UNIVERSITA' DEL SALENTO

DIPARTIMENTO DI SCIENZE E TECNOLOGIE BIOLOGICHE ED AMBIENTALI

DOTTORATO di RICERCA in

Scienze e Tecnologie Biologiche ed Ambientali

XXXV ciclo - Anno accademico di inizio: 2019/2020

SSD: ING-INF/05

MACHINE LEARNING FOR PREDICTING EXTREME WEATHER EVENTS IN A CHANGING CLIMATE

Tutor:
Chiar.mo Prof. ***Giovanni Aloisio***
Chiar.mo Prof. ***Luigi De Bellis***

Dottorando:
Dott. Ing. Marco Chiarelli

Il Coordinatore del Dottorato
Chiar.mo Prof. S. Marsigliante

Machine Learning for predicting Extreme Weather Events in a Changing Climate

ABSTRACT

Extreme Weather Events (EWEs) are defined as extremely rare events that bring any significant change in the atmospheric / oceanographic environment. They manifest as heavy rainfall, tropical cyclones, wildfires, marine heat weaves, thunderstorms, hail, etc. Typically, they have short duration and manifest at local scales.

Despite an internal variability of the Earth system, increasing carbon dioxide emissions (CO2) driven by human activities are strongly exacerbating the frequency, the duration and the severity of these events.

It is straightforward to understand the importance of correctly detecting and predicting them in a sufficient advance. However, due to their rarity, complexity and even difficulty to gather data in their exact location, the process is not trivial, at times very hard.

Traditional Climate Science methods for this task are based on the execution of Earth System Models (ESMs), in which they are embedded and resolved physical equations that describe the spatiotemporal evolution of several climatic variables.

However, these models are generally too complex, both to write and to maintain, and some physical processes may not be sufficiently known or not known all to be implemented. Moreover, they are typically computationally intensive and require a great amount of execution time to produce results.

Machine Learning techniques have already been demonstrated to be suited for applications in the Climate Science domain. They may overcome the main issues of physics-based frameworks, improving them from a computational / resources standpoint. In addition, being these methods fully data-driven, they don’t need to know anything about the physics behind the process to be described. In fact, they are able to reconstruct the input/output mapping based only or partially on data.

This dissertation confirms the convenience of using Machine Learning in Climate Science applications, by showing some use-cases mainly related to EWEs detections and prediction or which are preparatory for this task.

CONTENTS

To my family

ACKNOWLEDGEMENTS

I would like to express my gratitude to Professor Giovanni Aloisio and Professor Luigi De Bellis, for their motivation, support and for taking me and my research activities towards the right pathway and under constant improvement.

Moreover, I would like to thank my colleagues from the Advanced Scientific Computing (ASC) division of the CMCC Foundation: Gabriele Accarino, Francesco Immorlano, Valeria Aloisi, Andrea Gatto and Davide Donno for their precious support, knowledge and motivating me through this academic path. But first of all, I would like to thank them mainly for their genuine sense of friendship.
My deepest thanks go also to Antonio Aloisio for his helpful support in reviewing the scientific articles written during this PhD cycle, as well as the present thesis.

To my family. To my brother Michele, my parents Luciano and Monia, my aunt Caterina, Lidia, Sandra, my uncle Armando, Romeo, Giuseppe, my grandmothers Marcella and Maria. To my grandfathers Gino and Ninì, who look at me from the sky, and I believe they have supervised, encouraged and protected me in this path.
To my cousins Vincenzo, Federica, Valeria, Nicoletta and Giovanni.

To my best friends, Luka, Antonio, Francesco, Pierfrancesco, Alberto, Andrea. To all the friendships born in and after my university career: Paolo, Emanuele, Matteo, Dino.
I would like to express my grateful thanks to this people , who not only have motivated and supported me, but also have given me a lot of restoration and funny moments in order to break from the boring everyday routine.

0
Executive Summary

Nowadays Climate Change effects are evident throughout the world. The main effect is global warming, which is caused by a multitude of factors. Despite an internal variability of the Earth system, the increasing Greenhouse Gases (GHG) emissions (CO2, Nox, CH4, etc.), which led to *greenhouse effect*, are mainly caused by anthropogenic activities which intensified after the Industrial Era, and this has been demonstrated to be the primary factor of Climate Change.

Climate Change is responsible also for the intensification in terms of frequency, duration and severity of the Extreme Weather Events (EWEs). Between 2000 and 2020, these catastrophic disasters occurred 13345 times, and they were responsible for the death of 1.5 millions of people (Xu, 2020). It is trivial to understand that, detecting and predicting these catastrophic events in sufficient advance, may allow in the short term prompt policy makers action, while efficient adaptation and mitigation strategies may be implemented in the long term in order to save lives. In order to correctly detect and predict these catastrophic events, a great understanding of the Earth's climate is required.

So far, traditional models, named General Circulation Models (GCMs) have represented the state-of-the-art choice in modeling the evolution of several climatic variables as a response to different forcing. GCMs later evolved into Earth System Models (ESMs) including other Earth's climate components, such as biogeochemistry, aerosol, land surface, land/sea/ice coupling, etc.

However, these models are too complex, both to write and to maintain, because the underlying physical processes are highly non-linear, difficult to understand and model, or not known at all. These processes are represented through physical equations and embedded into the model under functions/routines. Processes not known at all, but whose influences are important for the evolution of the entire system are called parameterizations.

Moreover, these models are generally computationally intensive, they require massive supercomputing resources and typically they do not provide results in a reasonable amount of execution time, spending huge amounts of energy. Last but not least, they are explicitly based on the knowledge of the physical process to model.

Machine Learning techniques have already demonstrated to be suited to overcome the latter methods' limitations. They are more cost-effective, they do require less computational resources, less memory, less storage and they are entirely data-driven. They can learn the mapping between input/output data through a process called

Training, and if this is done correctly, then the ML algorithm is able to generalize to unseen data, mimicking the mapping / phenomena previously learned.

Finally, these algorithms typically have non-linearity entry points in their structure, giving the possibility to effectively model some inherent nonlinearity of the physical phenomena to be represented.

The present thesis work further demonstrates and remarks the usefulness and the convenience of using these algorithms in the Climate Science applications, especially in detecting and predicting some EWEs such as wildfires and sea level rise. Moreover, the so-called Downscaling procedure through ML, which is a preparatory technique for many climate science related use cases, including the aforementioned ones, has also been investigated during the PhD cycle.

This dissertation consists in five chapters and it is structured as follows: the first one presents an overview of climate change, on all the climate science investigation and methods which studies and models it, as well as its connection with the insurgence and the exacerbation of EWEs. The rest of the chapter gives a brief systematization of the current state-of-the-art Machine Learning algorithm related to climate science.

The following two chapters are focused on the EWEs. In particular, it is remarked on the importance of the task of detecting and predicting them, possibly in sufficient advance in order to adopt prompt actions and mitigation strategies. Classical solutions to EWEs prediction are discussed, as well as the various possibilities, both already developed and under developing, to exploit ML algorithms. Two of the most catastrophic EWEs, such as wildfires and sea level rise, are particularly discussed and stressed in these chapters, and most of the research of this work is focused on those.

Then, the other two chapters list my two research papers published under my PhD cycle. In particular, the first one co-authored with Gabriele Accarino, Sandro Fiore, Ivan Federico, Salvatore Causio, Giovanni Coppini and Giovanni Aloisio presents the application of the Long-Short Term Memory (LSTM) neural network to the problem of short-term sea level forecasting in the Southern Adriatic Northern Ionian (SANI) domain in the Mediterranean sea. In this work, a multi-model architecture based on LSTM networks has been trained to predict mean sea levels three days ahead, for different coastal locations. Predictions were compared with the observation data collected through the tide-gauge devices as well as with the forecasts produced by the Southern Adriatic Northern Ionian Forecasting System (SANIFS) developed at the Euro-Mediterranean Center on Climate Change (CMCC).

The second paper, co-authored with Gabriele Accarino, Francesco Immorlano, Valeria Aloisi and Andrea Gatto discusses a new architecture developed for statistical downscaling (SD), named MSG-GAN-SD, that allows interpretability and good stability during training, due to multi-scale gradient information. The proposed architecture, based on a Generative Adversarial Network (GAN), was applied to downscale ERA-Interim 2-m temperature fields, from 83.25 to 13.87 km resolution, covering the EURO-CORDEX domain within the 1979–2018 period. The training process involves seasonal and monthly dataset arrangements, in addition to different training strategies, leading to several models. The selected models were then tested on the 2015–2018 period using several metrics to identify the best training strategy and dataset arrangement, which finally produced several evaluation maps.

The final chapter discusses an enhancement of the Statistical Downscaling of 2-m temperature fields in the EURO-CORDEX domain using Residual Convolutional Neural Networks (CNN). This use case has been investigated in collaboration with Francesco Immorlano and Prof. Pierre Gentine of the *Department of Earth and Environmental Engineering* at the Columbia University in the City of New York, in which I have been hosted during my abroad research period from Mar 1st to May 1st of 2022.

1

Climate Change and Extreme Weather Events connection, Machine Learning convenience and opportunities of modeling

Climate Change is one of the main challenges that the Earth is experiencing. By definition, Climate change refers to long-term shifts in a given climatic variable (e.g. temperature or precipitation). Despite these shifts may be natural, depending for example on small variations of solar cycle or Earth's orbit, since half of the 1800s human activities, mainly based on the use of Fossil Fuels, were the primary driver of climate change.

Normally, nearly a third of the energy received from the sun is scattered back to space, due to the presence on our planet of reflecting surfaces like oceans, snow, ice, depending on their reflective power (albedo), whereas the remaining part is absorbed by the Earth.

However, among these received energy, the infrared radiation (long wavelength) is mainly absorbed by greenhouse gases (GHGs), and their continuous releasing in the atmosphere by anthropogenic sources (Solomon et al., 2007) leads to an increasing temperature all over the planet (Lacis et al., 2010). This effect is known as the *greenhouse effect* (Romm, 2016). Significant variations in temperature extremes are directly caused by the increasing GHGs emissions (Heim Jr, 2015).

Climate change effects span from sea ice losses (Wunderling et al., 2020), quick sea level rise (Nerem et al., 2018; Tebaldi et al., 2021; Priestley et al., 2021), prolonged heat waves (Marx et al., 2021), rainfalls intensification (Fowler et Al, 2021), changes in hydrological and water cycle (Trenberth, Kevin E, 2022) and oceans' currents (IPCC, 2019), oceans acidification (Guldberg et al., 2018; Doney et al., 2020), increasing drought periods (Douville et al., 2021). The insurgence of EWEs are exacerbated: more wild tropical cyclones and wildfire occur: the first because intensification of the water cycle (Hirabayashi et al., 2013), while the latter is directly because of global warming, thus promoting plants evaporation and dryness, and is also due to the prolonged temporally extension of the fire season, causing the fires to be more even more likely to occur (Dunne, Daisy, 2020).

Climate change has a direct effect also on people's health (Rocque et al., 2021), society (van der Geest et al., 2018), food security (Wheeler and von Braun, 2013) and economy, and contributes to global economic inequality (Diffenbaugh and Burke, 2019).

Forest trees play a particular role in climate change. Forests are natural carbon storage. *Carbon Sequestration* is the process of storing carbon in a carbon pool (IPCC, 2021). Vegetables absorb carbon dioxide from the atmosphere during their growing and it is converted into carbon and stored into the plant's branches, leaves, trunks, roots and in the soil. Unlike the carbon stored in deeper layers of the soil, which typically undergoes a slower cycle, the carbon stored in the forests is volatile and part of a quicker carbon cycle. Therefore, when a fire occurs, this stored carbon is immediately released into the atmosphere in the form of carbon dioxide, thus worsening global warming. It is of crucial importance of keeping the forest in a healthy state, promoting sustainable forest practices and preventing, detecting and mitigating harmful events such as wildfires.

In Climate Science, an *event* is defined as an occurrence of a significant change in a climate or a weather variable (KanimozhiSelvi and Sowmiya, 2019) exceeding a threshold close to the upper or lower ends of the usual range over a predefined duration

(Seneviratne et al., 2012). When an event is unlikely to happen, and is particularly severe in terms of intensity and duration, it is classified as an Extreme Weather Event (EWE). These events are responsible for catastrophic weather phenomena — such as short-term heavy rainfall, thunderstorms, gales, tornadoes, hail, etc. — that are usually sudden, have a short duration and manifest at local scales (Fang et al., 2021). Between 2000 and 2020, the cumulative number of disasters worldwide reached 13,345, and more than 1.5 million people died (Xu, 2020): this is why a lot of research has been devoted to EWEs in the scientific community.

Many studies showed that frequency, intensity, spatial extent, duration, and timing of heavy-to-extreme events have increased across the world, which could be because of global warming (Norouzi et al., 2019; Stott, 2016; Boo et al., 2006).

Detecting and predicting EWEs is challenging due to the rare occurrence of these events and consequently the lack of related historical data. Moreover, the gathering of data when the event manifest is not a straightforward process, due to the intrinsic difficulty of positioning and using acquisition systems (e.g., probes, sensors, etc.) in the exact location of the event (e.g Tropical Cyclones originating in open ocean water).

Besides the importance of an early prediction of these catastrophic events, it appears evident that, reducing or mitigating the primary causes of the climate change, that are the GHGs emissions induced by anthropogenic activities, remains the main option of reducing the frequency and the intensification of the EWEs.

1.1 General Circulation Models

General Circulation Models (GCMs) or Global Climate Model is a climate model employed to simulate the dynamic of the Earth’s atmosphere (Atmosphere GCM, AGCM) or the ocean (Ocean GCM, OGCM) (Goosse et al., 2010). It is based on complex mathematical equations to be resolved for various applications, from weather forecasting to understanding and also forecasting climate change.

Sea-ice or land-surface components might also be included as submodels, and when coupled with the former described GCMs, they form full climate models named Earth System Models (ESMs) (Sun and Hansen, 2003).

Several multi-dimensional variables, parameters and constraints are put into discrete equations, and the latters are integrated through time. When a process occurs only on finer scales, are too complex or not known at all to be directly described through mathematical equations, for example the convection, it is accounted for with a parameterization.

In order to discretize the intrinsic continuous equations, a spatial meshgrid over the atmosphere / ocean is imposed. Grids can be of various types, latitude / longitude grid, non-rectangular grids (e.g. icosahedral), variable resolution grid etc. The most simple and widely used grid is the lat / lon one, where the Earth is subdivided into cells of the same spatial extent, called horizontal resolution. Models' resolutions span from more than 500km to fractions of a degree (< 100km). Additionally, grid cells have several vertical layers aimed at simulating different climatic variables across the height and depth of the atmosphere and oceans. In these models there is also the concept of temporal resolution, which may be hourly, daily, monthly, annual, decadal, etc. Clearly, a certain horizontal or temporal resolution will be suited or not depending on the purpose of a simulation. For example, in order to study and comprehend the large-scale trend of a climatic variable and to assess its impact on climate and society, coarser resolutions will be required. However, when a risk / impact assessment process has to be done by policymakers, more high-resolution information will be needed.

1.2 Downscaling

Besides the intrinsic improvement that can be done on classical GCMs in order to enhance their horizontal resolution, a more rapid and powerful technique may be adopted: the Downscaling.

Downscaling is a procedure that allows making predictions at local scales, starting from climatic field information available at large scale. The climatic fields at large scales are resolved by the Global Climate Models (GCMs), whereas the fields at local scales are resolved by Regional Climate Models (RCMs). The mapping between these information is crucial to understand the local climate dynamics, which are often critical for assessing the impacts of a changing climate on society (Baño-Medina et al., 2018; Vandal et al., 2019).

Downscaling can be carried out through two classes of techniques: dynamical and statistical. The former is performed through a physics-based model, namely a regional climate model (RCM), that involves a set of physical equations for modeling different components of the climatic system and their interactions. The physical laws are numerically solved in order to simulate the outcomes for a series of different climatic fields at a finer resolution (Accarino et al. 2021). Unlike traditional dynamical approaches, statistical ones involve the learning of empirical statistical relationships between coarse GCM outputs and High-Resolution (HR) products, including in-situ observations (Baño-Medina et al., 2018; Vandal et al., 2019; Baño-Medina et al., 2020; Sachindra et al., 2018).

The link between Downscaling and Climate Change and, in particular with EWEs now appears evident. First of all, Downscaling allows resolving phenomenons that manifest only at local scales, such as local wildfires, rainfalls, etc., making prompt action and mitigation strategies more likely to be effective. Downscaling may also be useful in order to enhance the spatial resolution of some EWEs predictors, which may be poorly significant at a coarser resolution. This is the case, for example, of the Soil Moisture for wildfires predictions (Alemohammad et al., 2018).

Despite their strong usefulness, climate models are computationally intensive and require a vast amount of supercomputing resources to be executed. Most often, simulation times are very long and sufficiently compatible with the stringent use-case requirements. Last but not least, they consume a not negligible amount of energy.

In the last decade, there were vast efforts to improve the computational performance of these algorithms. The models underwent various optimizations, both at Sequential level with Cache Blocking, Loop Fusion etc., and both at parallel level: there have been implemented multi-node parallelization techniques with Message Passing Interface (MPI), hybridization techniques with multi-threading OpenMP API, GPU acceleration, etc. Effectively, performance has been improved at expense of the use of a massive code refactoring and more computing nodes. However, state-of-the-art climate models have become so optimized that there is no more room for improvement.

1.3 Machine Learning in Climate Research and Extreme Weather Event applications: A brief review

Machine Learning rapidly evolved and consolidated during the last decade. The great availability of powerful architectures, the ease in obtaining these computing resources through access to remote clusters, cloud, etc., as well as the developing of fast and highly-optimized software frameworks and libraries such as PyTorch, Tensorflow, Keras (Paszke et al., 2019; Abadi et al., 2016; Keras, 2015) were the main driver of this change.

Several Machine Learning techniques have been employed in several fields, such as Medicine, Industrial Engineering, Computer Vision, Speech and Text Recognition, Military, Economy, Society. Recently, there was a huge adoption of these algorithms also in Climate Science both to ameliorate existing physics-based solutions, to develop fully data-driven methods and also to mix them in a hybrid fashion.

The idea behind ML algorithms is to learn the function being approximated basing only on data. These data could come from climate model simulations, observation data gathered from sensors, satellite imagery, ground stations, etc. In supervised learning for example, input/output pairs of a given phenomena are fed to the algorithm during the training phase, in order to adjust and tune the weights of the various neurons of the network in order to give the best estimate of the underlying mapping. Then, in the test phase, there is a proof of generalization capabilities of the algorithm on unseen data. If the algorithm behaves sufficiently well also on test data, i.e. does not suffer from overfitting, then it is ready to use. There exist also other algorithms which belong to the unsupervised or semi-supervised learning paradigm, where an explicit binding between input/output pairs is not strictly required.

ML algorithms, in contrast to the so far described physics-based model (e.g. GCM), are fast to execute and produce results, at least in the test / operational phase. The training phase is typically slow. In fact, the algorithm has to see all the training dataset during one epoch, which is the atomic step of the training and is executed an arbitrary number of times. However, with the advent of GPU architectures and also of dedicated Neural Cores, i.e. Google TPU (Jouppi et al., 2017), their massive parallelism with respect to traditional CPU architectures, might be exploited in order to efficiently distribute the workload and dramatically reduce training time.

This one, along with the development of fast and easy-to-use powerful ML software libraries, are the main driver of the recent exponentially fast diffusion of ML in scientific applications, especially in those like Climate Science where an incredible amount of big data has to be dealt with. In Climate Science, ML might be used not only to overcome the more stringent computational requirements of physics-based models, but also to provide a "model-agnostic" black-box that approximate a certain phenomena, based only on data provided during the training phase. Though this can better describe complex relationships in data which could have not been captured in classical models, due to limited or lack of knowledge, or even reduce their biases, this black-box point of view is often criticized and sometimes labeled as dangerous (Rudin, 2019). ML solutions should be always diagnosed for correctness, robustness, and their outcomes should satisfy physical consistency. However, even when the algorithm is optimally trained and behaves well on a certain out-of-sample data (i.e. not seen during the training phase) there is no guarantee that it will behave well also on other unseen data. In fact, there is a strong assumption which states that the algorithm might generalize on unseen data, only if these data exhibit a probability distribution sufficiently close to those of training data previously trained upon. If ground truth predictions' values of new data are excessively far from training data values, or if the predictors' values have been provided with an insufficient quantity of variance in the training phase to extrapolate and well generalize to that former outcome, then the result might be incorrect and even physically inconsistent. Typically, limited training dataset or the lack of important predictors to describe the phenomena might be the cause of this behavior.

In order to deal with these problems yet still benefiting from ML advantages, there has been several attempts in Climate Science to embed physical knowledge into ML algorithms (Reichstein et al., 2019). These diverse techniques have been systematized in (Willard et al., 2022) into four categories: Physics-Guided Learning, Physics-Guided Design of Architecture, Residual Modeling, and Hybrid Physics-ML Modeling. By embedding physical knowledge into ML algorithms, the latters' inconsistency issues, deriving from the use of limited or poorly coherent datasets, the lack of some important modeling components, etc. could be solved or at least mitigated.

Physics-informed Neural Network (Raissi et al., 2019) are one of the most significant examples among these techniques. In these networks, several physical constraints might be added in order to enforce their solutions to satisfy them, thus leading to physical consistency and more truthful and precise outputs.

Another issue which affects ML algorithms is their lack of transparency during the training, which leads to a poor comprehension of the underlying learning. Despite the mathematics behind ML being generally well understood, the problem is to understand why the algorithm precisely makes one choice with respect to another (i.e. in classification problems), or why it predicts one number (i.e. in regression problems). Though the results could be correct too, the high-level misunderstanding of the process that governed that behavior generates distrust and suspicions in domain experts.

To this extent, a novel field called Explainable Artificial Intelligence (Explainable AI or XAI) has been developed and studied in order to give an explanation of why a ML algorithm obtained a certain solution. This is particularly important to critical applications for trust and transparency (Gohel et al., 2021).

In the context of Climate Science, since several attempts to use ML have been investigated in the beginning, nowadays there have been developed already working ML based systems.

