Advances in Land Surface Model-based Forecasting: A Comparison of LSTM, Gradient Boosting, and Feedforward Neural Networks as Prognostic State Emulators in a Case Study with ECLand

Marieke Wesselkamp[1], Matthew Chantry[2], Ewan Pinnington[2], Margarita Choulga[2], Souhail Boussetta[2], Maria Kalweit[3], Joschka Boedecker[3,4], Carsten F. Dormann[1], Florian Pappenberger[2], and Gianpaolo Balsamo[2,5]

1 Department of Biometry, University of Freiburg, Germany
2 European Centre for Medium-Range Weather Forecasts, Reading, United Kingdom
3 Department of Computer Science, University of Freiburg, Germany
4 BrainLinks-BrainTools, University of Freiburg, Germany
5 World Meteorological Organization, Geneva, Switzerland

Correspondence to: Marieke Wesselkamp (marieke.wesselkamp@biom.uni-freiburg.de)

Abstract

Most useful weather prediction for the public is near the surface. The processes that are most relevant for near-surface weather prediction are also those that are most interactive and exhibit positive feedback or have key roles in energy partitioning. Land surface models (LSMs) consider these processes together with surface heterogeneity and, when coupled with an atmospheric model, provide boundary and initial conditions. They forecast water, carbon and energy fluxes, which are an integral component of coupled atmospheric models. This numerical parametrization of atmospheric boundaries is computationally expensive and statistical surrogate models are increasingly used to accelerate experimental research. We evaluated the efficiency of three surrogate models in simulating land surface processes for speeding up experimental research. Specifically, we compared the performance of a Long-Short Term Memory (LSTM) encoder-decoder network, extreme gradient boosting, and a feed-forward neural network within a physics-informed multi-objective framework. This framework emulates key prognostic states of the ECMWF's Integrated Forecasting System (IFS) land surface scheme, ECLand, across continental and global scales. Our findings indicate that while all models on average demonstrate high accuracy over the forecast period, the LSTM network excels in continental long-range predictions when carefully tuned, XGB scores consistently high across tasks and the MLP provides an excellent implementation-time-accuracy trade-off. While their reliability is context dependent, the runtime reductions achieved by the emulators in comparison to the full numerical models are significant, offering a faster alternative for conducting experiments on land surfaces.

1 Introduction

While forecasting of climate and weather system processes has long been a task for numerical models, recent developments in deep learning have introduced competitive machine-learning (ML) systems for numerical weather prediction (NWP) (Bi et al., 2022; Lam et al., 2023; Lang et al., 2024). Land surface models (LSMs), even though being an integral part of numerical weather prediction, have not yet caught the attention of the ML-community. LSMs forecast water, carbon and energy fluxes and, in coupling with an atmospheric model, provide the lower boundary and initial conditions (Boussetta et al., 2021; De Rosnay et al., 2014). The parametrization of land surface states does not only affect predictability of earth and climate systems on sub-seasonal scales (Muñoz-Sabater et al., 2021), but also the short- and medium-range skill of NWP forecasts (De Rosnay et al., 2014). Beyond their online integration with NWPs, offline versions of LSMs provide research tools for experiments on the land surface (Boussetta et al., 2021), the diversity of which, however, are limited by substantial computational resources requirements and often moderate runtime efficiencies (Reichstein et al., 2019).

Emulators constitute statistical surrogates for numerical simulation models that, by approximating the latter, aim for increasing computational efficiency (Machac et al., 2016). While the construction of emulators can itself require substantial computational resources, their subsequent evaluation usually runs orders of magnitude faster than the original numerical model (Fer et al., 2018). For this reason, emulators have found application for example in modular parametrization of online weather forecasting systems (Chantry et al., 2021), in replacing the MCMC-sampling procedure in Bayesian calibration of ecosystem models (Fer et al., 2018), or in generating forecast ensembles of atmospheric states for uncertainty quantification (Li et al., 2023). Beyond their computational efficiency, surrogate models with high parametric flexibility have the potential to correct process mis-specification in a physical model when fine-tuned to observations (Wesselkamp et al., 2022).

Modelling approaches used for emulation range from low parametrized, auto-regressive linear models to highly non-linear and flexible neural networks (Baker et al., 2022; Chantry et al., 2021; Meyer et al., 2022; Nath et al., 2022). In the global land surface system M-MESMER, a set of simple AR1 regression models is used to initialize the numerical LSM, resulting in a modularized emulator (Nath et al., 2022). Numerical forecasts of gross primary productivity and hydrological targets were successfully approximated by Gaussian processes (Baker et al., 2022; Machac et al., 2016), the advantage of which is their direct quantification of prediction uncertainty. When it comes to highly diverse or structured data, neural networks have shown to deliver accurate approximations, for example for gravity wave drags and urban surface temperature (Chantry et al., 2021; Meyer et al., 2022). In most fields of machine learning, specific types of neural networks are now the best approach to representing fit and prediction. One exception is so-called tabular data, i.e. data without spatial or temporal interdependencies (as opposed to vision and sound), where extreme gradient boosting is still the go-to approach (Grinsztajn et al., 2022; Shwartz-Ziv and Armon, 2021).

ECLand is the land surface scheme that provides boundary and initial conditions for the Integrated Forecasting System (IFS) of the European Centre for Medium-range Weather Forecasts (ECMWF) (Boussetta et al., 2021). Driven by meteorological forcing and spatial climate fields, it has a strong influence on the NWP (De Rosnay et al., 2014) and also constitutes a standalone framework for offline forecasting of land surface processes (Muñoz-Sabater et al., 2021). The modular construction of ECLand offers potential for element-wise improvement of process representation and thus a stepwise development towards increased computational efficiency. Within the IFS, ECLand also forms the basis of the land surface data assimilation system, updating the land surface state with synoptic data and satellite observations of soil moisture and snow. Emulators of physical systems have been shown to be beneficial in data assimilation routines, allowing for a quick estimation and low maintenance of the tangent linear model (Hatfield et al., 2021). Together with the potential to run large ensembles of land surface states at a much-reduced cost, this would be a potential application of the surrogate models introduced here.

Long-short term memory networks (LSTMs) have gained popularity in hydrological forecasting as rainfall-runoff models, for predicting stream flow temperature and also soil moisture (Bassi et al., 2024; Kratzert et al., 2019b; Lees et al., 2022; Zwart et al., 2023). Research on the interpretability of LSTMs has found correlations between the model cell states and spatially or thematically similar hydrological units (Lees et al., 2022), suggesting the specific usefulness of LSTM for representing variables with dynamic storages and reservoirs (Kratzert et al., 2019a). As emulators, LSTMs have been shown useful for sea surface level projection in a variational manner with Monte Carlo dropout (Van Katwyk et al., 2023).

While most of these studies trained their models on observations or reanalysis data, our emulator learns the representation from ECLand simulations directly. To our knowledge, a comparison of models without memory mechanisms to an LSTM-based neural network for global land surface emulation has not been conducted before.

We emulate seven prognostic state variables of ECLand, which represent core land surface processes: soil water volume and soil temperature, each at three depth layers, and snow cover fraction at the surface layer. The represented variables would allow their coupling to the IFS, yet the emulators do not replace ECLand in its full capabilities. Yet, these three state variables represent the core of the current configuration of ECLand. We specifically focus on the utility of memory mechanisms, highlighting the development of a single LSTM-based encoder-decoder model compared to an extreme gradient boosting approach (XGB) and a multilayer perceptron (MLP), which all perform the same tasks. The LSTM architecture builds on an encoder-decoder network design introduced for flood forecasting (Nearing et al., 2024). To compare forecast skill systematically, the three emulators were compared in long-range forecasting against climatology (Pappenberger et al., 2015). In this work, the emulators are evaluated on ECLand simulations only, i.e. on purely synthetic data, while we anticipate their validation and fine-tuning on observations for future work.

2 Methods

2.1 The Land Surface Model: ECLand

ECLand is a tiled ECMWF Scheme for surface exchanges over land that represents surface heterogeneity and incorporates land surface hydrology (Balsamo et al., 2011; ECMWF, 2017). ECLand computes surface turbulent fluxes of heat, moisture and momentum and skin temperature over different tiles (vegetation, bare soil, snow, interception and water) and then calculates an area-weighted average for the grid-box to couple with the atmosphere (Boussetta et al., 2021). For the overall accuracy of the model, accurate land surface parameterizations are essential (Kimpson et al., 2023) as they e.g. determine the sensible and latent heat fluxes, and provide the lower boundary conditions for enthalpy and moisture equations in the atmosphere (Viterbo, 2002). We emulate three prognostic state variables of ECLand that represent core land surface processes: soil water volume ($\mathrm{m^3 m^{-3}}$) and soil temperature (K) at each three depth layers (each at 0 – 7 cm, 7 – 21 cm and 21 – 72 cm) and snow cover fraction (%), aggregated at the surface layer.

2.2 Data sources

As training data base, global simulation and reanalysis time series from 2010 to 2022 were compiled to *zarr* format at an aggregated 6-hourly temporal resolution. Simulations and climate fields were generated from ECMWFs development cycle CY49R2, ECLand forced by ERA-5 meteorological reanalysis data (Hersbach et al., 2020).

There are three main sources of data used for creation of the data base: The first is a selection of surface physiographic fields from ERA5 (Hersbach et al., 2020) and their updated versions (Boussetta et al., 2021; Choulga et al., 2019; Muñoz-Sabater et al., 2021), used as static model input features (X). The second is a selection of atmospheric and surface model fields from ERA5, used as static and dynamic model input features (Y). The third are ECLand simulations, constituting the model's dynamic prognostic state variables (z) and hence emulator input and target features. A total of 41 static, seasonal and dynamical features were used to create the emulators, see table 1 for an overview of input variables and details on the surface physiographic and atmospheric fields below.

2.2.1 Surface physiographic fields

Surface physiographic fields have gridded information of the Earth's surface properties (e.g. land use, vegetation type, and distribution) and represent surface heterogeneity in the ECLand of the IFS (Kimpson et al., 2023). They are used to compute surface turbulent fluxes (of heat, moisture, and momentum) and skin temperature over different surfaces (vegetation, bare soil, snow, interception, and water) and to calculate an area-weighted average for the grid box for coupling with the atmosphere. To trigger all different parametrization schemes, the ECMWF model uses a set of physiographic fields that do not depend on initial condition of each forecast run or the forecast step. Most fields are constant; surface albedo is specified for 12 months to describe the seasonal cycle. Depending on the origin, initial data come at different resolutions and different projections and are then first converted to a regular latitude–longitude grid (EPSG:4326) at ~ 1 km at Equator resolution and secondly to a required grid and resolution. Surface physiographic fields used in this work consist of orographic, land, water, vegetation, soil, albedo fields, see Table 1 for the full list of surface physiographic fields; for more details, see IFS documentation (ECMWF, 2023).

