Frontier Development Lab Technical Memorandum

Data-Driven and Physically-aware Precipitation Forecasting

FDL Europe 2020 – Digital Twin Earth

Researchers:
Daniele De Martini – University of Oxford
Christian Schroeder de Witt – University of Oxford
Catherine Tong – University of Oxford
Valentina Zantedeschi – GE Global Research

Mentors:
Piotr Biliński – University of Warsaw
Matthew Chantry – University of Oxford
Freddie Kalaitzis –
Duncan Watson-Parris – University of Oxford

September 2020

“Ad astra, per algoritmos"

Acknowledgments

This work has been enabled by Frontier Development Lab (FDL) Europe (https://fdleurope.org) a public / private partnership between the European Space Agency (ESA), Trillium Technologies and the University of Oxford (http://www.ox.ac.uk), and supported by Google Cloud, NVIDIA, SCAN, IBM, AIRBUS, Planet, IWMI and Catapult. We wish to thank our expert reviewers Julien Brajard, Peter Düben and Stephan Rasp for their helpful feedback throughout the project.

Abstract

This technical memorandum describes and discusses the work carried out by the Digital-Twin Earth (DTE) team during the Frontier Development Lab (FDL) sprint in the Summer of 2020. We propose a data-driven Deep Learning (DL) approach to precipitation prediction as a cheaper alternative to bespoke physics-based models that rely on some of the worlds largest supercomputers. We aim to enable data-driven investigations for global precipitation-intensity forecasting from satellite imagery. Our goal is to assess the challenges involved in moving from physics-based models to a data-driven digital twin of the Earth.

The proposed data-driven approached is based on a three-stage pipeline:

1. the state of the Earth is firstly extracted from satellite data, with special attention to the meaningful variables for precipitation prediction;
2. the current Earth state is propagated forward in time to a desired future state using some degree of atmospheric knowledge;
3. finally, a precipitation distribution is derived from this future state.

Moreover, as a by-product, we introduce **RainBench**, a multi-modal benchmark dataset for data-driven precipitation forecasting, which includes simulated satellite data, a selection of relevant meteorological data from the ERA5 reanalysis product and Integrated Multi-satellitE Retrievals (IMERG) precipitation data. Along with **RainBench**, we release **PyRain**, a library to process the three datasets efficiently, reducing time and hardware costs and thus lowering the barrier to entry into this field.

Contents

List of Acronyms

CDS Climate Data Storage.

CNN Convolutional Neural Network.

DA Data Assimilation.

DL Deep Learning.

DTE Digital-Twin Earth.

ECMWF European Centre for Medium-Range Weather Forecasts.

ERA5 ECMWF Re-Analysis, 5th edition.

FDL Frontier Development Lab.

GAN Generative Adversarial Network.

IFS Integrated Forecast System.

IMERG Integrated Multi-satellitE Retrievals.

LSTM Long Short-Term Memory.

ML Machine Learning.

RMSE Root-Mean-Square Error.

SimSat simulated satellite data.

SVG Stochastic Video Generation.

List of Figures

List of Tables

1. Introduction

Extreme precipitation events, such as violent rainfall and hail storms, routinely ravage economies and livelihoods around the developing world. Climate change further aggravates this issue (13). Data-driven deep learning approaches could widen the access to accurate multi-day forecasts, to mitigate against such events. Weather prediction is a huge challenge, assimilating hundreds of millions of observations every day. Currently this is carried out by bespoke physics-based models using some of the worlds largest supercomputers. In this work, we aim to enable data-driven investigations for global precipitation intensity forecasting from satellite imagery. Our goal is to assess the challenges involved in moving from physics-based models to a data-driven digital twin of the Earth.

To this end, we first introduce **RainBench**, a new multi-modal benchmark dataset for data-driven precipitation forecasting. It includes simulated satellite data, a selection of relevant meteorological data from the ERA5 reanalysis product, and IMERG precipitation data. We also present an extensive analysis of our novel dataset. The access patterns required for weather prediction, where both time and space have to be ingested simultaneously, mean that standard methods for data-loading produce bottlenecks for learning. Therefore, along with **RainBench**, we release **PyRain**, a library to process the three datasets efficiently, reducing time and hardware costs and thus lowering the barrier to entry into this field.

Using these tools we examine the challenges involved in end-to-end weather forecasting using two approaches. First, we consider the problem as a single task, where satellite and weather state are used to produce future time forecasts of precipitation. Second, we consider the problem as three elements. 1. The atmospheric state is estimated from recent satellite data, which in the weather forecasting field is called Data Assimilation (DA). 2. The atmospheric state is propagated forward in time. 3. The atmospheric state is used to derive the precipitation falling within a nearby time interval. For both of these approaches we develop baseline results.

Finally, we discuss the challenges for modelling weather over medium timescale and outline several fruitful avenues for future research.

Identified Need

Extreme precipitation events, such as violent rain and hail storms, can devastate crop fields and disrupt harvests (41; 20). These events can be locally forecasted with sophisticated numerical weather models that rely on extensive ground and satellite observations. However, such approaches require access to compute and data resources that developing countries in need - particularly in South America and West Africa - cannot afford (18; 12). The lack of advance planning for precipitation events impedes socioeconomic development and ultimately affects the livelihoods of millions around the world. Given the increase in global precipitation and extreme precipitation events driven by climate change (13), the need for accurate precipitation forecasts is ever more pressing.

Data-driven machine learning approaches circumvent the dependence on traditional resource-intensive numerical models, which typically take several hours to run (39), incurring a significant time lag. In contrast, deep learning models deployed on dedicated high-throughput hardware can produce inferences in a matter of seconds. However, while there have been a number of attempts in forecasting precipitation with neural networks, these modelling efforts have mostly been fragmented across different local regions, which hinders a systematic comparison into their performance.

In this work, we introduce **RainBench**, a multi-modal dataset to support data-driven forecasting of global precipitation from satellite imagery. We curate three types of datasets: simulated satellite data (SimSat), numerical reanalysis data (ERA5), and global precipitation estimates (IMERG). The use of satellite images to forecast precipitation globally would circumvent the need to collect ground station data, and hence they are key to our vision for widening the access to multi-day precipitation forecasts. Reanalysis data provide estimates of complete atmospheric state, and IMERG provides rigorous estimates of global precipitation. Access to these data opens up opportunities to develop more timely and potentially physics-informed forecast models, which so far could not have been studied systematically.

Most related to our work, (31) have developed WeatherBench, a benchmark environment for global data-driven medium-range weather forecasting. This dataset forms an excellent first step in weather forecasting. However, some important features of WeatherBench limit its use for end-to-end precipitation forecasts. WeatherBench does not include any observational raw data (e.g. satellite data) and only contains ERA5 reanalysis data, which have limited resolution of extreme precipitation events. Further, WeatherBench does not include a fast dataloading pipeline to train ML models, which we found to be a significant bottleneck in our model development and testing process. This gap prompted us to also release **PyRain**, a data processing and experimentation framework with fast and configurable multi-modal dataloaders.

Related Work

Weather forecasting systems have not fundamentally changed since they were first operationalised nearly 50 years ago. Current state-of-the-art operational weather forecasting systems rely on numerical models that integrate the physical atmospheric state in time based on a system of physical equations and parameterised subgrid processes (4). While global simulations typically run at grid sizes of $10\,\mathrm{km}$, regional models can reach $1.5\,\mathrm{km}$ (10) . For global simulations, skilled forecast lengths are usually limited to a maximum of 10 days, with a conjectured hard limit of 14 to 15 days (46). The skill is dependent upon the field of interest, with large-scale temperature patterns having much longer predictability time than precipitation events. *Nowcasting*, i.e. high-resolution weather forecasting only a few hours in advance, is currently limited by the several hours that numerical forecasting models take to run (39).

Given the huge amounts of data currently available from both numerical models and observations, new opportunities exist to train data-driven models to produce these forecasts. The current boom in Machine Learning (ML) has inspired several other groups to approach the problem of weather forecasting. Early work by (45) investigated using convolutional recurrent neural networks for precipitation nowcasting. More recently, (39) from Google proposed a "(weather) model free" approach, MetNet, which seeks to forecast precipitation in continental USA using geostationary satellite images and radar measurements as inputs. This approach performs well up to 7-8 hours, but inevitably runs into a forecast horizon limit as information from global or surrounding geographic areas is not incorporated into the system. This time window has value though it would not enable substantial disaster preparedness.

The prediction of extreme precipitation (and other extreme weather events) has a long history with traditional forecasting systems (17). More recent developments in ensemble weather forecasting systems surround the introduction of novel forecasting indices (47) and post-processing (11). There has also been other deep-learning based precipitation forecasting models as motivated by the monsoon prediction problem, for example, (34) and (35) use a stacked autoencoder to identify climatic predictors and an ensemble regression tree model, while (29) use kriging and multi-layer perceptrons to predict monsoon rainfall from ERA5 data.

WeatherBench (31) is a benchmark dataset for data-driven global weather forecasting, derived from data in the ERA5 archive. Its release has prompted a number of follow-up works to employ deep learning techniques for weather forecasting, although the variables considered have only been restricted to the forecasts of relatively static variables, such as 500 hPa geopotential and 850 hPa temperature (43; 44; 32; 7; 3). Unlike RainBench which incorporates the element of observational input data from (simulated) satellites, WeatherBench's data comes solely from the ERA5 reanalysis archive, and thus provides no route to producing an end-to-end forecasting system.