ML has been used for time-series forecasting, for example for temperature and precipitation (Papacharalampous et al., 2018; Cifuentes et al., 2020; Wolff et al., 2020; KHOSRAVI et al., 2022, Qian et al., 2022; Alizamir et al., 2020; He et al., 2022; Basha et al., 2020; Barrera-Animas et al., 2022).

Moreover, given the computational expensiveness of traditional physics-based climate models, often an ANN, called surrogate model, is trained on some of their component in order to approximate them in a cost-effective fashion (Reichstein, 2019; Lu and Ricciuto, 2019; Mansfield et al., 2020; Zahura et al., 2020). The most resonating example might be the use of ML to accelerate parameterizations models, which are extremely computationally expensive (Chantry et al., 2021), and the simulation of cloud processes by means of ANNs emulators (Gettelman et al., 2021; Gentine et al., 2018)

Climate satellite data, imagery maps and remote sensing data have also been widely exploited in order to train ML algorithms for several use-cases. From wheat yields predictions (Cai et al., 2019; Kamir et al., 2020; Gómez et al., 2021; Zhou et al., 2022; Bouras et al., 2021; Jhajharia et al., 2022) to streamflow estimation (Dayal et al., 2021), solar radiation estimation (Cornejo-Bueno et al., 2019) and pollutants prediction (Schneider et al., 2020).

Complex teleconnections phenomena may also be analyzed and predicted through ML, such as ENSO events (El Niño–Southern Oscillation) (Hernández et al., 2020; Pal et al., 2020; Lima et al,, 2015; Dijkstra et al., 2019; Nooteboom et al., 2020; Maher et al., 2022; Chen et al., 2021; Gupta et al., 2022; He et al., 2019; Mu et al., 2019).

ML can also be used for reconstructing missing information, through ML-based gap-filling techniques (Zhu et al., 2022; Sarafanov et al., 2020; Arriagada et al., 2021; Menzer et al., 2015; Sun et al., 2021; Tong et al., 2021; Bellido-Jiménez et al., 2021; Kim, Y, Johnson, MS, Knox, SH, et al. 2020; Khan et al., 2021; Tzanis et al., 2021; Chen et al., 2023).

Another field of interest in using ML for Climate Science is the adoption of such data-driven algorithms to develop early-warning systems for EWEs (McGovern et al., 2017). The challenge in predicting these events by traditional physics-based systems, such as GCMs, is that the latters exhibit a chaotic behavior in the short-term trend, whilst stabilizing at predicting long-term changes. Thus, their usefulness to predict EWEs remains limited. ML can overcome these difficulties by exploiting climate models' long-term information, trend changes in order to correctly predict the temporal instance when an EWE is about to occur. ML techniques have been used for predicting thunderstorm (Bala et al., 2017; Chakrabarty et al., 2015; Kamangir et al., 2020), tornadoes (Lagerquist et al., 2020; Trafalis et al., 2007; Aleskerov et al., 2020; Coffer et al., 2021; Adrianto et al., 2009; McGuire and Moore, 2022; Adrianto et al., 2010), hails (Burke et al., 2020; Czernecki et al., 2019; Gagne et al., 2017; Gagne II, David John, et al. 2015; Pullman et al., 2019; Zhang et al., 2021; Pulukool et al., 2020; Yao et al., 2020), tropical cyclones and fires. The ML state of the art tractation of the latters phenomena are further presented in the subsequent two chapters. ML algorithms have been developed also in order to classify, detect or segment an EWE rather than predicting its occurrence (Liu et al., 2016; Racah et al., 2017; Kurth et al., 2018).

The main issue in developing such techniques for EWEs prediction is their rarity. In fact, on one hand an imbalanced dataset cannot be used as is for training a ML algorithm and requires to be processed with various techniques, such as random undersampling, etc. On the other hand, the real world distribution of these events should be respected, and if a perfectly balanced EWEs dataset has been used for the training phase, it is unlikely to capture the effective, real-world imbalanced distribution of that event. Generally, a trade-off for this degree of dataset balance should be chosen, or additional validation techniques have to be implemented (Apostolakis et al., 2022).

Another trouble in climate models is represented by their uncertainty. In fact, different models exhibit slightly different long-term trends, given the same forcings. This is due to their different modeling strategies, which are obviously inherent sources of bias.

Traditionally, this issue is addressed by means of multi-model ensembles, which consist in running different climate models governed by the same forcings, and considering at the end several probabilistic indicators like mean, variance, etc. in order to decrease bias and other uncertainty sources (Tebaldi and Knutti, 2007). Machine Learning techniques have been demonstrated to be suited also for this task. For example, in (Monteleoni et al., 2011), 20 IPCC global climate models have been tested against over 100 years of historical temperature data in order to select the best model closer to the observations. In (Anderson and Lucas, 2018), a statistical model based on the random forests ML technique has been trained on a multiresolution perturbed parameter ensemble of a GCM, in order to make high-resolution model predictions of the global mean top-of-atmosphere energy flux and precipitation.

2

MSG-GAN-SD: A Multi-Scale Gradients GAN for Statistical Downscaling of 2-Meter Temperature over the EURO-CORDEX Domain

One of the most important open challenges in climate science is downscaling. It is a procedure that allows making predictions at local scales, starting from climatic field information available at large scale. Recent advances in deep learning provide new insights and modeling solutions to tackle downscaling-related tasks by automatically learning the coarse-to-fine grained resolution mapping. In particular, deep learning models designed for super-resolution problems in computer vision can be exploited because of the similarity between images and climatic fields maps. For this reason, a new architecture tailored for statistical downscaling (SD), named MSG-GAN-SD, has been developed, allowing interpretability and good stability during training, due to multi-scale gradient information. The proposed architecture, based on a Generative Adversarial Network (GAN), was applied to downscale ERA-Interim 2-m temperature fields, from 83.25 to 13.87 km resolution, covering the EURO-CORDEX domain within the 1979-2018 period. The training process involves seasonal and monthly dataset arrangements, in addition to different training strategies, leading to several models. Furthermore, a model selection framework is introduced in order to mathematically select the best models during the training. The selected models were then tested on the 2015-2018 period using several metrics to identify the best training strategy and dataset arrangement, which finally produced several evaluation maps. This work is the first attempt to use the MSG-GAN architecture for statistical downscaling. The achieved results demonstrate that the models trained on seasonal datasets performed better than those trained on monthly datasets. This study presents an accurate and cost-effective solution that is able to perform downscaling of 2 m temperature climatic maps.

This chapter has been published as:

Accarino, G., Chiarelli, M., Immorlano, F., Aloisi, V., Gatto, A. and Aloisio, G., "MSG-GAN-SD: A Multi-Scale Gradients GAN for Statistical Downscaling of 2-Meter Temperature over the EURO-CORDEX Domain", *AI*, vol. 2, no. 4, pp. 600-620, 2021. doi: 10.3390/ai2040036

2.1 Introduction

Downscaling is a procedure that allows making predictions at local scales, starting from climatic field information available at large scale. In climate science, a well-established representation of climatic fields involves the use of multi-dimensional structures (i.e., 2D, 3D or even 4D if the time dimension is considered) that can be treated as single images or image sequences (Vandal et al., 2017; Vandal et al., 2018a; Baño-Medina et al., 2018; Baño-Medina et al., 2020; Rocha Rodrigues et al., 2018). Therefore, the grid points in a climatic map can be represented as image pixels. This is the reason why the downscaling problem is closely related to the super-resolution (SR) problem in computer vision and image processing, corresponding to the enhancement of the spatial resolution of an image beyond its original resolution (Wood et al., 2004; Fowler et al., 2007; Maraun and Widmann, 2018; Yang et al., 2019; Leinonen et al., 2021; Wang et al., 2021). Thus, a prior extensive knowledge about the structure of the images at the target finer resolution is critical for SR models, and enables them to generate upsampled images that are coherent with the input data (Leinonen et al., 2021). The task of producing a super-resolution image (from now on called an HR image), starting from its lower resolution counterpart (from now on called an LR image), is recognized in the literature as single-image super resolution (SISR) (Leinonen et al., 2021). This problem is generally ill-posed because it does not have a unique solution, as many different HR images can be generated starting from the same LR image (Wang et al., 2021). In fact, the upsampling procedure involves the synthesis of artificial information which serves to scale-up the image towards the target resolution. (In this scenario, the terminology may lead to confusion as the terms “upsampling” and “upscaling” are both used in computer vision to indicate the process of increasing the number of pixels in an image, whereas the term “downsampling” indicates the inverse process. In climate science, the term “downscaling” refers to the generation of maps with a finer resolution (i.e., with a higher number of grid points), starting from their coarser resolution. This is due to the fact that the finer resolution maps, generated through a downscaling process, will consist of grid points with a smaller horizontal resolution.). The need for downscaling is generally motivated by the typically unsatisfactory coarse resolution of global climate models (GCMs). Although these models are used for a better understanding of climate change at global and up to continental scales, and provide information for a large number of climatic fields, they are not able to resolve processes that manifest at regional and local scales, whose dynamics are often critical for assessing the impacts of a changing climate on society (Baño-Medina et al., 2018; Vandal et al., 2018b). Downscaling can be carried out through two classes of techniques: dynamical and statistical. The former is performed through a physics-based model, namely a regional climate model (RCM), that involves a set of physical equations for modeling different components of the climatic system and their interactions. The physical laws are numerically solved in order to simulate the outcomes for a series of different climatic fields at a finer resolution. The statistical techniques, which are typically more accurate than the model’s raw output (Pan et al., 2019), involve the learning of empirical statistical relationships between coarse GCM outputs and HR products [(Baño-Medina et al., 2018; Baño-Medina et al., 2020; Vandal et al., 2018b; Sachindra et al., 2018). Inspired by the work presented in (Karnewar and Wang, 2020), a multi-scale deep architecture was developed for downscaling the 2-m temperature (T2M), from 0.75° × 0.75° up to 0.125° × 0.125° of spatial resolution, in the past 40 years (1979-2018), over the European domain by exploiting Era-Interim analysis data (ERA-Interim, 2021a). The aim was to provide a novel deep learning-based solution to the downscaling task, as an alternative to traditional dynamical and statistical techniques. This was also motivated by the cost-effectiveness of deep learning models that, once trained, provide outcomes through a limited amount of computing resources and execution time. The proposed architecture, named Multi-Scale Gradients GAN for Statistical Downscaling (from now on called MSG-GAN-SD), features a Generative Adversarial Neural Network (GAN) composed of Generator (from now on called G) and Discriminator (from now on called D) networks, that are both convolutional and set to work at multiple scales. Each of these networks is made up of several blocks exploiting multiple versions of the same image at different resolutions, depending on the specific scale. This allows the propagation of gradients coming from multiple scales during the training phase. As opposed to other similar works based on statistical downscaling of climatic fields, low-resolution T2M heatmaps were not artificially downsampled from the high-resolution counterpart, because images were directly gathered at low and high resolutions. Furthermore, the present study introduced an experimental multi-stage framework for evaluation purposes.

2.1.1 Related work

According to (Wilby et al., 2004), statistical downscaling is classified into three sub-categories: regression-based, weather classification-based, and weather generators-based approaches. Concerning regression-based approaches, several attempts at downscaling a variety of climatic fields— mainly temperature, precipitation and wind fields—have been proposed in recent years. Machine learning (ML) techniques have been widely adopted for downscaling. In particular, the LASSO regression was used in (Gao et al., 2014) for downscaling precipitation, whereas genetic programming (GP) was exploited in (Coulibaly, 2004) and (Sachindra and Kanae, 2019) for downscaling precipitation and temperature, and daily minimum and maximum temperature, respectively. Moreover, the random forest (RF) was used in (Bartkowiak et al., 2019) for land surface temperature, whereas a novel hybrid dynamical-statistical approach was presented in (Tran Anh and Taniguchi, 2018) focusing on the resolution of fine-scale rainfalls with lower computational costs, through a combination of dynamical and statistical downscaling. ANNs have been used in (Salimi et al., 2019) to perform precipitation downscaling, applying a further residual correction method based on Particle Swarm Optimization (PSO), an Imperialist Competitive Algorithm (ICA), and a Genetic Algorithm (GA). Furthermore, a Back-Propagation Neural Network (BPNN) and Support Vector Machine (SVM) fusion approach was adopted in (Min et al., 2020) to downscale precipitation. Several works moved towards deep architectures in the context of Deep Learning (DL), especially concerning Long-Short Term Memory (LSTM) networks, Convolutional Neural Networks (CNNs) and Generative Adversarial Networks (GANs). Since their introduction (Hochreiter and Schmidhuber, 1997), LSTMs have been proven to be suitable for recovering and bridging information arbitrarily far in time, while avoiding the vanishing gradient problem. LSTMs have also been widely adopted for time-series related problems and, in the context of climate downscaling, for statistical downscaling of precipitation (Misra et al., 2017) and rainfall forecasting (Tran Anh et al., 2019) in combination with Feed Forward Neural Networks (FFNNs). Convolutional Neural Networks (CNNs), due to their ability to deal with spatio-temporal multi-dimensional structures, have been demonstrated to be particularly suitable for accomplishing SR tasks. Several attempts to use deeper architectures have been proposed (Dong et al., 2014; Dong et al., 2016; Kim et al., 2016; Johnson et al., 2016; Kim et al., 2018; Park et al., 2019) for the extraction of high-level image characteristics and the downscaling of climatic fields. A deep neural network based on a CNN and a LSTM recurrent module was proposed in (Miao et al., 2019) to estimate precipitation based on well-resolved atmospheric dynamical fields. A novel architecture, named DeepSD, based on the super resolution framework, was presented in (Vandal et al., 2017; Vandal et al., 2018a) for downscaling precipitation fields, and a CNN-based approach was proposed in (Pan et al., 2019) as an alternative solution to the existing precipitation-related parameterization schemes for the numerical precipitation estimation. A CNN model for downscaling the occurrence of precipitation was also proposed in (Baño-Medina et al., 2018), whereas different configurations of CNN were adopted in (Sun and Lan, 2020) to downscale daily temperature and precipitation over China. A competitive DL framework based on a CNN was presented in (Huang, 2020) for downscaling temperature and precipitation, and it performed particularly well in generating spatio-temporal details at very fine-grid scales. A U-Net- type CNN was also used in (Kern et al., 2020) to learn a one-to-one mapping of low-resolution (input) to high-resolution (output) wind fields simulations data, and a conditional variational autoencoder (based on CNN) was exploited for learning the conditional distributions from data, assessing multimodalities and uncertainties. A CNN was adopted in (Shi, 2020) to perform smart dynamical downscaling (SDD) for extreme precipitation events, whereas a surrogate model, based on a Deep CNN (DCNN), was evaluated in (Sekiyama, 2020) for surface temperature, and was found to estimate image details that were not retained in the inputs. Recently, remarkable results were reported in several studies exploiting GANs for SR tasks in climate science. An Enhanced Super-Resolution GAN (ESRGAN) (Wang et al., 2019) was adopted and presented in (Singh et al., 2019) to downscale wind speeds from a coarse grid, capturing high frequency power spectra and high order statistics in the dataset, thus generating images of superior visual quality compared to the SR-CNN. A novel method (ClimAlign) was introduced in (Groenke et al., 2020) for unsupervised, generative downscaling of temperature and precipitation based on normalizing flows for variational inference. Further works

(Mendes and Marengo, 2009; Mouatadid et al., 2017; Chang et al., 2018; Liu et al., 2017; Sharifi et al., 2019; Baño-Medina et al., 2020; Höhlein et al., 2020; Xu et al., 2020; Li et al., 2020) opted for downscaling based on ML and DL, and they helped assess both strengths and weaknesses of such methods. The results reported in these studies show that downscaling models based on ML allow better performance with respect to the other statistical approaches presented in (Sachindra et al., 2012; Goly et al., 2014; Duhan and Pandey, 2014).

2.2 Materials and Methods

2.2.1 Data

The proposed work aimed at downscaling 2-meter temperature over the EURO-CORDEX domain (EURO-CORDEX, 2021), by learning a statistical relationship between HR images and their LR counterparts, each one representing a temperature map. Analysis data was gathered in NetCDF4 format (NetCDF, 2021) from the COPERNICUS ERA-Interim global atmospheric reanalysis dataset (ERA-Interim, 2021a), and treated as images throughout the various experiments, according to the SISR framework. All the background information can be found in (ERA-Interim, 2021b). From the global domain, the EURO-CORDEX subdomain was selected for both resolutions, from −48.5° E to 69.75° E of longitude and from 73.9° N to 20.15° N of latitude. The horizontal resolutions of HR and LR images are about 13.87 km (0.125° × 0.125°) and 83.25 km (0.75° × 0.75°) respectively, whereas the related image sizes are 431 × 947 and 72 × 158 pixels. The data covers a temporal range from January 1979 to December 2018 (40 years) and is made up of 6-hourly samples. Thus, each day of a year consists of four samples, at 00:00, 06:00, 12:00 and 18:00, respectively.

2.2.2 The Architecture: Multi-Scale Gradients GAN for Statistical Downscaling

The proposed Multi-Scale Gradients GAN architecture for statistical downscaling (MSG-GAN-SD, reported in Figure 2.1) is based on the fusion of two frameworks, that is, the super resolution GAN defined in (Ledig et al., 2017) and the Multi-Scale Gradients GAN proposed in (Karnewar and Wang, 2020). The multi-scale framework in (Karnewar and Wang, 2020) was considered in order to speed up both the development and the tuning of the model, overcoming the difficulties of a conventional GAN caused by its training instabilities. The nature of the statistical downscaling task required the employment of a SR architecture (Ledig et al., 2017), in which a LR version of the HR image (target) is used to feed G, instead of the random latent vector proposed in (Goodfellow et al., 2020). In (Karnewar and Wang, 2020), two implementations of the MSG framework were described: the MSG-ProGAN and the MSG-StyleGAN. They share the same architectural structure (layers) for G, but differ for D layers and the loss functions used. The present study selected the MSG-ProGAN (Karnewar and Wang, 2020) as the baseline architecture and used the Wasserstein GAN with Gradient Penalty (WGAN-GP) loss function (Gulrajani et al., 2017). Both G and D retain the ascending and descending complexity pattern of the MSG-ProGAN layers. At the early stages of development, the architectures reported in Tables A1 and A2 of Appendix A were adopted for G and D, respectively, in order to match the 2 × 4 low-resolution and 480 × 960 high-resolution padded images. Starting from a 2 × 4 input image, obtained by progressively undersampling the original LR image (see Section 2.3), G produces upsampled images at finer scales. Specifically, at the end of each block of G, an image is generated at the associated scale (see G0–G5 blocks in Table A1). The generated images have the property of being intelligible (once denormalized), thus representing the downscaled temperature map at a particular resolution. When discriminating the generated (fake) samples, images produced by G0–G5 are to be considered as auxiliary images that are further fed to the blocks of D matching the corresponding resolutions. By comparison, when D discriminates real samples, D blocks are additionally fed with the correspondingly downsampled version of higher resolution real images. In both cases, the auxiliary images in a certain block of D are concatenated with the activation volume coming from the preceding block, by means of a simple combine function. After each convolution, a Leaky ReLU (LReLU) with $\alpha = 0.2$ was used in order to keep positive values unchanged and lower the negative values. In addition, only for G, the PixelNormalization scheme (Karras et al., 2018) was used after each activated convolution, in order to make convergence faster and prevent signal escalation. A MinBatchStdDev (Denton et al., 2015; Dinh et al., 2015) layer was used at the beginning of each block of D to obtain statistics about the batch formed by the previous activation volume and the auxiliary image along with the different scales, thus improving sample diversity (Karnewar and Wang, 2020). Concerning the WGAN-GP loss, because D is a function of multi-scale input images produced by G or the corresponding ground truth HR images at different resolutions, the gradient penalty was modified to be the average of the penalties over each input. This dependence makes the multi-scale gradients able to flow between intermediate layers of both D and G (Karnewar and Wang, 2020).