2.2.2 ERA5

Climate reanalyses combine observations and modelling to provide calculated values of a range of climactic variables over time. ERA5 is the fifth-generation reanalysis from ECMWF. It is produced via 4D-Var data assimilation of the IFS cycle 41R2 coupled to a land surface model (ECLand, (Boussetta et al., 2021)), which includes lake parametrization by Flake (Mironov and Helmert, n.d.) and an ocean wave model (WAM). The resulting data product provides hourly values of climatic variables across the atmosphere, land, and ocean at a resolution of approximately 31 km with 137 vertical sigma levels up to a height of 80 km. Additionally, ERA5 provides associated uncertainties of the variables at a reduced 63 km resolution via a 10-member ensemble of data assimilations. In this work, ERA5 hourly surface fields at ~ 31 km resolution on the cubic octahedral reduced Gaussian grid (i.e. Tco399) are used. The Gaussian grid's spacing between latitude lines is not regular, but lines are symmetrical along the Equator; the number of points along each latitude line defines longitude lines, which start at longitude 0 and are equally spaced along the latitude line. In a reduced Gaussian grid, the number of points on each latitude line is chosen so that the local east–west grid length remains approximately constant for all latitudes (here, the Gaussian grid is N320, where N is the number of latitude lines between a pole and the Equator).

Table 1 Input and target features to all emulators from the data sources. The left column shows the observation-derived static physiographic fields, the middle column ERA5 dynamic physiographic and meteorological fields and the rightmost column ECLand generated dynamic prognostic state variables.

Climate fields	Units	Atmospheric forcing	Units	Prognostic states	Units
Vegetation cover (*low, high*)		Total precipitation fraction (*convective + stratiform*)		Soil water volume (*Layers 1-3*)	m^3m^{-3}
Type of vegetation (*low, high*)		Downward radiation (*long, short*)	W/m^2	Soil temperature (*Layers 1-3*)	K
Minimum stomatal resistance (*low, high*)		Seasonal LAI (*high, low*)		Snow cover fraction	%

Roughness length (*low, high)*		Wind speed (*v, u*)	m/s
Urban cover		Surface pressure	Pa
Lake cover		Skin temperature	K
Lake depth			
Orography (+ *std,* + *filtered*)	m^2/s^{-2}	Specific humidity	kg/kg
Photosynthesis pathways		Rainfall rate (*total*)	kg/m^2s
Soil type		Snowfall rate (*total)*	kg/m^2s
Glacier mask			
Permanent wilting point			
Field capacity			
Cell area			

2.3 Emulators

We compare a long-short term memory neural network (LSTM), extreme gradient boosting regression trees (XGB) and a feedforward neural network (that we here refer to as multilayer perceptron, MLP). To motivate this setup and pave the way for discussing effects of (hyper-)parameter choices, a short overview of all approaches is given. All analyses were conducted in Python. XGB was developed in dmlc's XGBoost python package[1]. The MLP and LSTM were developed in the PyTorch lightning framework for deep learning[2]. Neural networks were trained with the Adam algorithm for stochastic optimization (Kingma and Ba, 2017). Model architectures and algorithmic hyperparameters were selected through combined Bayesian hyperparameter optimization with the Optuna framework (Akiba et al., 2019) and additional manual tuning. The Bayesian optimization minimizes the neural network validation accuracy, specified here as mean absolute error (MAE), over a predefined search space for free hyperparameters with the Tree-structured Parzen Estimator (Ozaki et al., 2022).

[1] https://xgboost.readthedocs.io/en/stable/python/index.html
[2] https://lightning.ai/docs/pytorch/stable/

The resulting hyperparameter and architecture choices which were used for the different approaches are listed in the Supplementary Material.

2.3.1 MLP

For creation of the MLP emulator we work with a feed-forward neural network architecture of connected hidden layers with ReLU activations and dropout layers, model components which are given in detail in the Supplementary Material or in (Goodfellow et al., 2016). The MLP was trained with a learning rate scheduler. L2-regularization was added to the training objective via weight decay. Sizes and width of hidden layers as well as hyperparameters were selected together in the hyperparameter optimization procedure. Instead of forecasting absolute prognostic state variables $\boldsymbol{z_t}$, the MLP predicts the 6-hourly increment, $\frac{\widehat{dz}}{dt}$. It is trained on a stepwise rollout prediction of future state variables at a pre-defined lead time at given forcing conditions, see details in the section on optimization.

2.3.2 LSTM

LSTMs are recurrent networks that consider long-term dependencies in time series through gated units with input and forget mechanisms (Hochreiter and Schmidhuber, 1997). In explicitly providing time-varying forcing and state variables, LSTM cell states serve as long-term memory while LSTM hidden states are the cells' output and pass on stepwise short-term representations stepwise. In short notation (Lees et al., 2022), a one-step ahead forward pass followed by a linear transformation can be formulated as

$$\boldsymbol{h}_t, \boldsymbol{c}_t = f(\boldsymbol{x}_t, \boldsymbol{h}_{t-1}, \boldsymbol{c}_{t-1}, \boldsymbol{\theta})$$

$$\hat{\boldsymbol{z}}_t = \boldsymbol{A}\boldsymbol{h}_t + b$$

where $\boldsymbol{h}_{t-1}$ denotes the hidden state, i.e. output estimates from the previous time step, $\boldsymbol{c}_{t-1}$ the cell state from the previous time step, and $\boldsymbol{\theta}$ the time-invariant model weights. We stacked multiple LSTM cells to an encoder-decoder model with transfer layers for hidden and cell state initialization and for transfer to the context vector (see figure 1) (Nearing et al., 2024). A lookback l of the previous static and dynamic feature states are passed sequentially to the first LSTM cells in the encoder layer, while the l prognostic state variables $\boldsymbol{z}$ initialize the hidden state $\boldsymbol{h}_0$ after a linear embedding. The output of the first LSTM layer cells become the input to the deeper LSTM layer cells and the last hidden state estimates are the final output from the encoder. Followed by a non-linear transformation with hyperbolic tangent activation, the hidden cell states are transformed into a weighted context vector $\boldsymbol{s}$. Together with the encoder the cell state $(\boldsymbol{c}_t, \boldsymbol{s})$ initializes the hidden and cell states of the decoder. The decoder LSTM cells take as input again static and dynamic features sequentially at lead times $t = 1, \dots, \tau$, but not the prognostic states variables. These are estimated from the sequential hidden states of the last LSTM layer cells, transformed to target size with a linear forecast head before prediction. LSTM predicts absolute state variables $\boldsymbol{z}_t$ while being optimized on $\boldsymbol{z}_t$ and $d\hat{\boldsymbol{z}}_t$ simultaneously, see section on optimization.

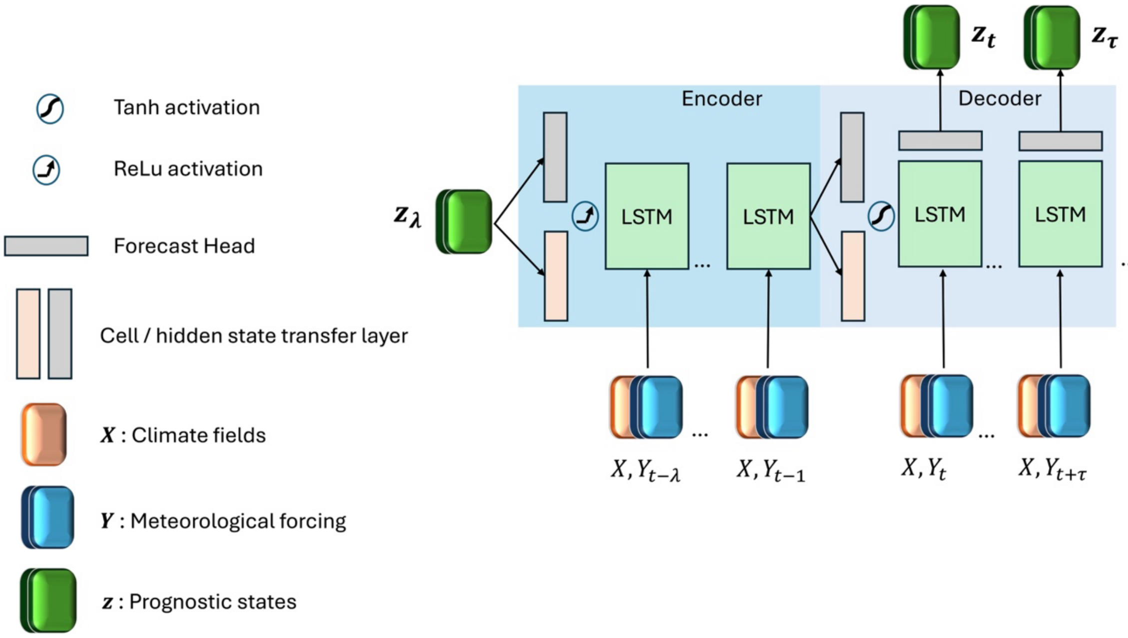


Figure 1: LSTM architecture. Blue shaded area indicates the encoder part, where the model is driven by a lookback λ of meteorological forcing and state variables. The light-blue shaded area indicates the decoder part that is initialized from the encoding to unroll LSTM forecasts from the initial time step t up to a flexibly long lead time of τ.

2.3.3 XGB

Extreme gradient boosting (XGB) is a regression tree ensemble method that uses an approximate algorithm for best split finding. It computes first and second order gradient statistics in the cost function, performing a similar to gradient descent optimization (Chen and Guestrin, 2016), where each new learner is trained on the residuals of the previous ones. Regularization and column sampling aim for preventing overfitting internally. XGB is known to provide a powerful benchmark for time series forecasting and tabular data (Chen and Guestrin, 2016; Chen et al., 2020; Shwartz-Ziv and Armon, 2021). Like the MLP, it is trained to predict the increment $\widehat{d\boldsymbol{z}}_{t,i}$ of prognostic state variables, but only for a one-step ahead prediction.

