure data (Saha et al., 2014).

Lorenz 1963 equations

The Lorenz system is a simplified model of Rayleigh-Benard convection, derived by E.N. Lorenz (Lorenz, 1963). It is an autonomous continuous dynamical system with three variables $\{x, y, z\}$ parametrizing respectively the convective motion, the horizontal temperature gradient and the vertical temperature gradient. It writes:

$$\begin{aligned} \frac{dx}{dt} &= \sigma(y-x) + \epsilon\xi_x(t) \\ \frac{dy}{dt} &= -xz + \varrho x - y + \epsilon\xi_y(t), \\ \frac{dz}{dt} &= xy - bz + \epsilon\xi_z(t), \end{aligned} \tag{13}$$

where σ, ϱ and b are three parameters, σ mimicking the Prandtl number and ϱ the reduced Rayleigh number and b the geometry of convection cells. The Lorenz model is usually defined using Eq. (13), with $\sigma = 10$, $\varrho = 28$ and $b = 8/3$. A deterministic trajectory of the system is shown in Figure 1a). It has been obtained via integrating numerically the Lorenz equations with an Euler scheme ($dt = 0.001$). We are aware that advanced time stepper (e.g. Runge Kutta) would provide better accuracy. However, when considering daily or 6-hourly data, as commonly done in climate sciences and analyses, we hardly are in the case of a smooth time stepper. We therefore stick to the Euler method for similarity with the climate data used in the last section of

the paper. The systems is perturbed via additive noise: $\xi_x(t), \xi_y(t)$ and $\xi_z(t)$ are independent and identically distributed (i.i.d.) random variables all drawn from a Gaussian distribution. The initial conditions are randomly selected within a long trajectory of $5{\cdot}10^6$ iterations. First, we study the dependence of the ESN on the training length in the deterministic system ($\epsilon = 0$, Figure 1b-d). We analyse the behavior of the rejection rate ϕ (panel b), the predictability horizon τ_s (panel c) and the initial error η (panel d) as a function of the training sample size. Our analysis suggests that $t \sim 10^2$ is a minimum sufficient choice for the training window. We compare this time to the typical time scales of the motion of the sytems, determined via the maximum Lyapunov exponent λ. For the Lorenz 1963 system, $\lambda = 0.9$, so that the Lyapunov time $\vartheta \approx \mathcal{O}\left(\frac{1}{\lambda}\right) \approx 1.1$. From the previous analysis we should train the network at least for $t > 100\vartheta$. For the other systems analysed in this article, we take this condition as a lower boundary for the training times.

To exemplify the effectiveness of the moving average filter in improving the machine learning performances, in Figure 2 we show 10 ESN trajectories obtained without moving average (black) and with (red) a moving average window $w = 0.01$ and compare them to the reference trajectory (blue) obtained with $\epsilon = 0.1$. The value of $w = 10dt = 0.01$ respects the condition $w \ll \vartheta$. Indeed, the APE averaged over the two groups of trajectories (Figure 2-b) shows an evident gain of accuracy (a factor of ~ 10) when the moving average procedure is applied. We now study in a more systematic way the dependence of the ESN performance on noise intensity ϵ, network size N and for three different averaging windows $w = 0$, $w = 0.01$, $w = 0.05$. We produce, for each combination, 100 ESN forecasts. Figure 3 shows ϕ (a), $\log(\tau_{s=1})$ (b) and $\log(\eta)$ (c) computed setting $u \equiv x$ variable of the Lorenz 1963 system (results qualitatively do not depend on the chosen variable). In each panel from left to right the moving average window is increasing, upper sub-Panels are obtained using the exact expression in Eq. 12 and lower panels using the residuals in Eq 10. For increasing noise intensity and for small reservoirs sizes, the performances without moving average (left subpanels) rapidly get worse. The moving average smoothing with $w = 0.01$ (central sub-panels) improves the performance for $\log(\tau_{s=1})$ (b) and $\log(\eta)$ (c), except when the noise is too large ($\epsilon = 1$). Hereafter we denote with $\log$ the natural logarithm. When the moving average window is too large (right panels), the performances of ϕ decrease. This failure can be attributed to the fact that residuals $\delta\boldsymbol{x}$ (Eq.10) are of the same order of magnitude of the ESN predicted fields for ϵ large. Indeed, if we use the formula provided in Eq. 12 as an alternative to step 6, we can evaluate the error introduced in the residuals. The results shown in Figure 3 suggest that residuals can be used without problems when the noise is small compared with the dynamics. When ϵ is close to one, the residuals overlay the deterministic dynamics and ESN forecast are poor. In this case, the exact formulation in Eq. 12 appears much better.

Pomeau-Manneville intermittent map

Several dynamical systems, including Earth climate, display intermittency, i.e., the time series of a variable issued by the system can experience sudden chaotic fluctuations, as well as a predictable behavior where the observables have small fluctuations. In atmospheric dynamics, such a behavior is observed in the switching between zonal and meridional phases of the mid-latitude dynamics if a time series of the wind speed at one location is observed: when a cyclonic structure passes through the area, the wind has high values and large fluctuations, when an anticyclonic structure is present the wind is low and fluctuations are

smaller (Weeks et al., 1997; Faranda et al., 2016). It is then of practical interest to study the performance of ESN in Pomeau Manneville predictions as they are a first prototypical example of the intermittent behavior found in climate data.

In particular, the Pomeau-Manneville (Manneville, 1980) map is probably the simplest example of intermittent behavior, produced by a 1D discrete deterministic map given by:

$$x_{t+1} = \mathrm{mod}(x_t + x_t^{1+a}, 1) + \epsilon\xi(t), \tag{14}$$

where $0 < a < 1$ is a parameter. We use $a = 0.91$ in this study and a trajectory consisting of $5{\times}10^5$ iterations. The system is perturbed via additive noise $\xi(t)$ drawn from a Gaussian distribution, and initial conditions are randomly drawn by a long trajectory, as for the Lorenz 1963 system. It is well known that Pomeau-Manneville systems exhibit sub-exponential separation of nearby trajectories and then the Lyapunov exponent is $\lambda = 0$. However, one can define a Lyapunov exponent for the non-ergodic phase of the dynamics and extract a characteristic time scale (Korabel and Barkai, 2009). From this latter reference, we can derive a value $\lambda \simeq 0.2$ for $a = 0.91$, implying $w < \tau \simeq 5$. For the Pomeau-Manneville map, we set $u(t) \equiv x(t)$. We find that the best match between ESN and equations in terms of the ϕ indicator are obtained for $w = 3$.

Results for the Pomeau-Manneville map are shown in Figure 4. We first observe that the ESN forecast of the intermittent dynamics of the Pomeau-Manneville map is much more challenging than for the Lorenz system as a consequence of the deterministic intermittent behavior of this system. For the simulations performed with $w = 0$, the ESN cannot simulate an intermittent behavior, for all noise intensities and reservoir sizes. This is reflected in the behavior of the indicators. In the deterministic limit, the ESN fails to reproduce the invariant density in 80% of the cases ($\phi \simeq 0.8$). We can therefore speculate that there is an intrinsic problem in reproducing intermittency driven by the deterministic chaos. For intermediate noise intensities $\phi > 0.9$ (Figure 4-a). The predictability horizon $log(\tau_{s=0.5})$ for the short term forecast is small (Figure 4d) and the initial error large (Figure 4g). It appears that in smaller networks, the ESN keeps better track of the initial conditions, so that the ensemble shows smaller divergences $log(\tau_{s=0.5})$. The moving average procedure with $w = 3$ partially improves the performances (Figure 4b,c,e,f,h,i) and it enables ESN to simulate an intermittent behavior (Figure 5). Performances are again better when using the exact formula in Eq. 12 (Figure 4b,e,h) than using the residuals $\delta\boldsymbol{x}$ (Figure 4c,f,i). Figure 5a) shows the intermittent behavior of the data generated with the ESN trained on moving averaged data of Pomeau-Manneville system (red) and compare to the target time series (blue). ESN simulations do not reproduce the intermittency in the average of the target signal, which shift from $x \sim 0$ in the non intermittent phase to $0.2 < x < 1$ in the intermittent. ESN simulations only show some second order intermittency in the fluctuations while keeping a constant average. Figure 5b) displays the power spectra showing in both cases a power law decay, which are typical of turbulent phenomena. Although the intermittent behavior is captured, this realization of ESN shows that the values are concentrated around $x = 0.5$ for the ESN prediction, whereas the non-intermittent phase peaks around $x = 0$ for the target data.