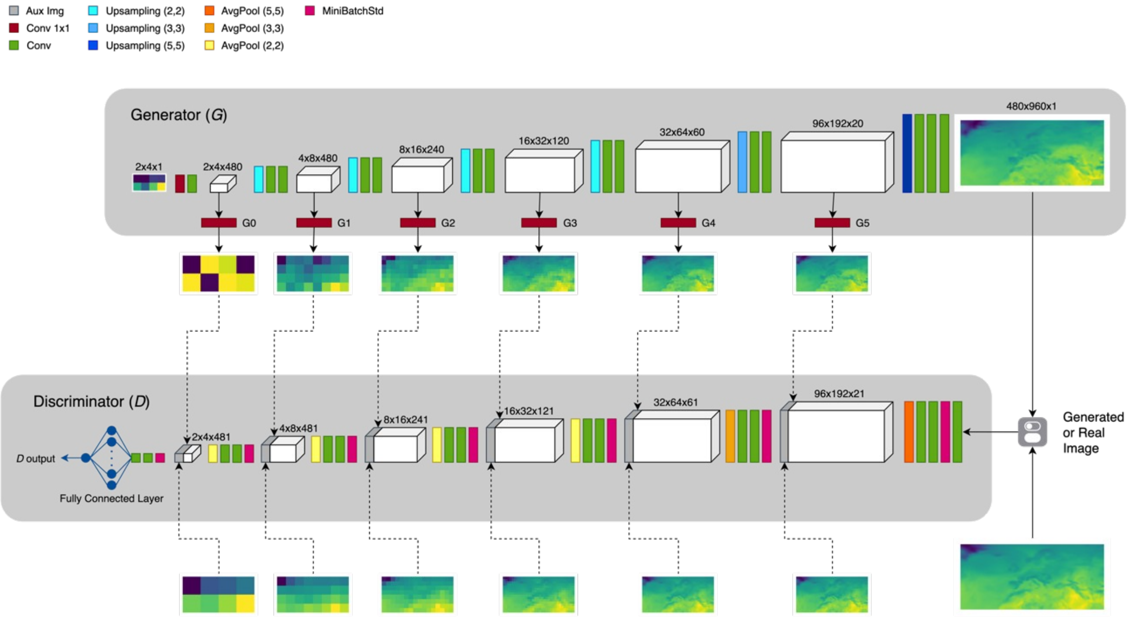


Figure 2.1 *The proposed MSG-GAN-SD architecture. The overall network architecture is composed of the Generator (G) and the Discriminator (D). The various multi-scale images entering as input to D are generated or real depending on the Discriminator (D). The various multi-scale images entering as input to D are generated or real depending on the sample being analyzed.*

2.2.3 Data pre-processing

In order to match architectural needs, HR images were artificially padded through edge-padding before training and test phases, going from 431 × 947 to 480 × 960 pixels. Edge-padding consists of adding a certain number of rows and columns at the image edges, by replicating the pixel values at the edges. Subsequently, starting from the 480 × 960 resolution, different scale versions of the padded images were progressively generated by means of the bilinear interpolation undersampling technique: 96 × 192, 32 × 64, 16 × 32, 8 × 16, 4 × 8, and 2 × 4. The edge-padding was also used for LR images, going from 72 × 158 to 80 × 160 pixels. Subsequently, the bilinear interpolation technique was applied to undersample padded LR images from 80 × 160 to 2 × 4 grid points, passing through the following intermediate resolutions: 40 × 80, 20 × 40, 10 × 20. The progressive undersampling procedure allows losing as little information as possible. Additionally, each resolution was chosen to maintain the same proportionality between rows (latitude) and columns (longitude), in order to satisfy the multi-scale framework requirements. Data was partitioned into three non-overlapping subsets, namely training, validation and test sets, that correspond to 87.5% (1979-2013, 35 years), 2.5% (2014, 1 year) and 10% (2015-2018, 4 years) of the whole dataset, respectively. Both HR and 2 × 4 images were normalized in the range [-1, 1] by computing their own maximum and minimum on the training set, and then by scaling all images in the training, validation and test sets, accordingly. Furthermore, because D takes multiple images at different resolutions as input, and these images were constructed from the HR image by applying progressive downsampling, the intermediate scales were also normalized by using HR image extrema. The same procedure was also applied for normalizing 2 × 4 images to be fed to G. Once the output is produced by G, a denormalization of the HR images is required to get back to the original temperature values. Afterwards, both generated and real HR images in full resolution were processed in order to remove the edge-padding. As explained in sub-section 2.2.4, two dataset arrangements were derived: the monthly set-up, where data is organized in twelve subsets, each one referring to a particular month, across different years (e.g., January 1979, January 1980, . . . , January 2018); and the season-based set-up, where data is organized in four subsets, each one referring to a particular season across different years (e.g., DJF 1979, DJF 1980, . . . , DJF 2018). In the remainder of the manuscript, seasons are intended to be DJF (December–January–February), MAM (March–April–May), JJA (June–July–August) and SON (September–October–November), according to the climate science scientific literature.

2.2.4 Experimental setup

Experiments were carried out exploiting the Marconi100 GPU cluster hosted by CINECA (CINECA, 2021). The HPC system is based on the IBM Power9 architecture with NVIDIA Volta GPUs. Specifically, each node hosts 2 × 16 cores IBM POWER9 AC922 at 3.1 GHz with 256 GB/node of RAM memory and 4 × NVIDIA Volta V100 GPUs per node, Nvlink 2.0, 16 GB (MARCONI100, 2021). Regarding the software adopted for our implementation, both architecture and training/test control flows were written in Python v3.8.2 based on the Keras API v2.4.3 (KERAS, 2021) and relying on the TensorFlow v2.2.0 (Abadi et al., 2016) backend. Training was performed in a distributed fashion by means of the TensorFlow Distributed Training (TDT, 2021) and MirroredStrategy. The results of the present study were achieved by running the model on just one node of the Marconi100 cluster exploiting all 4 GPUs available.

2.2.4.1 Training set arrangements

Because the dataset consists of 40 years of data, two dataset arrangements were derived: *(i)* monthly, where the dataset was organized in twelve subsets, each having data referring to a particular month across different years (e.g., January 1979, January 1980, . . . , January 2018); and *(ii)* season-based, where the dataset was organized in four subsets, each having data referring to a particular season across different years (e.g., DJF 1979, DJF 1980, . . . , DJF 2018). Clearly, the seasonal arrangement dataset is three times bigger than the monthly dataset because the same number of years is considered. Training the model on month- and season-based arrangements of the same dataset allows gaining valuable insights into the ability of the MSG-GAN-SD to capture intra-monthly, intra-seasonal and/or inter-annual climatic dynamics (Sachindra et al., 2018). In this manner, the models’ capability of capturing daily anomalies and extreme events is ensured, as reported in (Vandal et al., 2018b). Consequently, a total of 12 monthly and 4 seasonal generator models were collected after the training phase. Therefore, during the inference phase, each monthly model was able to generate a downscaled T2M map for the corresponding month, whereas each seasonal model was able to provide maps for each month belonging to the corresponding season.

2.2.4.2 Training configurations

In the literature, a well-known strategy for the WGAN-GP framework is to make D able to learn more quickly, leading it to be more powerful than G at a particular training epoch. This is based on the intuition that, during the training phase, if D is sufficiently accurate in the discrimination task, then its gradients flowing back and the subsequent update of G weights allow an improvement of the generation task (Gulrajani et al., 2017; Arjovsky et al., 2017). This strategy can be carried out in at least three ways: *(i)* raising D learning rates with respect to G; *(ii)* making D deeper, thus increasing its number of weights and enhancing its capacity; and *(iii)* using an imbalanced training, particularly suited for deep GANs. In this case, for each epoch, the Discriminator weights are updated more times with respect to those used for the Generator. The number of the discriminator updates is, from now on, referred to as DtrainUpdates. In the present study, a batch size of 64 and the RMSProp optimizer (Dauphin et al., 2015) with learning rates of 0.0003 and 0.0001 were used for G and D, respectively. However, because of memory issues, it was not possible to deepen D. Despite this, the discriminator was trained using three different update configurations (DtrainUpdates = 1, 2, 3 respectively). Therefore, a total of 48 models (12 monthly models × 3 DtrainUpdates + 4 seasonal models × 3 DtrainUpdates) were trained for 1000 epochs, saving them every 50 epochs.

2.2.4.3 The validation framework

The experimental workflow consisted of training the proposed architecture on the four climatological seasons (i.e., DJF, MMA, JJA, SON) and all twelve months, for all DtrainUpdates configurations (i.e., 1, 2, 3). The validation framework was based on three phases: (1) extraction of the best models through the minimization of a mathematical expression aiming to select the epoch in which the architecture performed better, by looking at training and cross-validation errors; (2) an evaluation procedure, which assessed the quality of the previously selected models by computing several metrics on the test set; and (3) a final test procedure, in which insightful climatological aggregated maps were created, using the previously selected best models.

(1) *Best Models Selection*

For each seasons/DtrainUpdates and months/DtrainUpdates combination, it was necessary to select a model at the training epoch in which the network was sufficiently mature. This means that it well approximated the mapping between the ground truth and the generated distributions, without incurring a lack of generalization. Therefore, this epoch should be the point where the trade-off between underfitting and overfitting is reached. At the same time, accuracy and variety (or sample diversity) capabilities also need to be considered. In order to consider these factors during the training phase, different random batches of samples, coming from both training and validation sets, were monitored by computing the batch-averaged MSE between generated and real HR images, across scales. By using these variable batches, it was possible to check both the accuracy and the variety of the ongoing trained models at the same time. Ideally, the epoch at which the training and validation sets MSEs are sufficiently close was supposed to be the best epoch. In fact, there is usually a sort of convergence of the two aforementioned errors beyond this point. However, as a consequence of the GAN's training instabilities, although both MSEs could be close in multiple epochs, they can report high values in these epochs. Even if MSG-GAN mitigates these issues by increasing the stability during training through the use of additional multi-scale gradient information, the choice of the best epoch in which the model performed better is not a trivial task. In order to tackle this problem, the following expression was proposed:

$$e_{best} = \underset{e}{argmin}\left(\lambda_{tr} MSE_{tr,e} + \lambda_{tr-cv} \left| MSE_{tr,e} - MSE_{cv,e} \right| \right) \tag{2.1}$$

where $MSE_{tr,e}$ and $MSE_{cv,e}$ are the MSEs computed at each epoch e on a random batch of 64 training and validation samples, respectively; λ_{tr} and λ_{tr-cv} (both set to 1) weight the $MSE_{tr,e}$ and the difference term $\left| MSE_{tr,e} - MSE_{cv,e} \right|$, respectively; and e_{best} represents the point where the trade-off between underfitting and overfitting is reached. Further details on Equation (2.1) are reported in Appendix B. Once the selection of the best-epoch model was completed, a total of 48 models for the various seasons/DtrainUpdates and months/DtrainUpdates combinations were analyzed.

(2) *Evaluation Procedure*

The previously described selection procedure leads to a pool of model candidates, which are further tested by making predictions on the test set and computing evaluation metrics such as:

- Mean Squared Error (MSE):

$$MSE = \frac{1}{N} \sum_{i=1}^{N} \left(T_{real,i} - T_{gen,i} \right)^2$$

- Peak Signal-to-Noise Ratio (PSNR):

$$PSNR = 10\log_{10}\left(\frac{T_{gen}}{MSE\left(T_{real,}, T_{gen}\right)}\right)\ [dB]$$

- Log Spectral Distance (LSD):

$$LSD = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(10\log_{10}\frac{T_{real,i}}{T_{gen,i}}\right)^2}$$

- Structural Similarity Index Measure (SSIM):

$$SSIM(x,y) = \frac{\left(2\mu_x\mu_y + c_1\right)\left(2\sigma_{xy} + c_2\right)}{\left(\mu_x^2 + \mu_y^2 + c_1\right)\left(\sigma_x^2 + \sigma_y^2 + c_2\right)}$$

- Fréchet Inception Distance (FID):

$$FID = d^2\left((m,C),(m_w,C_w)\right) = \left\|m - m_w\right\|_2^2 + tr\left(C + C_w - 2\left(C C_w\right)^{\frac{1}{2}}\right)$$

where:

- T_{real} and T_{gen} are real and generated T2M maps, respectively
- x and y refer to square windows of fixed size
- μ_x and σ_x are mean intensity and standard deviation of the x window (similarly for y)
- σ_{xy} is the covariance between x and y
- c_1 and c_2 are non-negative constants used to stabilize the division with weak denominator
- $d^2\left((m,C),(m_w,C_w)\right)$ represents the Fréchet distance between the Gaussian with mean (m,C obtained from the probability of generating model data and the Gaussian (m_w,C_w obtained from the probability of observing real world data.

Further details about SSIM and FID metrics are reported in (Wang et al., 2004) and (Heusel et al., 2017), respectively.

Moreover, several temporal metrics are recorded, such as the total predictions' elapsed time and mean prediction elapsed time for the sample. When testing the models on the test set, the model that reaches the best trade-off on the quality metrics for each season/month is referred to as the best model. All these evaluation metrics were used to define a novel comprehensive metric called 5-fold Accuracy Perceptivity Product ($5f_{APP}$) reported in Equation (2.2).

$$5f_{APP\square} := Accuracy \times Perceptivity = \frac{\left(\frac{\lambda_{MSE}}{MSE}\right)\left(\lambda_{PSNR}PSNR\right)\left(\lambda_{SSIM}SSIM\right)}{\left(\lambda_{FID}FID\right)\left(\lambda_{LSD}LSD\right)} \quad (2.2)$$

All the λ values in Expression (2.2) can be arbitrarily selected. In the present setup these parameters were set to 1 in order to equally weight all the metrics involved in the computation. Specifically, the higher the λ value, the more the corresponding metric affects the $5f_{APP\square}$ result. More details on Equation (2.2) are reported in Appendix B. At this point, it is possible to claim whether seasonal or monthly training performed better. Furthermore, the best models were used to produce T2M maps that were compared with the corresponding real high-resolution maps. Pixel-wise Mean Absolute Error (MAE), and Pearson ($\rho_{X,Y}$) and Spearman (r_s) correlations between generated and real images, were also assessed and reported, according to the following formulations:

$$MAE = \frac{1}{d}\sum_{i=1}^{d}\left|X_i - Y_i\right|$$

$$\rho_{X,Y} = \frac{cov\left(X,Y\right)}{\sigma_X \sigma_Y}$$

$$r_s = \frac{cov\left(R\left(X\right), R\left(Y\right)\right)}{\sigma_{R\left(X\right)}\sigma_{R\left(Y\right)}}$$

where:

- X and Y represent vectors of T2M values for the same pixel location in the generated and real images, respectively. The dimension of these vectors is
    - d = # daily samples x # days in a month (for the monthly arrangement)
    - d = # daily samples x # days in a season (for the seasonal arrangement);
- $cov(X, Y)$ is the covariance between X and Y;
- σ_X, σ_Y are the standard deviations of X and Y, respectively;
- R(X), R(Y) are ranks of X and Y, respectively;
- $\sigma_{R(X)}$, $\sigma_{R(Y)}$ are the standard deviations of R(X) and R(Y), respectively.

Additionally, the statistical significance of the Spearman correlation was reported using the Spearman's associated *p-value* (CI 95%).

(3) *Final Test Procedure*

In this phase, monthly samples were temporally averaged. In order to identify the interannual trends, this was done for each month of every test set year. The monthly maps were further averaged along the test set years for obtaining interannual monthly maps, which allowed the assessment of the overall T2M averaged value in the considered period. This kind of aggregation was useful to capture large scale climate trends for different temporal scales (Vandal et al., 2018b). The same procedure was also applied to correlation maps, such as Pearson, Spearman, and Spearman's associated p-value between generated and ground truth samples.

2.3 Results and Discussion

This section presents the results of the training phase and evaluation procedure. For the best model selection among the various training epochs, both λ_{tr} and λ_{tr-cv} in Equation (2.1) have been set to 1. After the validation procedure, the best model was found to be the one trained on the JJA season (from now on, it will be indicated as the JJA model) with DtrainUpdates = 1.

2.3.1 Training results

Figure 2.2 reports the MSE calculated on random training batches of the aforementioned model. Both errors exhibit an early decreasing behavior that leads to a substantial stable convergence in the last epochs. This indicates that the mapping was learnt very quickly, starting around the 50^{th} epoch. The remaining training time was used to improve the model perception capabilities, learning complex details, geometric structures, and high- frequency details, and to enhance sample diversity. Despite this, in the last epochs, the errors did not change significantly. The simultaneous training of all MSG-GAN-SD multi- scale layers makes each epoch slower (especially for models with DtrainUpdates equal to 2 or 3), but fewer epochs were needed for reaching convergence. This means that the overall time required to train the model was reduced compared to that of traditional GANs (Karnewar and Wang, 2020).

Figure 2.3 reports the resulting generated images of a single fixed sample along the training epochs and resolutions. Throughout the training, all the scales synchronize with each other and mature together during the epochs in terms of accuracy and perceptivity. Concerning the training time performance, the average execution time of each experiment is reported in Table 2.1. It must be remarked that the experiments were run under the operative conditions explained in Section 2.2.

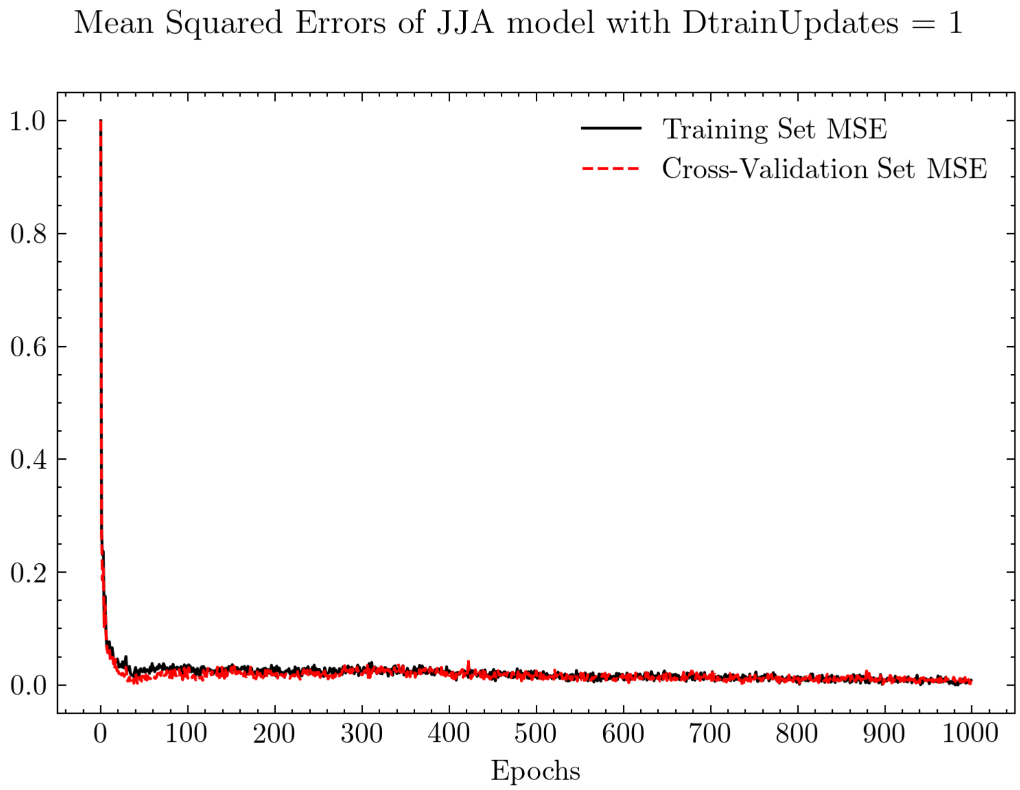


Figure 2.2 *Comparison between training and cross-validation MSEs along training epochs. The training is related to the JJA model with DtrainUpdates = 1. At each epoch, MSEs have been averaged over random batches of 64 samples and scaled in the range [0,1].*

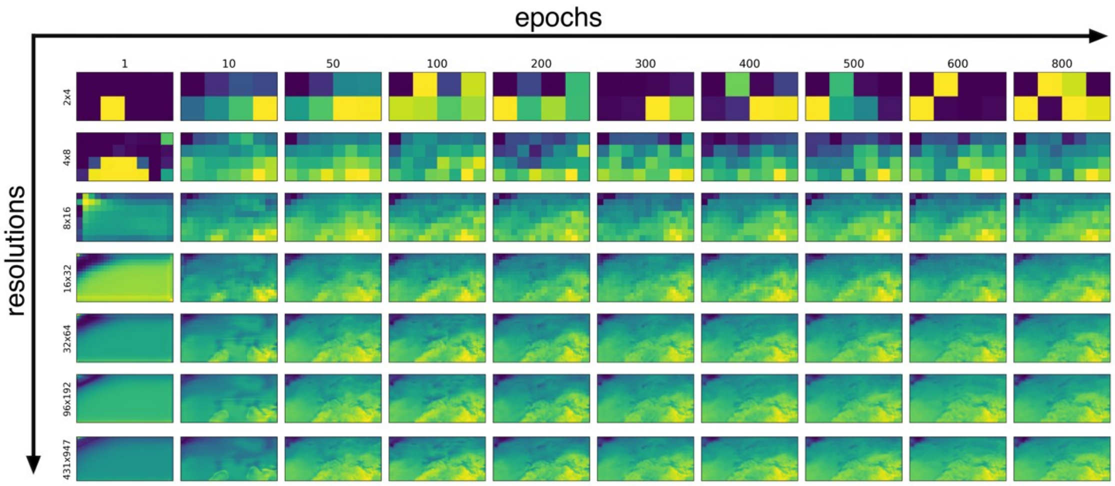


Figure 2.3 *Fixed generated sample along different training epochs and resolutions. The training is related to the JJA model with DtrainUpdates = 1.*

Table 2.1 Training execution time performance.

Training set arrangements	Month-based	Season-based
DtrainUpdates = 1	~1 day	~2 days
DtrainUpdates = 2	~1 day	~3 days
DtrainUpdates = 3	~2 days	~4 days

2.3.2 Evaluation procedure

Table A3 reported in Appendix B shows the number of discriminator updates in the training phase (DtrainUpdates) and the selected epoch of the best models, for both month- and season-based arrangements. For each of the three DtrainUpdates models, the best epoch was obtained by applying Equation (2.1), then the best model was the one that produced the highest $5f_{APP□}$ score after a test cycle. As shown in Table A3, there was no evidence of the improvement caused by training D more times than G in a single epoch.

Tables A4 and A5 in Appendix B report the outputs of the evaluation procedure on the test set, for both monthly and seasonal models' outcomes, respectively. For each considered month, it is evident from the $5f_{APP□}$ metric (higher is better) that the seasonal models perform better against the respective monthly models during the test set years (2015–2018). Evidently, still looking at the $5f_{APP□}$metric, they reach a sort of best trade-off among all the metrics involved in the Accuracy and Perceptivity terms. Thus, the higher number of samples in seasonal dataset arrangements was of greater benefit than the more specific climate dynamics exhibited by the monthly arrangements. Moreover, from a practical standpoint, it is more convenient to manage four seasonal models rather than twelve monthly ones. According to the $5f_{APP□}$, it can be easily noted that the JJA model was found to be the best among the seasonal models, because the metric scores on the average are higher than those reported for the months in the other seasons (see Table A5 in Appendix B). Additionally, the JJA model reached the highest $5f_{APP□}$ score in predicting the August month.