2.4 Experimental setup

We distinguish the experimental analysis into three parts that vary in the usage of the training database: (1) model development, (2) model testing, and (3) global model transfer.
The models were developed and for the first time evaluated on a low state resolution (ECMWF's TCO199 reduced gaussian grid, see section on data sources) and temporal subset from the training data base, i.e. on a bounding box of 7715 grid cells over Europe with time series of six years from 2016 to 2022. For details on the development data base, model selection and model performances, see Supplementary Material S3.
The selected models were recreated on a high state resolution (TCO399) continental scale European subset with 10 051 grid cells. Models were trained on five years 2015-2020 with the year 2020 as validation split and evaluated on the year 2021 for the scores we report in the main part. Note that for computation of forecast horizons, the two test years 2021 and 2022 were used, see details in section on forecast horizons. With this same data splitting setup, the analysis was repeated in transferring the candidates to the low resolution (TCO199) global data set with a total of 47892 grid cells. The low global resolution on one hand allowed a systematic comparison of the three models, because high resolution training with XGB was prohibited by the required working memory. On the other hand, this extrapolation scenario created an unseen problem for the models that were selected on a continental and high-resolution scale which is reflected in the resulting scores.

2.5 Optimization

2.5.1 Loss functions

The basis of the loss function $\mathcal{L}$ for the neural network optimization was PyTorch's SmoothL1Loss[3], a robust loss function that combines L1-norm and L2-norm and is less sensitive to outliers than pure L1-norm (Girshick, 2015). Based on a pre-defined threshold parameter β, smooth L1 transitions from L2-norm to L1-norm above the threshold.

[3] https://pytorch.org/docs/stable/generated/torch.nn.SmoothL1Loss.html

SmoothL1Loss $\mathcal{L}$ is defined as

$$\mathcal{L}(\hat{z},\ z) = 0.5(\hat{z} - z)^2 \frac{1}{\beta} \text{ if } |\hat{z} - z| < \beta \text{ and}$$

$$\mathcal{L}(\hat{z},\ z) = |\hat{z} - z| - 0.5\,\beta \text{ otherwise,}$$

here with $\beta = 1$. All models were trained to minimize the incremental loss $\mathcal{L}_s$ that is the differences between the estimates of the seven prognostic states increments $\widehat{d\boldsymbol{z}_t}$ and the full model's prognostic states increments $d\boldsymbol{z}_t$ simultaneously as the sum of losses over all states. We opted for a loss function equally weighted by variables to share inductive biases among the non-independent prognostic states (Sener and Koltun, 2018). When aggregating over all training lead times $t = 1, \dots, \tau$, $\mathcal{L}_s$ and grid cells $i = 1, \dots, p$ is

$$\mathcal{L}_s(\widehat{d\boldsymbol{z}}, d\boldsymbol{z}) = \sum_t^{\tau} \sum_i^{p} \mathcal{L}_t(\widehat{d\boldsymbol{z}}_{t,i}, d\boldsymbol{z}_{t,i}),$$

Whereas when computing a rollout loss $\mathcal{L}_r$ stepwise,

$$\mathcal{L}_r(\widehat{d\boldsymbol{z}}, \boldsymbol{z}) = \frac{1}{\tau} \sum_t^{\tau} \sum_i^{p} \mathcal{L}_t(z_{t-1,i} + \widehat{d\boldsymbol{z}}_{t,i}, z_{t,i})$$

Prognostic state increments are essentially the first differences from one to the next timestep that are normalized again by the global standard deviation of the model's states increments, s_{dz} before computation of the loss (Keisler, 2022). Due to the forecast models' structural differences, loss functions were individually adapted:

MLP The combined loss function for the MLP is the sum of the incremental loss $\mathcal{L}_s$ and the rollout loss $\mathcal{L}_r$. For the rollout loss $\mathcal{L}_r$, $\mathcal{L}$ was aggregated over grid cells p and accumulated after an auto-regressive rollout over lead times τ, before being averaged out by division by τ (Keisler, 2022).

LSTM The combined loss function for the LSTM is the sum of the incremental loss $\mathcal{L}_s$, where the $d\hat{\boldsymbol{z}}_t$ were derived from $\hat{\boldsymbol{z}}_t$ after the forward pass, and the loss $\mathcal{L}$ computed on decoder estimates of prognostic states variables, a functionality that leverages the potential of our LSTM structure.

XGB Trained only from one to the next time step, i.e. at a lead time of $\tau = 1$, the incremental loss $\mathcal{L}_s = \mathcal{L}_r$. Without a SmoothL1Loss implementation provided in dmlc's XGBoost, we trained XGB with both the Huber-Loss and the default L2-loss. The latter initially providing better results, we chose the default L2-norm as loss function for XGB with the regularization parameter $\lambda = 1$.

2.5.1 Normalization

As prognostic target variables are all lower bounded by zero, we tested both z-scoring and max-scoring. The latter yielded no significant improvement; thus we show our results with z-scored target variables. For neural network training but not for fitting XGB, static, dynamic and prognostic state variables were all normalized with z-scoring towards the continental or global mean $\bar{z}$ and unit standard deviation s_z as

$z_{t,n} = \frac{z_{t,n} - \bar{z}}{s_z}$.

Prognostic target state increments were normalized again by the global standard deviation of increments computing the loss (see section 2.5.1) to smooth magnitudes of increments (Keisler, 2022). State variables were back transformed to original scale before evaluation.

2.5.3 Spatial and temporal sampling

Sequences were sampled randomly from the training data set, while validation happened sequentially. MLP and XGB were trained on all grid cells simultaneously in both the continental and global setting, while LSTM was trained on the full continental data set but was limited by GPU memory in the global task. We overcame this limitation by randomly subsetting grid cells in the training data into largest possible, equally sized subsets which were then loaded along with the temporal sequences during the batch sampling.

2.6 Evaluation

Three scores are used for model validation during the model development phase and in validating architecture and hyperparameter selection, being the root mean squared error ($RMSE$), the mean absolute error (MAE) and the anomaly correlation coefficient (ACC). First, scores were assessed objectively in quantifying forecast accuracy of the emulators against ECLand simulations directly with RMSE and MAE. Doing so, scores were aggregated over lead times, grid cells or both. The total RMSE was computed as

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{\tau,p}(z - \hat{z})^2}{n}},$$

As the mean absolute error in prognostic state variable prediction over the total of n grid cells p times lead times τ. Equivalently, MAE was computed as

$$\mathrm{MAE} = \frac{\sum_{t,p} |z - \hat{z}|}{n},$$

Beyond accuracy, the forecast skill of emulators was assessed using a benchmark model: the ACC (see below) as index of the long-term naïve climatology c of ECLand, forced by ERA5 (see section 2.2). More specifically, this is the 6-hourly mean of prognostic state variables over the last 10 years preceding the test year, i.e. the years 2010 to 2020. While climatology is a hard-to-beat benchmark specifically in long-term forecasting, the persistence is a benchmark for short-term forecasting (Pappenberger et al., 2015). For verification against climatology, we compute the anomaly correlation coefficient (ACC) over lead times as

$$\mathrm{ACC}(t) = \frac{\overline{(\hat{z} - c)(z - c)}}{\sqrt{\overline{(\hat{z} - c)^2}\ \overline{(z - c)^2}}}$$

at each t = 1, …, τ where the overbar denotes averaging over grid cells $p = i, \ldots, n$. This way, the nominator represents the average spatial covariance of emulator and numerical forecasts with climatology as expected sample mean. Hence, it indicates the mean squared skill error towards climatology, and the denominator indicates its variability. The aggregated scores that are shown in tables 3-5 represent the temporally arithmetic mean of ACC(t). ACC is bounded between 1 and -1, and an ACC of 1 indicates perfect representation of forecast error variability, an ACC of 0.5 indicates a similar forecast error to that of the climatology, an ACC of 0 indicates that forecast error variability dominates and the forecast has no value and an ACC approaching -1 indicates that the forecast has been very unreliable (ECMWF, n.d.). ACC is undefined when the denominator is zero. This is the case either when mean squared emulator or ECLand anomaly, or both are zero because forecast and climatology perfectly align, or because they cancel out at summation to the mean.

2.6.1 Forecast horizons

Forecast horizons of the emulators are defined by the decomposition of the RMSE (Bengtsson et al., 2008) into the emulator's variability around climatology (i.e. anomaly), ECLand's variability around climatology and the covariance of both. The horizon is the point in time at which the forecast error reaches saturation level, that is when the covariance of emulator and ECLand anomalies approaches zero, as does the ACC.

We analysed predictive ability and predictability by computing the ACC for all lead times from 6 hours to approx. one year, i.e. lead times $t = 1, \ldots, \tau$, τ being 1350. As this confounds the seasonality with the lead time, we compute these for every starting point of the prediction, requiring two test years (2021 and 2022).

Forecast horizons based on the emulators' skill in standardized anomaly towards persistence were equivalently computed but with persistence as a benchmark for shorter time scales, this was only done for three months, from January to March 2021.

The analysis was conducted on two exemplary regions in northern and southern Europe that represent very different conditions orography and in prognostic land surface states, specifically in snow cover. For details on the regions and on the horizons computed with standardized anomaly skill, see Appendices A1 and A4 respectively.

3 Results

The improvement in evaluation runtimes achieved by emulators toward the numerical ECLand were significant. Iterating the forecast over a full test year at 30 km spatial resolution, XGB evaluates in 5.4 minutes, LSTM in 3.09 minutes and MLP in 0.05 minutes (i.e. 3.2 seconds) on average. In contrast, ECLand integration over a full test year on 16 CPUs at 30 km spatial resolution takes approximately 240 minutes (i.e. four hours). The slow runtime of the LSTM compared to the MLP emulator is caused by a spatial chunking procedure that was not optimise for this work but could be improved in the future.

3.1 Aggregated performances

Europe. All emulators approximated the numerical LSM with high average total accuracies (all RMSEs < 1.58 and MAEs < 0.84) and confident correlations (all ACC > 0.72) (see table 2 and figure 2). The LSTM emulator achieved the best results across all total average scores on the European scale. It decreased the total average MAE by ~25% towards XGB and by ~37% towards the MLP and the total average RMSE by ~42% towards XGB and ~38% towards the MLP. In total average ACC, the LSTM scored 20% higher than the MLP and 15% than XGB, also being the only emulator that achieved an ACC > 0.9. While the MLP outperforms XGB in total average RMSE by ~5%, XGB scores better than the MLP in MAE by ~27%.