The Lorenz 1996 system

Before running the ESN algorithm on actual climate data, we test our idea in a more sophisticated, and yet still idealized, model of atmospheric dynamics, namely the Lorenz 1996 equations (Lorenz, 1996). This model explicitly separates two scales and therefore will provide a good test for our ESN algorithm. The Lorenz 1996 system consists of a lattice of large-scale resolved variables X, coupled to small-scale variables Y, whose dynamics can be intermittent. The model is defined via two sets of equations:

$$\begin{aligned} \frac{dX_i}{dt} &= X_{i-1}(X_{i+1} - X_{i-2}) - X_i + F - \frac{hc}{b}\sum_{j=1}^{J} Y_{j,i}, \\ \frac{dY_{j,i}}{dt} &= cbY_{j+1,i}(Y_{j-1,i} - Y_{j+2,i}) - cY_{j,i} + \frac{hc}{b}X_i \end{aligned} \tag{15}$$

where $i = 1, \ldots, I$ and $j = 1, 2, \ldots, J$ denote respectively the number of large-scale X and small-scale Y variables. Large-scale variables are meant to represent the meanders of the jet-stream driving the weather at mid-latitudes. The first term on the right-hand side represents advection, the second diffusion, while F mimics an external forcing. The system is controlled via the parameters b and c (the time scale of the fast variables compared to the small variables) and via h (the coupling between large and small scales). From now on, we fix $I = 30, J = 5$ and $F = b = 10$ as these parameters are typically used to explore the behavior of the system (Frank et al., 2014). We integrate the equations with an Euler scheme ($dt = 10^{-3}$) from the initial conditions $Y_{j,i} = X_i = F$, where only one mode is perturbed as $X_{i=1} = F + \varepsilon$ and $Y_{j,i=1} = F + \varepsilon^2$. Here $\varepsilon = 10^{-3}$. We discard about $2 \cdot 10^3$ iterations to reach a stationary state on the attractor, and we retain $5 \cdot 10^4$ iterations. When c and h vary, different interactions between large and small scales can be achieved. A few examples of simulations of the first mode X_1 and Y_1 are given in Figure 6. Figure 6a,c show simulations obtained for $h = 1$ by varying c: the larger c the more intermittent the behavior of the fast scales. Figure 6.b,d) show simulations obtained for different coupling h at fixed $c = 10$: when $h = 0$, there is no small-scale dynamics.

For the Lorenz 1996 model, we do not need to apply a moving average filter to the data, as we can train the ESN on the large-scale variables only. Indeed, we can explore what happens to the ESN performances if we turn on and off intermittency and/or the small-to-large-scale coupling, without introducing any additional noise term. Moreover, we can also learn the Lorenz 1996 dynamics on the X variables only, or learn the dynamics on both X and Y variables. The purpose of this analysis is to assess whether the ESN are capable of learning the dynamics of the large-scale variables X alone, and how this capability is influenced by the coupling and the intermittency of the small-scale variables Y. Using the same simulations presented in Figure 6, we train the ESN on the first $2.5 \cdot 10^4$ iterations, and then perform, changing the initial conditions 100 different ESN predictions for $2.5 \cdot 10^4$ more iterations. We apply our performance indicators not to the entire I-dimensional X variable $(X_1, \ldots, X_I)$, as the χ^2 test becomes intractable in high dimensions, but rather to the average of the large-scale variables X. Consistently with our notation, it means that $u(t) \equiv \sum_{i=1}^{I} X_i(t)$ The behavior of each variable X_i is similar, so the average is representative of the collective behavior. The rate of failure ϕ is very high (not shown) because even when the dynamics is well captured by the ESN in terms of characteristics time and spatial scales, the predicted variables are not scaled and centered

as those of the original systems. For the following analysis, we therefore replace ϕ with the χ^2 distance Σ (Eq. (5)). The use of Σ allows for better highlighting the differences in the ESN performance with respect to the chosen parameters. The same considerations also apply to the analysis of the sea-level pressure data reported in the next paragraph.

Results of the ESN simulations for the Lorenz 1996 system are reported in Figure 7. In Figure 7a,c,e) ESN predictions are obtained by varying c at fixed $h = 1$, while in Figure 7b,d,f) by varying h at fixed $c = 10$. The continuous lines refer to results obtained feeding the ESN with only the X variables, dotted lines with both X and Y. For the χ^2 distance Σ (Figure 7a,b), performances show a large dependence on both intermittency c and coupling h. First of all, we remark that learning both X and Y variables lead to higher distances Σ, except for the non intermittent case, $c = 1$. For $c > 1$, the dynamics learnt on both X and Y never settles on a stationary state resembling that of the Lorenz 1996 model. When $c > 1$ and only the dynamics of the X variables is learnt, the dependence on N when h is varied is non monotonic and better performances are achieved for $800 < N < 1200$. For this range, the dynamics settles on stationary states whose spatio-temporal evolution resembles that of the Lorenz 1996 model, although the variability of time and spatial scales is different from the target. An example is provided in Figure 8, for $N = 800$. This figure shows an average example of the performance of ESN in reproducing Lorenz 1996 system when the fit succeeds. For comparison, we refer to the results by Vlachas et al. (2020) which shows that better fits of the Lorenz 1996 dynamics can be obtained using back-propagation algorithms.

Let us now analyse the two indicators of short-term forecasts. Figure 7c,d) display the predictability horizon τ_s with $s = 1$. The best performances are achieved for the non-intermittent case $c = 1$ and learning both X and Y. When only X is learnt, we again get better performances in terms of τ_s for rather small network sizes. The performances for $c > 1$ are better when only X variables are learnt. The good performance of ESN in learning only the large-scale variables X are even more surprising when looking at initial error η (Figure 7), which is one order of magnitude smaller when X, Y are learnt. Despite this advantage in the initial conditions, the ESN performances on (X, Y) are better only when the dynamics of Y is non-intermittent. We find clear indications that large intermittency ($c = 25$) and strong small-to-large scale variables coupling ($h = 1$) worsen the ESN performances, supporting the claims made for the Lorenz 1963 and the Pomeau-Manneville systems.

The NCEP sea-level pressure data

We now test the effectiveness of the moving average procedure in learning the behavior of multiscale and intermittent systems on climate data issued by reanalysis projects. We use data from the National Centers for Environmental Prediction (NCEP) version 2 (Saha et al., 2014) with a horizontal resolution of 2.5°. We adopt the global 6 hourly sea-level pressure (SLP) field from 1979 to 31/08/2019 as the meteorological variable proxy for the atmospheric circulation. It traces cyclones (resp. anti-cyclones) with minima (resp. maxima) of the SLP fields. The major modes of variability affecting mid-latitudes weather are often defined in terms of the Empirical Orthogonal Functions (EOF) of SLP and a wealth of other atmospheric features (Hurrell, 1995; Moore et al., 2013), ranging from teleconnection patterns to storm track activity to atmospheric blocking can be

diagnosed from the SLP field.

The dataset consists therefore of a gridded time series $SLP(t)$, consisting of $\sim$ 33000 time realization of the pressure field over a grid of spatial size 72 longitudes $\times$73 latitudes. Our observable $u(t) \equiv \langle SLP(t) \rangle_{lon,lat}$ where brackets indicate spatial average. In addition to the time moving average filter, we also investigate the effect of spatial coarse-graining the SLP fields by a factor c and perform the learning on the reduced fields. We use the nearest neighbor approximation, which consist in taking from the original dataset the closest value to the coarse grid. Compared with methods based on averaging or dimension reduction techniques such as EOFs, the nearest neighbors approach has the advantage of not removing the extremes (except if the extreme is not in one of the closest gridpoint) and preserve cyclonic and anticyclonic structures. For $c = 2$ we obtain a horizontal resolution of 5° and for $c = 4$ a resolution 10°. For $c = 4$ the information on the SLP field close to the poles is lost. However, in the remaining of the geographical domain, the coarse grained fields still capture the positions of cyclonic and anticyclonic structures. Indeed, as shown in Faranda et al. (2017), this coarse grain field still preserves the dynamical properties of the original one. There is therefore a certain amount of redundant information on the original 2.5° horizontal resolution SLP fields.

The dependence of the quality of the prediction for the sea-level pressure NCEPv2 data on the coarse graining factor c and on the moving average window size w is shown in Figure 9. We show the results obtained using the residuals (Eq. 10) as the exact method is not straightforwardly adaptable to systems with both spatial and temporal components. Figure 9a-c show the distance from the invariant density, using the χ^2 distance Σ. Here it is clear that by increasing w, we get better forecast with smaller network sizes N. A large difference for the predictability expressed as predictability horizon τ_s, $s = 1.5$ hPa (Figure 9d-f) emerges when SLP fields are coarse grained. We gain up to 10h in the predictability horizon with respect to the forecasts performed on the original fields ($c = 0$). This gain is also reflected by the initial error η (Figure 9g-i). Note also that Σ blow-up for larger N is related to the long-time instability of the ESN. The blow only affects global indicator Σ and not τ_s and η which are computed as short term properties. From the combination of all the indicators, after a visual inspection, we can identify the best-set of parameters: $w = 12$ h, $N = 200$ and $c = 4$. Indeed this is the case such that, with the smallest network we get almost the minimal χ^2 distance T, the highest predictability (32 h) and one of the lowest initial errors. We also remark that, for $c = 0$ (panels (c) and (i)), the fit always diverges for small network sizes.

We compare in details the results obtained for two 10-year predictions with $w = 0$h and $w = 12$h at $N = 200$ and $c = 4$ fixed. At the beginning of the forecast time (Supplementary Video 1), the target field (panel a) is close to both that obtained with $w = 0$h (panel b) and $w = 12$h (panel c). When looking at a very late time (Supplementary Video 2), of course we do not expect to see agreement among the three datasets. Indeed we are well beyond the predictability horizon. However, we remark that the dynamics for the run with $w = 0$h is steady: positions of cyclones and anticyclones barely evolve with time. Instead, the run with $w = 12$h shows a richer dynamical evolution with generation and annihilation of cyclones. A similar effect can be observed in the ESN prediction of the Lorenz 96 system shown in Figure 8b) where the quasi-horizontal patterns indicate less spatial mobility than the original system (Figure 8a).

In order to assess the performances of the two ESNs with and without moving average in a more quantitative way, we present the probability density functions for $u(t) \equiv \langle SLP(t) \rangle_{lon,lat}$ in Figure 10a. The distribution obtained for the moving average $w = 12$h matches better than the run $w = 0$h that of the target data. Figure 10b-d shows the Fourier power spectra for the target data, with the typical decay of turbulent climate signal. The non-filetered ESN simulation $W = 0$ show a spectrum with very low energy for high frequency and an absence of the daily cycle (no peak at value 10^0). The simulation with $w = 12$h also shows a lower energy for weekly or monthly time-scales but it is the correct peak for the daily cycle and the right energy at subdaily time scales. Therefore, also the spectral analysis shows a real improvment in using moving average data.