2.3.3 Test results

Figure 2.4 shows the results generated by the JJA model in August (left panel), in addition to the ground truth (center panel) and the MAE (right panel). In all cases, the generated and real maps appear to be nearly indistinguishable. In fact, the MAE maps in Figure 2.4 are mainly blue or dark blue, representing a low error. However, there are recurrent regions in which the behavior is not as good with errors greater than 1 degree, as shown by critical yellow and light blue zones. Additionally, the highest errors (red hotspots) are mainly located in the North-West zone, where very low T2M values are recorded in the ground truth images. Consequently, the weight of the unavoidable error in the network forecasting phase is greater at very low real values. These errors may also depend on the use of a large amount of data during the network training phase. High temporal resolution data (6-hourly) has actually been used, which allowed for training the deep architecture introduced in the present work. By doing this, overfitting could be avoided but, at the same time, it inevitably led to fitting the noise due to day and night cycles that characterize the daily temperature trend. An alternative solution may be considering data at a coarser time resolution (e.g., daily or monthly) with an inevitable reduction in the number of samples available for the training phase. It would therefore be necessary to extend the time interval of the analysis (longer than 40 years) or look for simpler architectural solutions requiring fewer data. In addition, as a pre-processing step, each data sample may be expressed as an anomaly with respect to the daily or monthly average, in order to potentially fasten convergence. Nonetheless, the critical hotspots are mitigated by the interannual mean shown in Figure 2.5, at the expense of an increased error in the remaining zones. Figures 2.6 and 2.7 present, respectively, the monthly and interannual monthly means of Pearson (left panel) and Spearman (center panel) correlation metrics, along with the Spearman associated p-value with a confidence interval (CI) of 95% (right panel). The latter values were computed between the ground truth samples and those generated by the JJA model in August. Looking at these maps, it can be observed that there exists a positive moderate-to-strong correlation —of both Pearson and Spearman types— between generated and real samples, mostly in land areas. This correlation is higher than in sea areas, where there still exists a negative moderate-to-weak correlation. However, there are also some infrequent sea area hotspots in which a modest correlation exists. Considering the Spearman associated p-value, there exist some border-like zones, mostly located in the Western and Northern regions, where the non-linear correlation is not statistically significant (p-value > 0.05).

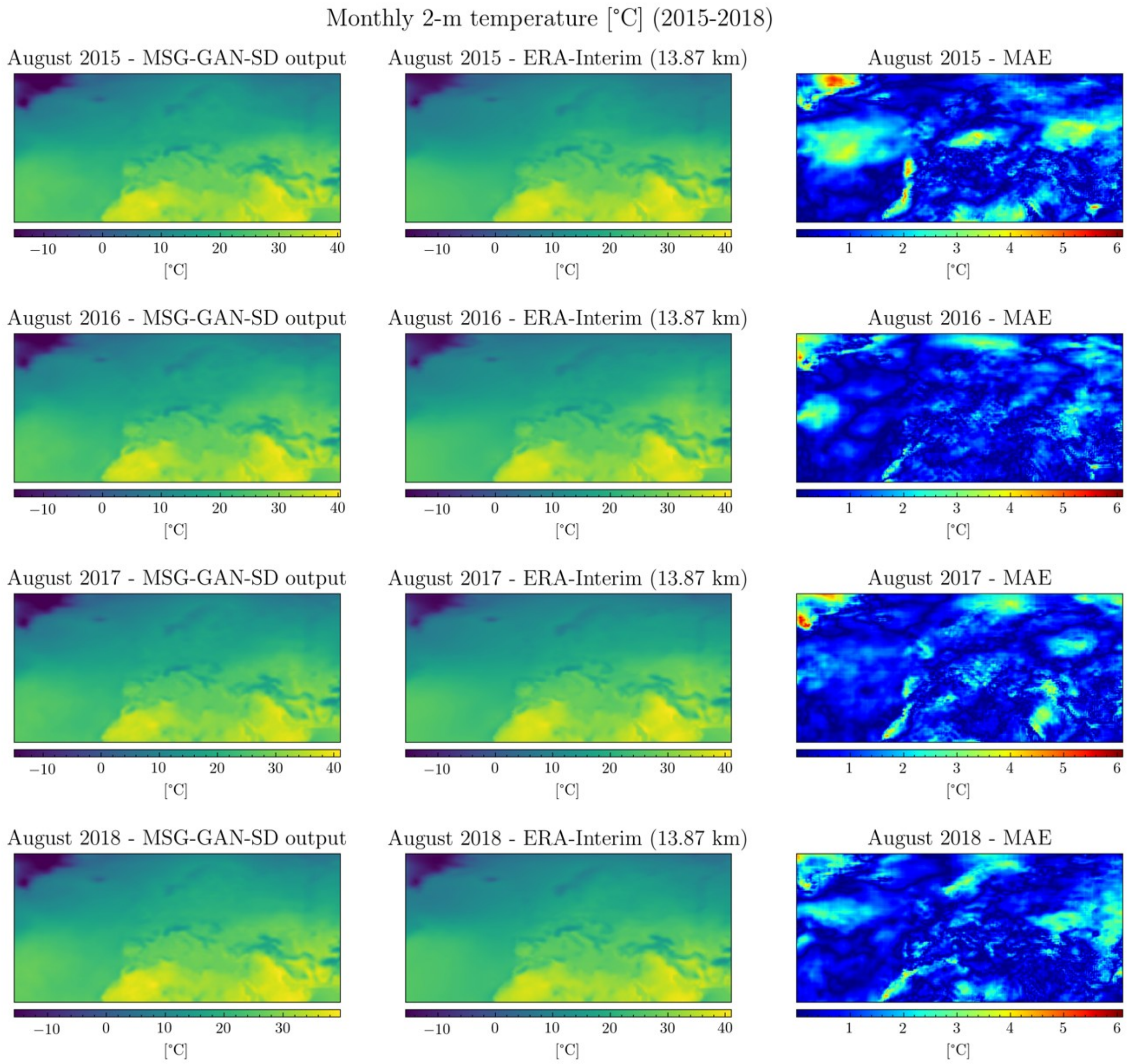

Monthly 2-m temperature [°C] (2015-2018)
August 2015 - MSG-GAN-SD output
August 2015 - ERA-Interim (13.87 km)
August 2015 - MAE
August 2016 - MSG-GAN-SD output
August 2016 - ERA-Interim (13.87 km)
August 2016 - MAE
August 2017 - MSG-GAN-SD output
August 2017 - ERA-Interim (13.87 km)
August 2017 - MAE
August 2018 - MSG-GAN-SD output
August 2018 - ERA-Interim (13.87 km)
August 2018 - MAE
−10 0 10 20 30 40
1 2 3 4 5 6
[°C]

Figure 2.4 *Monthly means comparison (for each August of every year from 2015 to 2018) among MSG-GAN-SD generated, ground truth (ERA-Interim 13.87 km) and MAE maps. The model used is the JJA with DtrainUpdates = 1.*

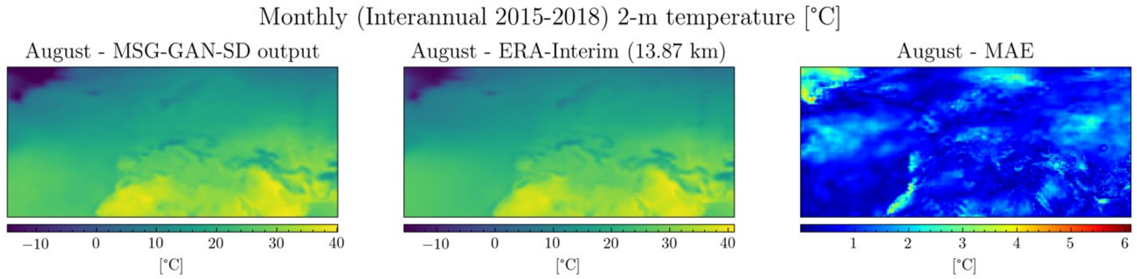

Monthly (Interannual 2015-2018) 2-m temperature [°C]
August - MSG-GAN-SD output
August - ERA-Interim (13.87 km)
August - MAE
−10 0 10 20 30 40
1 2 3 4 5 6
[°C]

Figure 2.5 *Interannual monthly means comparison (for each August of every year from 2015 to 2018) among MSG-GAN-SD generated, ground truth (ERA-Interim 13.87 km) and MAE samples. The model used is the JJA with DtrainUpdates = 1.*

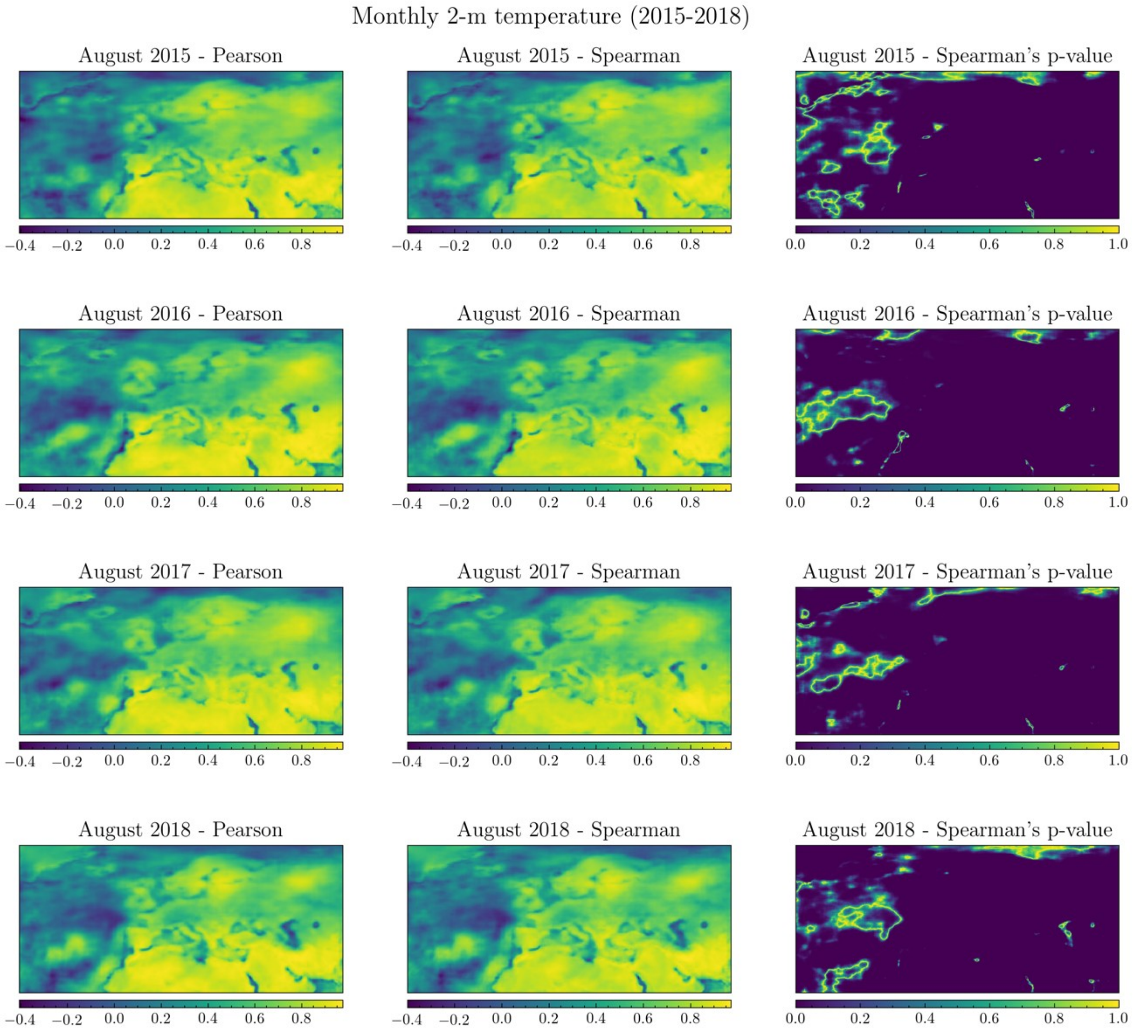


Figure 2.6 *Monthly means of Pearson, Spearman correlation metrics and the Spearman associated p-value (for each August of every year from 2015 to 2018) between MSG-GAN-SD generated and ground truth (ERA-Interim 13.87 km) samples. The model used is the JJA with DtrainUpdates = 1.*

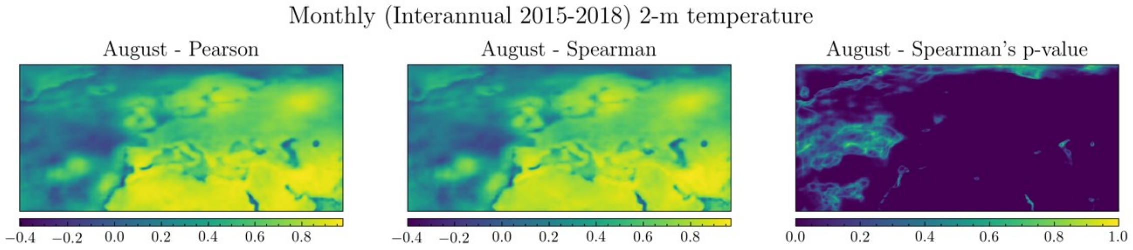


Figure 2.7 *Interannual Monthly means of Pearson, Spearman correlation metrics and the Spearman associated p-value (for each August of every year from 2015 to 2018) between MSG-GAN-SD generated and ground truth (ERA-Interim 13.87 km) samples. The model used is the JJA with DtrainUpdates = 1.*

The complete collection of the rest of the monthly and interannual monthly means comparisons is available at: https://drive.google.com/drive/folders/1GenkyhZHDGxfTF2K12lw2-5dxbkjFLpW?usp=sharing, (accessed on 23 December 2021). The considered architecture required about 43.942 s (on average) to predict a particular month for all test set years. Each atomic sample was processed in about 0.090 s on average. The aforementioned times were computed by averaging the execution times of all test runs. All these tests have been carried out by exploiting only one GPU on a single node of the cluster mentioned in Section 2.2.

2.4 Conclusions

This work is the first attempt to use the MSG-GAN architecture for statistical downscaling. The proposed architecture was used for downscaling 2 m temperature from the resolution of 83.25 to 13.87 km over the EURO-CORDEX domain. The season-based training set arrangement was found to overcome the monthly one. Formally, this was confirmed by means of a novel metric—the $5f_{APP}$□—introduced in the present study to simultaneously take both Accuracy and Perceptivity issues into account. In fact, the quality of the images generated by MSG-GAN-SD is very high as they appear nearly indistinguishable from the ground truth samples. However, some critical hotspots were highlighted in the North-West area of the EURO-CORDEX domain, and these are worthy of further investigations in order to improve the overall MSG-GAN-SD accuracy. From a computational standpoint, the training of a seasonal model on 36 years of the corresponding season required three days in the worst case (DtrainUpdates = 3), exploiting all the four GPUs of a single node. Clearly, the use of more compute nodes would have led to a significant reduction in the training time. Moreover, the inference phase for each month required less than a minute, considering the four years of the test set. The present work is not aimed at providing a comparison between MSG-GAN-SD output and other downscaling approaches, and the authors leave this kind of investigation for future work. Obviously, errors should be further reduced for this approach to be operationally adopted in climate science. Additional architectural variants shall therefore be taken into consideration as future work. Moreover, a new dataset with either a daily or a monthly temporal resolution will also be considered to avoid the noise coming from daily temperature cycles. Overall, the solution presented in this work paves the way for possible scenarios regarding the climate science context and the use of DL techniques coming from the SISR image processing domain, which offer flexible, powerful and computationally convenient solutions. The authors also intend to experiment with the introduction of additional climatic fields providing more information as inputs, in addition to the explicit embedding of the temporal dimension.

3

AR-CNN-SD: A climate change aware Residual CNN architecture for statistical downscaling of 2-meter temperature over the EURO-CORDEX domain

The present chapter discusses an enhancement of the work presented in the previous chapter. In particular, a Statistical Downscaling of 2-m temperature fields in the EURO-CORDEX domain using Residual Convolutional Neural Networks (CNN) is performed and results are evaluated. This use case has been investigated in collaboration with Francesco Immorlano and Prof. Pierre Gentine of the *Department of Earth and Environmental Engineering* at the Columbia University in the City of New York, in which I have been hosted during my abroad research period from Mar 1st to May 1st of 2022, granted by the GEMCLIME (Global Excellence in in Modelling of Climate and Energy) European project, which has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 681228.

Despite the satisfying results, the architecture employed in the previous work, The ***Multi-Scale Gradients GAN for Statistical Downscaling*** (**MSG-GAN-SD**) seems to be too complex for the considered problem frame and dataset. In fact, The MSG-GAN generator performs very well on the training set, but it is unclear why it lacks generalization capabilities (i.e. when predicting on the test set). An intuition arising by looking at the gap between interannual averaged generated maps and ground truth ones

(Figure 2.5) is that some kind of subtle overfitting has been affecting the network. The cause might reside in the fact the architecture is too complex with respect to the relatively small size of the training set. It may seem trivial, but diagnosing overfitting in this complex architecture with more than ten millions of parameters is very difficult, despite the good Best Models Selection practice and the strong mathematical diagnosis framework employed in the previous work (Chapter 2, Section 2.2.4.3).

In this investigation, a generator extracted from a vanilla SR-GAN, a simpler architecture has been re-engineered with the introduction of skip-connection residual information, thus being still able to extract meaningful and complex patterns in the data but with a more limited number of parameters. Moreover, its generalization capabilities on a different, bigger test set resulting from a new partition of the previous dataset, has been assessed.

3.1 Materials and Methods

3.1.1 Data

The dataset used in this research is the same as the one discussed in (Chapter 2, Section 2.2.1): the variable 2-meter temperature, both at LR (83.25 km i.e. 0.75° × 0.75°) and HR (13.87 km i.e. 0.125° × 0.125°) resolution has been taken from the ERA-Interim global atmospheric reanalysis dataset (ERA-Interim, 2021a). The data covers a temporal range from January 1979 to December 2018 (40 years) and is made up of 6-hourly samples. Data has been partitioned into three non-overlapping subsets, namely training, validation and test sets, which are 50.0% (1979–1998, 20 years), 2.5% (1999, 1 year) and 47.5% (2000–2018, 19 years) of the whole dataset, respectively. This subdivision is different from classical neural networks training best practice, where the training set is far more bigger than the test set. This was done for two reasons: (i) prevent the model from learning climate change signals that became more intense after 1999, (ii) assessing the model prediction capabilities on unseen data from 2000, characterized by a more significant climate change trend.

3.1.2 Experimental setup

Experiments have been carried out exploiting the Ginsburg hybrid (CPU-GPU) cluster hosted by Columbia University in the City of New York. The HPC system is a Dell architecture equipped with 87 Intel Xeon nodes, 30 high memory Intel Xeon nodes, 18 Nvidia RTX 8000 GPU nodes and 4 Nvidia V100S GPU nodes. Specifically, each standard node hosts 2 x 16 cores Intel Xeon Gold 6226 at 2.9 GHz with 192 GB/node of RAM memory. Each high memory node has 768 GB/node of RAM memory instead. The first 18 GPU nodes have 2 x Nvidia RTX 8000 GPU modules, whereas the latter 4 GPU nodes have 2 x Nvidia V100S GPU. Both these GPU nodes have the same computing / memory characteristics of the former CPU nodes.

Concerning the software, both architecture and training/test control flows have been written in Python v3.9.7 exploiting the Keras API v2.4.3 (Keras, 2015) relying upon TensorFlow v2.4.1 (Abadi et al., 2016) backend. Training has been performed in a distributed fashion by using the TensorFlow Distributed Training (TensorFlow Distributed, 2020) and adopting the MirroredStrategy. Also, ImageDataGenerator facilities have been used, in order to process the data only when required from the training algorithm. In particular, for the purposes of the present study, just one node per single experiment has been used, by consuming all the available resources (2 x GPUs).

3.1.3 The Architecture: Auto-Regressive CNN for Statistical Downscaling

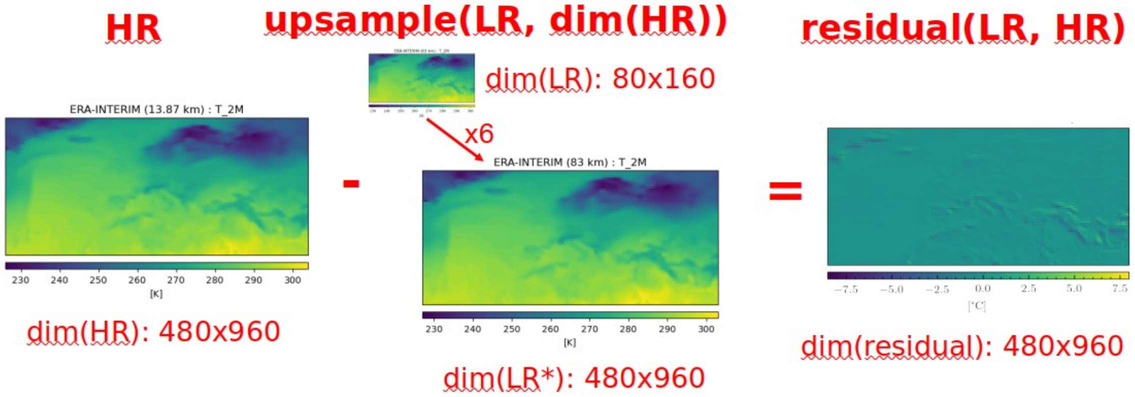


Figure 3.1 *A Residual Map (on the right), obtained by subtracting from the HR map (on the left) an upsampled version of the LR map (on center).*

A first preparatory investigation was the assessment of whether residual information might help in the training phase. A *Residual Map* is an HR image obtained by subtracting from the HR map an upsampled version of the LR map (to the resolution of the HR), as can be seen in (Figure 3.1).

Then, three types of network have been trained:

- *Baseline Networks* start from LR maps and generate pure HR maps;
- *Residual Networks* start from LR maps and generate Residual HR maps;
- *Added Networks* start from LR maps and generate HR maps consisting in pure HR maps plus the corresponding Residual HR maps.

Singularly taken, the former two networks are useless. But, they have to be combined in a so-called *Added-Residual Network* (this is a Virtual Network, it is not a physical network at all), such that the output of the Residual Network is subtracted from the corresponding Added Network, as shown in (Figure 3.2). These Networks have been demonstrated to outperform the Baseline ones throughout the various conducted experiments.