At variable level, results differentiate into model specific strengths. In soil water volume, XGB outperforms the neural network emulators by up to 60% ($m^3 m^{-3}$) in the first and second layer MAEs towards the LSTM and up to over 40% ($m^3 m^{-3}$) for towards the MLP (see table 3). While the representation of anomalies by specifically the LSTM decreases towards lower soil layers with an ACC of only 0.6214 at the third soil layer, it remains consistently higher for XGB with an ACC still > 0.789 at soil layer three.
In soil temperature approximation, LSTM achieves best accuracies at higher soil levels with up to 7% (K) improvement in MAE towards XGB and ACCs > 0.92, but XGB outperforms LSTM at the third soil level with a close to 50% (K) improvement (see table 4). The MLP doesn't stand out by high scores on the continental scale. However, in terms of accuracy we found an inverse ranking in the model development procedure during which LSTM outscored XGB in soil water volume but struggled with soil temperature approximations, for the interested reader we refer to the supplementary information.
In snow cover approximation, the LSTM emulator enhances accuracies by over ~50% in MAE towards both the XGB and the MLP emulator and scores highest in anomaly representation with an ACC of ~0.87 compared to an ACC of ~0.66 for the MLP and only ~0.74 for the XGB (see table 5).
Globe. Score ranking on the global scale varies strongly from the continental scale (see table 2). In total average accuracies, the MLP outperforms XGB by over 30% and LSTM by up ~25% in RMSE and improves MAE more than 15% towards both. In anomaly correlation however it scores last, whereas XGB achieves the highest total average of over 0.75. Consistent with scores on the continental scale is XGBs high performance in soil temperature (see table 3). It significantly outperforms the LSTM by ~60% (K) in RMSE and nearly up to 75% (K) in MAE in all layers and the MLP by up to 50% (K) in MAE at the top layer. Anomaly persistence for all models degrade visibly towards the lower soil layers, while that of the LSTM most relative to MLP and XGB. Like on the continental scale, XGB also outperforms the other candidates in soil temperature forecasts in all but the medium layer, where the MLP gets higher scores in MAE and RMSE but not in ACC (see table 4). LSTM doesn't stand out with any scores on the global scale.

3.2 Spatial and temporal performances

Europe. When summarizing temporally aggregated scores as boxplots to a total distribution over space (see figure 2, A), the long tails of XGB scores become visible, whereas the MLP indicates most robustness. This is reflected in the geographic distribution of scores at the example of ACC (see figure 2, bottom), where the area of low anomaly correlation is largest for XGB, ranging over nearly all northern Scandinavia, while MLP and LSTM have smaller and more segregated areas of clearly low anomaly correlation. The LSTM shows a homogenously high ACCs over most of central Europe but the Alps, while also seems to be challenged in areas of relative to the central Europe extreme weather conditions at the Norwegian and Spanish coasts.

Globe. Like the results from the continental analysis, we find again long upper tails of outliers for XGB in total spatial distribution of accuracies, both in RMSE and MAE and only few outliers for MLP and LSTM. The anomaly correlation distribution changed towards longer lower tails for MLP and LSTM and a shorter lower tail for XGB. We should, however, take the results of total average ACC with care as it remains largely undefined in regions without much noise in snow cover or soil water volume and globally represents mainly patterns of soil temperature.

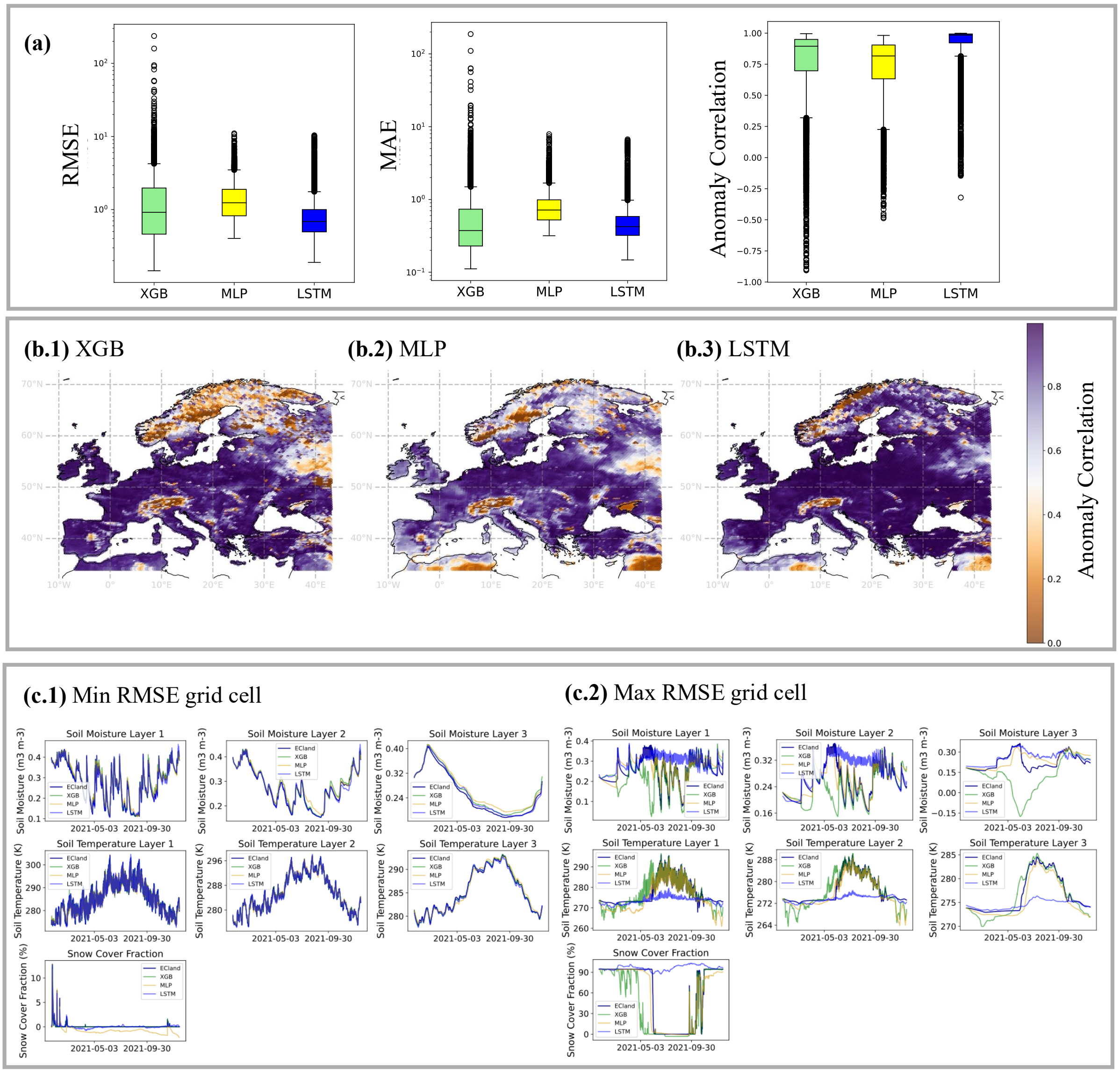


*Figure 2: **a**: Total aggregated distributions of (log) scores averaged over lead times, i.e. displaying the variation among grid cells. **b:** The distribution of the anomaly correlation in space on the European subset (b.1: XGB, b.2: MLP, b.3: LSTM). **c:** Model forecasts over test year 2021 for grid cell with minimum and maximum RMSE values (LSTM).*

Table 2: Emulator total average scores (unitless), aggregated over variables, time and space from the European and Global model testing.

Variable	**Model**	**RMSE**		**MAE**		**ACC**	
		Europe	Globe	Europe	Globe	Europe	Globe
All variables	XGB	1.575	2.611	0.695	1.601	0.765	**0.755**
	MLP	1.486	**1.699**	0.832	**1.189**	0.728	0.569
	LSTM	**0.918**	2.252	**0.526**	1.787	**0.925**	0.647

Table 3: Emulator average scores (RMSE, MAE in m^3m^{-3}*) on soil water volume forecasts for the European subset, aggregated over space and time from the European and Global model testing.*

Variable	**Layer**	**Model**	**RMSE**		**MAE**		**ACC**	
			Europe	Globe	Europe	Globe	Europe	Globe
Soil water volume	1	XGB	**0.013**	**0.015**	**0.01**	**0.01**	**0.908**	**0.92**
		MLP	0.019	0.029	0.015	0.023	0.856	0.791
		LSTM	0.029	0.048	0.023	0.04	0.847	0.729
	2	XGB	**0.011**	**0.012**	**0.008**	**0.009**	**0.901**	**0.884**
		MLP	0.019	0.023	0.014	0.018	0.789	0.77
		LSTM	0.029	0.05	0.023	0.042	0.79	0.617
	3	XGB	**0.015**	**0.014**	**0.011**	**0.01**	**0.789**	**0.777**
		MLP	0.02	0.02	0.017	0.016	0.576	0.667
		LSTM	0.033	0.051	0.027	0.043	0.621	0.475

Table 4: Emulators' average scores (RMSE, MAE in K) on soil temperature forecasts for the European subset, aggregated over space and time.

Variable	**Layer**	**Model**	**RMSE**		**MAE**		**ACC**	
			Europe	Globe	Europe	Globe	Europe	Globe
Soil temperature	1	XGB	1.154	4.539	0.744	3.278	0.806	**0.769**
		MLP	1.628	**2.606**	1.188	**2.072**	0.674	0.581
		LSTM	**0.931**	3.152	**0.682**	2.626	**0.938**	0.735
	2	XGB	0.901	2.501	**0.51**	1.772	0.812	**0.797**
		MLP	1.134	**1.851**	0.784	**1.452**	0.718	0.606
		LSTM	**0.734**	2.87	0.541	2.4	**0.928**	0.699
	3	XGB	**0.714**	**1.287**	**0.482**	**0.933**	**0.722**	**0.711**
		MLP	1.128	1.375	0.821	1.071	0.416	0.514
		LSTM	1.141	3.466	0.918	3.002	0.598	0.406

Table 5: Emulators' average scores (RMSE, MAE in %) on snow cover forecasts for the European subset, aggregated over space and time.