4 Discussion

We have analysed the performance of ESN in reproducing both the short and long-term dynamics of observables of geophysical flows. The motivation for this study came from the evidence that a straightforward application of ESN to high dimensional geophysical data (such as the 6 hourly global gridded sea-level pressure data) does not yield to the same results quality obtained by (Pathak et al., 2018) for the Lorenz 1963 and the Kuramoto-Sivashinsky models. Here we have investigated the causes for this behavior and identified two main bottlenecks: (i) intermittency and (ii) the presence of multiple dynamical scales, which both appear in geophysical data. In order to illustrate this effect, we have first analysed two low dimensional systems, namely the Lorenz (1963) and the Manneville (1980) equation. To mimic multiple dynamical scales, we have added noise terms to the dynamics. The performance of ESN in predicting rapidly drops when the systems are perturbed with noise. Filtering the noise allows to partially recover predictability. It also enables to simulate some qualitative intermittent behavior in the Pomeau-Manneville dynamics. This feature could be explored by changing the degree of intermittency in the Pomeau-Manneville map as well as performing parameter tuning in ESN. This is left for future work. Our study also suggests that deterministic ESN with smooth, continuous activation function cannot be expected to produce trajectories that look spiking/stochastic/rapidly changing. Most previous studies on ESNs (e.g., Pathak et al., 2018) were handling relatively smooth signals, and not such rapidly changing signals. Although it does not come as a surprise that utilizing the ESN on the time averaged dynamics and then adding a stochastic residual improves performance, the main insights is the intricate dependence of the ESN performance on the noise structure and the fact that, even for non-smooth signal, ESN with hyperbolic tangent functions can be used to study systems that have a intermittent or multiscale dynamics. Here we have used a simple moving-average filter and shown that a careful choice of the moving-average window can enhance predictability. As an intermediate step between the low-dimensional models and the application to the sea-level pressure data, we have analysed the ESN performances on the Lorenz (1996) system. This system was introduced to mimic the behavior of the atmospheric jet at mid-latitude, and features a lattice of large-scale variables, each connected to small-scale variables. Both the coupling between large and small scales and intermittency can be tuned in the model, giving rise to a plethora of behaviors. For the Lorenz 1996 model, we did not have to apply a moving average filter to the data, as we can train the ESN on the large-scale variables only. Our computations have shown that, whenever the small scales are intermittent, or the coupling is strong, learning the dynamics of the coarse grained variable is more effective, both in terms of computation time and performances. The results also apply to geophysical datasets:

here we analysed the atmospheric circulation, represented by sea-level pressure fields. Again we have shown that both a spatial coarse-graining and a time moving-average filter improve the ESN perfomances.

Our results may appear rather counter-intuitive, as the weather and climate modelling communities are moving towards extending simulations of physical processes to small scales. As an example, we cite the use of highly-resolved convection-permitting simulations (Fosser et al., 2015) as well as the use of stochastic (and therefore non-smooth) parameterizations in weather models (Weisheimer et al., 2014). We have, however, a few heuristic arguments on why the coarse-gaining and filtering operations should improve the ESN performances. First, the moving-average operation helps both in smoothing the signal and by providing the ESN with a wider temporal information. In some sense, this is reminiscent of the embedding procedure (Cao, 1997), where the signal behavior is reconstructed by providing not only information on the previous time step, but on previous times depending on the complexity. The filtering procedure can also be motivated by the fact that the active degrees of freedom for the sea-level pressure data are limited. This has been confirmed by Faranda et al. (2017) via coarse-graining these data and showing that the active degrees of freedom are independent on the resolution, in the same range explored in this study. Therefore, including small scales in the learning of sea-level pressure data, does not provide additional information on the dynamics and push towards over-fitting and saturating the ESN with redundant information. The latter consideration also poses some caveats on the generality of our results: we believe that this procedure is not beneficial whenever a clear separation of scales is not achievable, e.g. in non-confined 3-D turbulence. Moreover, in this study, three sources of stochasticity were present: (i) in the random matrices and reservoir, (ii) in the perturbed initial conditions and (iii) in the ESN simulations when using moving average filtered data with sampled $\delta\boldsymbol{x}$ components. The first one is inherent to the model definition. The perturbations of the starting conditions allow characterizing the sensitivity of our ESN approach to the initial conditions. The stochasticity induced by the additive noise $\delta\boldsymbol{x}$ provides a distributional forecast at each time t. Although this latter noise can be useful to simulate multiple trajectories and evaluate their long-term behaviour, in practice, i.e., in the case where an ESN would be used operationally to generate forecasts, one might not want to employ a stochastic formulation with an additive noise, but rather the explicit and deterministic formulation in Eq. 12. This exemplifies the interest of our ESN approach for possible distinction between forecasts and long-term simulations, and therefore makes it flexible to adapt to the case of interest.

Our results, obtained on ESN, should also be distinguished from those obtained using other RNN approaches. Vlachas et al. (2020); Albers et al. (2018) suggest that, although ESNs have the capability for memory, they often struggle to represent it when compared to fully-trained RNNs. This essentially defeats the purpose of ESNs, as they are supposed to *learn* memory. In particular, it will be interesting to test whether all experiments reported here could be repeated with a simple artificial neural network, Gaussian Processes regression, Random Feature Map, or other data-driven function approximator that does not have the dynamical structure of RNNs/ESNs (Cheng et al., 2008; Gottwald and Reich, 2021). In future work, it will be interesting to use other learning architectures and other methods of separating large- from small-scale components (Wold et al., 1987; Froyland et al., 2014; Kwasniok, 1996). Finally, our results give a more formal framework for applications of machine learning techniques on geophysical data. Deep-learning approaches have proven useful in performing learning at different time and

spatial scales whenever each layer is specialized in learning some specific features of the dynamics (Bolton and Zanna, 2019; Gentine et al., 2018). Indeed, several difficulties encountered in the application of machine learning on climate data could be overcome if the appropriate framework is used, but this requires a critical understanding of the limitations of the learning techniques.

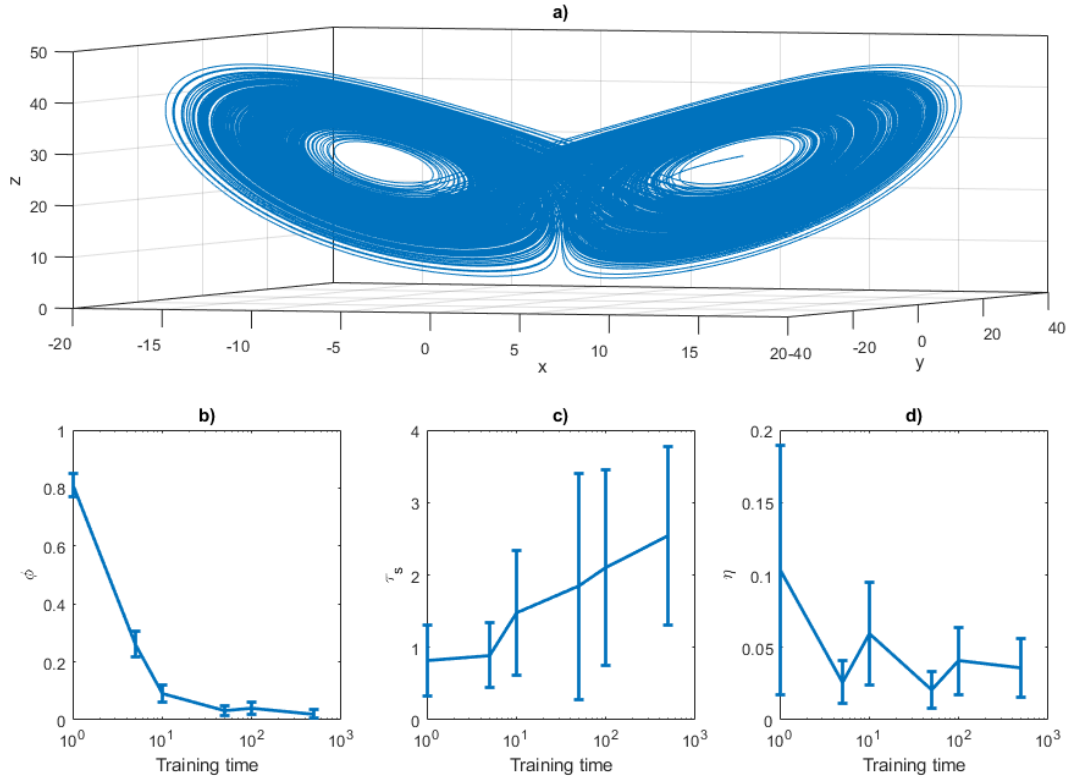

Figure 1. a) Lorenz 1963 attractor obtained with a Euler scheme with $dt = 0.001$, $\sigma = 10$, $r = 28$ and $b = 8/3$. Panels b-d) show the performances indicator as a function of the training time. b) the rejection rate ϕ of the invariant density test for the x variable; c) the first time t such that the APE$>$1; d) the initial error η. The error bar represents the average and the standard deviation of the mean over 100 realizations.