2. Data

In this section, we introduce RainBench, which consists of data derived from three publicly-available sources: (1) European Centre for Medium-Range Weather Forecasts (ECMWF) simulated satellite data (SimSat), (2) the ECMWF Re-Analysis, 5th edition (ERA5) reanalysis product, and (3) IMERG global precipitation estimates.

SimSat We use simulated satellite data in place of real satellite imagery to minimise data processing requirements and to simplify the prediction task. SimSat data are model-simulated satellite data generated from ECMWF's high-resolution weather-forecasting model using the RTTOV radiative transfer model (36). SimSat emulates three spectral channels from the Meteosat-10 SEVIRI satellite (2). SimSat provides information about global cloud cover and moisture features and has a native spatial resolution of about 0.1° – i.e. about $10\,\mathrm{km}$ – at three-hourly intervals. The product is available from April 2016 to present (with a lag time of $24\,\mathrm{h}$). Using simulated satellite data provides an intermediate step to using real satellite observations as the images are a global nadir view of Earth, avoiding issues of instrument error and large numbers of missing values. Here we aggregate the data to 0.25° – about $30\,\mathrm{km}$ – to be consistent with the ERA5 dataset.

ERA5 We use ERA5 as it is an accurate and commonly used reanalysis product familiar to the climate science community (31). ERA5 reanalysis data provides hourly estimates of a variety of atmospheric, land and oceanic variables, such as specific humidity, temperature and geopotential height at different pressure levels (14). Estimates cover the full globe at a spatial resolution of 0.25° and are available from 1979 to present, with a lag time of five days.

IMERG IMERG is a global half-hourly precipitation estimation product provided by NASA (15). Specifically we use the Final Run product which primarily uses satellite data from multiple polar-orbiting and geo-stationary satellites. This estimate is then corrected using data from reanalysis products (MERRA2, ERA5) and rain-gauge data. IMERG is produced at a spatial resolution of 0.1° – about $10\,\mathrm{km}$ – and is available from June 2000 to present, with a lag time of about three to four months.

To facilitate efficient experimentation, we convert all data from their original resolutions to lower resolutions using bilinear interpolation. Throughout this memo, we predominately consider data at 5.625°.

RainBench provides precipitation values in two forms: ERA5 precipitation and IMERG precipitation. The ERA5 precipitation is accumulated precipitation over the last hour – in m – and is calculated as an averaged quantity over a grid-box. IMERG precipitation has here been aggregated into hourly accumulated precipitation – in mm – and should be considered as a point estimate of the precipitation.

Figure 2.1 shows the distribution of precipitation for the years 2000-2017 with both ERA5 and IMERG. Their different distributions indicate that the quality of global precipitation estimates, in particular related to extreme precipitation events, varies with the choice of precipitation data. IMERG has significantly larger rainfall tails than ERA5, and these tails rapidly vanish with decreasing dataset resolution. The underestimation of extreme precipitation events in ERA5 is clearly visible.

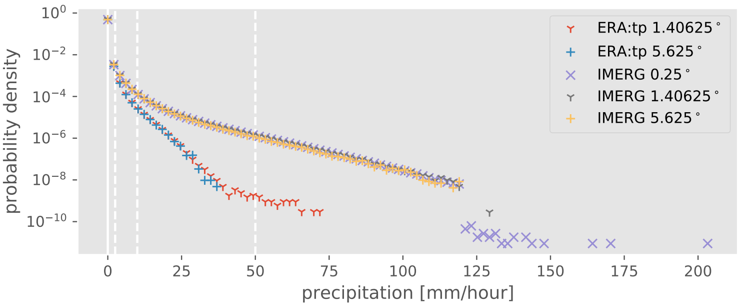


Figure 2.1: Precipitation histogram for the years 2000-2017 with both ERA5 and IMERG at different resolutions. Vertical lines delineate convection rainfall types: slight ($0–2\,\mathrm{mm\,h^{-1}}$), moderate ($2–10\,\mathrm{mm\,h^{-1}}$), heavy ($10–50\,\mathrm{mm\,h^{-1}}$), and violent (over $50\,\mathrm{mm\,h^{-1}}$) (24).

Data Analysis

To analyse the dependencies between all RainBench variables, we calculate pairwise Spearman's rank correlation indices over latitude band from -60 to $60°$ and date range from April 2016 to December 2019 (see Figure 2.2). In contrast to Pearson's correlation coefficient, Spearman's correlation coefficient is significant if there is a, potentially non-linear, monotonic relationship between variables, while Pearson's considers only linear correlations. This allows to capture relationships between variables such as between temperature and absolute latitude. Comparing correlations at altitude pressure levels $300\,\mathrm{hPa}$ (about $10\,\mathrm{km}$) and $850\,\mathrm{hPa}$ ($1.5\,\mathrm{km}$), we can see that they are almost identical, save for a few exceptions: Specific humidity, q, and geopotential height, z, correlate strongly at $300\,\mathrm{hPa}$ but not at $850\,\mathrm{hPa}$, cloud ice water content, ciwc, generally correlates more strongly at higher altitude (and cloud liquid water content, clwc, vice versa). A careful examination of the underlying physical dependencies results in the realisation that all of these asymmetries stem mostly from latitudinal correlations or effects related to cloud formation, e.g. ice and liquid form in clouds at different temperatures/altitudes.

As we are particularly interested in variables that have predictive skill on precipitation, we note that all SimSat spectral channels moderately anti-correlate with both ERA5 and IMERG precipitation estimates. Interestingly, SimSat signals correlate much stronger with specific humidity and cloud ice water content at higher altitude, which might be a consequence of spectral penetration depth. ERA5 state variables that correlate most with either precipitation estimates are specific humidity and temperature. Cloud ice water content correlates moderately strongly with precipitation estimates at high altitude, but not at all at lower altitude (where ice water content tends to be much lower). Interestingly, a number of time-varying ERA5 state variables correlate more strongly with IMERG precipitation than ERA5 precipitation, as do SimSat signals. Conversely, a number of constant variables, such as land-sea mask, orography and soil type are significantly anti-correlated with ERA5 precipitation, but not at all correlated with IMERG. Overall, we find that all variables that are significantly correlated or anti-correlated with both ERA5 tp and IMERG are also correlated or anti-correlated with SimSat clbt:0-2, suggesting that precipitation prediction from simulated satellite data alone may be feasible.

We also study in Figure 2.3 the precipitation class occurrences derived from IMERG at native

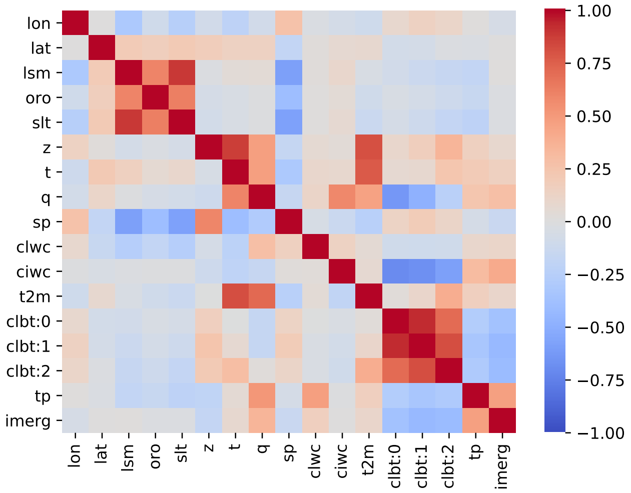
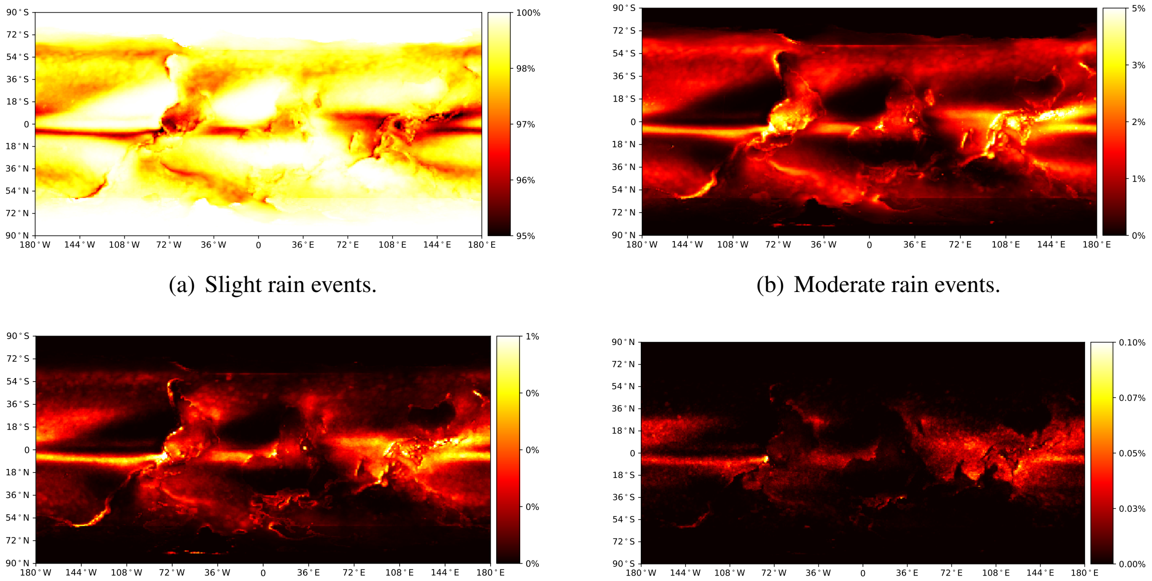


Figure 2.2: Spearman's correlation of RainBench variables from April 2016 to December 2019 at a spatial resolution of $5.625°$ in latitude band $[-60°, 60°]$ at pressure levels $300\,\mathrm{hPa}$ (about $10\,\mathrm{km}$) (upper triangle) and $850\,\mathrm{hPa}$ ($1.5\,\mathrm{km}$) (lower triangle). Legend: lon: longitude, lat: latitude, lsm: land-sea mask, oro: orography (topographic relief of mountains), lst: soil type, z: geopotential height, t: temperature, q: specific humidity, sp: surface pressure, clwc: cloud liquid water content, ciwc: cloud ice water content, $t2m$: temperature at 2m, clbt:i: ith SimSat channel, tp: ERA5 total precipitation, imerg: IMERG precipitation. All correlations in this plot are statistically significant ($p < 0.05$).

resolution ($0.1°$) with max-pooling as downscaling to preserve pixel-wise extremes. Global occurrences are defined as the probability of a given precipitation class for each location on the globe and time-independent. We notice that extreme events are mostly concentrated in the tropics while slight rain events occur evenly around the Earth. Once again, this analysis highlights the extreme imbalance between types of precipitation events.