Variable	**Layer**	**Model**	**RMSE**		**MAE**		**ACC**	
			Europe	Globe	Europe	Globe	Europe	Globe

Snow top cover	XGB	8.219	9.906	3.099	5.196	0.746	**0.707**
	MLP	6.449	**5.995**	2.986	**3.671**	0.66	0.618
	LSTM	**3.526**	6.127	**1.47**	4.357	**0.877**	0.698

3.3 Forecast horizons

Forecast horizons were computed for two European regions, of which the northern one represents the area of lowest emulators' skill (see figure 2, B.1-3) and the southern one an area of stronger emulators' skill. Being strongly correlated with soil water volume, these two regions differ specifically in their average snow cover fraction (see figure 3). The displayed horizons were computed over all prognostic state variables simultaneously, while their interpretation is related to horizons computed for prognostic state variables separately, for the figures of which we refer to the Supplementary Material.

In the North, predictive skill depended on an interaction of how far ahead a prediction was made (the lead time) and the day of year to which the prediction was made. In the best case, the LSTM, summer predictions were poor (light patches in figure 3 heat maps), but only when initialised in winter. Or, in other words, one can make good predictions starting in winter, but not to summer. Vertical structures indicate a systematic model error that appears at specific initialisation times and that is independent of prediction date, for example in XGB forecasts that are initialized in May (see figure 3, northern region). Diagonal light structures in the heat maps indicate a temporally consistent error and can be interpreted as physical limits of system predictability, where the different initial forecast time doesn't affect model scores.

All models show stronger limits in predictability and predictive ability in the northern European region (see figure 3, left column). MLP and XGB struggled with representing seasonal variation towards climatology at long lead times, while LSTM is strongly limited by a systematic error in certain regions. Initializing the forecast the 1 January 2021, MLP drops below an ACC of 80% repeatedly from initialization on and then to an ACC below 10% at the beginning of May. LSTMs performance is more robust in the beginning of the year but depletes strongly later to less than 10% ACC in mid-May. On the one hand, this represents two different characteristics of model errors: MLP forecasts for snow cover fraction are less than zero for some grid cells while LSTM forecasts for snow cover fraction remain falsely at very high levels for some grid cells, not predicting the snowmelt in May (see Supplementary Material, S4.1). On the other hand, this represents a characteristic error due to change in seasonality: the snowmelt in this region in May happens abruptly and all emulators repeatedly over- or underpredict the exact date.

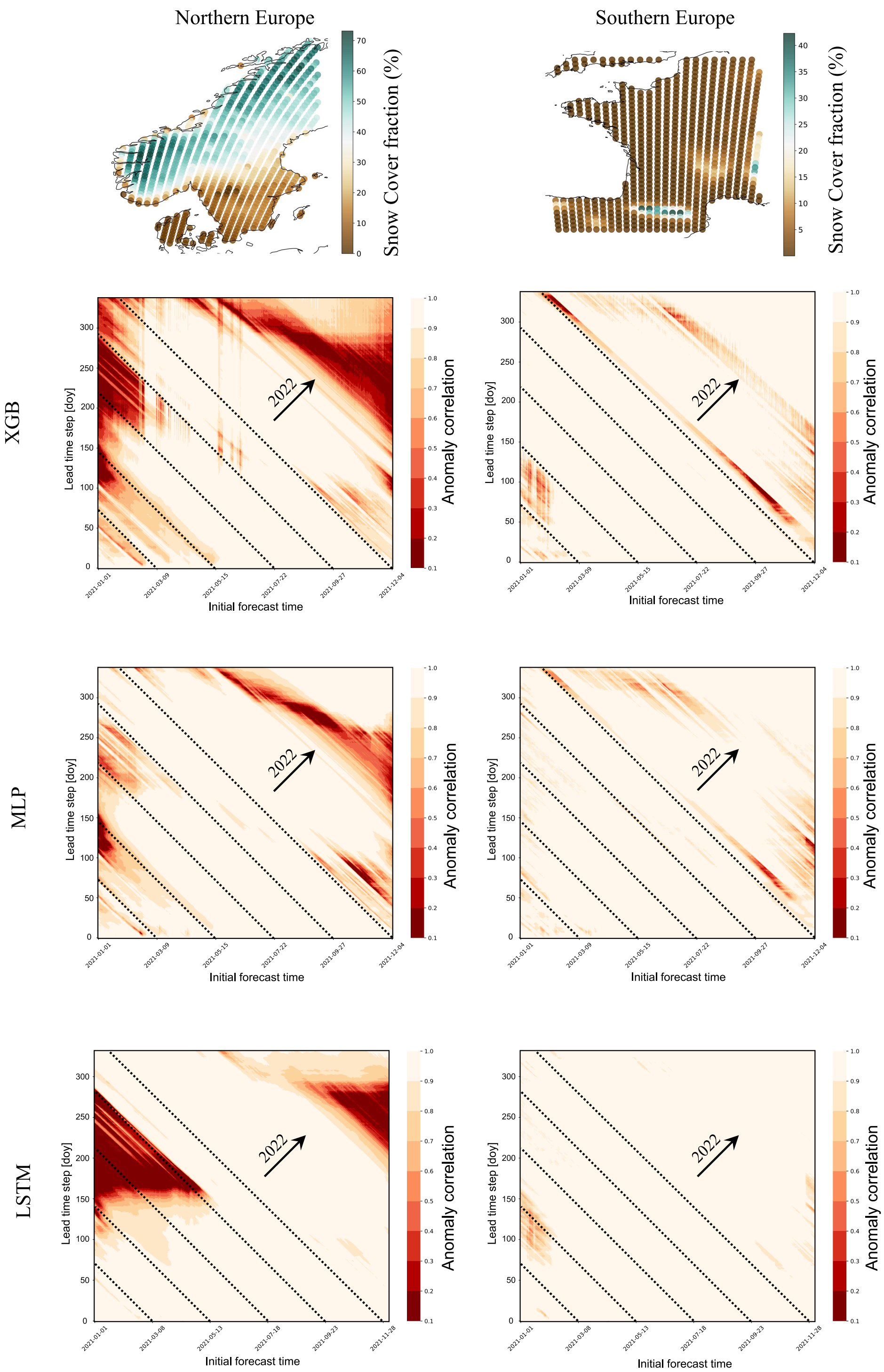
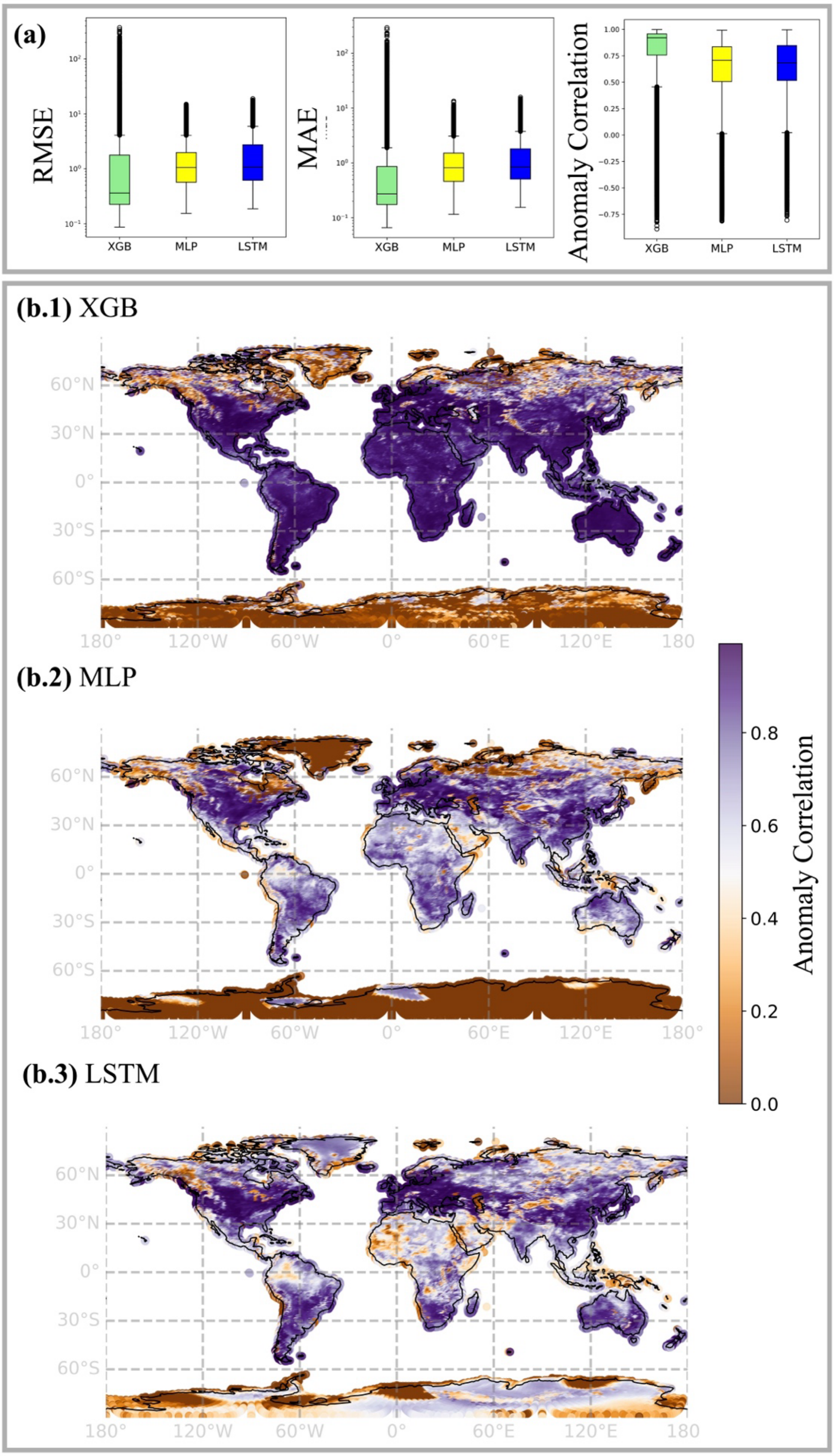


*Figure 3: **Top row**: European subregions for computations of forecast skill horizons and their yearly average snow cover fraction (%), predicted by ECLand. **Rows 2-4**: Emulator forecast skill horizons in the subregions, aggregated over prognostic state variables, computed with the anomaly correlation coefficient (ACC) at 6-hourly lead times (y-axis) over approx. one*

year, displayed as a function of the initial forecast time (x-axis). The horizon is the time at which the forecast has no value at all, i.e. when ACC is 0 (or below 10%). The diagonal dashed lines indicate the day of the test year 2021 as labelled on the x-axis, the arrows indicate where forecasts reach the second test year 2022.