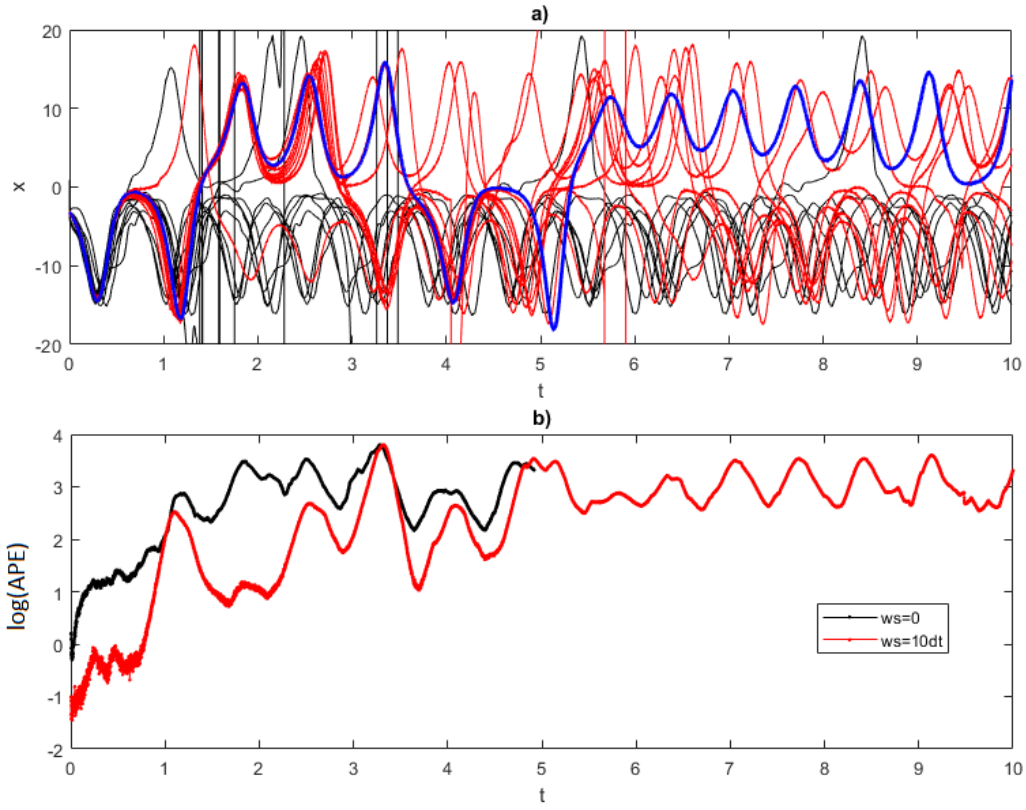

Figure 2. a) Trajectories predicted using ESN on the Lorenz 1963 attractor for the variable x. The attractor is perturbed with Gaussian noise with variance $\epsilon = 0.1$. The target trajectory is shown in blue. 10 trajectories obtained without moving average (black) show an earlier divergence compared to 10 trajectories where the moving average is performed with a window size of $w = 10dt = 0.01$ (red). Panel (b) shows the evolution of the log(APE), averaged over the trajectories for the cases with $w = 0.01$ (red) and $w = 0$ (black). The trajectories are all obtained after training the ESN for the same training set consisting of a trajectory of 10^5 time-steps. Each trajectory consists of 10^4 time steps.

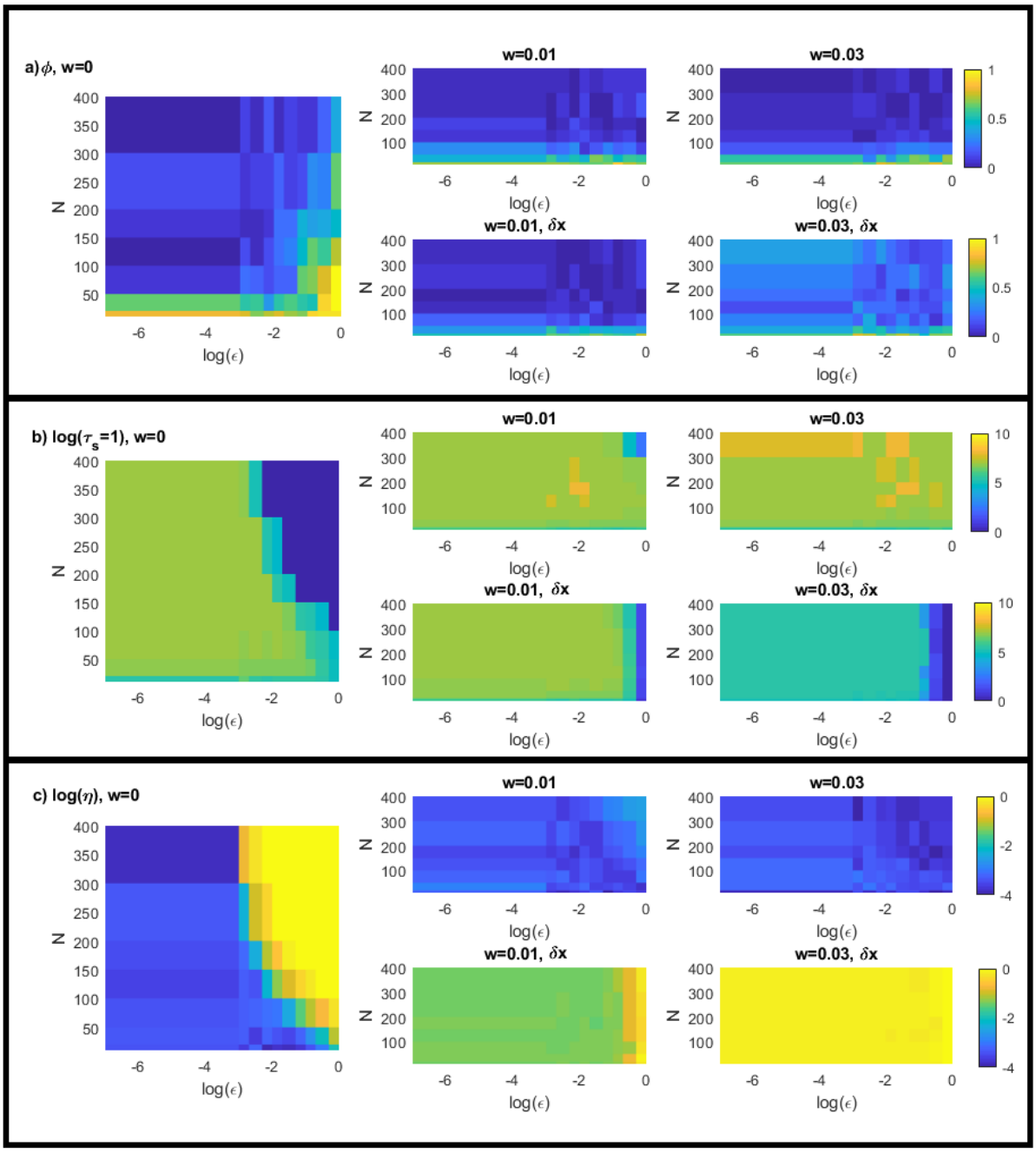

Figure 3. Lorenz 1963 analysis for increasing noise intensity ϵ (x-axes), and number of neurons N (y-axes). The colorscale represents: ϕ the rate of failure of the χ^2 test (size $\alpha = 0.05$) (a); the logarithm of predictability horizon $\log(\tau_{s=1})$ (b); the logarithm of initial error $\log(\eta)$ (c). These diagnostics have been computed on the observable $u(t) \equiv x(t)$. All the values are averages over 30 realizations. Left sub-panels refer to results without moving average, central sub-panels with averaging window $w = 0.01$, right hand-side panels with averaging window $w = 0.03$. Upper sub-panels are obtained using the exact expression in Eq. 12 and lower sub-panels using the residuals in Eq 10. The trajectories are all obtained after training the ESN for 10^5 time-steps. Each trajectory consists of 10^4 time steps.