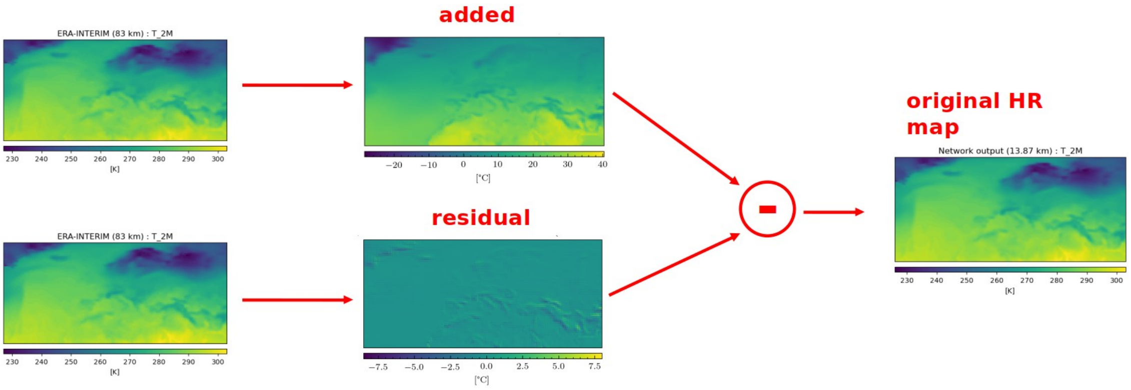


Figure 3.2 *A linear combination (i.e. a subtraction in this case) of the Added and the Residual Maps, generated by the respective Added and Residual Networks by starting from LR maps, leads to a better prediction of the original HR map.*

This fact has led to an intuition: the correction of HR maps with pure LR information again might help. Therefore, a new type of network starting from a vanilla SR-GAN generator has been developed: the *Auto-Residual Convolutional Neural Network* (AR-CNN, Figure 3.3), in which the convolutional volume is added to an upsampled version of the LR map. Being these added layers learnable, the CNN was able to decide by herself how to adjust the HR with information newly coming from the LR, whose details and patterns might have been lost during the main de-convolutional branch. This new network outperformed also the Added - Residual one.

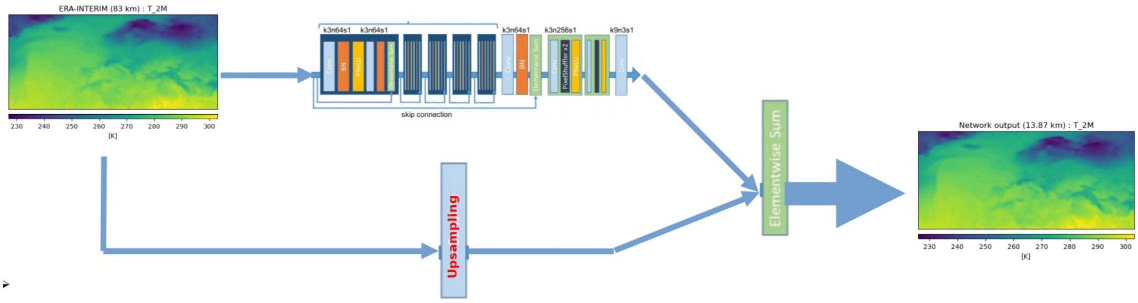


Figure 3.3 *Auto-Residual Convolutional Neural Network (AR-CNN). The baseline architecture is that of a vanilla SR-GAN generator, in which a skip-connection branch departing from the LR input and consisting in an upsampling layer, has been added in order to contribute to the output through an Elementwise Sum Layer. The abbreviations in the de-convolutional architecture depicted in the upper branch are only indicative and have been put only for exemplary purposes, thus they don't represent the real generator blocks composition.*

3.2 Results and Discussion

This section presents the results of the training phase and evaluation procedure. For the best model selection among the various training epochs, both λ_{tr} and λ_{tr-cv} in Equation (2.1) have been set to 1. After the validation procedure, the best model was found to be the one trained on the September month (from now on, it will be indicated as the Sep model).

3.2.1 Evaluation procedure

Table D1 reported in Appendix D shows the selected epoch of the best models, for both month- and season-based arrangements. The best epoch is obtained by applying the Eq. 1, then the best model is the one that produces the highest 5fAPP score after a test cycle. Tables D2 and D3 in Appendix D report the outputs of the evaluation procedure on the test set, for both monthly and seasonal models' outcomes, respectively. It is evident that $5f_{APP}$ values are far more than higher than those reported in Tables B2 and B3 of

Appendix B, referring to the previous work performance evaluation, indicating that AR-CNN-SD architecture outperforms the previous MSG-GAN-SD one in terms of generalization capabilities, due to being less prone to overfitting.

3.2.2 Test Results

Figure 3.4 shows the results generated by the Sep model in September (left panel), in addition to the ground truth (center panel) and the MAE (right panel). In all cases, the generated and real maps appear to be indistinguishable. In practice, all the MAE maps in Figure 3.4 appear all blue everywhere. The previous architecture issue, that has led to the rise of hotspots in various zones (Figure 2.17, 2.18), has been correctly resolved.

The interannual mean shown in Figure 3.5 further reduces the noise components in the predictions, being very few yet still present. Figures 3.6 and 3.7 present, respectively, the monthly and interannual monthly means of Pearson (left panel) and Spearman (center panel) correlation metrics, along with the Spearman associated p-value with a confidence interval (CI) of 95% (right panel). The latter values were computed between the ground truth samples and those generated by the Sep model in September. Looking at these maps, it can be observed that there exists a positive strong correlation —of both Pearson and Spearman types— between generated and real samples, everywhere in the maps, thus improving to the most possible degree the moderate correlation exhibited by the previous architecture in (Figure 2.19, 2.20).

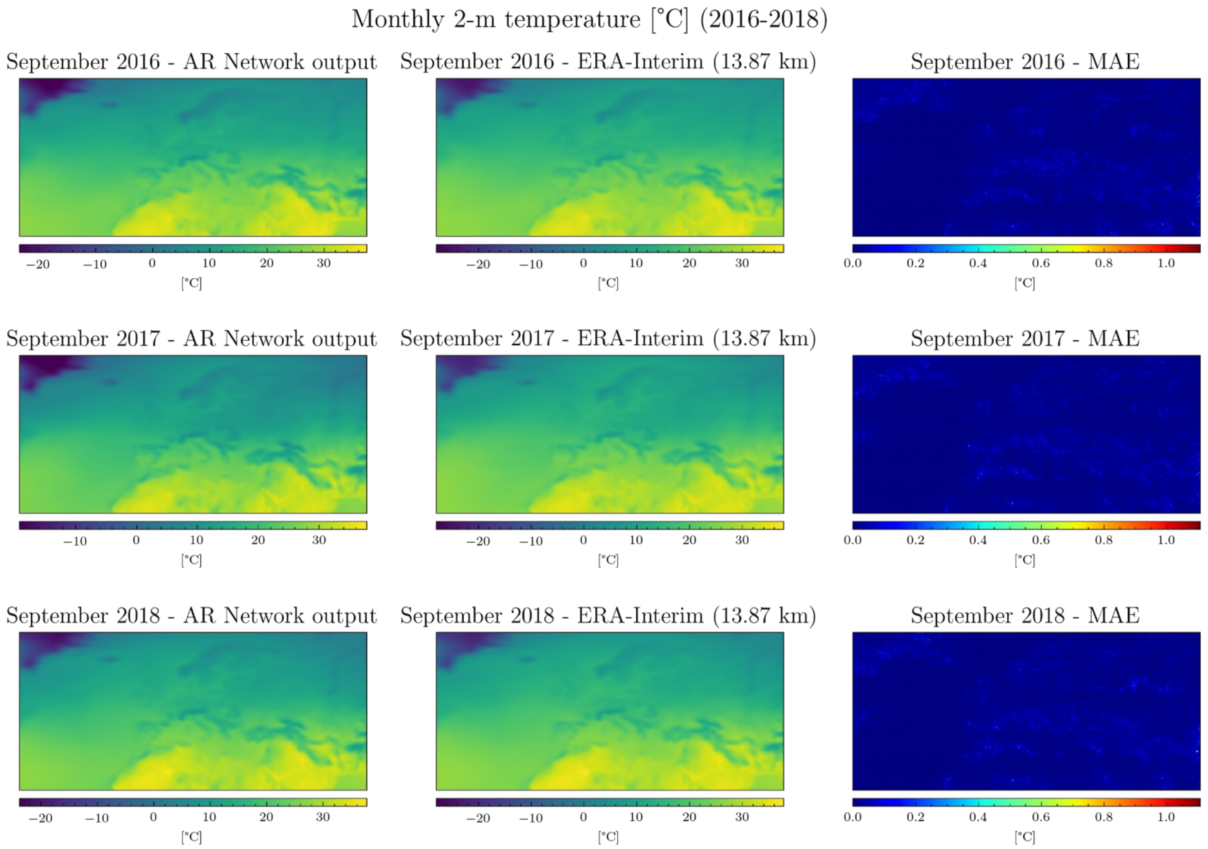

Monthly 2-m temperature [°C] (2016-2018)
September 2016 - AR Network output
September 2016 - ERA-Interim (13.87 km)
September 2016 - MAE
September 2017 - AR Network output
September 2017 - ERA-Interim (13.87 km)
September 2017 - MAE
September 2018 - AR Network output
September 2018 - ERA-Interim (13.87 km)
September 2018 - MAE
−20
−10
0
10
20
30
[°C]
0.0
0.2
0.4
0.6
0.8
1.0
[°C]

Figure 3.4 *Monthly means comparison (for each September of every year from 2016 to 2018) among the AR-CNN-SD generated, the ground truth (ERA-Interim 13.87 km) and the MAE maps. The model used is Sep.*

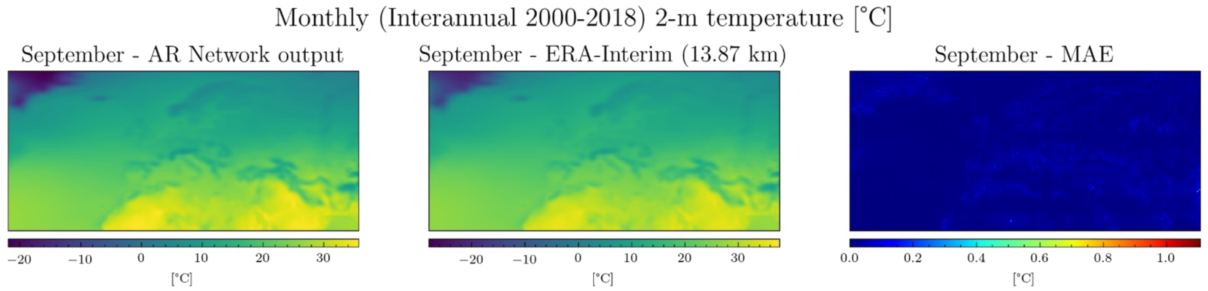

Monthly (Interannual 2000-2018) 2-m temperature [°C]
September - AR Network output
September - ERA-Interim (13.87 km)
September - MAE
−20
−10
0
10
20
30
[°C]
0.0
0.2
0.4
0.6
0.8
1.0
[°C]

Figure 3.5 *Interannual monthly means comparison (for each September of every year from 2000 to 2018) among the AR-CNN-SD generated, the ground truth (ERA-Interim 13.87 km) and the MAE samples. The model used is Sep.*

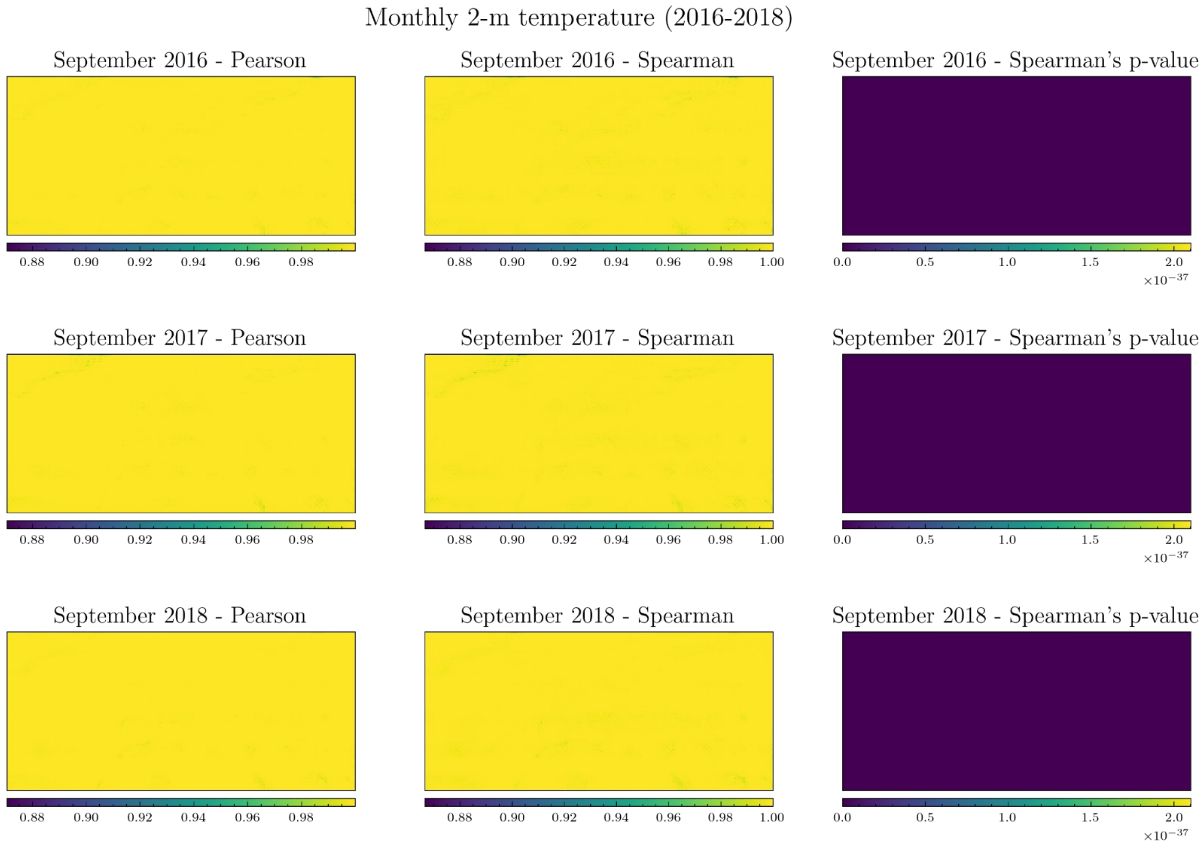

Monthly 2-m temperature (2016-2018)
September 2016 - Pearson
September 2016 - Spearman
September 2016 - Spearman's p-value
September 2017 - Pearson
September 2017 - Spearman
September 2017 - Spearman's p-value
September 2018 - Pearson
September 2018 - Spearman
September 2018 - Spearman's p-value
0.88 0.90 0.92 0.94 0.96 0.98 1.00
0.0 0.5 1.0 1.5 2.0
×10−37

Figure 3.6 *Monthly means of Pearson, Spearman and Spearman's associated p-value correlation metrics (for each September of every year from 2016 to 2018) between the AR-CNN-SD generated and the ground truth (ERA-Interim 13.87 km) samples. The model used is Sep.*

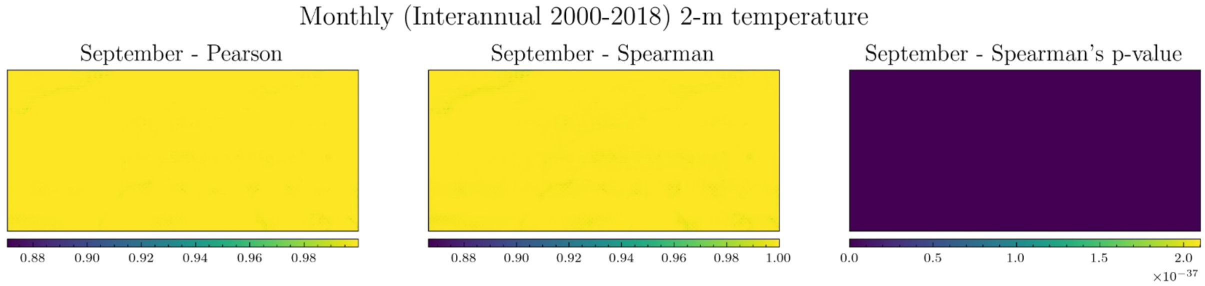


Figure 3.7 *Interannual Monthly means of Pearson, Spearman and Spearman's associated p-value correlation metrics (for each September of every year from 2000 to 2018) between the AR-CNN-SD generated and the ground truth (ERA-Interim 13.87 km) samples. The model used is Sep.*

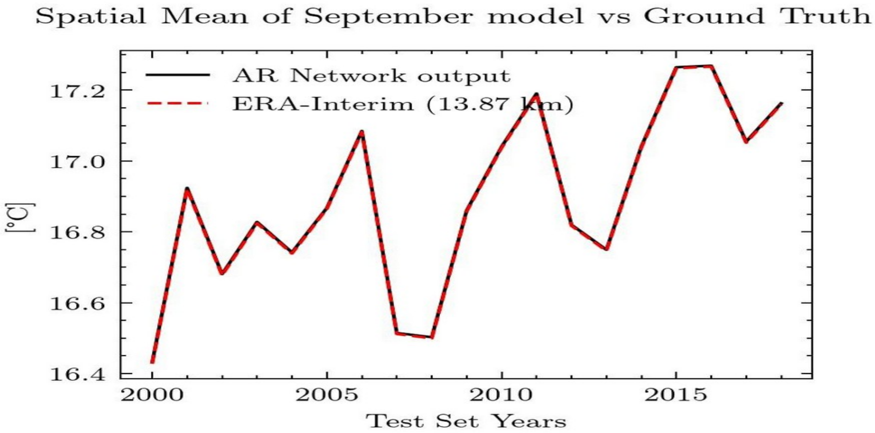


Figure 3.8 *Spatial Mean of the interannual monthly means (for each September of every year from 2000 to 2018) among the AR-CNN-SD generated (in black) and the ground truth (ERA-Interim 13.87 km) (in red). The model used is Sep.*

In (Figure 3.8), the interannual monthly means depicted in (Figure 3.5) were furtherly averaged, collapsing in one numerical value for each year. The trend is respected in a way that makes the two lines perfectly overlapped and thus indistinguishable.

4

Extreme Weather Event: Wildfires

Global warming and climate change are increasingly modifying the ecosystem and the occurrence of climate extremes, during the last decades. Among extreme events, wildfires are some of the most significant, since they are strongly correlated with climate change. High-intensity fires irreparably damage both flora and fauna and produce pyro-clouds, encouraging greenhouse gas emissions (*Fire Effects on the Environment*).

Fires are, in general, natural events that improve the fertility of the plants, cracking seeds (*Fire Effects on Plants and Animals*) and regenerating the soil (*Fire Effects on the Environment; Fire Effects on the Soil*). However, a consistent number of studies show that climate change encourages the development and spread of wildfires throughout the world (*Barbero et al., 2015; Global Warming and Extreme Wildfires, 2022; Global Wildfire Crisis, 2022; Will global warming produce more frequent and more intense wildfires?; Fire risk will increase by 2050, 2022)*. The European geographical domain is particularly sensitive to fire spread and development, due to the high cultural heritage as well as potential damages from economic and natural points of view.

In order to prevent wildfire occurrence, it is relevant to measure fire danger susceptibility in this area of interest. Among traditional fire risk assessment products, Canadian Fire Weather Index (FWI) is very popular, due to its high practical usefulness. By computing five intermediate indices (estimating soil moisture levels as well as the potential heat that could be spread in case of fire ignition) FWI can be evaluated on a global scale.

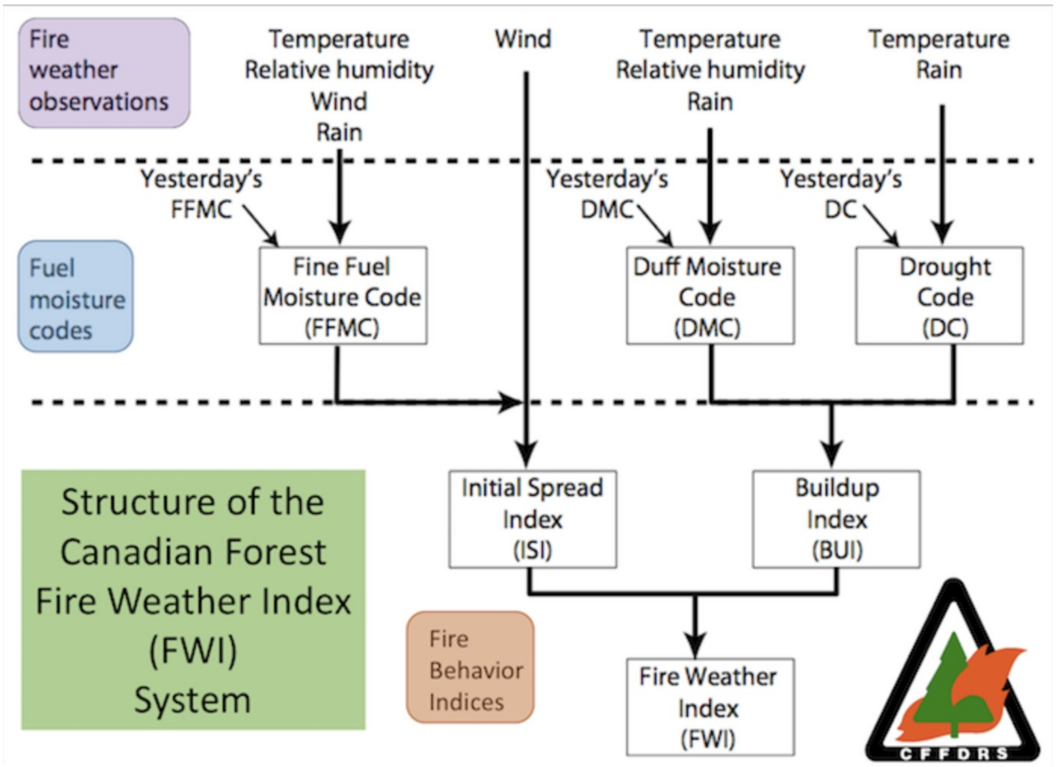

Fire weather observations
Temperature Relative humidity Wind Rain
Wind
Temperature Relative humidity Rain
Temperature Rain
Yesterday's FFMC
Yesterday's DMC
Yesterday's DC
Fuel moisture codes
Fine Fuel Moisture Code (FFMC)
Duff Moisture Code (DMC)
Drought Code (DC)
Initial Spread Index (ISI)
Buildup Index (BUI)
Structure of the Canadian Forest Fire Weather Index (FWI) System
Fire Behavior Indices
Fire Weather Index (FWI)
CFFDRS

Figure 4.1 *FWI computational model*

As depicted in (Figure 4.1), FWI is an index consisting of 5 different intermediate indices: Fine Fuel Moisture Code (FFMC), Duff Moisture Code (DMC), Drought Code (DC), Initial Spread Index (ISI) and Build-Up Index (BUI) (Ricotta et al.). The first three components estimate soil moisture content on three layers of the soil, while the last two estimate the potential heat that could be generated if a fire started at that moment. In particular, FWI is computed by taking in input 4 different weather forcings: 2 meter temperature, relative humidity, cumulative precipitation over the last 24 hours and 10 meter wind speed.