(a) Slight rain events.

(b) Moderate rain events.

(c) Heavy rain events.

(d) Violent rain events.

Figure 2.3: Global occurrences of the four defined classes of precipitation intensity.

3. Proposed Methodology

Our goal is to propose a data-driven, medium-term ($8\,\mathrm{h}$ - $5\,\mathrm{d}$), precipitation-forecasting system from satellite images at global scale. The key aspect of the proposed system is that it is trained to mimic (*emulate*) a physical simulator. This will be achieved, as briefly shown in Figure 3.2, by injecting physical knowledge through data coming from the real-world physical simulator ERA5.

We illustrate two different approaches for building such a system. We first describe an end-to-end approach, which directly estimates precipitation rates at the given lead time from historical data. As we will see in Section 5, directly forecasting precipitation is an intrinsically hard task, as datasets are not available at the same times. This problem significantly reduces the training set, which consequently limits the capacity of the trained models. For these reasons, we also propose a three-steps approach, which firstly estimates the present atmospheric state from historical Earth observations, secondly forecasts the atmospheric state at the chosen lead time given a stream of estimated states, and lastly predicts the precipitation rate at the same time step. This approach has the advantage of leveraging all available data, as each step is tackled independently at first, allowing to train larger models that are able to capture all different dynamics. This also allows easier comparison with existing baselines.

End-to-end Approach

For the end-to-end approach, we consider three different input data settings as historical data: SimSat, reanalysis data (ERA5), or both. From the ERA5 dataset, we select a subset of variables as input to the forecast model based on our data analysis results; the inputs are geopotential (z), temperature (t), humidity (q), cloud liquid water content (clwc), cloud ice water content (ciwc), each sampled at $300\,\mathrm{hPa}$, $500\,\mathrm{hPa}$ and $850\,\mathrm{hPa}$ geopotential heights; to these we add the surface pressure and the 2-meter temperature (t2m), as well as static variables that describe the location and surface of the Earth, i.e. latitude, longitude, land-sea mask, orography and soil type. From the SimSat dataset, the inputs are cloud-brightness temperature (clbt) taken at three wavelengths. We normalize each variable with its global mean and standard deviation.

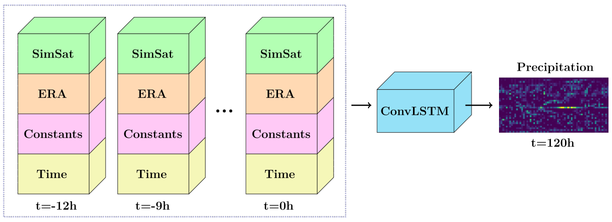


Figure 3.1: Modelling setup for the benchmark forecasting tasks.

We perform experiments with a neural network based on Convolutional LSTMs, which have been shown to be effective for regional precipitation nowcasting (45). We structure our forecasting task based on MetNet's configurations (39), where a single model is trained conditioned on time and is capable of forecasting at different lead times.

The network's input is composed of a time series $\{x_t\}$, where each x_t is the set of standardized features at time t, sampled in regular intervals Δt from $t = -T$ to $t = 0$; the output is a precipitation forecast y at lead time $t = \tau \leq \tau_L$. In addition to the aforementioned atmospheric features, static features (e.g. latitude) along with three time-dependant features (hour, day, month) are repeated per timestep. The input vector is then concatenated with a lead-time one-hot vector x_τ. In our experiments, we adopt $T = 12$ h, $\Delta t = 3$ h and forecasts at 24-hour intervals up to $\tau_L = 120$ h. We note that we do not include precipitation as an input temporal feature. An overview of this setup is shown in Figure 3.1.

We consider two tasks, with the ground truth precipitation values taken from either ERA5 or IMERG, and approach them as a regression problem. Following (31), we use the mean latitude-weighted Root-Mean-Square Error (RMSE) as loss and evaluation metric. We compare the results to two standard baselines: (1) a *persistence* forecast in which the precipitation at $t = 0$ is used as prediction at $t = \tau$, and (2) a *climatology* forecast in which the mean precipitation in the training data is used as prediction.

Three-step Approach

The approach is composed by three steps, which are self-contained and can be designed and trained separately; a final fine-tuning process can be used to harmonise them for the task at hand.

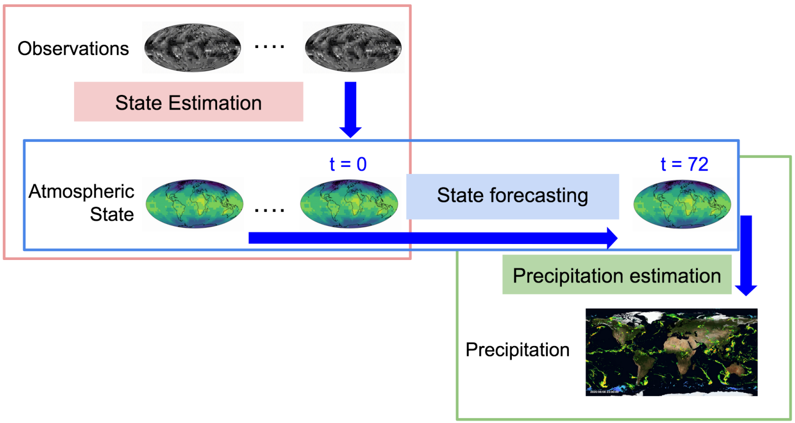


Figure 3.2: System overview. The three steps – state estimation, state forecasting and precipitation estimation – are highlighted by different colours.

We will refer to the three steps as *state estimation*, *state forecasting* and *precipitation estimation*; we describe them in the next sections.

State Estimation

The first aspect of the system will carry out the extraction from weather-satellite imagery of a compact representation of the state of the atmosphere, which in essence is a DA problem. To achieve this, we use a historic sequence of SimSat data as input (in Δt-hour intervals from $t = -T$ to $t = 0$) to predict the atmospheric state at $t = 0$.

We are interested in predicting the atmospheric-state variables which would be helpful in the latter steps of the precipitation forecasting pipeline. Specifically, we consider an output vector consisting of the following variables taken from the ERA5 reanalysis product: geopotential, temperature and humidity (sampled at $300\,\mathrm{hPa}$, $500\,\mathrm{hPa}$ and $850\,\mathrm{hPa}$), cloud liquid water content and cloud ice water content (sampled at $300\,\mathrm{hPa}$ and $500\,\mathrm{hPa}$), as well as the surface pressure (`sp`) and 2-metre temperature (`t2m`).

The setup so far assumes only the availability of satellite imagery at inference time $t = 0$. It is reasonable to expect that other forms of data which are conducive to this state extraction task will also be available at $t = 0$. Near-ground observations such as `sp` and `t2m` are frequently measured. Further, we can reasonably assume that the ERA atmospheric state vector from 24 hours ago could be useful for inference at $t = 0$. As a result, we consider three alternative setups for the input to a machine learning model:

1. **SimSat only.** Here the input sequence is formed of the 3 frequency bands of SimSat data ($clbt : 0 - 2$). We take $\Delta t = 3$ and $T = 12$.

2. **SimSat and ground observations.** The input sequence is formed of $clbt : 0 - 2$ and ground observations (namely `sp` and `t2m`). We take $\Delta t = 3$ and $T = 12$.

3. **SimSat, ground observations and historic atmospheric states.** We take an input sequence $\Delta t = 24$ and $T = 72$. At time steps $t = -72$ to $t = -24$, the input vector is formed of $clbt : 0 - 2$, ground observations and the ERA state variables. At $t = 0$, the input vector is formed only of $clbt : 0 - 2$ and ground observations.

Each of the input sequences specified above is then concatenated with static variables (latitude, longitude, land-sea mask, orography, soil type), repeated per time step.

For this task, we train a Convolutional LSTM model (45) with an Adam Optimizer, using latitude-weighted mean squared error as the loss function. As baselines, we use Persistence (i.e. using the ERA state at $t = -24$ as prediction output), and Climatology (i.e. using the training set mean as prediction output per ERA variable) for comparison.

State Forecasting

The second part of the system propagates the atmospheric state forward in time. Using ERA5 as training data will implicitly allow the model to emulate the physical dynamics of the atmosphere. This ability to probe the weather state provides a route to surpass the forecast horizon of MetNet (39).