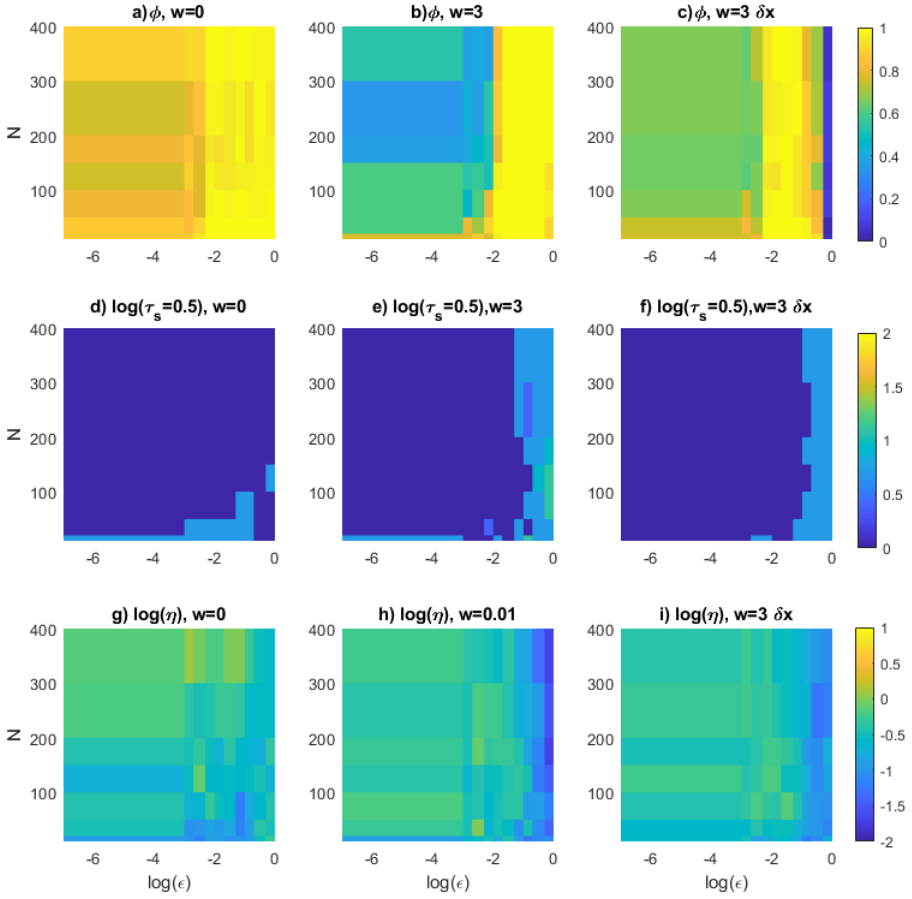

Figure 4. Analysis of the Pomeau-Manneville system for increasing noise intensity ϵ (x-axes), and number of neurons N (y-axes). The colorscale represents: ϕ the rate of failure of the χ^2 test (size $\alpha = 0.05$) (a-c); the logarithm of predictability horizon $\log(\tau_{s=0.5})$ (d-f); the logarithm of initial error $\log(\eta)$ (g-i). These diagnostics have been computed on the observable $u(t) \equiv x(t)$ All the values are averages over 30 realizations. Panels a,d,g) refer to results without moving average, b,c,e,f,h,i) with averaging window $w = 3$, c,f,i). Panels b,e,h) are obtained using the exact expression in Eq. (12) and c,f,i) using the residuals $\delta \boldsymbol{x}$ in Eq (10). The trajectories are all obtained after training the ESN for 10^5 time-steps. Each trajectory consists of 10^4 time steps.

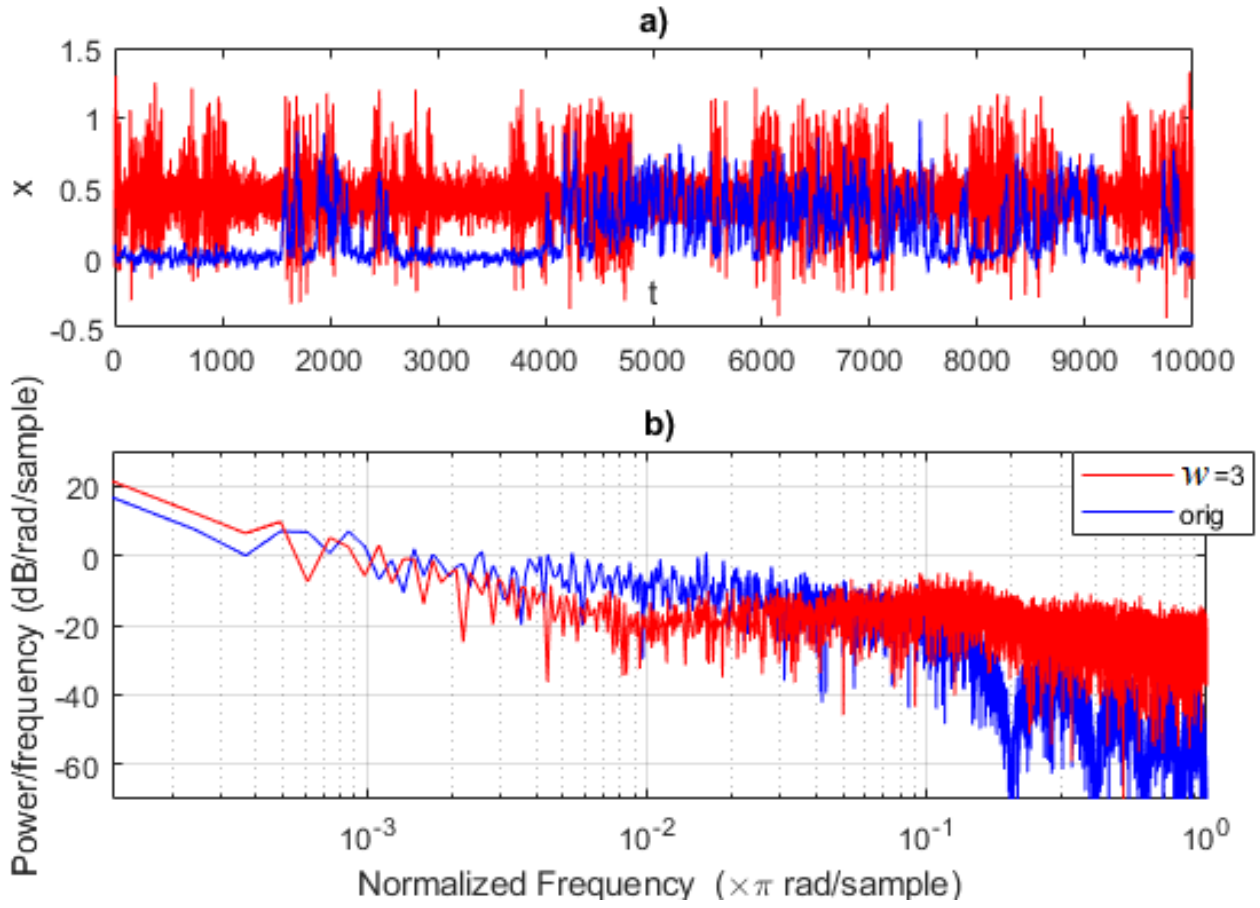

Figure 5. Pomeau-Manneville ESN simulation (red) showing an intermittent behavior and compared to the target trajectory (blue). The ESN trajectory is obtained after training the ESN for 10^5 time-steps using the moving average time series with $w = 3$. It consists of 10^4 time steps. Cases $w = 0$ are not shown as trajectories always diverge. Evolution of trajectories in time (a) and Fourier power spectra (b).

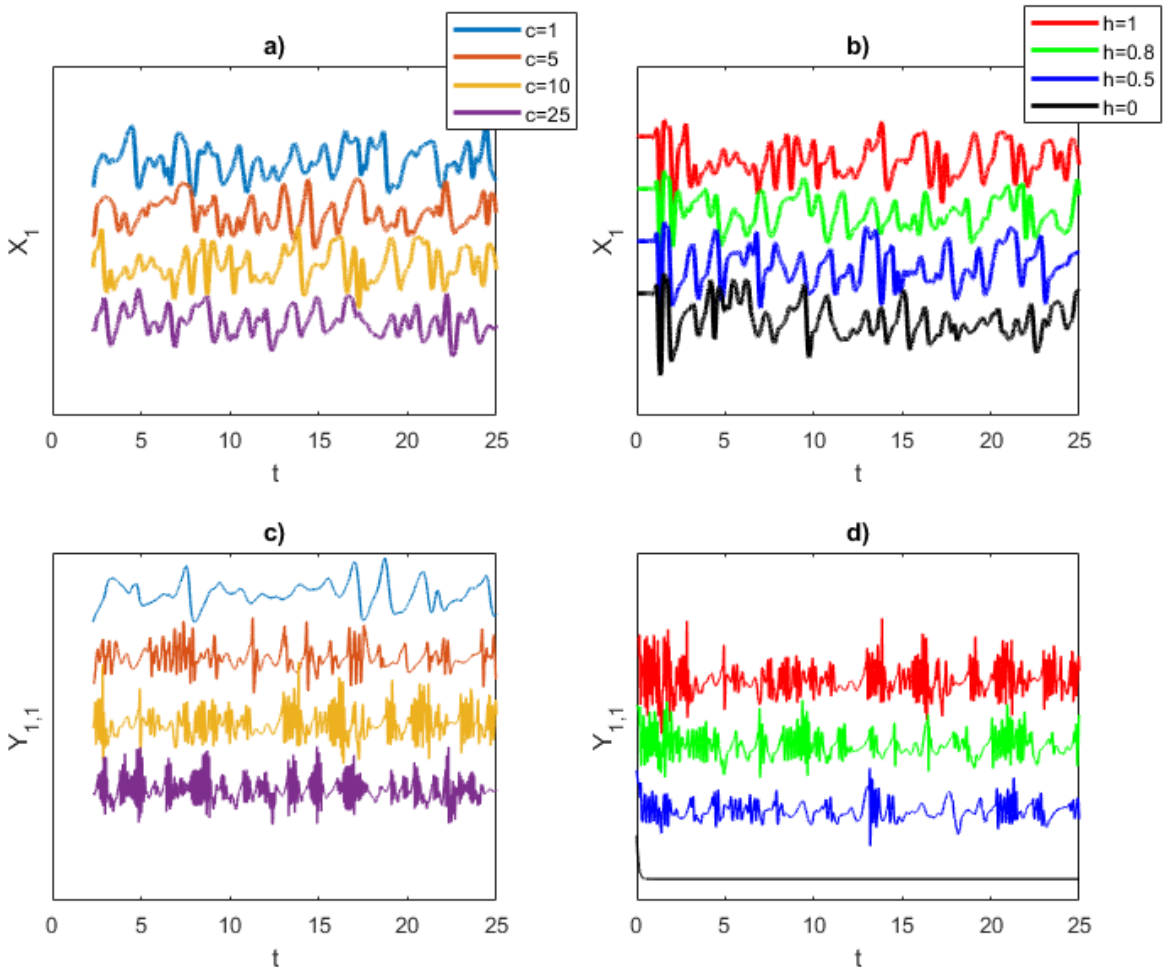

Figure 6. Lorenz 1996 simulations for the large-scale variable X_1 (a,b) and small-scale variable $Y_{1,1}$ (c,d). Panels (a,c) show simulations varying c at fixed $h = 1$. The larger c, the more intermittent the behavior of the fast scales. Panels (b,d) show simulations varying the coupling h for fixed $c = 10$. When $h = 0$, there is no small-scale dynamics. y-axes are in arbitrary units, time-series are shifted for better visibility.