ML algorithms can be used to improve the computational efforts of deterministic FWI, by providing reliable synthetic FWI maps. Given daily input climatic drivers, they might be capable of capturing seasonal climatic trends and providing consistent FWI maps.

Common ML approaches to fire risk prediction can be divided into two macro categories: burned area extension estimation cases and FWI risk classification works. Accounting for the first case, (Omar et al., 2021) proposed a Long Short Term Memory algorithm (LSTM) in order to forecast the total burned area (in hectares, ha) in the region of Montesano Park by taking in input 12 different features. Among the features, they considered the FWI indices as well as other climate drivers.

(Safi et al., 2013) proposed a work very close to the previous one. As (Omar et al., 2021), they provided a Multi-Layer Perceptron (MLP) architecture that takes in input 12 features related to both FWI and weather conditions. The region of interest was Montesinho Park in this case.

With regards to FWI risk classification, works like (Kosović et al., 2021) and (Rosadi et al. 2022) are extremely meaningful. More in detail, in the first one several state-of-the-art ML algorithms are developed in order to classify FWI risk, such as Random Forest (RF), Logistic Regression (LR), MLP and others. They make use of a 8 feature vector of climate variables to predict FWI value and therefore its corresponding class. At last, (Rosadi et al. 2022) focused on the estimation of the burned area that could be burnt if nothing is done to prevent fire spread. The novelty of their approach is the use of 12 different climate drivers to develop a Fuzzy C-Means (FCM) algorithm.

Since FWI does not take into account neither vegetation distribution and indexes nor geomorphology and physical composition of the region, a step further in fire danger assessment could be the prediction of a Fire Danger Probability Map through ML approaches. A novel and cutting-edge approach relies on the use of historical fire ignition events to train a ML-based classifier that takes into account both climatic and geomorphological driver maps.

In this chapter, a ML approach to generate daily FWI risk maps is briefly presented. This work has been co-developed by Davide Donno and presented in his Master of Science Degree thesis in Computer Engineering at the Department of Engineering and Innovation in the University of Salento.

This approach involves the use of a Generative Neural Network algorithm and the daily weather acquisitions of 2m temperature, 10m wind speed, relative humidity and total precipitation. A Pix2Pix Conditional Generative Adversarial Network (cGAN) has been developed for the aforementioned task.

4.1 Data Sources and Pre-Processing

The dataset needed for the development of the use case involved the download of the following variables:

- Fire Weather Index risk maps;
- Temperature at 2 meters above the surface;
- Relative Humidity;
- Total Precipitation in 24 hours;
- Wind Speed component at 10 meters above the surface.

All these data are made available by the Copernicus service (About Copernicus). For this task, temperature, relative humidity, cumulative rain and wind speed are taken as input to the ML algorithm, whereas FWI is the output. It is fundamental that both the input maps and output maps match on the spatial domain in terms of resolution and grid.

The considered spatial domain is the EURO-CORDEX domain which spans from -48.5°E to 69.75°E in longitude and from 73.9°N to 20.15°N in latitude.

In particular, the dataset has been downloaded from three different sources: *ERA5 single levels dataset*, 2021, *ERA5 pressure levels dataset*, 2021 for input drivers (temperature, cumulative rain, relative humidity and wind speed), and *Fire Danger Indices Dataset*, 2021 for the output (FWI risk maps).

In order to correct a mismatching temporal resolution between hourly input climate variables and the daily fire hazard maps, an intermediate pre-processing phase has been required. In particular, for each day, the average map that holds the entire information of the day is computed, except for the precipitation whose hourly values had to be summed instead. In addition, downloaded dataset has a temporal coverage equal of 42 years, from January 3*rd*, 1979 to December 31*st*, 2020.

Moreover, it is noticeable that the data concerning each variable are available through NetCDF4 files in the form of a masked array, meaning that it has an associated mask along the data. As a result, since FWI maps are not significant along the sea, all the data along it is masked with a sentinel value.

In the ERA5 single levels dataset, 2021, there is no reference to wind speed download. In particular, it is relevant to evaluate just the wind speed, regardless of its direction. In particular, 10 meters u and v components have been used for this purpose. They are the horizontal wind-speed that flows at 10 meters to the East and the horizontal wind-speed that flows at 10 meters to the North, respectively. The two components are related to wind speed, $\vec{V}$, as follows:

$$u=-\left|\vec{V}\right| \vee \sin(\phi)$$

$$v=-\left|\vec{V}\right| \vee \cos(\phi)$$

Where ϕ is the wind direction. Since ERA5 provides just u and v components, wind magnitude can be computed as:

$$\left|\vec{V}\right|=\sqrt{v^2+u^2}$$

After the download of the maps of u and v wind components, the two maps can be merged together according to the previous equation, obtaining the component of wind speed pixel-wise for each day.

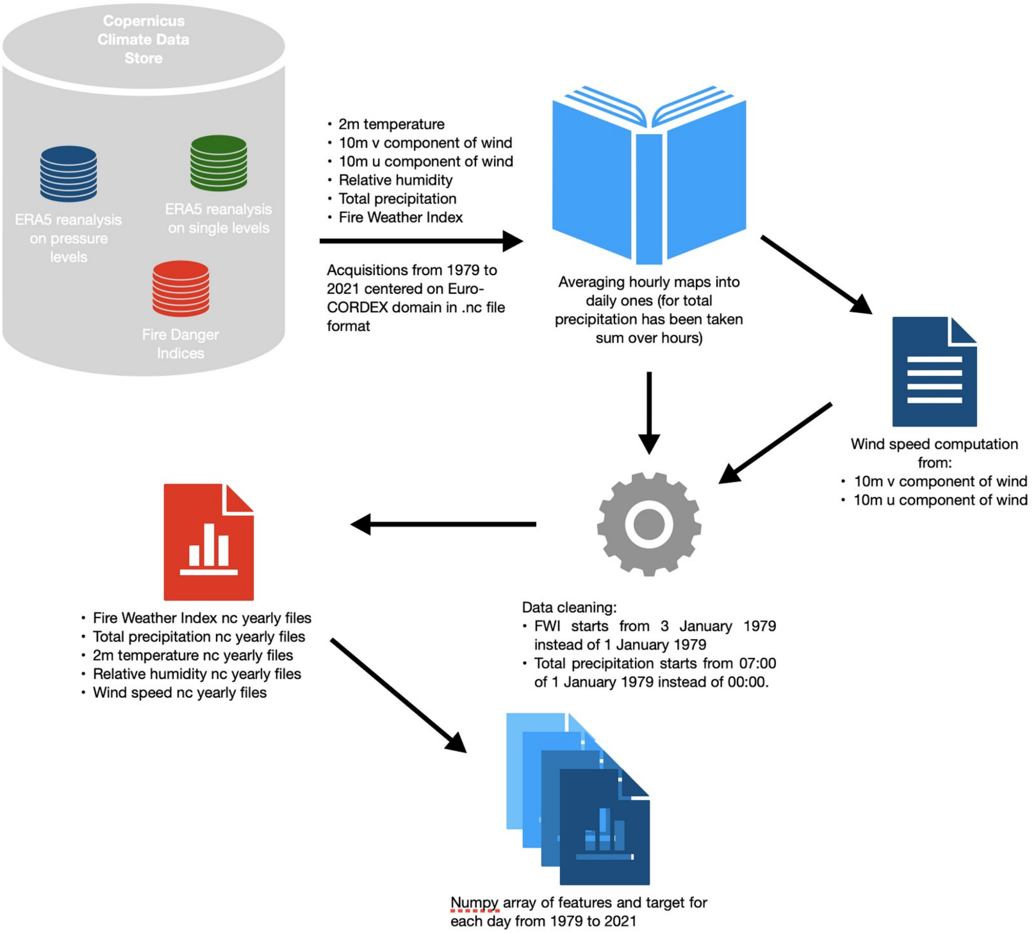

Copernicus Climate Data Store
ERA5 reanalysis on pressure levels
ERA5 reanalysis on single levels
Fire Danger Indices
• 2m temperature
• 10m v component of wind
• 10m u component of wind
• Relative humidity
• Total precipitation
• Fire Weather Index
Acquisitions from 1979 to 2021 centered on Euro-CORDEX domain in .nc file format
Averaging hourly maps into daily ones (for total precipitation has been taken sum over hours)
Wind speed computation from:
• 10m v component of wind
• 10m u component of wind
Data cleaning:
• FWI starts from 3 January 1979 instead of 1 January 1979
• Total precipitation starts from 07:00 of 1 January 1979 instead of 00:00.
• Fire Weather Index nc yearly files
• Total precipitation nc yearly files
• 2m temperature nc yearly files
• Relative humidity nc yearly files
• Wind speed nc yearly files
Numpy array of features and target for each day from 1979 to 2021

Figure 4.2 *Download and processing pipeline of the used dataset. Starting from the download of raw data, they pass through a series of processing steps which lead to final file-arrays of daily features and ground truth.*

Figure 4.4 shows the entire data pre-processing chain to obtain the resulting dataset constituted of daily samples stored according to the procedure explained in the following paragraph.

The downloaded dataset has been divided into 42 years, in a temporal window that spans from 1979 to 2020. Each year contains 365 or 366 FWI maps, as well as weather observations, (except for 1979 which contains just 363 objects).

Moreover, these years have been divided into 4 decades: [1979-1988], [1989-1998], [1999-2008], [2009-2018] with 2 spurious years: 2019 and 2020. Then, the following partitioning strategy has been adopted: it has been taken each median year in each decade for both validation and testing partitions, in addition to the two remaining spurious years: 2019 as well as 2020. By choice, it has taken the floor of the decade-median for validation on one side, and the ceiling of the decade-median for testing, on the other side. This means that 1983, 1993, 2003, 2013, 2019 belong to the validation set, whilst 1984, 1994, 2004, 2014 and 2020 belong to the test set, instead. Hence, the remaining 32 years belong to the training set.

Furthermore, during each training phase's epoch of the model, each year has been shuffled and then years are concatenated in a chronological order. The reason behind this partitioning choice is to ensure the physical consistency of the entire training procedure. Having all the observations across a certain year in the same dataset and possibly in the same batch, ensures to catch the seasonality of FWI risk, leading to a balance in the samples contained in all three sets. Furthermore, weather observation as well as fire weather index risk both change over the time, meaning that each decade represents a different meteorological situation. Hence, having median years for cross-validation and testing ensures to balance all the sets, sampling the 42 years span equally.

4.2 Machine Learning Algorithm

This chapter describes the proposed machine learning approach developed. As previously said, the aim is to find out a neural network design and implementation in order to generate synthetic FWI maps by taking in input 4 drivers used for FWI deterministic computation (temperature, relative humidity, total precipitation and wind speed).

4.2.1 Neural Network Design

The Neural Network Algorithm used for this thesis use case is a Pix2Pix CGAN (Isola et al., 2016). As its name suggests, it is a Conditional Generative Adversarial Network, meaning that it learns conditionally on the target supplied to both the generator and discriminator (Mirza et al., 2014). CGANs learn how to generate data based on additional information given by ground truth data, which can be either a class label or an image (Mirza et al., 2014). In particular, the latter is the case that has been developed in this work.

Therefore, Pix2Pix learns how to translate an input image to an output image. It has a huge variety of different applications (Isola et al., 2016). For instance, they span from map to aerial photo conversion as well as photo colorization to day-night inversion.

Dataset features for the present task are a stack of maps containing temperature, relative humidity, cumulative rain and wind speed, while ground truth is a FWI risk map. Weather acquisitions are fed to the network along FWI targets and Pix2Pix learns how to generate corresponding FWI maps. In the following, it will be referred to the climate variables as features and fire maps as target, respectively.

Pix2Pix CGAN is composed of a U-Net-shaped (Ronneberger et al., 2015) generator characterized by skip connections and a PatchGAN (Isola et al., 2016) discriminator suited for real-fake classification, as well. Moreover, in (Isola et al., 2016) they have proved that PatchGAN is able to determine whether a NxN patch is real or not with a good probability, even if N is much smaller than the original size of the input matrix.

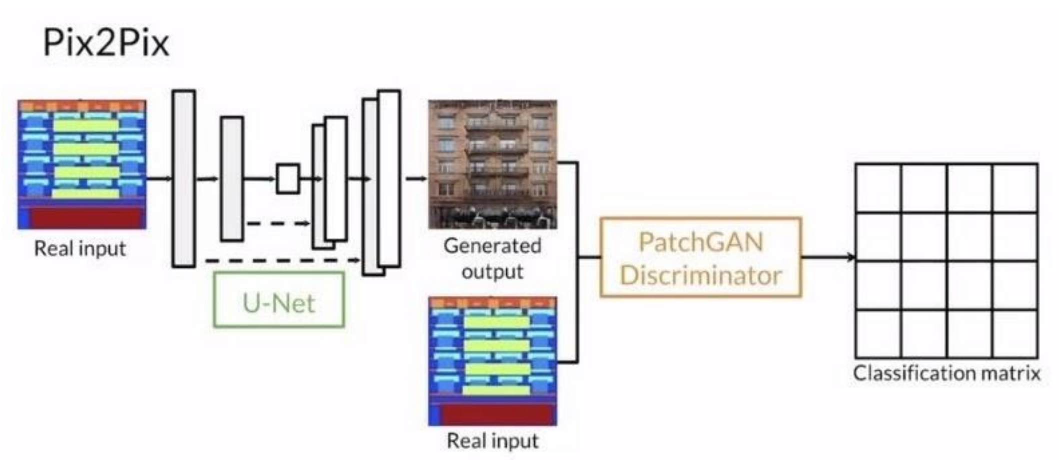


Figure 4.3 *Pix2Pix architecture. On the left, a U-Net-shaped generator creates fake samples. On the right, a PatchGAN discriminator classifies the generated output.*

Regarding the loss function, it is required to mention both generator and discriminator loss functions. On one side, generator loss is a combination of two losses: the first one is a L1 loss ($L1_{loss}$) (L1 and L2 functions, 2019) which is a mean absolute error (MAE, 2016) between the ground truth image and the predicted one, whilst the second is a sigmoid cross entropy loss (Sigmoid, 2019) (SCE_{loss}) between the patch obtained by the generated target and an array of ones of the same size of the patch itself (Isola et al., 2016). Therefore, the generator total loss (GEN_{loss}) is given by the following formula:

$$GEN_{loss} = SCE_{loss} + \lambda L1_{loss}$$

Where λ is an experimentally determined hyper-parameter which has been set to 100 by the authors of the paper.

On the other side, discriminator loss is the addition of two sigmoid cross-entropy losses (Isola et al., 2016).

- $DISC_{real-loss}$ is the loss between the NxN patch on the real target image and an array of ones.

- $DISC_{gen-loss}$ is the loss between the patch on the predicted output and an array of ones.

Hence, the discriminator loss ($DISC_{loss}$) is:

$$DISC_{loss} = DISC_{real-loss} + DISC_{gen-loss}$$

Finally, in order to check whether or not the training is going well and both generator and discriminator are correctly working (i.e. none of them is better than the other) $DISC_{loss}$, as well as SCE_{loss}, should be about $\log(2) = 0.69$ (Pix2pix Implementation, 2022). In this way, there is almost the same probability that the discriminator is deceived by the generator in classifying a generated target map.

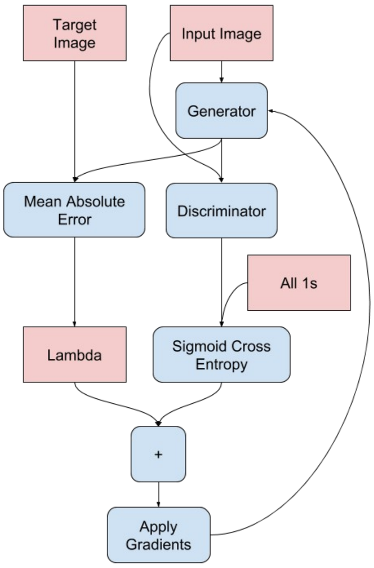


Figure 4.4 *Pix2Pix generator training step and gradient update.*

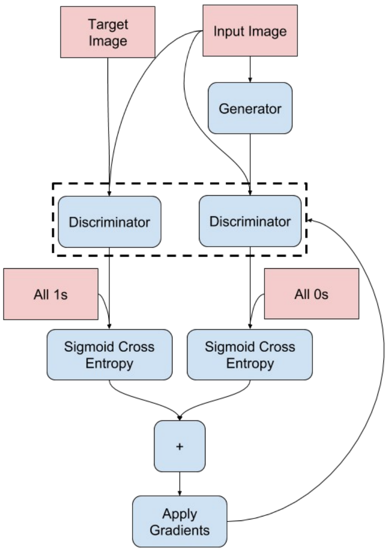


Figure 4.5 *Pix2Pix discriminator training step and gradient update.*

A customized version of Pix2Pix architecture was designed and implemented. Even though the general skeleton of the network is still the same, some tricks have been taken into account. First of all, the original network makes use of a Sigmoid activation

(Activation Functions Explained, 2017) on the output layer, which squashes the values of the network in the range $[0,1]$ in a non-linear way. Instead, since FWI can range into an upper unbounded interval which has a minimum equal to 0, it is more suitable to use a ReLU activation function (Activation Functions Explained, 2017).

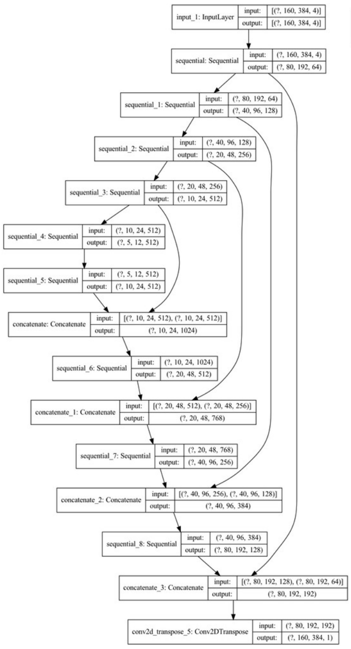


Figure 4.6 *the generator structure*

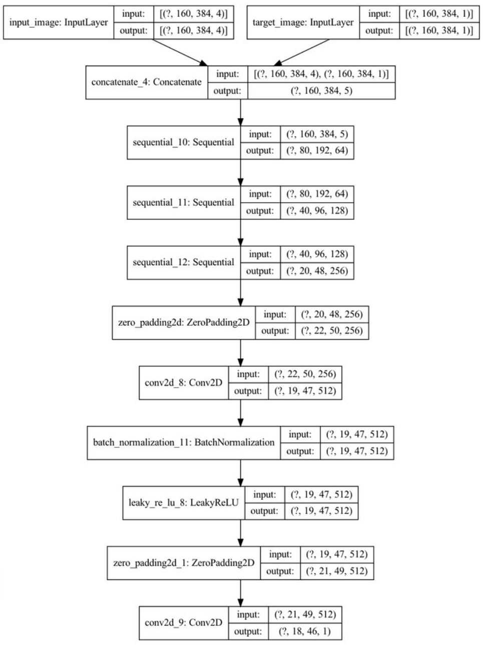


Figure 4.7 *the discriminator layers displacement*

Furthermore, as shown on Figure 4.6, the number of layers has been decreased in the UNet-shaped generator, making it with a less number of parameters.

4.2.2 Physics Informed Machine Learning

These are not the only updates to the neural network, some constraints have been taken into account in order to make the experiments physically consistent. First of all, intrinsic consistency of the experiment has been guaranteed by the former train, validation and test split, as specified before (paragraph 4.1.4). Hence, similarly to what

(Manepalli et al., 2019), did in their work, the sea-surface mask was taken in order to enhance the physical consistency of the results.

More in detail, sea mask is an array consisting of 0 values in the land and 1 values along the sea. Since fire risk on the sea surface is obviously null, there is the need to focus on the meaning of a null value of the index. In particular, a null risk of fire is different between land and sea, as well. Respectively, the first one indicates that the climate variables setup provide no risk of wildfire injection, while the second one shows the impossibility for the fire to build up and spread in the sea. Hence, in order to incorporate this information into the model, a sentinel value has been chosen for fire weather index along the sea. Several tests have been carried out and the best value is -1.

In conclusion, each FWI output map has been updated with -1 values along the sea, before feeding it to the neural network.

4.2.3 Data Augmentation

A crucial step of the entire workflow is the data augmentation applied on both training and validation sets. Specifically, the first data transformation has been a preliminary resize of the images, in order to let them flow along the network. The original size was 215×474, whereas Pix2Pix model has been designed to accept as inputs maps of size 160×384. Since features exhibit very different scales of values (e.g., the temperature at 2 m registered values greater than 250 °K, while the total precipitation values, expressed in meters, resulted in very small values) a Min Max Scaler has been used to re-scale both features and ground truth images to the range $[0,1]$. However, as it will be more detailed in the following paragraph, target scaling may also be discarded, since FWI value is unbounded.