For the backbone methodology for this step we looked into a *stochastic video prediction* approach, which we will describe in the next sections.

Stochastic Video Prediction

In the last years, video prediction has seen a lot of interest from the computer vision community and, more recently, from the robotics community (26); the ability of predicting future outcomes from past knowledge is seen as a key component of intelligent decision-making systems and the progresses in Deep Learning (DL) software and hardware opened the possibility to process enough data to achieve promising results.

From Figure 3.3 we can see one of the main problems with predicting the future with deterministic algorithms, like neural networks are: the uncertainty of the process is reflected in uncertainty of the outcome of the prediction, leading to an averaged future. Many steps have been taken to solve such problems; we are going to focus on the usage of probabilistic approaches since the process we want to predict is inherently stochastic.

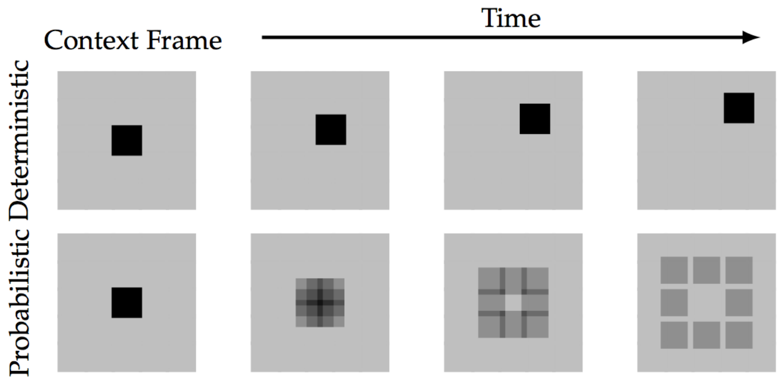


Figure 3.3: Effects of predicting uncertain futures with deterministic algorithims.

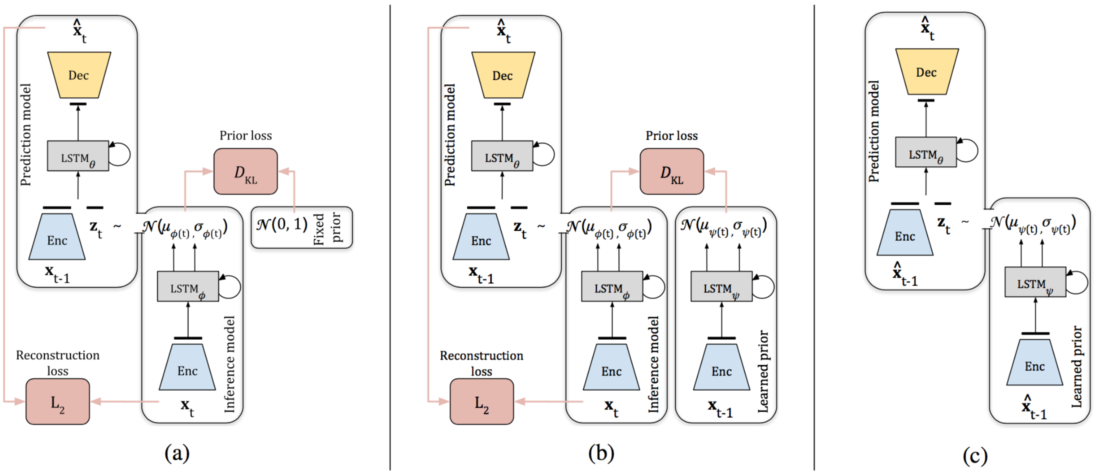


Figure 3.4: The structure of the Stochastic Video Generation methodology described in (9). In (a) the structure for training the version using a *fixed prior*, e.g. a unary gaussian distribution. (b) & (c) instead depict the version using a *learnt prior* in training and inference respectively.

Selected Methodology We based our solution on the one described in (9). This methodology, called Stochastic Video Generation (SVG) with *learnt prior* and depicted in Figure 3.4, is composed by two main parts: an encoder-decoder structure which carries out the feature extraction from the input video and the reconstruction, and a Variational Encoder which encodes the *learnt prior*, i.e. a stochastic distribution that mirrors the uncertainty of the possible futures.

Interesting is the way the training procedure is carried out. In order to learn a meaningful prior, the encoder-decoder network is conditioned using a prior obtained from the future frame (available at training but not at inference); this helps the network to converge and to learn a probabilistic distribution which is helpful for video prediction (called *posterior*), which is ensured by a reconstruction loss on the future frame. Since the future frame is not available at inference, though, a second Variational Encoder is trained using only the available frames and constrained to learn a prior which follows the posterior by means of a KL loss.

Proposed adaptations We made two main adaptations: firstly a structural one to cope with the data streams and an architectural one to add a adversarial losses to the learning process.

Since the methodology is inherently iterative, i.e. to predict 3 steps ahead the method has to predict iteratively the two steps before, the network has been adapted to predict all the non-fixed variables and the training procedure to produce a sequence of predictions as long as the lead time and to combine the losses altogether. As for the network adaptations for the fixed variables, e.g. latitude and longitude, the inference procedure is modified to produce an output tensor of different dimensions than the input one, as well as to concatenate dynamically the fixed variables. This has the advantage to diminish the network dimensions and to mitigate the differences in dynamics that lead the various variables.

Lastly, we followed (42; 19; 1; 23) and added a set of multi-resolution discriminators to the architecture to enhance the crispness of the results. We implemented single-channel and multi-channel – i.e. looking at the single variable or at the atmospheric state as a whole – discriminators for single image and multi-image results – i.e. looking at the single timestamp or at the evolution of the variables. Unfortunately, due to lack of time, we could test only the static discriminators for the atmospheric state.

Lastly, modifications to the network bottleneck have been developed place to use a Wavenet-like (25) approach at fusing the past-timestamps information but no tests have taken place. This has been done since Long Short-Term Memory (LSTM) cells are notoriously delicate to train.

Precipitation Estimation

The final step consists in estimating precipitation intensity given the predicted state of the atmosphere. For this task, we make use of IMERG as target ground-truth and of ERA5's temperature (t), humidity (q), cloud liquid water content (clwc), cloud ice water content (ciwc), at $300\,\mathrm{hPa}$, $500\,\mathrm{hPa}$ and $850\,\mathrm{hPa}$ geopotential levels and surface temperature (sp), land-sea mask (lsm) and orography. Both datasets are retrieved at $0.25°$ resolution. We implement a gridcell-wise neural network as global or regional contexts are not required when estimating the precipitation at the current time. Considered that at a given time t IMERG provides the precipitation accumulated over the period $[t, t+1]$, the model takes as inputs the feature values for both time steps: $X_t = [x_t; x_{t+1}]$. The model architecture itself is a concatenation of a batch normalization, five fully connected layers with Swish activation (30) and dropout (40), and a final fully connected layer.

In order to deal with the large output imbalance, we define the following four classes of precipitation intensity accumulated over three hours:

1. No-precipitation: precipitation intensity below $1\,\mathrm{mm}$;
2. Drizzle precipitation: precipitation intensity between 1 and $7.5\,\mathrm{mm}$;
3. Light precipitation: precipitation intensity between 7.5 and $22.8\,\mathrm{mm}$;
4. Heavy precipitation: precipitation intensity above $22.8\,\mathrm{mm}$.

These classes are useful for defining data sampling strategies to balance the target distribution at training and for defining metrics for assessing the quality of predictions fairly across the types of precipitation. Figure 3.5 reports the obtained class distributions for IMERG at $0.25°$ resolution. The choice of a gridcell-wise model allows us to balance the defined classes at training, through a masking process when evaluating the chosen loss function: given an input X_t and a target label mask y_t, we select all pixels of the minority class within the input X_t and randomly select an equal amount of pixels for each of the other classes, masking-out the remaining pixels. This procedure ensures that classes are equally represented at training and that the model can be easily fine-tuned together with the models of the previous steps. To further address the data imbalance, in the experiments we train the model by minimizing the Focal Loss (21). This loss is a modified version of the Cross Entropy loss that decreases the weights of well-classified examples, so that the training focuses on hard examples which are typically the ones from under-represented classes.

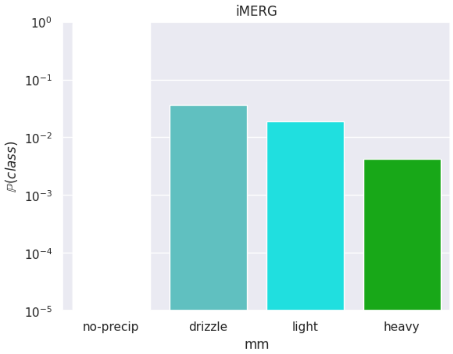


Figure 3.5: Precipitation class distributions for IMERG $0.25°$. Densities are reported in logarithmic scale.

4. Replicability

Data availability

The project has been built on three datasets all of which are publicly available.

ERA5 describes the atmospheric state (14). The data can be downloaded from the Climate Data Storage (CDS)[1]. The WeatherBench project (31) provides tools and detailed instructions on downloading and regridding the data and is our recommended starting location. In addition to the variables included in the WeatherBench dataset we download and use cloud ice water content and cloud liquid water content, both of which can be downloaded with the WeatherBench tools or directly from CDS.

Simulated satellite data is produced[2] from the ECMWF high-resolution forecasting system (38). The data has been downloaded at $0.25°$ then regridded using the WeatherBench tools to the desired resolution. ECMWF have recently adopted the Creative Commons 4.0 for all past forecasts, meaning data will soon be available for public download. Until then the data will be made available upon request from the authors.