531



532 *Figure 4: a) Total average scores, representing spatial variation among grid cells. B) Total*
533 *average ACC in space. Note that ACC remained undefined for regions of low signal in snow*
534 *cover and soil water volume, see Supplementary Material.*

4 Discussion

In the comparative analysis of emulation approaches for land surface forecasting, three primary models—LSTM (Long Short-Term Memory networks), MLP (Multi-Layer Perceptrons), and XGB (Extreme Gradient Boosting)—have been evaluated to understand their effectiveness across different operational scenarios. Evaluating emulators over the test period yielded a significant runtime improvement toward the numerical model for all approaches (see section 3). While all models achieved high predictive scores, they differ in their demand of computational resources (Cui et al., 2021) and each one offers unique advantages and faces distinct challenges, impacting their suitability for various forecasting tasks. In this work we present the first steps towards enabling quick offline experimentation on the land surface with ECMWF's land surface scheme ECLand and towards decreasing computational demands in, i.e. coupled data assimilation.

4.1 Approximation of prognostic land surface states

The total evaluation scores of our emulators indicate good agreement with ECLand simulations. Among the seven individual prognostic land surface states, emulators achieve notably different scores and in the transfer from the high-resolution continental to the low-resolution global scale, their performance ranking change. On average, neural network performances degrade towards the deeper soil layers, while XGB scores remain relatively stable. Also, the neural networks scores drop in the extrapolation from continental to global scale, while XGB scores also for this task remain constantly high.

In a way, these findings are not surprising. It is known that neural networks are highly sensitive to selection bias (Grinsztajn et al., 2022) and tuning of hyper-parameters (Bouthillier et al., 2021), suboptimal choices of which may destabilise variance in predictive skill. Previous and systematic comparisons of XGB and deep neural networks have demonstrated that neural networks can hardly be transferred to new data sets without performance loss (Shwartz-Ziv and Armon, 2021). On tabular data, XGB still outperforms neural networks in most cases (Grinsztajn et al., 2022), unless these models are strongly regularized (Kadra et al., 2021). The disadvantage of neural networks might lay in the rotational invariance of MLP-like architectures, due to which information about the data orientation gets lost, as well as in their instability regarding uninformative input features (Grinsztajn et al., 2022).

Inversely to expectations and preceding experiments, on the European data set relative to the two other models the LSTM scored better in the upper layer soil temperatures than in forecasting soil water volume and decreased in scores towards lower layers with slower processes. For training on observations, the decreasing LSTM predictive accuracy for soil moisture with lead time is discussed (Datta and Faroughi, 2023), but reasons arising from the engineering side remain unclear. In an exemplary case of a single-objective, deterministic streamflow forecast, a decrease in recurrent neural network performance has been related with an increasing coefficient of variation (Guo et al., 2021). In our European subregions, the signal-to-noise ratio of the prognostic state variables (computed as the averaged ratio of mean and standard deviation) is up to ten times higher in soil temperature than in soil water volume states (see Supplementary Material, S2.1). While a small signal of the latter may induce instability in scores, it does not explain the decreasing performance towards deeper soil layers with slow processes, where we expected an advantage of the long-term memory.

Stein's paradox tells us that joint optimization may lead to better results if the target is multi-objective, but not if we are interested in single targets (James and Stein, 1992; Sener and Koltun, 2018). While from a process perspective multi-objective scores are less meaningful than single ones, this is what we opted for due to efficiency. The unweighted linear loss combination might be suboptimal in finding effective parameters across all prognostic state variables (Chen et al., 2017; Sener and Koltun, 2018), yet being strongly correlated, we deemed their manual weighting inappropriate. An alternative to this provides adaptive loss weighting with gradient normalisation (Chen et al., 2017).

4.2 Evaluation in time and space

We used aggerated MAE and RMSE accuracies as a first assessment tool to conduct model comparison, but score aggregation hides model specific spatio-temporal residual patterns. Further, both scores are variance dependent, favouring low variability in model forecasts even though this may not be representative of the system dynamic (Thorpe et al., 2013). Assessing the forecast skill over time as the relative proximity to a subjectively chosen benchmark helps disentangling areas of strengths and weaknesses in forecasting with the emulators (Pappenberger et al., 2015). The naïve 6-hourly climatology as benchmark highlights periods where emulators long-range forecasts on the test year are externally limited by seasonality, i.e. system predictability, and where they are internally limited by model error, i.e. the model's predictive ability. Applying this strategy in two exemplary European subregions showed that all emulators struggle most in forecasting the period from late summer to autumn, unless they are initialized in summer (see figure 3). Because forecast quality is most strongly limited by snow cover (see Supplementary Material, A4.1), we interpret this as the unpredictable start of snow fall in autumn. External predictability limitations seem to affect the LSTM overall less than the two other models, and specifically XGB drifts at long lead times.

From a geographical perspective inferred from the continental scale, emulators struggle in forecasting prognostic state variables in regions with complicated orography and strong environmental gradients. XGB scores vary seemingly random in space, while neural networks scores exhibit spatial autocorrelation. A meaningful inference about this, however, can only be conducted in assessing model sensitivities to physiographic and meteorological fields through gradients and partial dependencies. While the goal of this work is to introduce our approach to emulator development, this can be investigated in future analyses.

4.3 Emulation with memory mechanisms

Without much tuning, XGB challenges both LSTM and MLP for nearly all variables (see tables 2-4). In training on observations for daily short-term and real-time rainfall-runoff prediction, XGB and LightXGB were shown before to equally performed as, or outperformed LSTMs (Chen et al., 2020; Cui et al., 2021). Nevertheless, models with memory mechanism such as the encoder-decoder LSTM remain a promising approach for land surface forecasting regarding their differentiability (Hatfield et al., 2021), their flexible extension of lead times, for exploring the effect of long-term dependencies or for inference from the context vector that may help identifying the process relevant climate fields (Lees et al., 2022).

The LSTM architecture assumes that the model is well defined in that the context vector perfectly informs the hidden decoder states. If that assumption is violated, potential strategies are to create a skip-connection between context vector and forecast head, or to consider input of time-lagged variables or self-attention mechanisms (Chen et al., 2020). With attention, the context vector becomes a weighted sum of alignments that relates neighbouring positions of a sequence, a feature that could be leveraged for forecasting quick processes such as snow cover or top-level soil water volume.

Comparing average predictive accuracies across different training lead times indicates that training at longer lead times may enhance short-term accuracy of the LSTM at the cost of training runtime (see Supplementary Material, S2). A superficial exploration of encoder length indicates no visible improvement on target accuracies if not a positive tendency towards shorter sequences. This needs an extended analysis for understanding, yet without a significant improvement by increased sequence length, GRU cells might provide a simplified and less parameterized alternative to LSTM cells. They were found to perform equally well on streamflow forecast performance before, while reaching higher operational speed (Guo et al., 2021).

4.4 Emulators in application

LSTM networks with a decoder structure are valued for their flexible and fast lead time evaluation, which is crucial in applications where forecast intervals are not consistent. The structure of LSTM is well-suited for handling sequential data, allowing it to perform effectively over different temporal scales (Hochreiter and Schmidhuber, 1997). They provide access to gradients, which facilitates inference, optimization and usage for coupled data assimilation (Hatfield et al., 2021). Nevertheless, the complexity of LSTMs introduces disadvantages: Despite their high evaluation speed and accuracy under certain conditions, they require significant computational resources and long training times. They are also highly sensitive to hyperparameters, making them challenging to tune and slow to train, especially with large datasets.

MLP models stand out for their implementation, training and evaluation speed with yet rewarding accuracy, making them a favourable choice for scenarios that require rapid model deployment. They are tractable and easy to handle, with a straightforward setup that is less demanding computationally than more complex models. MLPs also allow for access to gradients, aiding in incremental improvements during training and quick inference (Hatfield et al., 2021). Despite these advantages, MLPs face challenges with memory scaling during training at fixed lead times, which can hinder their applicability in large-scale or high-resolution forecasting tasks.

XGB models are highly regarded for their robust performance with minimal tuning, achieving high accuracy not only in sample applications, but also in transfer to unseen problems (Grinsztajn et al., 2022; Shwartz-Ziv and Armon, 2021). Their simplicity makes them easy to handle, even for users with limited technical expertise in machine learning. However, the slow evaluation speed of XGB becomes apparent as dataset complexity and size increase. Although generally more interpretable than deep machine learning tools, XGB is not differentiable, limiting its application in coupled data assimilation (Hatfield et al., 2021) even though research on differentiable trees is ongoing (Popov et al., 2019).

4.5 Experimentation with Emulators

In the IFS, the land surface is coupled to the atmosphere via skin temperature (ECMWF, 2023), the predictability of which is known to be influenced by specifically by soil moisture (Dunkl et al., 2021). This is the interface with the numerical model where a robust surrogate could act online to improve forward (i.e. parametrization (Brenowitz et al., 2020)) or backward (i.e. data assimilation (Hatfield et al., 2021)) procedures, and it motivates the experiment from the perspective of hybrid forecasting models (Irrgang et al., 2021; Slater et al., 2023). However, because an offline training ignores the interaction with the atmospheric model, emulator scores will not directly translate to the coupled performance and of course additional experiments would be necessary (Brenowitz et al., 2020). As the current stand-alone models, emulators provide a pre-trained model-suite (Gelbrecht et al., 2023) and can be used for experimentation on the land surface. The computation of forecast horizons is an example for such an experiment, seen as a step toward a predictability analysis of land surface processes. Full predictability analyses are commonly conducted with model ensembles (Guo et al., 2011; Shukla, 1981), the simulation of which can quicker be done with emulators than with the numerical model (see evaluation runtimes, section 3).