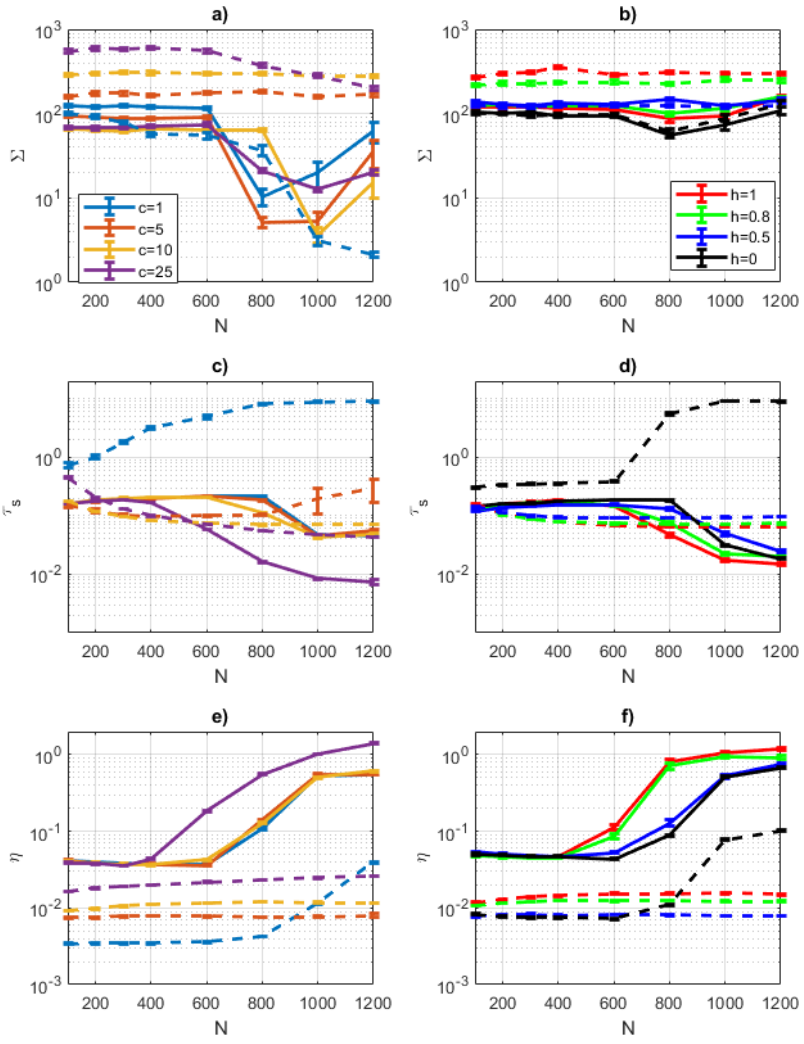

Figure 7. Lorenz 1996 ESN prediction performance for $u(t) \equiv \sum_{i=1}^{I} X_i(t)$. a,b) χ^2 distance Σ; (c,d) the predictability horizon τ_s with $s = 1$. (e,f) the initial error η in hPa. In (a,c,e) ESN predictions are made varying c at fixed $h = 1$. In (b,d,f) ESN predictions are made varying h at fixed $c = 10$. Continuous lines show ESN prediction performance made considering X variables only, dotted lines considering both X and Y variables.

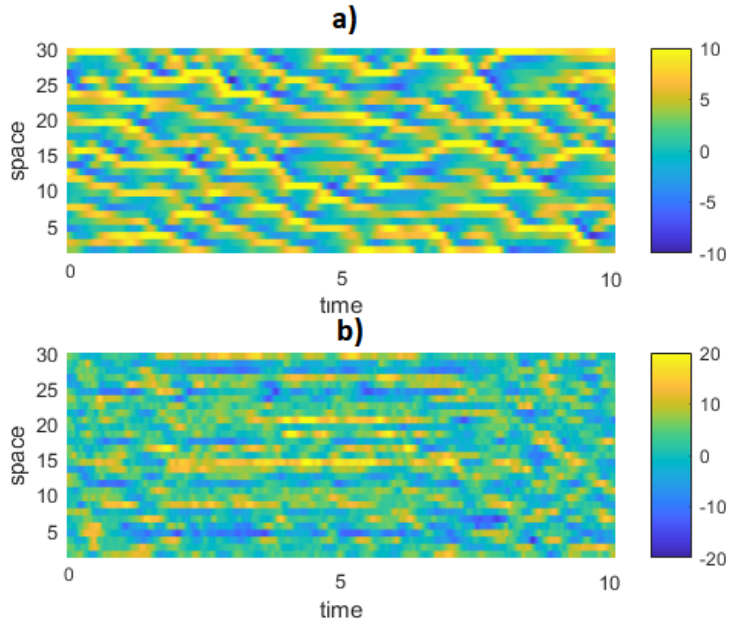

Figure 8. Example of (a) target Lorenz 1996 spatio-temporal evolution of large-scale variables X for $c = 1, h = 1$ and (b) ESN prediction realized with $N = 800$ neurons. Note that the colors are not on the same scale for the two panels.

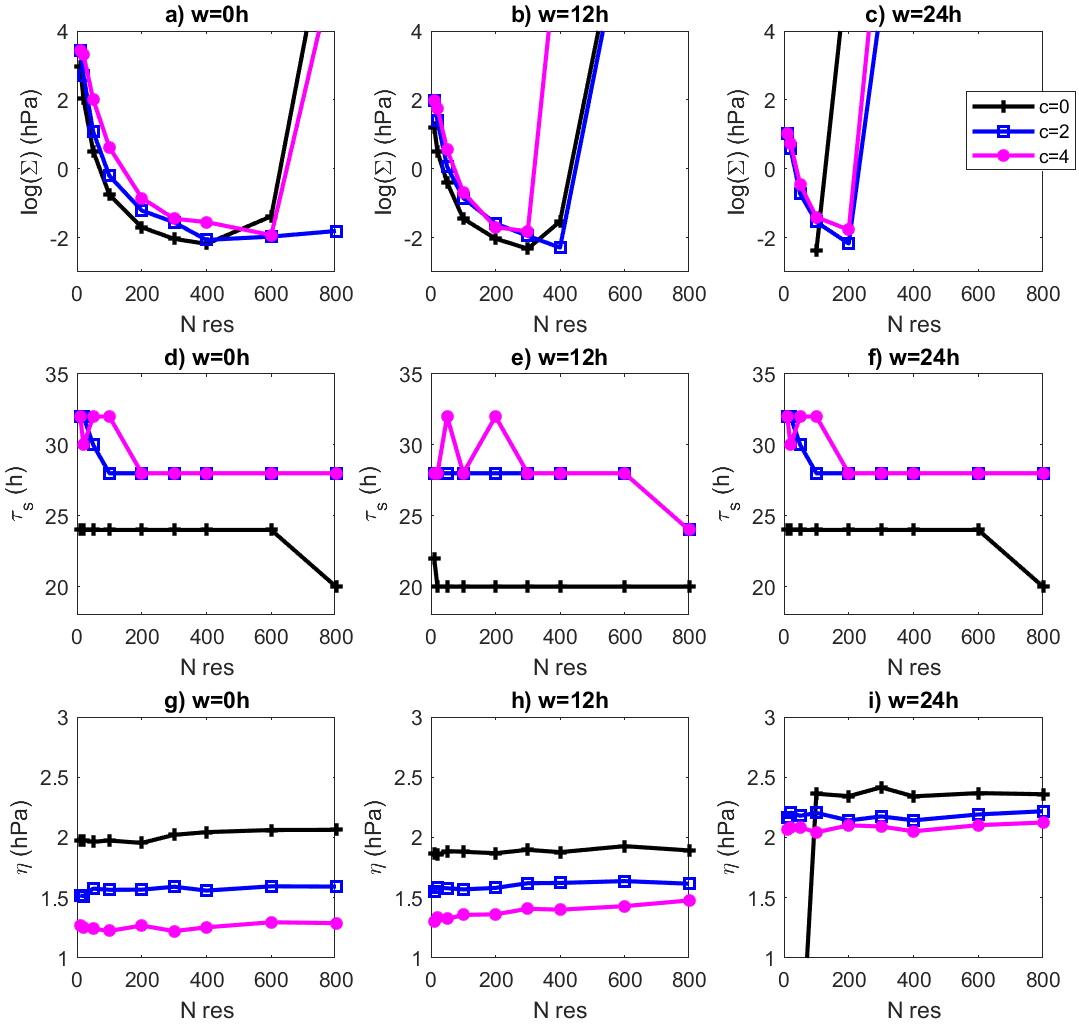

Figure 9. Dependence of the quality of the results for the prediction of the sea-level pressure NCEPv2 data on the coarse graining factor c and on the moving average window size w. The observable used is $u(t) \equiv \langle SLP(t) \rangle_{lon,lat}$. a-c) χ^2 distance $log(\Sigma)$; d-f) predictability horizon (in hours) τ_s, $s = 1.5$ hPa; g-i) logarithm of initial error η. Different coarse grain factor c are shown with different colors. a,d,g) $w = 0$, b,e,h) $w = 12$ h, c,f,i) $w = 24$ h.