IMERG is our precipitation dataset (15). Specifically, we use the calibrated precipitation product from the level 3 final run version of IMERG. Data is aggregate from the native half-hourly time resolution to either hourly or three hourly accumulated precipitation depending on the task. Spatially, we regrid the data from $0.1°$ to the desired resolution using the tools provided with the WeatherBench project (31).

Tools, Compute, Software Environment

We extensively made use of Google Cloud Platform and of Scan's NVIDIA DGX-1 for pre-processing the data and for running the experiments. Namely, we relied on DGX-1, with its 80 CPUs, 500 GB memory and 8 NVIDIA Tesla V100, on six Virtual Machines with 96 vCPUs, 360 GB memory and 8 NVIDIA Tesla V100 each, and on several SSDs of 40TB total capacity. Given the amount of data we were dealing with, these resources were fundamental to speed-up our extremely time and memory consuming tasks. Concerning the software, we utilized many scientific Python packages, principally pytorch, scikit-learn, numpy, iris, xarray, and JupyterLab.

[1] https://cds.climate.copernicus.eu/

[2] https://confluence.ecmwf.int/pages/viewpage.action?pageId=55127736

Table 4.1: Number of data samples loaded per second using PyRain versus a conventional NetCDF framework. Typical configurations assumed and performed on a NVIDIA DGX1 server with 64 CPUs.

	NetCDF	PyRain	Speedup
16 workers	40	2410	60.3×
64 workers	70	1930	27.6×

PyRain

To support efficient data-handling and experimentation on Rainbench, we release PyRain, an out-of-the-box experimentation framework.

PyRain introduces an efficient dataloading pipeline for complex sample access patterns that scales to the terabytes of spatial timeseries data typically encountered in the climate and weather domain. Previously identified as a decisive bottleneck by the Pangeo community[3], PyRain overcomes existing dataloading performance limitations through an efficient use of NumPy *memmap* arrays[4] in conjunction with optimised software-side access patterns.

In contrast to storage formats requiring *read* system calls, including HDF5[5], Zarr[6] or xarray[7], memory-mapped files use the *mmap* system call to map physical disk space directly to virtual process memory, enabling the use of *lazy* OS demand paging and circumventing the kernel buffer. While less beneficial for chunked or sequential reads and spatial slicing, memmaps can efficiently handle the fragmented random access inherent to the randomized sliding-window access patterns along the primary axis as required in model training.

In Table 4.1, we compare PyRain's memmap data reading capcity against a NetCDF+Dask [8] (33) dataloader. We find empirically that PyRain's memmap dataloader offers significant speedups over other solutions, saturating even SSD I/O with few process workers when used with PyTorch's (27) inbuilt dataloader.

Note that explicitly storing each training sample is not only slow and inflexible for research settings, but requires twenty to fifty times more storage and so usually comes at a higher cost than constructing samples on the fly. Thus e.g. the option of writing samples in TFRecord format (43; 1) seems only sensible for highly distributed training in production settings.

PyRain's dataloader is easily configurable and supports both complex multimodal item compositions, as well as periodic (39) and sequential (44) train-test set partitionings. Apart from its data-loading pipeline, PyRain also supplies flexible raw-data conversion tools, a convenient interface for data-analysis tasks, various data-normalisation methods and a number of ready-built training settings based on PyTorch Lightning[9]. While being optimised for use with RainBench, PyRain is also compatible with WeatherBench.

[3] https://pangeo.io/index.html (2021)

[4] https://docs.python.org/3/library/mmap.html (2021)

[5] https://portal.hdfgroup.org/display/HDF5/HDF5(2021)

[6] https://zarr.readthedocs.io/en/stable/ (2021)

[7] http://xarray.pydata.org/en/stable/ (2021)

[8] https://www.unidata.ucar.edu/software/netcdf/ (2021)

[9] https://pytorch-lightning.readthedocs.io/en/latest/ (2021)

5. Experimental Results

In this section, we report our models' performance on the end-to-end precipitation forecasting task, which highlights the difficulty in forecasting precipitation values on IMERG. We then present the experiments for the three-step approach, reporting our results for each step.

End-to-end Precipitation Forecasting

Since each data source is available for a different time window, we use the subset of data available from April 2016 to train all models for the benchmark tasks, unless specified otherwise. We use data from 2018 and 2019 as validation and test sets respectively. To make sure no overlap exists between training and evaluation data, the first evaluated date is 6 January 2019 while the last training date is 31 December 2017.

Table 5.1 shows our neural model baseline for the two forecasting tasks. Using the ERA5 precipitation as target, Table 5.2 shows that all model results outperform the two simple baselines. We observe that training from SimSat alone gives the worst results across the data settings. This confirms the difficulty in precipitation forecast from satellite data alone, which does not contain as much information about the atmospheric state as sophisticated reanalysis data such as ERA5. Importantly, the complementary benefits of utilizing data from both sources is already visible despite our simple concatenation setup, as training from both SimSat and ERA5 achieves the best results across all lead times (when holding the number of training instances constant).

Figure 5.1 shows example forecasts from one random input sequence across the different data settings for predicting ERA5 precipitation. We observe that the forecasts can capture the general precipitation distribution across the globe, but there is various degrees of blurriness in the outputs. As we shall discuss later in the paper, considering probabilistic forecasts would be a promising solution to blurriness, which might have arisen as the mean predicted outcome.

We also see the importance in using a large training dataset, since extending the considered training instances to the full ERA5 dataset outperforms the baselines further in the 1-day forecasting regime (shown in the last rows). A key limitation in our current setup is that we have only used the overlapping time frames (from 2016 onwards) of ERA5, IMERG and SimSat as our training data. This observation suggests that there is still significant room for improvement above the presented baselines, especially by developing alternative modelling setups that adequately make use of the full available data from each source.

Table 5.3 shows the forecast results when using the IMERG precipitation as target. The similar performance between the climatology baseline and neural model suggests that this is a considerably more difficult task. In contrast to the previous task, forecasting skill based on ERA5 input is only mildly better than the climatology baselines for 1-day and 3-day forecasts. We believe that the difficulty in this task is closely tied to IMERG featuring more extreme precipitation events (Figure 2.1).

Table 5.1: Precipitation forecasts evaluated with Latitude-weighted RMSE (mm). All rows except the last shows models trained with data from 2016 onwards.

Table 5.2: Predicting ERA Precipitation

Inputs	1-day	3-day	5-day
Persistence	0.6249	0.6460	0.6492
Climatology	0.4798	0.4802	0.4803
SimSat	0.4610	0.4678	0.4691
ERA	0.4562	0.4655	0.4677
SimSat + ERA	**0.4557**	**0.4655**	**0.4675**
ERA (Since 1979)	0.4485	0.4670	0.4699

Table 5.3: Predicting IMERG Precipitation

Inputs	1-day	3-day	5-day
Persistence	1.1321	1.1497	1.1518
Climatology	0.8244	0.8249	0.8246
SimSat	0.8166	0.8201	0.8198
ERA	0.8182	0.8224	0.8215
SimSat + ERA	**0.8134**	**0.8185**	**0.8185**
ERA (Since 2000)	0.8085	0.8194	0.8214

Three-step Approach

State Extraction

To compare the model performance given by the three input data setups, we compute their latitude-weighted RMSE results on two important atmospheric state variables, temperature at 850hPa (t-850) and humidity at 500hPa (q-500).

In Table 5.4, we see that the three setups are able to reconstruct the state variables at $t = 0$, all of which achieve superior performance when compared to persistence and climatology baselines. As expected, the setup which also takes in past ERA5 input gives the best performance out of the three, most noticeably for $t - 850$ with RMSE at 1.81K. We also observe an encouraging result when only Simsat is used as input, which predicts $q - 850$ similarly (4.88×10^{-4}) to the third setup (4.82×10^{-4}).

To understand the state extraction results further, Figure 5.2 visualizes the prediction given by the third setup (Simsat, Ground obs. and past ERA5 as input) of a randomly chosen time step. It is clear that the main variations across the globe in the two variables are visible in both the ground truth and prediction.

State Forecasting

For this task, we split the ERA5 – in $5.625°$ resolution dataset – into training/validation sets using the following time intervals: for training, 2010 to 2016 inclusive; for validation, 2017 and 2018. Five

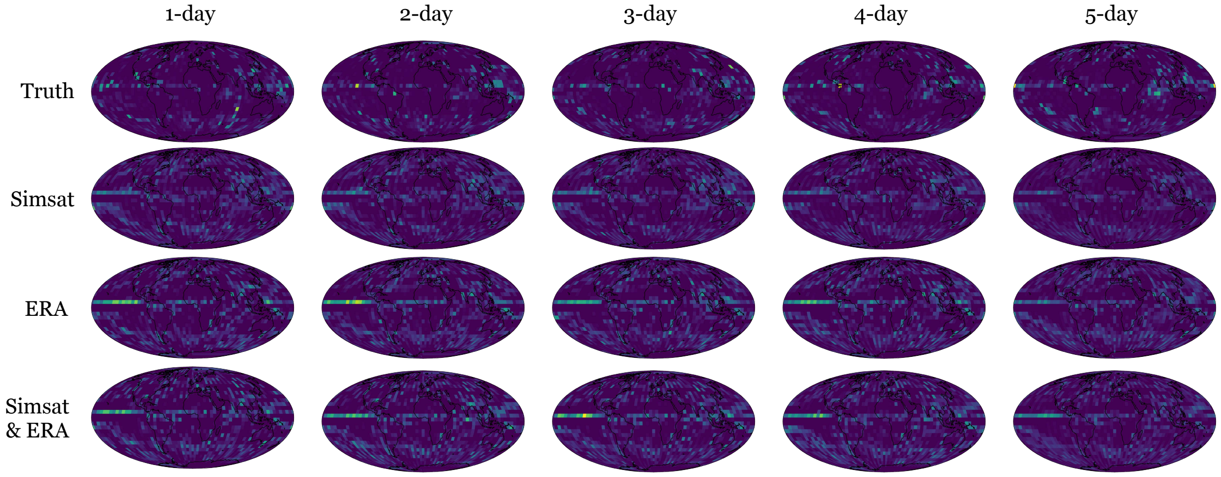


Figure 5.1: ERA5 Precipitation forecasts on one random sample.