We want to stress at this point that to avoid misleading statements, evaluation of the emulators on observations is required. In the context of surrogate models, two inherent sources of uncertainty are specifically relevant: First, the structural uncertainty by statistical approximation of the numerical model and second, the uncertainty arising by parameterization with synthetic (computer model generated) data (Brenowitz et al., 2020; Gu et al., 2017). Both sources can cause instabilities in surrogate models that could translate when coupled with the IFS (Beucler et al., 2021), but that also should be quantified when drawing conclusions from the stand-alone models outside of the synthetic domain. Consequently, a reliable surrogate model for online or offline experimentation requires validation, and enforcing additional constraints may be advantageous for physical consistency (Beucler et al., 2021).

5 Conclusion

To conclude, the choice between LSTM, MLP, and XGB models for land surface forecasting depends largely on the specific requirements of the application, including the need for speed, accuracy, and ease of use. Each model's computational demands, flexibility, and operational overhead must be carefully considered to optimize performance and applicability in diverse forecasting environments. When it comes to accuracy, combined model ensembles of XGB and neural networks have been shown to yield the best results (Shwartz-Ziv and Armon, 2021), but accuracy alone will not determine a single best approach (Bouthillier et al., 2021). Our comparative assessment underscores the importance of selecting the appropriate emulation approach based on a clear understanding of each model's strengths and limitations in relation to the forecasting tasks at hand. By developing the emulators for ECMWF’s numerical land surface scheme ECLand, we path the way towards a physics-informed ML-based land surface model that on the long run can be parametrized with observations. We also provide a pretrained model suite to improve land surface forecasts and future land reanalyses.

Code and data availability

Code for this analysis is published on OSF (DOI: 10.17605/OSF.IO/8567D) or at https://github.com/MWesselkamp/land-surface-emulation. Training data is published at 10.21957/n17n-6a68 (Tco199) and 10.21957/pcf3-ah06 (Tco399).

Author contribution

MW, MCha, EP, FP and GB conceived the study. MW and EP conducted the analysis. MW, MCha, MK, EP discussed and took technical decisions. SB advised on process decisions. MW, MCho and FP wrote the manuscript. MW, MCha, EP, MCho, SB, MK, CFD, FP reviewed the analysis and/or manuscript.

Competing interest

The authors declare that they have no conflict of interest.

Acknowledgements

This work profited from discussion with Linus Magnusson, Patricia de Rosnay, Sina R. K. Farhadi and Karan Ruparell and many more. MW thankfully acknowledges ECMWF for providing two research visit stipendiates over the course of the collaboration. EP was funded by the CERISE project (grant agreement No101082139) funded by the European Union. Views and opinions expressed are however those of the authors only and do not necessarily

reflect those of the European Union or the Commission. ChatGPT version 4.0 was used for coding support.

References

Akiba, T., Sano, S., Yanase, T., Ohta, T., and Koyama, M.: Optuna: A Next-generation Hyperparameter Optimization Framework, in: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD '19: The 25th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Anchorage AK USA, 2623–2631, https://doi.org/10.1145/3292500.3330701, 2019.

Baker, E., Harper, A. B., Williamson, D., and Challenor, P.: Emulation of high-resolution land surface models using sparse Gaussian processes with application to JULES, Geosci. Model Dev., 15, 1913–1929, https://doi.org/10.5194/gmd-15-1913-2022, 2022.

Balsamo, G., Boussetta, S., Dutra, E., Beljaars, A., and Viterbo, P.: Evolution of land-surface processes in the IFS, 2011.

Bassi, A., Höge, M., Mira, A., Fenicia, F., and Albert, C.: Learning Landscape Features from Streamflow with Autoencoders, https://doi.org/10.5194/hess-2024-47, 20 February 2024.

Bengtsson, L. K., Magnusson, L., and Källén, E.: Independent Estimations of the Asymptotic Variability in an Ensemble Forecast System, Mon. Weather Rev., 136, 4105–4112, https://doi.org/10.1175/2008MWR2526.1, 2008.

Beucler, T., Pritchard, M., Rasp, S., Ott, J., Baldi, P., and Gentine, P.: Enforcing Analytic Constraints in Neural Networks Emulating Physical Systems, Phys. Rev. Lett., 126, 098302, https://doi.org/10.1103/PhysRevLett.126.098302, 2021.

Bi, K., Xie, L., Zhang, H., Chen, X., Gu, X., and Tian, Q.: Pangu-Weather: A 3D High-Resolution Model for Fast and Accurate Global Weather Forecast, https://doi.org/10.48550/ARXIV.2211.02556, 2022.

Boussetta, S., Balsamo, G., Arduini, G., Dutra, E., McNorton, J., Choulga, M., Agustí-Panareda, A., Beljaars, A., Wedi, N., Munõz-Sabater, J., De Rosnay, P., Sandu, I., Hadade, I., Carver, G., Mazzetti, C., Prudhomme, C., Yamazaki, D., and Zsoter, E.: ECLand: the ECMWF land surface modelling system, Atmosphere, 12, 723, https://doi.org/10.3390/atmos12060723, 2021.

Bouthillier, X., Delaunay, P., Bronzi, M., Trofimov, A., Nichyporuk, B., Szeto, J., Sepah, N., Raff, E., Madan, K., Voleti, V., Kahou, S. E., Michalski, V., Serdyuk, D., Arbel, T., Pal, C., Varoquaux, G., and Vincent, P.: Accounting for Variance in Machine Learning Benchmarks, http://arxiv.org/abs/2103.03098, 1 March 2021.

Brenowitz, N. D., Henn, B., McGibbon, J., Clark, S. K., Kwa, A., Perkins, W. A., Watt-Meyer, O., and Bretherton, C. S.: Machine Learning Climate Model Dynamics: Offline versus Online Performance, https://doi.org/10.48550/ARXIV.2011.03081, 2020.

Chantry, M., Hatfield, S., Duben, P., Polichtchouk, I., and Palmer, T.: Machine learning emulation of gravity wave drag in numerical weather forecasting, https://doi.org/10.48550/ARXIV.2101.08195, 2021.

Chen, T. and Guestrin, C.: XGBoost: A Scalable Tree Boosting System, https://doi.org/10.48550/ARXIV.1603.02754, 2016.

Chen, X., Huang, J., Han, Z., Gao, H., Liu, M., Li, Z., Liu, X., Li, Q., Qi, H., and Huang, Y.: The importance of short lag-time in the runoff forecasting model based on long short-term memory, J. Hydrol., 589, 125359, https://doi.org/10.1016/j.jhydrol.2020.125359, 2020.

Chen, Z., Badrinarayanan, V., Lee, C.-Y., and Rabinovich, A.: GradNorm: Gradient Normalization for Adaptive Loss Balancing in Deep Multitask Networks, https://doi.org/10.48550/ARXIV.1711.02257, 2017.

Choulga, M., Kourzeneva, E., Balsamo, G., Boussetta, S., and Wedi, N.: Upgraded global mapping information for earth system modelling: an application to surface water depth at the ECMWF, Hydrol. Earth Syst. Sci., 23, 4051–4076, https://doi.org/10.5194/hess-23-4051-2019, 2019.

Cui, Z., Qing, X., Chai, H., Yang, S., Zhu, Y., and Wang, F.: Real-time rainfall-runoff prediction using light gradient boosting machine coupled with singular spectrum analysis, J. Hydrol., 603, 127124, https://doi.org/10.1016/j.jhydrol.2021.127124, 2021.

Datta, P. and Faroughi, S. A.: A multihead LSTM technique for prognostic prediction of soil moisture, Geoderma, 433, 116452, https://doi.org/10.1016/j.geoderma.2023.116452, 2023.

De Rosnay, P., Balsamo, G., Albergel, C., Muñoz-Sabater, J., and Isaksen, L.: Initialisation of Land Surface Variables for Numerical Weather Prediction, Surv. Geophys., 35, 607–621, https://doi.org/10.1007/s10712-012-9207-x, 2014.

Dunkl, I., Spring, A., Friedlingstein, P., and Brovkin, V.: Process-based analysis of terrestrial carbon flux predictability, Earth Syst. Dyn., 12, 1413–1426, https://doi.org/10.5194/esd-12-1413-2021, 2021.

ECMWF: IFS Documentation CY43R3 - Part IV: Physical processes, https://doi.org/10.21957/EFYK72KL, 2017.

ECMWF: IFS Documentation CY48R1 - Part IV: Physical Processes, https://doi.org/10.21957/02054F0FBF, 2023.

ECMWF: Forecast User Guide, Anom. Correl. Coeff., n.d.

Fer, I., Kelly, R., Moorcroft, P. R., Richardson, A. D., Cowdery, E. M., and Dietze, M. C.: Linking big models to big data: efficient ecosystem model calibration through Bayesian model emulation, Biogeosciences, 15, 5801–5830, https://doi.org/10.5194/bg-15-5801-2018, 2018.

Gelbrecht, M., White, A., Bathiany, S., and Boers, N.: Differentiable programming for Earth system modeling, Geosci. Model Dev., 16, 3123–3135, https://doi.org/10.5194/gmd-16-3123-2023, 2023.

Girshick, R.: Fast R-CNN, https://doi.org/10.48550/ARXIV.1504.08083, 2015.

Goodfellow, I., Bengio, Y., and Courville, A.: Deep learning, The MIT Press, Cambridge, Massachusetts, 775 pp., 2016.

Grinsztajn, L., Oyallon, E., and Varoquaux, G.: Why do tree-based models still outperform deep learning on tabular data?, https://doi.org/10.48550/ARXIV.2207.08815, 2022.

Gu, M., Wang, X., and Berger, J. O.: Robust Gaussian Stochastic Process Emulation, https://doi.org/10.48550/ARXIV.1708.04738, 2017.

Guo, Y., Yu, X., Xu, Y.-P., Chen, H., Gu, H., and Xie, J.: AI-based techniques for multi-step streamflow forecasts: application for multi-objective reservoir operation optimization and performance assessment, Hydrol Earth Syst Sci, 2021.

Guo, Z., Dirmeyer, P. A., and DelSole, T.: Land surface impacts on subseasonal and seasonal predictability: LAND IMPACTS SUBSEASONAL PREDICTABILITY, Geophys. Res. Lett., 38, n/a-n/a, https://doi.org/10.1029/2011GL049945, 2011.