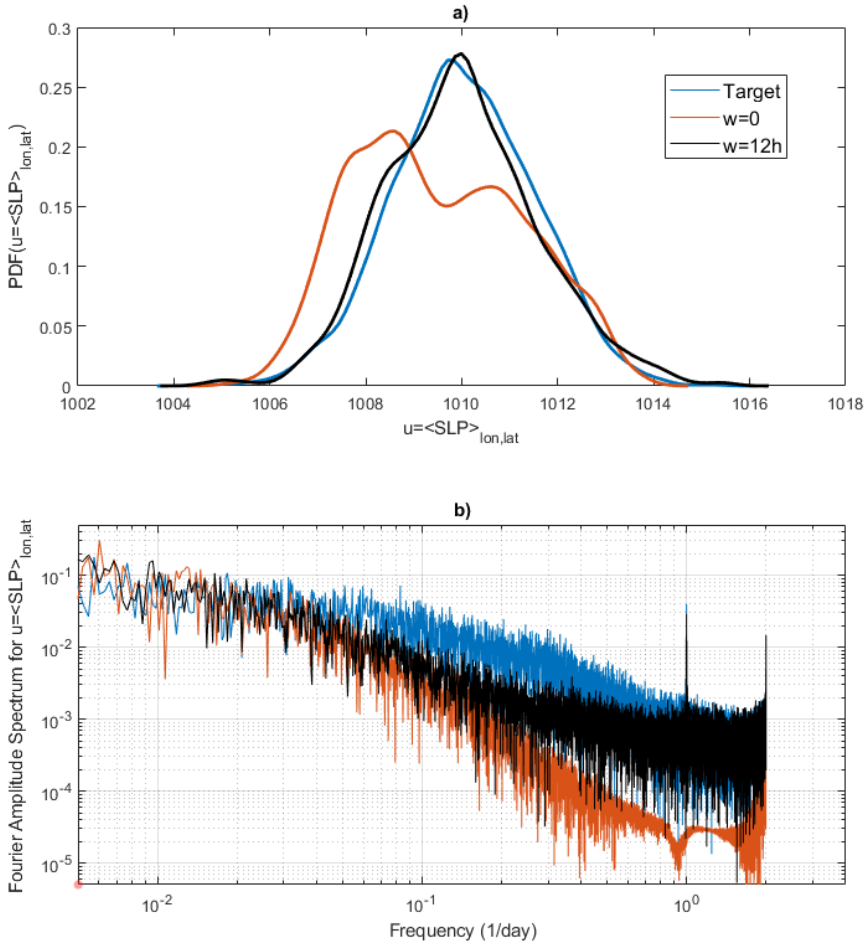

Figure 10. a) probability density function and b) Fourier power spectra for $u(t) \equiv \langle SLP(t) \rangle_{lon,lat}$ for the target NCEPv2 SLP data (blue), an ESN with $c = 4$ and $w = 0$ h (red), and an ESN with $c = 4$ and $w = 12$ h (black).

Appendix A: Numerical code

We report here the MATLAB code used for the computation of the Echo State Network. This code is adapted from the original code available here: https://mantas.info/code/simpleesn

A1 ESN Training

```
function [Win, W, Wout]=ESN_training(data,Nres)
%This function train the Echo State network using the data provided.
%INPUTS:
%data: a matrix of the input data to train, arranged as space X time
%Nres: the number of neurons N to be used in the training
%OUTPUTS:
%Win: the input weight matrix which consists of random weights
%W: the network of neurons
%Wout: the output weights, they are adjusted to match the next iterations
inSize = size(data,1);
trainLen= size(data,2);
Win = (rand(Nres,1+inSize)-0.5) .* 1;
W = rand(Nres,Nres)-0.5;
% normalizing and setting spectral radius
opt.disp = 0;
rhoW = abs(eigs(W,1,'LM',opt));
W = W .* ( 1.25 /rhoW);
% memory allocation
X = zeros(1+inSize+Nres,trainLen-1);
Yt = data(:,2:end)';
x = zeros(Nres,1);
for t = 1:trainLen-1
u = data(:,t);
x = tanh( Win*[1;u] + W*x );
X(:,t) = [1;u;x];
end
reg = 1e-8; % regularization coefficient
Wout = ((X*X' + reg*eye(1+inSize+Nres)) \ (X*Yt))';
end
```

A2 ESN Prediction

```
function [Y_pred]=ESN_prediction(data,Win, W, Wout)
% This function returns the recurrent Echo State Network prediction
%INPUT:
%data: the full data matrix of the data to predict in the form (space*time)
%Win: input weights
%W: neurons matrix
%Wout: output weights
%OUTPUT:
%Y_pred: the ESN prediction
Y_pred = zeros(size(data,1),size(data,2) );
x = zeros(size(W,1),1);
u=data(:,1);
for t = 1:size(data,2)
x = tanh( Win*[1;u] + W*x );
y = Wout*[1;u;x];
Y_pred(:,t) = y;
u = y;
end
end
```

Code and data availability. The numerical code used in this article is provided in Appendix A

Author contributions. Davide Faranda, Mathieu Vrac, Pascal Yiou and Soulivanh Thao conceived this study. Davide Faranda, Adnane Hamid, Cedric Nguounge Langue and Giulia Carella performed the analysis. Flavio Pons designed and performed the statistical tests. All the authors contributed to writing and discussing the results of the paper.

Competing interests. The authors declare that there is no conflict of interest.

Acknowledgements. We acknowledge Barbara D'Alena, Julien Brajard, Venkatramani Balaji, Berengere Dubrulle, Robert Vautard, Nikki Vercauteren, Francois Daviaud, Yuzuru Sato for useful discussions. We also acknowlegde Matthew Levine and three anonymous referes for useful suggestions on the manuscript. This work is supported by the CNRS INSU-LEFE-MANU grant "DINCLIC".

References

Albers, D. J., Levine, M. E., Stuart, A., Mamykina, L., Gluckman, B., and Hripcsak, G.: Mechanistic machine learning: how data assimilation leverages physiologic knowledge using Bayesian inference to forecast the future, infer the present, and phenotype, Journal of the American Medical Informatics Association, 25, 1392–1401, 2018.

Bassett, D. S. and Sporns, O.: Network neuroscience, Nature neuroscience, 20, 353, 2017.

Basu, A., Shioya, H., and Park, C.: Statistical inference: the minimum distance approach, Chapman and Hall/CRC, 2019.

Berkson, J. et al.: Minimum chi-square, not maximum likelihood!, The Annals of Statistics, 8, 457–487, 1980.

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, 2019.

Bonferroni, C.: Teoria statistica delle classi e calcolo delle probabilita, Pubblicazioni del R Istituto Superiore di Scienze Economiche e Commericiali di Firenze, 8, 3–62, 1936.

Bony, S., Stevens, B., Frierson, D. M., Jakob, C., Kageyama, M., Pincus, R., Shepherd, T. G., Sherwood, S. C., Siebesma, A. P., Sobel, A. H., et al.: Clouds, circulation and climate sensitivity, Nature Geoscience, 8, 261, 2015.

Brenowitz, N. D. and Bretherton, C. S.: Prognostic validation of a neural network unified physics parameterization, Geophysical Research Letters, 45, 6289–6298, 2018.

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, 2019.

Brockwell, P. J. and Davis, R. A.: Introduction to time series and forecasting, springer, 2016.

Buchanan, M.: The limits of machine prediction, Nature Physics, 15, 2019.

Cao, L.: Practical method for determining the minimum embedding dimension of a scalar time series, Physica D: Nonlinear Phenomena, 110, 43–50, 1997.

Cheng, J., Wang, Z., and Pollastri, G.: A neural network approach to ordinal regression, in: 2008 IEEE international joint conference on neural networks (IEEE world congress on computational intelligence), pp. 1279–1284, IEEE, 2008.

Cochran, W. G.: The $\chi 2$ test of goodness of fit, The Annals of Mathematical Statistics, pp. 315–345, 1952.

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, 2018.

Eckmann, J.-P. and Ruelle, D.: Ergodic theory of chaos and strange attractors, in: The theory of chaotic attractors, pp. 273–312, Springer, 1985.

Eguchi, S. and Copas, J.: Interpreting kullback–leibler divergence with the neyman–pearson lemma, Journal of Multivariate Analysis, 97, 2034–2040, 2006.

Eskridge, R. E., Ku, J. Y., Rao, S. T., Porter, P. S., and Zurbenko, I. G.: Separating different scales of motion in time series of meteorological variables, Bulletin of the American Meteorological Society, 78, 1473–1484, 1997.

Faranda, D., Lucarini, V., Turchetti, G., and Vaienti, S.: Generalized extreme value distribution parameters as dynamical indicators of stability, International Journal of Bifurcation and Chaos, 22, 1250 276, 2012.

Faranda, D., Masato, G., Moloney, N., Sato, Y., Daviaud, F., Dubrulle, B., and Yiou, P.: The switching between zonal and blocked mid-latitude atmospheric circulation: a dynamical system perspective, Climate Dynamics, 47, 1587–1599, 2016.

Faranda, D., Messori, G., and Yiou, P.: Dynamical proxies of North Atlantic predictability and extremes, Scientific reports, 7, 41 278, 2017.

Fosser, G., Khodayar, S., and Berg, P.: Benefit of convection permitting climate model simulations in the representation of convective precipitation, Climate Dynamics, 44, 45–60, 2015.

Frank, M. R., Mitchell, L., Dodds, P. S., and Danforth, C. M.: Standing swells surveyed showing surprisingly stable solutions for the Lorenz'96 model, International Journal of Bifurcation and Chaos, 24, 1430 027, 2014.