Table 5.4: State extraction results at $t = 0$ from data available on the satellite.

	t-850	q-500
Persistence	3.0941	0.0008781
Climatology	13.744	0.001286
Simsat	2.2289	0.0004875
Simsat and Ground obs.	2.4499	0.0005246
Simsat, Ground obs. and past ERA5	**1.8145**	**0.0004821**

frames, separated by three hours each, are used as input and the output is the state at three hours in the future; the prediction is carried out iteratively on a lead time of 72 h.

As encoder, we used a three-layer Convolutional Neural Network (CNN), linked to a LSTM block made by two layers and a decoder with skip connection from the features of the last image in time. The posterior and prior networks share the same feature extractor that is connected to a single LSTM block.

As a qualitative result of the methodology, we can refer to Figure 5.3. Here the variable `t2m` has been depicted. In the lower part, two different future of the variable on Australia are presented. The futures have been sampled using the prior as described in (9).

Figure 5.4 instead shows some quantitative results on the RMSE of two different variables: `t-850` and `q-500`. The first variable is compared to the results obtained with a persistence model, Weatherbench (31) and Integrated Forecast System (IFS) from ECMWF; the second variable, since it is not included in Wetherbench, is compared only against the persistence model. The results show that the methodology has great potential for the task in hand since it performs comparable with the other methodologies in the field.

Precipitation Estimation

For this task, we split the ERA5 and IMERG datasets into training/validation/test sets using the following time intervals: for training, 2010 to 2016 inclusive; for validation, 2017 and 2018; for testing, 2019. We pre-process the target values by accumulating precipitations over three-hours

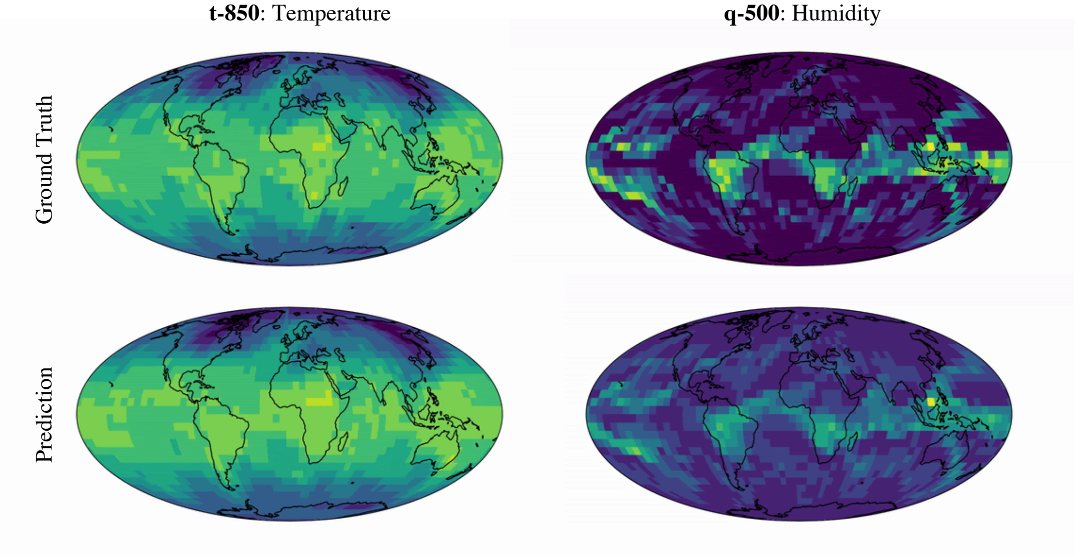


Figure 5.2: State Estimation Result of a randomly chosen time point.

Table 5.5: Precipitation estimation results as measured by class F1 scores (the higher the better). Best results are in bold.

	no-precip	drizzle	light	heavy
Ours - FCNN	**0.9607**	0.2997	**0.2114**	**0.0641**
ERA5	0.9572	**0.4085**	0.1784	0.0632

periods (the frequency at which the Atmospheric State is estimated in the previous step) always in the perspective of fine-tuning the models learned in the three steps together. Figure 5.5(a) provides an example of the obtained ground-truth global precipitations and Figure 5.5(b) reports the predictions of our gridcell-wise Fully Connected Neural Network (FCNN) for the same time step.

Visually, we notice that our FCNN is accurate in distinguishing between rainy and non-rainy cells, but it has a tendency to overestimating precipitation rates. To better analyze the obtained predictions, we study the obtained F1 scores on the test set, in particular compared to ERA5's baseline. We report them in Table 5.5. Performance is assessed by per-class F1 scores in order to equally study precision and recall, and to highlight results over rainy classes. Overall, the FCNN is able to achieve performance similar to ERA's one, and it even out-performs this numerical model on all classes but drizzle. However, both models show limited predictive capabilities on minority classes, for the opposite reasons: ERA5 underestimates precipitations, as shown in Figure 2.1, while the trained FCNN overestimates them. We conjecture that FCNN's performance could be improved by (i) extending the input feature set, (ii) carefully tuning the class weights and proportions, and (iii) extending the training set by adding more years.

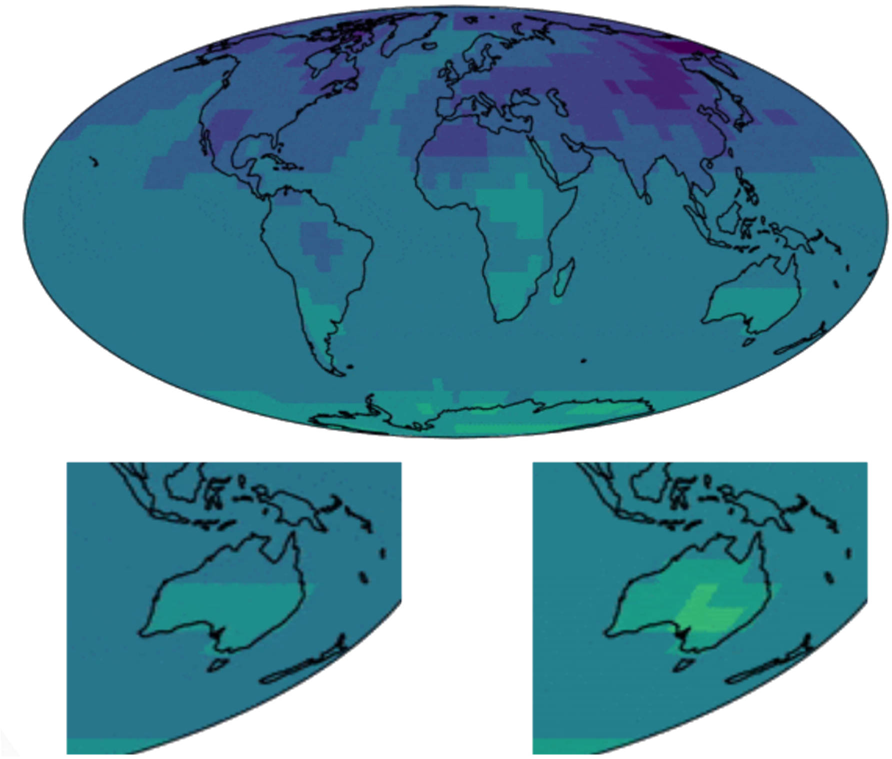


Figure 5.3: Result of the stochastic video prediction algorithm on the `t2m` variable. The reader can note the different outcomes on Australia where two different futures have been sampled.

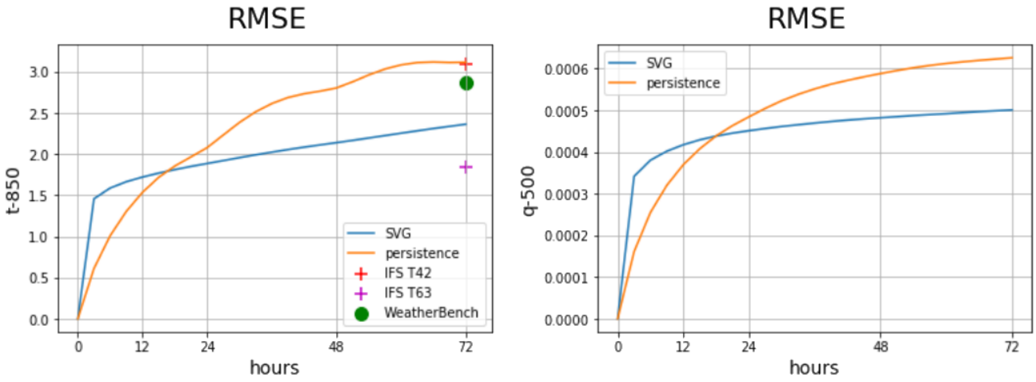
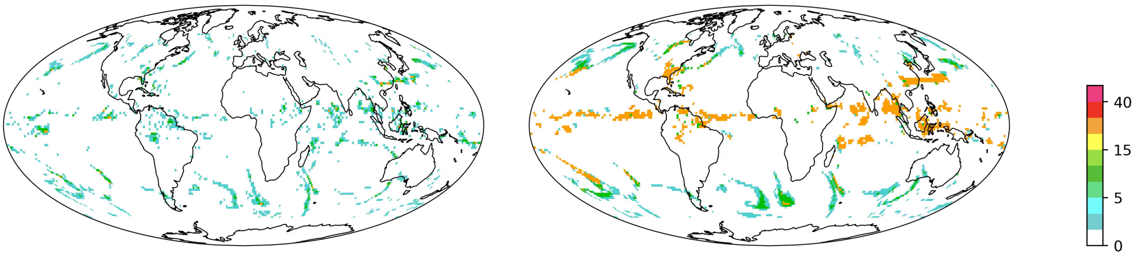


Figure 5.4: Quantitative results for state prediction. On the left the variable `t-850` compared to Weatherbanch (31), persistence and Integrated Forecast System; on the right `q-500` compared to the persistence model.