Hatfield, S., Chantry, M., Dueben, P., Lopez, P., Geer, A., and Palmer, T.: Building Tangent-Linear and Adjoint Models for Data Assimilation With Neural Networks, J. Adv. Model. Earth Syst., 13, e2021MS002521, https://doi.org/10.1029/2021MS002521, 2021.

Hersbach, H., Bell, B., Berrisford, P., Hirahara, S., Horányi, A., Muñoz-Sabater, J., Nicolas, J., Peubey, C., Radu, R., Schepers, D., Simmons, A., Soci, C., Abdalla, S., Abellan, X., Balsamo, G., Bechtold, P., Biavati, G., Bidlot, J., Bonavita, M., De Chiara, G., Dahlgren, P., Dee, D., Diamantakis, M., Dragani, R., Flemming, J., Forbes, R., Fuentes, M., Geer, A., Haimberger, L., Healy, S., Hogan, R. J., Hólm, E., Janisková, M., Keeley, S., Laloyaux, P., Lopez, P., Lupu, C., Radnoti, G., De Rosnay, P., Rozum, I., Vamborg, F., Villaume, S., and Thépaut, J.: The ERA5 global reanalysis, Q. J. R. Meteorol. Soc., 146, 1999–2049, https://doi.org/10.1002/qj.3803, 2020.

Hochreiter, S. and Schmidhuber, J.: Long Short-Term Memory, Neural Comput., 9, 1735–1780, https://doi.org/10.1162/neco.1997.9.8.1735, 1997.

Irrgang, C., Boers, N., Sonnewald, M., Barnes, E. A., Kadow, C., Staneva, J., and Saynisch-Wagner, J.: Towards neural Earth system modelling by integrating artificial intelligence in Earth system science, Nat. Mach. Intell., 3, 667–674, https://doi.org/10.1038/s42256-021-00374-3, 2021.

James, W. and Stein, C.: Estimation with Quadratic Loss, in: Breakthroughs in Statistics, edited by: Kotz, S. and Johnson, N. L., Springer New York, New York, NY, 443–460, https://doi.org/10.1007/978-1-4612-0919-5_30, 1992.

Kadra, A., Lindauer, M., Hutter, F., and Grabocka, J.: Well-tuned Simple Nets Excel on Tabular Datasets, http://arxiv.org/abs/2106.11189, 5 November 2021.

Keisler, R.: Forecasting Global Weather with Graph Neural Networks, http://arxiv.org/abs/2202.07575, 15 February 2022.

Kimpson, T., Choulga, M., Chantry, M., Balsamo, G., Boussetta, S., Dueben, P., and Palmer, T.: Deep learning for quality control of surface physiographic fields using satellite Earth observations, Hydrol. Earth Syst. Sci., 27, 4661–4685, https://doi.org/10.5194/hess-27-4661-2023, 2023.

Kingma, D. P. and Ba, J.: Adam: A Method for Stochastic Optimization, 2017.

Kratzert, F., Herrnegger, M., Klotz, D., Hochreiter, S., and Klambauer, G.: NeuralHydrology -- Interpreting LSTMs in Hydrology, https://doi.org/10.48550/ARXIV.1903.07903, 2019a.

Kratzert, F., Klotz, D., Shalev, G., Klambauer, G., Hochreiter, S., and Nearing, G.: Towards learning universal, regional, and local hydrological behaviors via machine learning applied to large-sample datasets, Hydrol. Earth Syst. Sci., 23, 5089–5110, https://doi.org/10.5194/hess-23-5089-2019, 2019b.

Lam, R., Sanchez-Gonzalez, A., Willson, M., Wirnsberger, P., Fortunato, M., Alet, F., Ravuri, S., Ewalds, T., Eaton-Rosen, Z., Hu, W., Merose, A., Hoyer, S., Holland, G., Vinyals, O., Stott, J., Pritzel, A., Mohamed, S., and Battaglia, P.: Learning skillful medium-range global weather forecasting, Science, 382, 1416–1421, https://doi.org/10.1126/science.adi2336, 2023.

Lang, S., Alexe, M., Chantry, M., Dramsch, J., Pinault, F., Raoult, B., Clare, M. C. A., Lessig, C., Maier-Gerber, M., Magnusson, L., Bouallègue, Z. B., Nemesio, A. P., Dueben, P. D., Brown, A., Pappenberger, F., and Rabier, F.: AIFS - ECMWF's data-driven forecasting system, http://arxiv.org/abs/2406.01465, 3 June 2024.

Lees, T., Reece, S., Kratzert, F., Klotz, D., Gauch, M., De Bruijn, J., Kumar Sahu, R., Greve, P., Slater, L., and Dadson, S. J.: Hydrological concept formation inside long short-term memory (LSTM) networks, Hydrol. Earth Syst. Sci., 26, 3079–3101, https://doi.org/10.5194/hess-26-3079-2022, 2022.

Li, L., Carver, R., Lopez-Gomez, I., Sha, F., and Anderson, J.: SEEDS: Emulation of Weather Forecast Ensembles with Diffusion Models, https://doi.org/10.48550/ARXIV.2306.14066, 2023.

Machac, D., Reichert, P., and Albert, C.: Emulation of dynamic simulators with application to hydrology, J. Comput. Phys., 313, 352–366, https://doi.org/10.1016/j.jcp.2016.02.046, 2016.

Meyer, D., Grimmond, S., Dueben, P., Hogan, R., and Van Reeuwijk, M.: Machine Learning Emulation of Urban Land Surface Processes, J. Adv. Model. Earth Syst., 14, e2021MS002744, https://doi.org/10.1029/2021MS002744, 2022.

Mironov, D. and Helmert, J.: Parameterization of Lakes in NWP and Climate Models, n.d.

Muñoz-Sabater, J., Dutra, E., Agustí-Panareda, A., Albergel, C., Arduini, G., Balsamo, G., Boussetta, S., Choulga, M., Harrigan, S., Hersbach, H., Martens, B., Miralles, D. G., Piles, M., Rodríguez-Fernández, N. J., Zsoter, E., Buontempo, C., and Thépaut, J.-N.: ERA5-Land: a state-of-the-art global reanalysis dataset for land applications, Earth Syst. Sci. Data, 13, 4349–4383, https://doi.org/10.5194/essd-13-4349-2021, 2021.

Nath, S., Lejeune, Q., Beusch, L., Seneviratne, S. I., and Schleussner, C.-F.: MESMER-M: an Earth system model emulator for spatially resolved monthly temperature, Earth Syst. Dyn., 13, 851–877, https://doi.org/10.5194/esd-13-851-2022, 2022.

Nearing, G., Cohen, D., Dube, V., Gauch, M., Gilon, O., Harrigan, S., Hassidim, A., Klotz, D., Kratzert, F., Metzger, A., Nevo, S., Pappenberger, F., Prudhomme, C., Shalev, G., Shenzis, S., Tekalign, T. Y., Weitzner, D., and Matias, Y.: Global prediction of extreme floods in ungauged watersheds, Nature, 627, 559–563, https://doi.org/10.1038/s41586-024-07145-1, 2024.

Ozaki, Y., Tanigaki, Y., Watanabe, S., Nomura, M., and Onishi, M.: Multiobjective Tree-Structured Parzen Estimator, J. Artif. Intell. Res., 73, 1209–1250, https://doi.org/10.1613/jair.1.13188, 2022.

Pappenberger, F., Ramos, M. H., Cloke, H. L., Wetterhall, F., Alfieri, L., Bogner, K., Mueller, A., and Salamon, P.: How do I know if my forecasts are better? Using benchmarks in hydrological ensemble prediction, J. Hydrol., 522, 697–713, https://doi.org/10.1016/j.jhydrol.2015.01.024, 2015.

Popov, S., Morozov, S., and Babenko, A.: Neural Oblivious Decision Ensembles for Deep Learning on Tabular Data, https://doi.org/10.48550/ARXIV.1909.06312, 2019.

Reichstein, M., Camps-Valls, G., Stevens, B., Jung, M., Denzler, J., Carvalhais, N., and Prabhat: Deep learning and process understanding for data-driven Earth system science, Nature, 566, 195–204, https://doi.org/10.1038/s41586-019-0912-1, 2019.

Sener, O. and Koltun, V.: Multi-Task Learning as Multi-Objective Optimization, https://doi.org/10.48550/ARXIV.1810.04650, 2018.

Shukla, J.: Dynamical Predictability of Monthly Means, J. Atmospheric Sci., 38, 2547–2572, https://doi.org/10.1175/1520-0469(1981)038<2547:DPOMM>2.0.CO;2, 1981.

Shwartz-Ziv, R. and Armon, A.: Tabular Data: Deep Learning is Not All You Need, https://doi.org/10.48550/ARXIV.2106.03253, 2021.

Slater, L. J., Arnal, L., Boucher, M.-A., Chang, A. Y.-Y., Moulds, S., Murphy, C., Nearing, G., Shalev, G., Shen, C., Speight, L., Villarini, G., Wilby, R. L., Wood, A., and Zappa, M.:

Hybrid forecasting: blending climate predictions with AI models, Hydrol. Earth Syst. Sci., 27, 1865–1889, https://doi.org/10.5194/hess-27-1865-2023, 2023.

Thorpe, A., Bauer, P., Magnusson, L., and Richardson, D.: An evaluation of recent performance of ECMWF's forecasts, https://doi.org/10.21957/HI1EEKTR, 2013.

Van Katwyk, P., Fox-Kemper, B., Seroussi, H., Nowicki, S., and Bergen, K. J.: A Variational LSTM Emulator of Sea Level Contribution From the Antarctic Ice Sheet, J. Adv. Model. Earth Syst., 15, e2023MS003899, https://doi.org/10.1029/2023MS003899, 2023.

Viterbo, P.: Land_surface_processes, 2002.

Wesselkamp, M., Moser, N., Kalweit, M., Boedecker, J., and Dormann, C. F.: Process-guidance improves predictive performance of neural networks for carbon turnover in ecosystems, https://doi.org/10.48550/ARXIV.2209.14229, 2022.

Zwart, J. A., Oliver, S. K., Watkins, W. D., Sadler, J. M., Appling, A. P., Corson-Dosch, H. R., Jia, X., Kumar, V., and Read, J. S.: Near-term forecasts of stream temperature using deep learning and data assimilation in support of management decisions, JAWRA J. Am. Water Resour. Assoc., 59, 317–337, https://doi.org/10.1111/1752-1688.13093, 2023.