Froyland, G., Gottwald, G. A., and Hammerlindl, A.: A computational method to extract macroscopic variables and their dynamics in multiscale systems, SIAM Journal on Applied Dynamical Systems, 13, 1816–1846, 2014.

Gentine, P., Pritchard, M., Rasp, S., Reinaudi, G., and Yacalis, G.: Could machine learning break the convection parameterization deadlock?, Geophysical Research Letters, 45, 5742–5751, 2018.

Gettelman, A., Gagne, D. J., Chen, C.-C., Christensen, M., Lebo, Z., Morrison, H., and Gantos, G.: Machine Learning the Warm Rain Process, 2020.

Gottwald, G. A. and Reich, S.: Supervised learning from noisy observations: Combining machine-learning techniques with data assimilation, Physica D: Nonlinear Phenomena, 423, 132 911, 2021.

Grover, A., Kapoor, A., and Horvitz, E.: A deep hybrid model for weather forecasting, in: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 379–386, ACM, 2015.

Hastie, T., Tibshirani, R., and Wainwright, M.: Statistical learning with sparsity: the lasso and generalizations, Chapman and Hall/CRC, 2015.

Haupt, S. E., Cowie, J., Linden, S., McCandless, T., Kosovic, B., and Alessandrini, S.: Machine Learning for Applied Weather Prediction, in: 2018 IEEE 14th International Conference on e-Science (e-Science), pp. 276–277, IEEE, 2018.

Hinaut, X.: Réseau de neurones récurrent pour le traitement de séquences abstraites et de structures grammaticales, avec une application aux interactions homme-robot, Ph.D. thesis, Thèse de doctorat, Université Claude Bernard Lyon 1, 2013.

Hurrell, J. W.: Decadal trends in the North Atlantic Oscillation: regional temperatures and precipitation, Science, 269, 676–679, 1995.

Hyman, J. M. and Nicolaenko, B.: The Kuramoto-Sivashinsky equation: a bridge between PDE's and dynamical systems, Physica D: Nonlinear Phenomena, 18, 113–126, 1986.

Jaeger, H.: The "echo state" approach to analysing and training recurrent neural networks-with an erratum note, Bonn, Germany: German National Research Center for Information Technology GMD Technical Report, 148, 13, 2001.

Jordan, M. I. and Mitchell, T. M.: Machine learning: Trends, perspectives, and prospects, Science, 349, 255–260, 2015.

Korabel, N. and Barkai, E.: Pesin-type identity for intermittent dynamics with a zero Lyaponov exponent, Physical review letters, 102, 050 601, 2009.

Krasnopolsky, V. M. and Fox-Rabinovitz, M. S.: Complex hybrid models combining deterministic and machine learning components for numerical climate modeling and weather prediction, Neural Networks, 19, 122–134, 2006.

Krasnopolsky, V. M., Fox-Rabinovitz, M. S., and Chalikov, D. V.: New approach to calculation of atmospheric model physics: Accurate and fast neural network emulation of longwave radiation in a climate model, Monthly Weather Review, 133, 1370–1383, 2005.

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, Advances in Artificial Neural Systems, 2013, 2013.

Kwasniok, F.: The reduction of complex dynamical systems using principal interaction patterns, Physica D: Nonlinear Phenomena, 92, 28–60, 1996.

LeCun, Y., Bengio, Y., and Hinton, G.: Deep learning, nature, 521, 436, 2015.

Li, D., Han, M., and Wang, J.: Chaotic time series prediction based on a novel robust echo state network, IEEE Transactions on Neural Networks and Learning Systems, 23, 787–799, 2012.

Liu, J. N., Hu, Y., He, Y., Chan, P. W., and Lai, L.: Deep neural network modeling for big data weather forecasting, in: Information Granularity, Big Data, and Computational Intelligence, pp. 389–408, Springer, 2015.

Lorenz, E. N.: Deterministic nonperiodic flow, Journal of the atmospheric sciences, 20, 130–141, 1963.

Lorenz, E. N.: Predictability: A problem partly solved, in: Proc. Seminar on predictability, vol. 1, 1996.

Manneville, P.: Intermittency, self-similarity and 1/f spectrum in dissipative dynamical systems, Journal de Physique, 41, 1235–1243, 1980.

Moore, G., Renfrew, I. A., and Pickart, R. S.: Multidecadal mobility of the North Atlantic oscillation, Journal of Climate, 26, 2453–2466, 2013.

Mukhin, D., Kondrashov, D., Loskutov, E., Gavrilov, A., Feigin, A., and Ghil, M.: Predicting critical transitions in ENSO models. Part II: Spatially dependent models, Journal of Climate, 28, 1962–1976, 2015.

Paladin, G. and Vulpiani, A.: Degrees of freedom of turbulence, Physical Review A, 35, 1971, 1987.

Panichi, F. and Turchetti, G.: Lyapunov and reversibility errors for Hamiltonian flows, Chaos, Solitons & Fractals, 112, 83–91, 2018.

Pathak, J., Lu, Z., Hunt, B. R., Girvan, M., and Ott, E.: Using machine learning to replicate chaotic attractors and calculate Lyapunov exponents from data, Chaos: An Interdisciplinary Journal of Nonlinear Science, 27, 121 102, 2017.

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, Physical review letters, 120, 024 102, 2018.

Quinlan, C., Babin, B., Carr, J., and Griffin, M.: Business research methods, South Western Cengage, 2019.

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, 2018.

Saha, S., Moorthi, S., Wu, X., Wang, J., Nadiga, S., Tripp, P., Behringer, D., Hou, Y.-T., Chuang, H.-y., Iredell, M., et al.: The NCEP climate forecast system version 2, Journal of Climate, 27, 2185–2208, 2014.

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

Scher, S. and Messori, G.: Predicting weather forecast uncertainty with machine learning, Quarterly Journal of the Royal Meteorological Society, 144, 2830–2841, 2018.

Scher, S. and Messori, G.: Weather and climate forecasting with neural networks: using general circulation models (GCMs) with different complexity as a study ground, Geoscientific Model Development, 12, 2797–2809, 2019.

Schertzer, D., Lovejoy, S., Schmitt, F., Chigirinskaya, Y., and Marsan, D.: Multifractal cascade dynamics and turbulent intermittency, Fractals, 5, 427–471, 1997.

Schneider, T., Lan, S., Stuart, A., and Teixeira, J.: Earth system modeling 2.0: A blueprint for models that learn from observations and targeted high-resolution simulations, Geophysical Research Letters, 44, 12–396, 2017.

Seleznev, A., Mukhin, D., Gavrilov, A., Loskutov, E., and Feigin, A.: Bayesian framework for simulation of dynamical systems from multidimensional data using recurrent neural network, Chaos: An Interdisciplinary Journal of Nonlinear Science, 29, 123 115, 2019.

Shi, X., Gao, Z., Lausen, L., Wang, H., Yeung, D.-Y., Wong, W.-k., and Woo, W.-c.: Deep learning for precipitation nowcasting: A benchmark and a new model, in: Advances in neural information processing systems, pp. 5617–5627, 2017.

Shi, Z. and Han, M.: Support vector echo-state machine for chaotic time-series prediction, IEEE Transactions on Neural Networks, 18, 359–372, 2007.

Sprenger, M., Schemm, S., Oechslin, R., and Jenkner, J.: Nowcasting foehn wind events using the adaboost machine learning algorithm, Weather and Forecasting, 32, 1079–1099, 2017.

Vlachas, P. R., Pathak, J., Hunt, B. R., Sapsis, T. P., Girvan, M., Ott, E., and Koumoutsakos, P.: Backpropagation algorithms and reservoir computing in recurrent neural networks for the forecasting of complex spatiotemporal dynamics, Neural Networks, 126, 191–217, 2020.

Wang, J.-X., Wu, J.-L., and Xiao, H.: Physics-informed machine learning approach for reconstructing Reynolds stress modeling discrepancies based on DNS data, Physical Review Fluids, 2, 034 603, 2017.

Weeks, E. R., Tian, Y., Urbach, J., Ide, K., Swinney, H. L., and Ghil, M.: Transitions between blocked and zonal flows in a rotating annulus with topography, Science, 278, 1598–1601, 1997.

Weisheimer, A., Corti, S., Palmer, T., and Vitart, F.: Addressing model error through atmospheric stochastic physical parametrizations: impact on the coupled ECMWF seasonal forecasting system, Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 372, 20130 290, 2014.

Weyn, J. A., Durran, D. R., and Caruana, R.: Can machines learn to predict weather? Using deep learning to predict gridded 500-hPa geopotential height from historical weather data, Journal of Advances in Modeling Earth Systems, 11, 2680–2693, 2019.

Wold, S., Esbensen, K., and Geladi, P.: Principal component analysis, Chemometrics and intelligent laboratory systems, 2, 37–52, 1987.

Wolf, A., Swift, J. B., Swinney, H. L., and Vastano, J. A.: Determining Lyapunov exponents from a time series, Physica D: Nonlinear Phenomena, 16, 285–317, 1985.

Wu, J.-L., Xiao, H., and Paterson, E.: Physics-informed machine learning approach for augmenting turbulence models: A comprehensive framework, Physical Review Fluids, 3, 074 602, 2018.

Xingjian, S., 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, pp. 802–810, 2015.

Xu, M., Han, M., Qiu, T., and Lin, H.: Hybrid regularized echo state network for multivariate chaotic time series prediction, IEEE transactions on cybernetics, 49, 2305–2315, 2018.

Young, L.-S.: What are SRB measures, and which dynamical systems have them?, Journal of Statistical Physics, 108, 733–754, 2002.

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.