(a) IMERG ground-truth.

(b) FCNN prediction.

Figure 5.5: Comparison of ground-truth and predicted precipitations, accumulated over 3-hour period in 2019. Globally our model accurately captures the rain no-rain dichotomy. Our best networks struggle with over prediction of heavy rain, particularly in the tropics.

6. Discussion and Conclusions

We outline the key challenges in global precipitation forecasting, our proposed solutions, we also discuss promising research avenues that can build on our work.

Challenges

From our experiments, we identified a number of challenges inherent to data-driven extreme precipitation forecasting.

Class imbalance. Extreme precipitation events, by their nature, rarely occur (see Figure 2.1). In the context of supervised learning, this manifests as a class imbalance problem, in which a model might rarely predict extreme values. Designing an appropriate class sampling strategy (e.g. inverse frequency sampling) can mitigate this imbalance, as shown in Figure 5. Further, we believe that a mixture of pixelwise-weighting and balanced sampling could be a potential solution. This is a much harder problem in the context of end-to-end forecasting, where the model is required to produce global predictions of precipitation. However, this problem is still present in the state forecasting setting, where humidity variables will capture some of the class imbalance and skew that the precipitation field displays.

Probabilistic forecasts. A key driver in improved weather forecasting skill in the last twenty years has been the adoption of probabilistic forecasting. In physics-based models, this is achieved through ensembles of forecasts, where initial conditions and model parameters are varied in an attempt to cover the inherent uncertainties. A challenge with these forecasts is to be reliable, which in this context means that the ensemble spreads (on average) at the same rate as model error grows. A future challenge is to ensure that the stochastic video prediction (or alternate methods) can produce reliable probabilistic forecasts.

Data normalisation. Feature scaling is common data-processing step for training machine learning models and well-understood to be advantageous (6). Our baseline approach currently normalizes input variables using each variable's climatology mean and standard deviation: nevertheless, this approach disregards any local spatial differences, which is important for modelling local weather patterns (43). Previous work has suggested that a patch-wise normalisation procedure may be more appropriate (11). We suggest studying a refinement to LAS, which adjusts the kernel size with latitude such that the spatial normalisation context remains constant (*Latitude-Adjusted LAS*, or LALAS) per-channel image-size normalisation.

Data topology. Lastly, the spherical input and output data topology of global forecasting contexts poses interesting questions to neural network architecture. While a multitude of approaches to handle spherical input topologies have been suggested, see (22) for an overview, it seems yet unclear which approach works best. Our results and dataset might constitute valuable benchmarks for such research.

Future research avenues

Apart from overcoming the challenges outlined above, we have identified a variety of opportunities for further research.

Coupling the three step approach. For the three step approach to weather forecasting we have designed and assessed the performance of each step in isolation. An obvious next step is to couple all three components together to produce an end-to-end forecasting system. We envisage this will require a further step of learning to fine-tune the performance. In particular, the state forecasting component has a tendency to produce blurred images when forecasting for longer times, a commonly observed with state forecasting (31). Future work will balance the Generative Adversarial Network (GAN) loss to produce sharp images throughout the prediction window. This will be key as the precipitation estimate step has been trained with sharp images of the state from the ERA5 dataset as inputs.

Physics-informed multi-task learning. Apart from using reanalysis data for model training, we do not currently exploit the fact that many aspects of weather forecasting are well-understood from a physical perspective. For the end-to-end approach one way of informing model training of physical constraints would be to train precipitation forecasting concurrently with prediction of physical state variables, including temperature and specific humidity, in a multi-task setting, e.g. through using separate decoder heads for different variables (similarly to (8)). This approach promises to combine the advantages of data-driven learning with low-level feature regularisation through a physics-informed inductive bias. Multi-task learning can also be regarded as a form of data augmentation (37), promising to further increase forecasting performance using real or simulated satellite data without requiring access to reanalysis data at inference time. For the three step approach, the physical state is already a target of the prediction system. However there are additional methods being developed to further incorporate physical knowledge. One example is enforcing global or local conservation of physical quantities, if appropriate. Methods for achieving this have already been tested in the context of kernels of weather forecasting models (5) but not in complete data-driven models.

Increasing spatial resolution. Data at higher spatial resolution tends to capture heavy and extreme precipitation events better but poses a number of challenges. Large sample batch sizes may lead to network activation storage that exceeds GPU global memory capacity even for distributed training. Apart from exploring TPU or nvlink-based solutions, another way would be to switch to mixed-precision or half-precision or employ techniques that trade-off memory for compute such as gradient checkpointing (28). PyRain's dataloader efficiently maximises total disk throughput, which may itself become a bottleneck at very high resolutions. Storing all or part of the training data memmaps on one or several high-speed local SSDs may increase disk throughput a few-fold. Apart from memory and disk throughput, there is also a lack of suitably highly resolved historical climate data for pre-training (31). One possible way of overcoming this would be to integrate high-resolution local forecasting model or sensor data into the training process (10), another exciting approach spearheaded in computational fluid dynamics (16) is to employ a multi-fidelity approach, where hierarchical variance-reduction techniques are employed to enable training to be performed at lower-resolution data as often as possible, thus minimising the need for training on high-resolution data.

Conclusion

We have examined the challenges involved in end-to-end forecasting of precipitation, as a first step towards building a digital twin of the Earth. This is a bold initiative, aiming to replace bespoke models which have been developed and improved for over 50 years. There are three key outcomes from this project that we hope will accelerate future research in this exciting area.

Firstly, we have outlined datasets and challenges for end-to-end weather forecasting. This has been captured in **RainBench**, a novel benchmark suite for data-driven extreme precipitation forecasting. This crucially builds on the WeatherBench dataset by providing satellite imagery and improved precipitation output.

Secondly, to accelerate progress in the field of data-driven weather forecasting we release **PyRain**, an associated rapid experimentation framework with a fast dataloader. Both RainBench and PyRain are open source and well-documented.

Thirdly, we consider two methodologies for building medium-range precipitation forecasting systems at global scale, combining Earth observational data with reanalysis data. One is an end-to-end approach which aims to use satellite data to predict precipitation directly. The second divides this task into several tasks, with the long-term goal being the coupling of these systems into an end-to-end forecasting model. Machine learning models are revealed to be difficult to train with a end-to-end approach, partly due to limited data availability. The three-step approach enables easier interpretablility and can overcome the limited data overlap.

State extraction, as the first of the three-step approach, demonstrates a realistic strategy for extracting atmospheric state information from satellite imagery; our results confirm that there is enough information contained in such data to reconstruct important atmospheric state variables for the downstream precipitation forecasting task.

The usage of SVG for state forecasting shows great potential for the task in hand, with comparable results to state-of-the-art methodologies[1] in the field and successfully incorporates uncertainty modelling.

The final step estimates precipitation from same-time atmospheric state variables. Excitingly, this outperformed the ERA5 estimate on three of the precipitation classes, but was still inadequate to accurately predict extreme events.

For each of the three steps, above baseline models were produced, but there is clearly room for improvement in the modelling of each step. In the near future, we plan to fine-tune together the models trained on each of the described subtasks and compare the performance of the final model to the one of the model trained with the end-to-end methodology.

We hope that our benchmarks and framework will lower the barrier of entry for the global research community such that our work contributes to rapid progress in data-driven weather prediction, democratisation of access to adequate weather forecasts and, ultimately, help protect and improve livelihoods in a warming world.

[1] when run at comparable spatial resolution

Bibliography

[1] Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mane, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viegas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng, X. TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems. *arXiv:1603.04467 [cs]* (Mar. 2016).

[2] Aminou, D. Msg's seviri instrument. *ESA Bulletin(0376-4265)*, 111 (2002), 15–17.

[3] Arcomano, T., Szunyogh, I., Pathak, J., Wikner, A., Hunt, B. R., and Ott, E. A machine learning-based global atmospheric forecast model. *Geophysical Research Letters 47*, 9 (2020), e2020GL087776.

[4] Bauer, P., Thorpe, A., and Brunet, G. The quiet revolution of numerical weather prediction. *Nature 525*, 7567 (2015), 47–55.

[5] Beucler, T., Rasp, S., Pritchard, M., and Gentine, P. Achieving conservation of energy in neural network emulators for climate modeling. *arXiv preprint arXiv:1906.06622* (2019).