Since feature maps contain weather parameters, it is not worth applying color transformations (i.e., those involving brightness, hue, saturation and so on) because they would change the values of those climatic variables and would compromise the effectiveness of the prediction, as well as its consistency. Instead, it is much more useful to augment images from a geometrical perspective. In fact, random flips on both the x or y axis have been performed. This procedure would increase generator ability in identifying land shapes and in separating them from the sea (where fire risk value can be only null, according to what has been said in 4.2.2), as well as in increasing the number of different samples. Furthermore, since target files contain zero values of the fire risk along the sea, in accordance to paragraph 4.2.2, each target map has been updated before feeding it to the neural network according to the following equation:

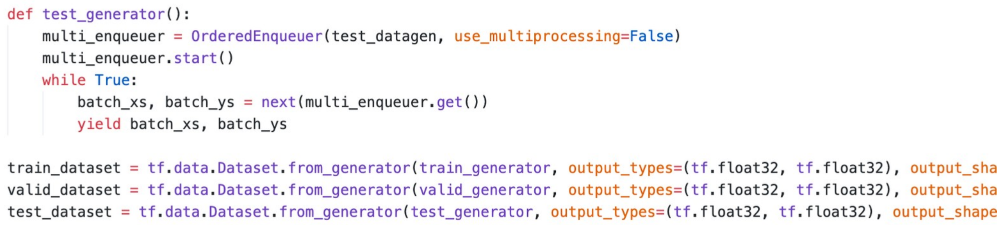
$$Y = Y + k * M$$

where Y is the target image, k is the sentinel value of fire risk in the sea and M is the binary mask. In particular, for the experiments, k has been set to -1.

Two main data augmentation techniques have been adopted. The first one consists of a random flip on x or y axis according to a probability parameter. The second one is the addition of the flipped features along the real ones to the training batch. Moreover, according to the value of a binary mask, it is possible to select which flipped version must be added to the batch: flip along x, along y or along both axes. From now on, the second augmentation technique will be referred to as full.

4.2.3 Data Generators and Distributed Environment

Data generators are the main tool used for the implementation of the dataset processing pipeline. As already described, the dataset has been divided into train, validation and test sets, according to the considerations described in Section 4.1.4. There is no simple way to implement this scenario with standard Tensorflow Dataset approaches. As a result, a data-generator has been implemented in order to load data batches on-the-fly in memory, process them and then discard the samples. Every data generator is a subclass of the *keras.utils.Sequence* class. However, in order to make this object as an *Iterable*, it must be embedded into an *OrderedEnqueuer* (Ordered Enqueuer doc, 2022).



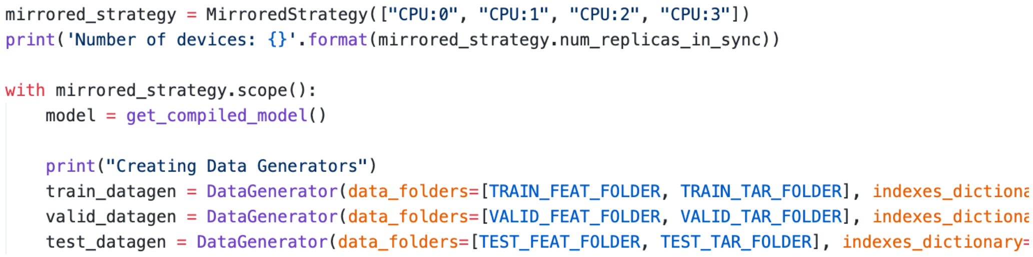
```
def test_generator():
    multi_enqueuer = OrderedEnqueuer(test_datagen, use_multiprocessing=False)
    multi_enqueuer.start()
    while True:
        batch_xs, batch_ys = next(multi_enqueuer.get())
        yield batch_xs, batch_ys

train_dataset = tf.data.Dataset.from_generator(train_generator, output_types=(tf.float32, tf.float32), output_sha
valid_dataset = tf.data.Dataset.from_generator(valid_generator, output_types=(tf.float32, tf.float32), output_sha
test_dataset = tf.data.Dataset.from_generator(test_generator, output_types=(tf.float32, tf.float32), output_shape
```

Figure 4.8 *Data generator iterable with an OrderedEnqueuer and tf.data.Dataset implementation.*

Figure 4.8 shows the implementation of an *OrderedEnqueuer* used to create a *tf.data.Dataset* (TF Dataset doc, 2022) from a generator. In this way, the researcher may have a fine control on every single aspect of the training phase, from the batch loading to the data feeding to the network.

During the development of the model, training routines have been implemented and run on the M100-Cineca supercomputer. As already stated in Appendix, the user account provided on M100 lets researchers run programs with the help of 4 Volta100 GPUs. However, it is necessary to set up the distributed processing pipeline through the Tensorflow Python library. This framework makes available to users several distributed training routines, each of which have their meaning and usage. In particular, in this case it has been widely used the *Mirrored Strategy* (Mirrored Strategy doc, 2022).



```
mirrored_strategy = MirroredStrategy(["CPU:0", "CPU:1", "CPU:2", "CPU:3"])
print('Number of devices: {}'.format(mirrored_strategy.num_replicas_in_sync))

with mirrored_strategy.scope():
    model = get_compiled_model()

    print("Creating Data Generators")
    train_datagen = DataGenerator(data_folders=[TRAIN_FEAT_FOLDER, TRAIN_TAR_FOLDER], indexes_dictiona
    valid_datagen = DataGenerator(data_folders=[VALID_FEAT_FOLDER, VALID_TAR_FOLDER], indexes_dictiona
    test_datagen = DataGenerator(data_folders=[TEST_FEAT_FOLDER, TEST_TAR_FOLDER], indexes_dictionary=
```

Figure 4.9 *Example of variable definitions into the Mirrored Strategy scope.*

As shown in (Figure 4.9), the Mirrored Strategy definition comes with an optional parameter, the CPUs or GPUs defined by the strategy. If the parameter is not provided, then the framework determines itself the maximum number of available processing units. All the replica-variables must be defined into the scope of the strategy. These kinds of variables are copied onto all the distributed devices, as well as they are kept synchronized at each step of the algorithm. Moreover, it has been chosen to define into the scope the data generators, too. In this way each generator is copied on each GPU (or CPU) in order to correctly distribute the dataset. As a result, the training batch size (also called global batch size) of the data is divided among each Processing Unit (PU) (called local batch size). For instance, if a global batch size provided is 256 and there are 4 GPUs, each GPU holds a local batch size of 64 samples.

As explained by the documentation (Mirrored Strategy doc, 2022) about distribution strategies, it is important to replicate the model along each unit, in order to keep it synchronized. As a matter of fact, at the beginning of each epoch, every PU stores in its memory its own model and processes batches on it. At the end of the epoch, PUs merge their local copies through an algorithm of all-reduce.

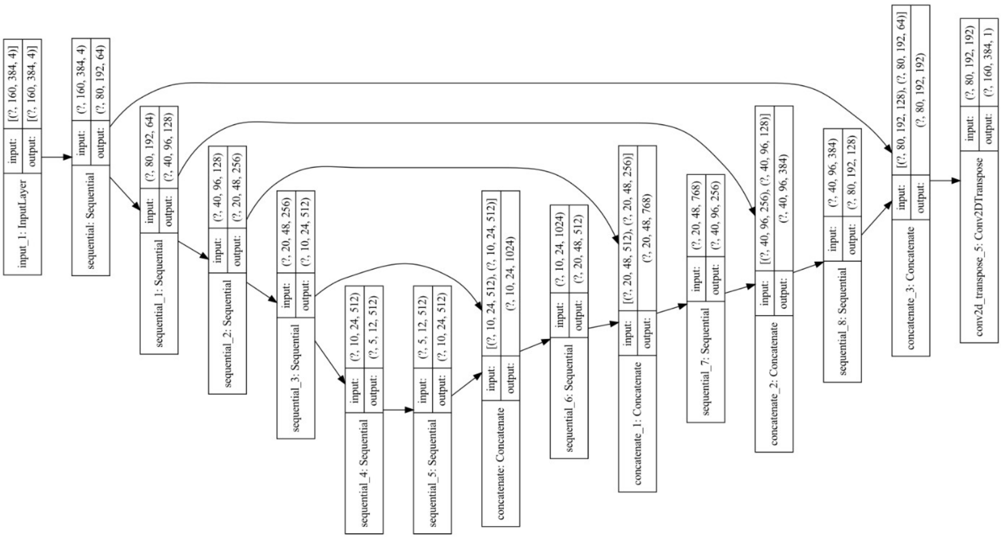
4.3 Results

The following section will describe the training procedure and will present the results from both a qualitative and a quantitative point of view.

4.3.1 Models Training

The training procedure of the model underwent many different steps. The first one was aimed at finding out a training guideline by setting up several hyperparameter configurations and trying out the training on a few epochs. It is worth mentioning that deep neural networks like Pix2Pix cGAN must be trained by making little forward steps. As a result, the learning rate of the network has been fixed to a little value: 0.0001 . Furthermore, as loss function it was chosen the same described in (Isola et al., 2016), whereas the chosen gradient optimizer was Adam, with a $\beta_1 = 0.5$. The filter-sizes of the U-Net Generator are detailed in Figure 4.10.

Figure 4.10 *U-Net Generator of the Deep Pix2Pix model.*



Hence, the resulting Generator contains more than 27 million parameters, whereas the Discriminator has just 2.7 million.

Since temperature is the main driver that increases FWI, and this variable could theoretically assume infinite values, FWI values are unbounded too. An important question that has been addressed is the eventual need for target scaling. As a matter of fact, four main approaches have been carried on:

1) Without target scaling and without data augmentation techniques: in this case, the activation function for the last layer is a PReLU;
2) Without target scaling and with data augmentation: also in this case, the activation function for the last layer activation is a PReLU;
3) With target scaling and with data augmentation: a ReLU activation function has been considered for the last layer;
4) With target scaling, without data augmentation techniques: a ReLU is the last-layer activation function.

All the four training procedures have been developed on 30 epochs, with a batch size of 128 or 64 elements for implementations without and with full data augmentation, respectively. In addition, other variants of the same approaches have been tested out, changing the last activation function. As shown in Figure 4.11, all the models which do not use the target scaling predict more blurred samples if compared to those produced by models which use target scaling.

In addition to these four models, two other training procedures have been added. The former has a linear last-layer activation function and involves target scaling, as well as data augmentation. The latter has a sigmoid activation, instead.

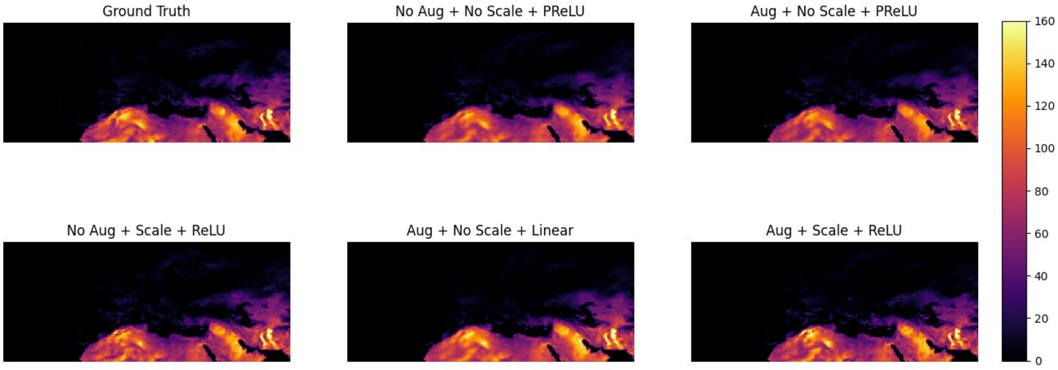


Figure 4.11 *Prediction comparison between several network configurations.*

Conversely, those models who scale target variables are subject to an implicit constraint: the maximum possible value of the trainable fire weather index is bounded to the maximum value observed across the time series. As results show, the guiding line for the training procedure is dictated by target MinMax scaling as well as the data augmentation with the addition of augmented samples along real FWI target maps.

After having chosen the best initial model, the next step consisted in training the deep model on a much higher number of epochs. Therefore, the model has been trained over an initial baseline of 120 epochs, with full online data augmentation. The results provided by this model are quite impressive since, as shown in (Figure 4.12), all the main characteristics of the images are correctly captured.

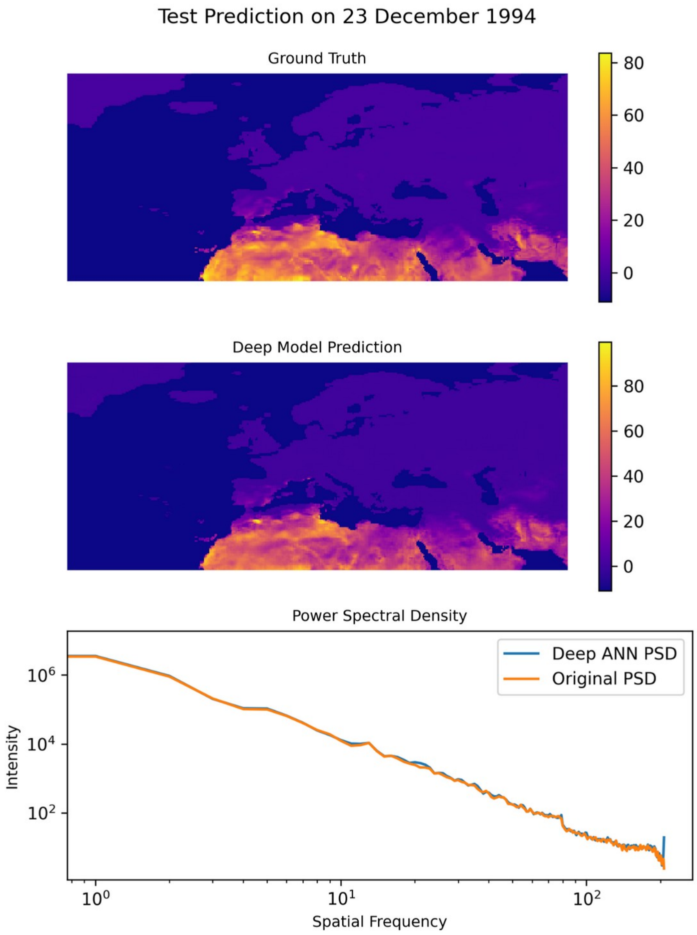
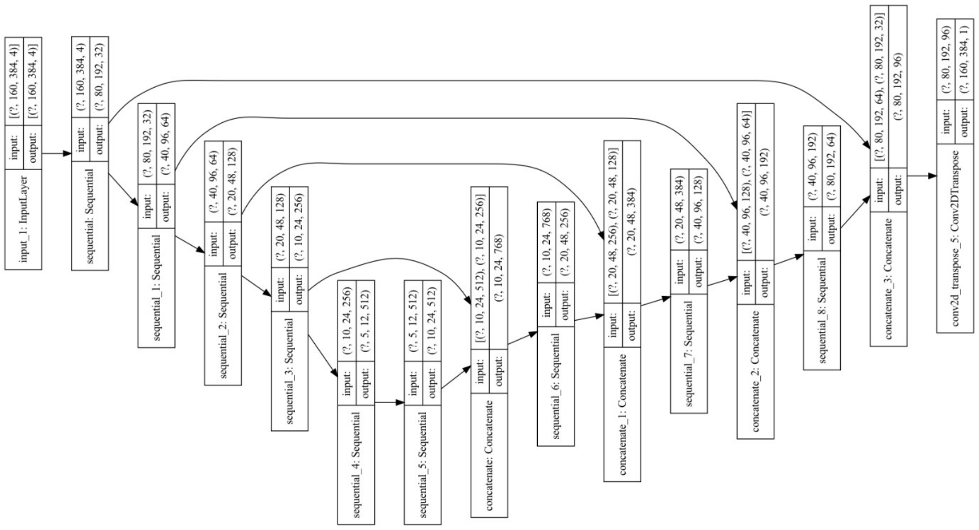


Figure 4.12 *Comparison between ground truth (first image) and generated sample (second image) on a test set sample taken recording 23 December 1994 FWI risk. Third image shows the comparison between ground truth and predicted PSDs.*

Moreover, the Power Spectral Density (PSD) of both the target and predicted FWI maps are very similar, capturing details over all the spatial frequencies.

However, another important model update has been performed. Thus, a less deep model has been designed and implemented according to (Figure 4.13), which details the filter-sizes of the U-Net Encoder.

Figure 4.13 *U-Net Generator of the Light Pix2Pix model.*



This less deep model contains just 11 million parameters on the Generator. Moreover, it has been trained over 90 epochs following the same hyperparameters configuration of the best model obtained according to the previously explained procedure. This means that the model was trained over 90 epochs by using a full data augmentation with a global batch size of 64 elements and by applying MinMax scaling in $[0,1]$ on the target. Hence, the training of this light version of Pix2Pix has produced the following results shown in (Figure 4.14).

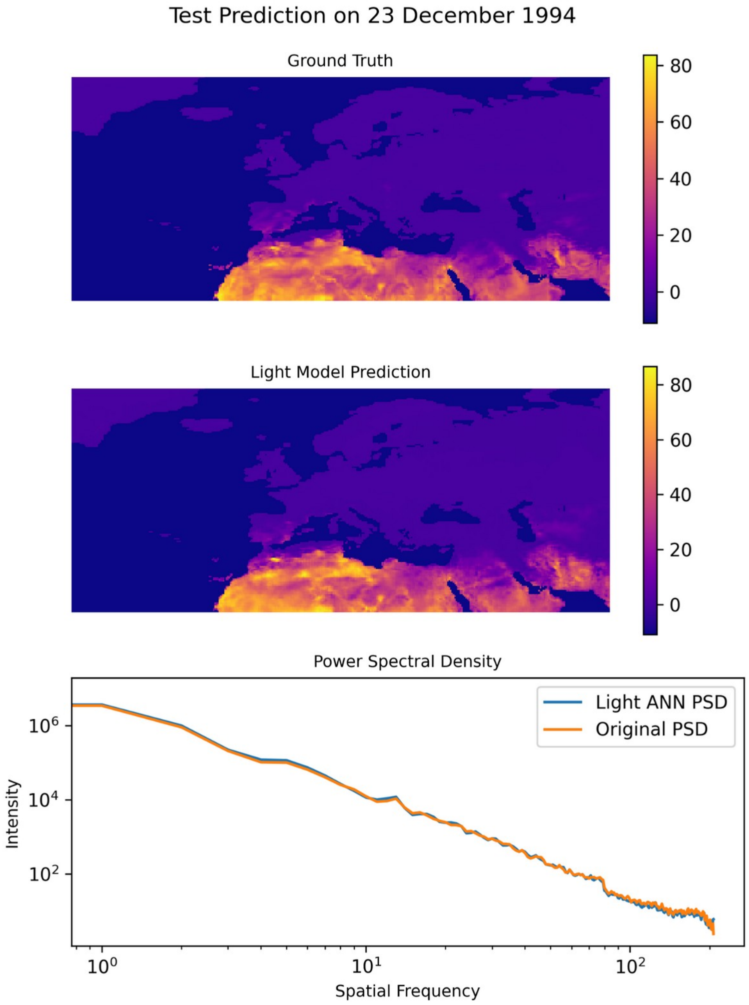


Figure 4.14 *Ground truth (23 december 1994 test sample) and predicted FWI comparison (first and second image, respectively) on the lighter GAN model. On the bottom image the comparison between ground truth PSD and predicted PSD is represented.*

From a qualitative point of view, the lighter model correctly captures all the spatial frequencies and generates accurate FWI maps. However, the following paragraph will describe more in detail all the metrics evaluated on these models and the quantitative results.

4.3.2 Models Comparison

Both the deep and the light models produce accurate FWI maps, capturing all the spatial information of the index. From now on, the deep model will be referred to as *DeepP2P*, whilst the light model as *LightP2P*. However, the deeper model is difficult to be handled, leading to more time for the generation of a forecast. On the other hand, the lighter version of the model is faster in its computations.

As a result, on the same machine, the total time spent by the DeepP2P in forecasting 1000 FWI maps is 162.61 seconds, with an average time spent for a single instance of 0.162 seconds. LightP2P model, instead, produces 1000 FWI maps in 74.85 seconds, with an average of 0.074 seconds. Even though both running times are extremely low, the lighter model performs faster than the deeper one.

4.3.3 Training Curves

A critical point of the training is the evaluation of loss functions training and validation curves. As already discussed in (Section 4.2.1), the Pix2Pix model contains 3 different loss functions: the Discriminator Loss, the Generator Sigmoid Cross Entropy (SCE) Loss and the Generator L1 Loss. As reported in (Pix2pix Implementation, 2022), it is important to minimize the L1 loss of the generator in order to check the correctness of the training procedure. While, in contrast, both the Discriminator Loss and Generator SCE Loss should settle around $\ln(2) = 0.69$. In this way, it is guaranteed that neither the generator nor the discriminator overwhelms the other and a balance in generation-discrimination of samples is established.