[6] Bhanja, S., and Das, A. Impact of Data Normalization on Deep Neural Network for Time Series Forecasting. *arXiv:1812.05519 [cs, stat]* (Jan. 2019). arXiv: 1812.05519.

[7] Bihlo, A. A generative adversarial network approach to (ensemble) weather prediction. *arXiv preprint arXiv:2006.07718* (2020).

[8] Caruana, R. Multitask Learning. *Machine Learning 28*, 1 (July 1997), 41–75.

[9] Denton, E., and Fergus, R. Stochastic video generation with a learned prior. *arXiv:1802.07687 [cs, stat]* (Mar 2018). arXiv: 1802.07687.

[10] Franch, G., Maggio, V., Coviello, L., Pendesini, M., Jurman, G., and Furlanello, C. TAASRAD19, a high-resolution weather radar reflectivity dataset for precipitation nowcasting. *Scientific Data 7*, 1 (July 2020), 234. Number: 1 Publisher: Nature Publishing Group.

[11] Grönquist, P., Yao, C., Ben-Nun, T., Dryden, N., Dueben, P., Li, S., and Hoefler, T. Deep Learning for Post-Processing Ensemble Weather Forecasts.

[12] Gubler, S., Sedlmeier, K., Bhend, J., Avalos, G., Coelho, C., Escajadillo, Y., Jacques-Coper, M., Martinez, R., Schwierz, C., de Skansi, M., et al. Assessment of ecmwf seas5 seasonal forecast performance over south america. *Weather and Forecasting 35*, 2 (2020), 561–584.

[13] Gupta, A. K., Yadav, D., Gupta, P., Ranjan, S., Gupta, V., and Badhai, S. Effects of climate change on agriculture. *Food and Agriculture Spectrum Journal 1*, 3 (2020).

[14] Hersbach, H., Bell, B., Berrisford, P., Hirahara, S., Horányi, A., Muñoz-Sabater, J., Nicolas, J., Peubey, C., Radu, R., Schepers, D., et al. The era5 global reanalysis. *Quarterly Journal of the Royal Meteorological Society 146*, 730 (2020), 1999–2049.

[15] Huffman, G., Stocker, E., Bolvin, D., Nelkin, E., and Tan, J. Gpm imerg final precipitation l3 half hourly 0.1 degree x 0.1 degree v06. Tech. rep., 2019. ftp://arthurhou.pps.eosdis.nasa.gov/gpmdata/.

[16] Jabarullah Khan, N. K., and Elsheikh, A. H. A Machine Learning Based Hybrid Multi-Fidelity Multi-Level Monte Carlo Method for Uncertainty Quantification. *Frontiers in Environmental Science 7* (2019). Publisher: Frontiers.

[17] Lalaurette, F. Early detection of abnormal weather conditions using a probabilistic extreme forecast index. *Quarterly Journal of the Royal Meteorological Society 129*, 594 (2003), 3037–3057.

[18] Le Coz, C., and van de Giesen, N. Comparison of rainfall products over sub-saharan africa. *Journal of Hydrometeorology 21*, 4 (2020), 553–596.

[19] Lee, A. X., Zhang, R., Ebert, F., Abbeel, P., Finn, C., and Levine, S. Stochastic adversarial video prediction. *arXiv:1804.01523 [cs]* (Apr 2018). arXiv: 1804.01523.

[20] Li, Y., Guan, K., Schnitkey, G. D., DeLucia, E., and Peng, B. Excessive rainfall leads to maize yield loss of a comparable magnitude to extreme drought in the United States. *Global Change Biology 25*, 7 (2019), 2325–2337.

[21] Lin, T.-Y., Goyal, P., Girshick, R., He, K., and Dollár, P. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision* (2017), pp. 2980–2988.

[22] Llorens Jover, I. Geometric deep learning for medium-range weather prediction, June 2020. Master's Thesis.

[23] Luc, P., Clark, A., Dieleman, S., Casas, D. d. L., Doron, Y., Cassirer, A., and Simonyan, K. Transformation-based adversarial video prediction on large-scale data. *arXiv:2003.04035 [cs]* (Mar 2020). arXiv: 2003.04035.

[24] MetOffice. Fact sheet 3 — Water in the atmosphere. Tech. rep., MetOffice UK, 2012.

[25] Oord, A. v. d., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., Kalchbrenner, N., Senior, A., and Kavukcuoglu, K. Wavenet: A generative model for raw audio. *arXiv:1609.03499 [cs]* (Sep 2016). arXiv: 1609.03499.

[26] Oprea, S., Martinez-Gonzalez, P., Garcia-Garcia, A., Castro-Vargas, J. A., Orts-Escolano, S., Garcia-Rodriguez, J., and Argyros, A. A review on deep learning techniques for video prediction. *arXiv:2004.05214 [cs, eess]* (Apr 2020). arXiv: 2004.05214.

[27] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d. Alché-Buc, E. Fox, and R. Garnett, Eds. Curran Associates, Inc., 2019, pp. 8024–8035.

[28] Pinckaers, H., van Ginneken, B., and Litjens, G. Streaming convolutional neural networks for end-to-end learning with multi-megapixel images. *arXiv:1911.04432 [cs]* (Nov. 2019). arXiv: 1911.04432.

[29] Praveen, B., Talukdar, S., Shahfahad, Mahato, S., Mondal, J., Sharma, P., Islam, A. R. M. T., and Rahman, A. Analyzing trend and forecasting of rainfall changes in India using non-parametrical and machine learning approaches. *Scientific Reports 10*, 1 (June 2020), 10342. Number: 1 Publisher: Nature Publishing Group.

[30] Ramachandran, P., Zoph, B., and Le, Q. V. Searching for activation functions. *arXiv preprint arXiv:1710.05941* (2017).

[31] Rasp, S., Dueben, P. D., Scher, S., Weyn, J. A., Mouatadid, S., and Thuerey, N. WeatherBench: A benchmark dataset for data-driven weather forecasting. *arXiv:2002.00469 [physics, stat]* (June 2020). arXiv: 2002.00469.

[32] Rasp, S., and Thuerey, N. Purely data-driven medium-range weather forecasting achieves comparable skill to physical models at similar resolution.

[33] Rocklin, M. Dask: Parallel computation with blocked algorithms and task scheduling. In *Proceedings of the 14th Python in Science Conference* (2015), K. Huff and J. Bergstra, Eds., pp. 130 – 136.

[34] Saha, M., Mitra, P., and Nanjundiah, R. S. Deep learning for predicting the monsoon over the homogeneous regions of India. *Journal of Earth System Science 126*, 4 (June 2017), 54.

[35] Saha, M., Santara, A., Mitra, P., Chakraborty, A., and Nanjundiah, R. S. Prediction of the Indian summer monsoon using a stacked autoencoder and ensemble regression model. *International Journal of Forecasting* (May 2020).

[36] Saunders, R., Hocking, J., Turner, E., Rayer, P., Rundle, D., Brunel, P., Vidot, J., Roquet, P., Matricardi, M., Geer, A., et al. An update on the rttov fast radiative transfer model (currently at version 12). *Geoscientific Model Development 11*, 7 (2018).

[37] Shorten, C., and Khoshgoftaar, T. M. A survey on Image Data Augmentation for Deep Learning. *Journal of Big Data 6*, 1 (July 2019), 60.

[38] Simmons, A., Burridge, D., Jarraud, M., Girard, C., and Wergen, W. The ecmwf medium-range prediction models development of the numerical formulations and the impact of increased resolution. *Meteorology and atmospheric physics 40*, 1-3 (1989), 28–60.

[39] Sønderby, C. K., Espeholt, L., Heek, J., Dehghani, M., Oliver, A., Salimans, T., Agrawal, S., Hickey, J., and Kalchbrenner, N. Metnet: A neural weather model for precipitation forecasting. *arXiv preprint arXiv:2003.12140* (2020).

[40] SRIVASTAVA, N., HINTON, G., KRIZHEVSKY, A., SUTSKEVER, I., AND SALAKHUTDINOV, R. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research 15*, 1 (2014), 1929–1958.

[41] VOGEL, E., DONAT, M. G., ALEXANDER, L. V., MEINSHAUSEN, M., RAY, D. K., KAROLY, D., MEINSHAUSEN, N., AND FRIELER, K. The effects of climate extremes on global agricultural yields. *Environmental Research Letters 14*, 5 (2019), 054010.

[42] WANG, Y., BILINSKI, P., BREMOND, F., AND DANTCHEVA, A. Imaginator: Conditional spatio-temporal gan for video generation. In *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)* (Mar 2020), IEEE, p. 1149–1158.

[43] 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*, 8 (2019), 2680–2693.

[44] WEYN, J. A., DURRAN, D. R., AND CARUANA, R. Improving data-driven global weather prediction using deep convolutional neural networks on a cubed sphere. *arXiv:2003.11927 [physics, stat]* (Mar. 2020). arXiv: 2003.11927.

[45] 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* (2015), pp. 802–810.

[46] ZHANG, F., SUN, Y. Q., MAGNUSSON, L., BUIZZA, R., LIN, S.-J., CHEN, J.-H., AND EMANUEL, K. What Is the Predictability Limit of Midlatitude Weather? *Journal of the Atmospheric Sciences 76*, 4 (Apr. 2019), 1077–1091. Publisher: American Meteorological Society.

[47] ZSÓTÉR, E. Recent developments in extreme weather forecasting, 2006. Issue: 107 Pages: 8-17 Publisher: ECMWF.