The DeepP2P model has the following training and validation curves:

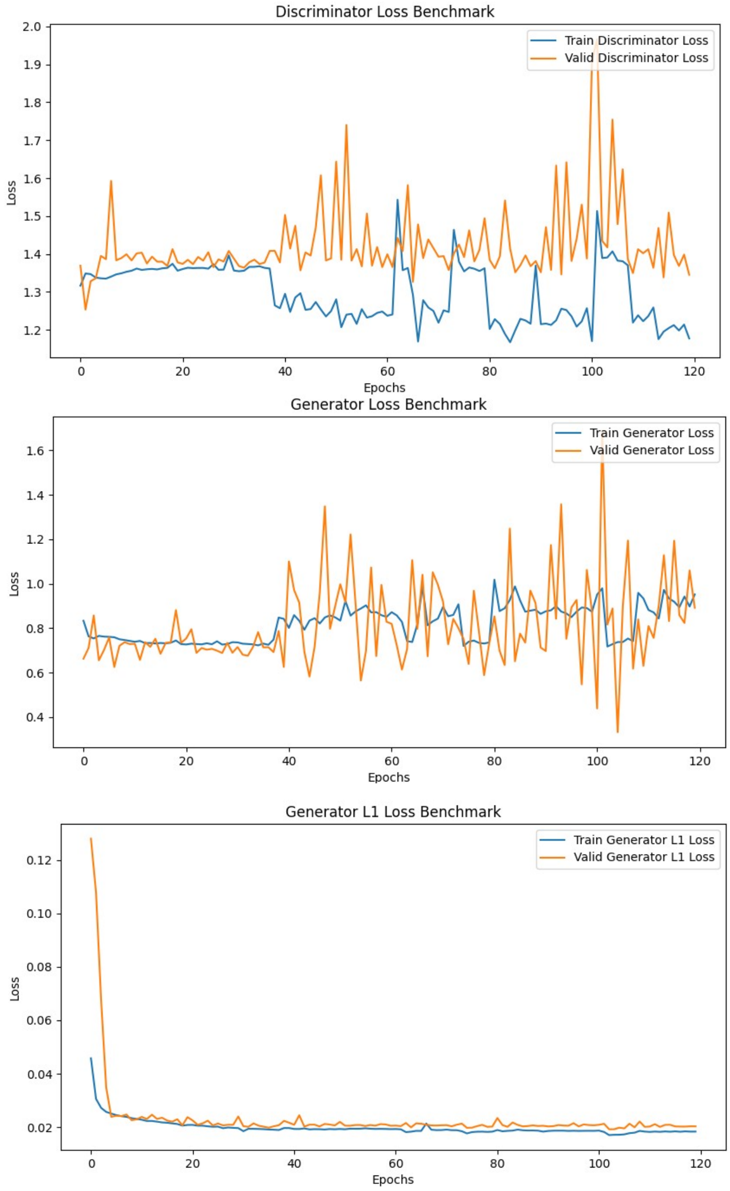


Figure 4.15 *DeepP2P loss curves. In order of appearance: training and validation curves for discriminator, generator and L1 losses.*

As shown in Figure 4.15, the training discriminator loss settles around an average value of 1.28, while the generator one lies around a value of 0.9. However, even though these losses do not resemble the ones explained in (Isola et al., 2016), the L1 loss almost perfectly follows the expected training and validation curves.

On the other hand, the LightP2P model produces the following curves:

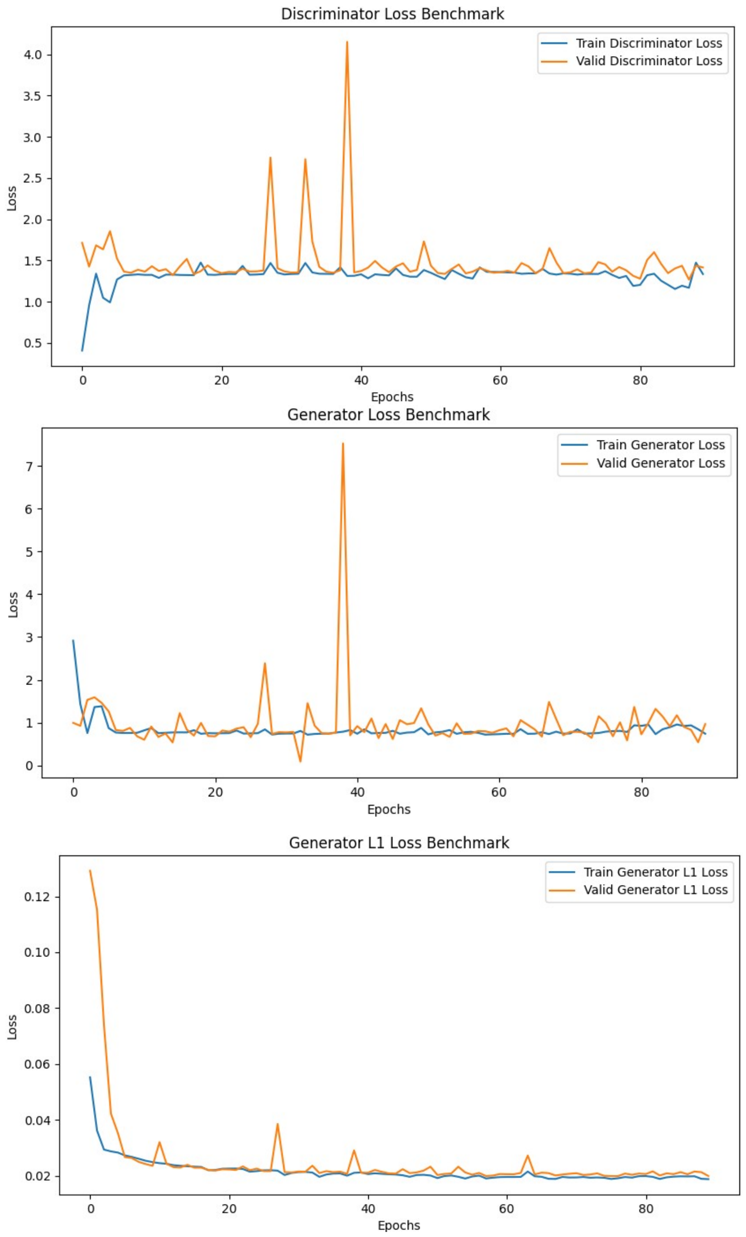


Figure 4.16 *LightP2P loss curves. In order of appearance: training and validation curves for discriminator, generator and L1 losses.*

In this case, both discriminator and generator (SCE) losses have some spikes in the validation curves. Discriminator loss settles around a value of 1.2, exactly as in the DeepP2P case, whilst the generator loss varies around 0.9. However, as in the DeepP2P case, the generator L1 loss curves constantly decrease up to a value of 0.02, following the correct behavior for the generation of fake samples.

4.3.3 Fire Weather Index Map Generation

In order to better compare the results obtained from the two ML approaches, it is suitable to make average predictions over a specified temporal window. As already mentioned, the Test set contains the daily observations of the 4 drivers and of the FWI maps over the following years: 1984, 1994, 2004, 2014 and 2020. Hence, the dataset has been divided into the four seasons accordingly:

- December - January - February (DJF);
- March - April - May (MAM);
- June - July - August (JJA);
- September - October - November (SON);

In particular, each of the above datasets contains inter-annual observations. Furthermore, for each dataset, it has been computed the predicted FWI daily map for both the DeepP2P and LightP2P, taking the average over the number of samples. In addition, another average map has been taken over the entire 5 years dataset. At the end of the procedure, 3 average maps for each season and 3 average maps for the entire test dataset were obtained: FWI ground truth average map, FWI LightP2P predicted average map, FWI DeepP2P predicted average map.

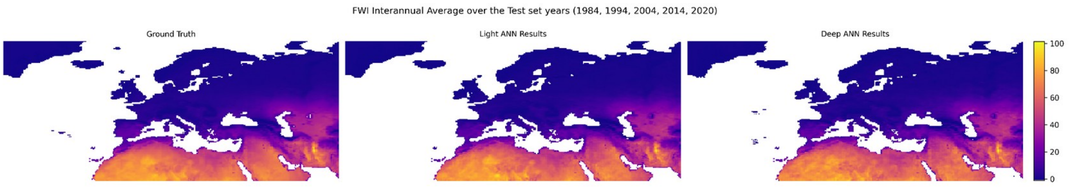

FWI Interannual Average over the Test set years (1984, 1994, 2004, 2014, 2020)
Ground Truth
Light ANN Results
Deep ANN Results
100
80
60
40
20
0

Figure 4.17 *FWI Interannual Average over Test set years (1984, 1994, 2004, 2014, 2020). Ground Truth (left image) compared to LightP2P average prediction (center image) and DeepP2P average prediction (right image).*

As shown in (Figure 4.17), from a first visual investigation, average predicted maps over the entire 5 years Test Dataset are almost identical with each other. On average, both the Artificial Neural Networks (ANN) produce FWI predictions very similar to the original one and all the main trends and features of the FWI maps are correctly captured from a qualitative point of view.

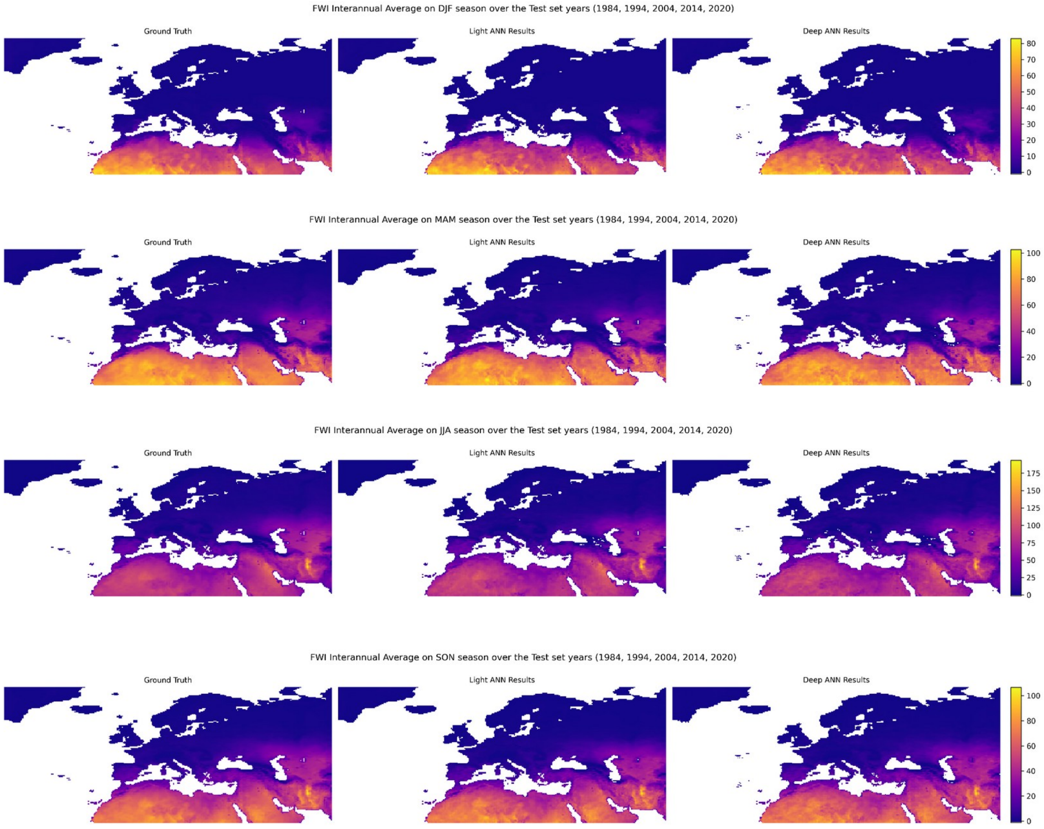

FWI Interannual Average on DJF season over the Test set years (1984, 1994, 2004, 2014, 2020)
Ground Truth
Light ANN Results
Deep ANN Results
80
70
60
50
40
30
20
10
0
FWI Interannual Average on MAM season over the Test set years (1984, 1994, 2004, 2014, 2020)
Ground Truth
Light ANN Results
Deep ANN Results
100
80
60
40
20
0
FWI Interannual Average on JJA season over the Test set years (1984, 1994, 2004, 2014, 2020)
Ground Truth
Light ANN Results
Deep ANN Results
175
150
125
100
75
50
25
0
FWI Interannual Average on SON season over the Test set years (1984, 1994, 2004, 2014, 2020)
Ground Truth
Light ANN Results
Deep ANN Results
100
80
60
40
20
0

Figure 4.18 *DJF (first row), MAM (second row), JJA (third row), SON (fourth row) FWI Interannual Averages over Test set years (1984, 1994, 2004, 2014, 2020). Ground Truth (left images) compared to LightP2P average prediction (center images) and DeepP2P average prediction (right images)*

As depicted in Figure 4.18, the resulting predictions averaged over each season represent correctly the main characteristics of the original FWI target map, for both the DeepP2P and LightP2P. This means that, even though a certain generated daily FWI map may be different from the original, both the two ANNs are capable of forecasting future fire risk by taking in input the 4 drivers.

4.3.4 Fire Weather Index Classification

Another key aspect related to the Fire Weather Index is the classification of the risk. In order to timely react to a possible fire ignition, it is not necessary to determine the specific real value of FWI risk, since the predicted value could float around the real one. Hence, it is much more useful to understand the category of risk of fire spread in a specific region of interest. As the Climate ADAPT documentation (FWI Classification, Climate Adapt) about FWI states, the fire risk can be classified according to its specific value into the following categories:

- Very Low: $0 \leq fwi \leq 5.2$;
- Low : $5.2 < fwi \leq 11.2$;
- Moderate : $11.2 < fwi \leq 21.3$;
- High : $21.3 < fwi \leq 38.0$;
- Very High : $38.0 < fwi \leq 50.0$;
- Extreme : $fwi > 50.0$.

Moreover, another class has been artificially added to the aforementioned ones – the Water class – in order to differentiate the land from the sea. Indeed, all pixels belonging to the sea need to be marked as Water. Both Water and Very Low classes express the near-impossibility of wildfire to build-up and spread. However, they are two physically different situations. The first one is the real impossibility for fire ignition, since there is no fuel along the sea in standard conditions. The second one is the very low probability of wildfire spread due to the high moisture levels in the vegetation.

Furthermore, in this case, it is not so important that the predicted image looks like the original one for each pixel since it is much more useful having a generated map where each pixel belongs to the same fire danger class indicated in the original map. Consequently, a Python script has been developed in order to convert both target and generated maps into classified ones. The script has been applied to interannual average maps, in order to obtain a risk classification of each map.

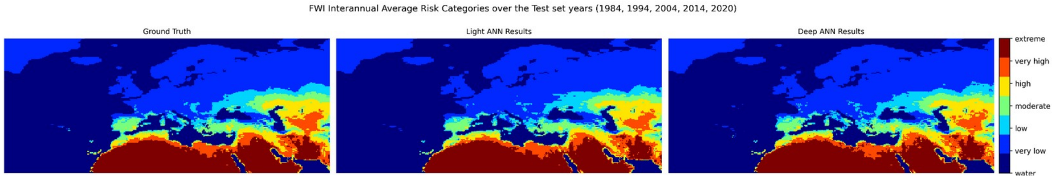

FWI Interannual Average Risk Categories over the Test set years (1984, 1994, 2004, 2014, 2020)
Ground Truth
Light ANN Results
Deep ANN Results
extreme
very high
high
moderate
low
very low
water

Figure 4.19 *FWI Interannual Average Risk Categories over Test set years (1984, 1994, 2004, 2014, 2020). Ground Truth (left image) compared to LightP2P average prediction (center image) and DeepP2P average prediction (right image).*

The resulting classified maps are depicted in Figure 4.19. Both the two networks succeed in correctly classifying the main average risk categories over the entire European territory. However, both of them misclassify some features along the south-eastern side of the map, even though the main baseline of the risk is correctly caught. Moreover, DeepP2P provides much more jagged contours along Africa’s risk categorization. By the way, both the two approaches provide excellent results over the average inter-annual datasets.

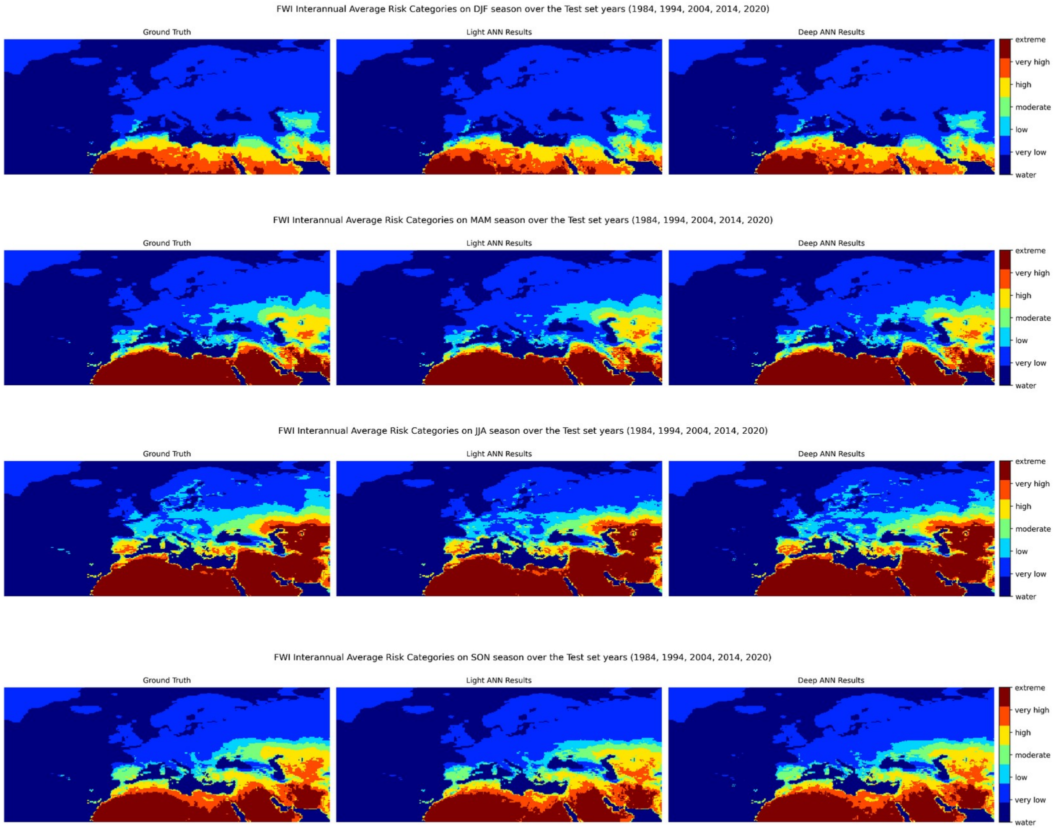

FWI Interannual Average Risk Categories on DJF season over the Test set years (1984, 1994, 2004, 2014, 2020)
Ground Truth
Light ANN Results
Deep ANN Results
extreme
very high
high
moderate
low
very low
water
FWI Interannual Average Risk Categories on MAM season over the Test set years (1984, 1994, 2004, 2014, 2020)
Ground Truth
Light ANN Results
Deep ANN Results
extreme
very high
high
moderate
low
very low
water
FWI Interannual Average Risk Categories on JJA season over the Test set years (1984, 1994, 2004, 2014, 2020)
Ground Truth
Light ANN Results
Deep ANN Results
extreme
very high
high
moderate
low
very low
water
FWI Interannual Average Risk Categories on SON season over the Test set years (1984, 1994, 2004, 2014, 2020)
Ground Truth
Light ANN Results
Deep ANN Results
extreme
very high
high
moderate
low
very low
water

Figure 4.20 *DJF (first row), MAM (second row), JJA (third row), SON (fourth row) FWI Interannual Average Risk Categories over Test set years (1984, 1994, 2004, 2014, 2020). Ground Truth (left images) compared to LightP2P average prediction (center images) and DeepP2P average prediction (right images).*

Figure 4.20 represents the average risk classification over the datasets DJF, MAM, JJA and SON, respectively. Both the two ANNs follow the main characteristics provided by the real FWI classes along each seasonal dataset. From a visual perspective, the results are quite impressive, however some little imperfections can appear along the images. For instance, LightP2P fails at determining the risk class in the central-northern side of Europe in the JJA average map, while DeepP2P better succeeds in doing it. In the SON dataset, both the Networks fail at determining the amplitude of the features in the south-eastern part of the map.

4.3.5 Accuracy Metrics

As explained in the previous two sections, 5 different average FWI ground truth and generated maps have been obtained over the Test Dataset. Then, a series of four different accuracy metrics on these five average FWI predictions have been evaluated, in order to quantitatively monitor if the two Artificial Neural Networks (DeepP2P and LightP2P) are successful in making predictions on unseen data.

First of all, it is important to assess the meaning of these accuracy metrics in terms of features that they may capture. As described in the following, each metric has its own strengths and weaknesses.

- Absolute Error (AE): it is calculated by taking the absolute value of the difference between the ground truth FWI map and the predicted one, according to the following equation:

  $$AE=\left|GT_{FWI}-PR_{FWI}\right|$$

  where GT_{FWI} is the Ground Truth FWI map and PR_{FWI} is the Predicted FWI map. It is a useful metric to understand the pixel-wise difference between two images. In this way, the differences between two visually similar images can be enhanced. Lower values correspond to more similar images.

- Kullback Leibler Divergence (KLD) (Kullback Leibler Divergence computation, 2019): it considers the two images as distribution probabilities and quantitatively computes the difference between them. It is not a symmetric evaluation metric, hence it must be computed accordingly. The KLD value can be computed as:

$$KLD(P \vee Q) = \sum_{i}^{\square} P(i) \log_2 \left(\frac{P(i)}{Q(i)} \right)$$

where P and Q are two probability distributions (in this case they represent two FWI maps). As the above formula shows, KLD is not symmetrical, meaning that it expresses how distribution P is different from Q. In addition, the smaller the KLD, the more similar the two images are.

- Power Spectral Density (PSD): it is the measure of the intensity content of an image along the different spatial frequencies. It can be useful to determine whether or not the generated maps correctly capture the distribution of the data along all the spatial frequencies. On a 2D image, it is computed according to the following steps:
  - Computation of the 2D Fast Fourier Transform;
  - Shift of the 0 frequency component of the spectrum on the center of the image;
  - Compute the absolute value of the power of previous result;
  - Compute the azimuthally averaged radial profile.

The resulting 1D vector represents the one-dimensional Power Spectral Density of the input image.

- Pearson Correlation Index (PCI): Pearson Correlation measures how well two distributions are related to each other. In particular, PCI provides a correlation value which is in the range $[-1,1]$. A value equal to -1 means that the two distributions are negatively related, whereas a value equal to 0 means that there is no relation and a value of 1 indicates that the two distributions are positively correlated. It is computed as:

$$r = \frac{n\sum xy - (\sum x)(\sum y)}{\sqrt{[n\sum x^2 - (\sum x^2)][n\sum y^2 - (\sum y^2)]}}$$

In this case, PCI has been evaluated for each pixel, computing a correlation map. For each pixel of coordinate (i, j), the daily temporal variation along the years provided by the dataset has been collected. This procedure has been repeated for the Target FWI maps, as well as for the synthetic data provided by both the LightP2P and DeepP2P. As a result, the correlation map for each land point has been computed, showing the correlation between real data and generated one for each point of the EURO-CORDEX geographical domain.

Similarly to what has been shown in section 4.3.3 and 4.3.4, 5 different accuracy plots have been produced: a plot is related to the entire averaged Test Dataset and the other 4 ones concern each season of the 5 years span, respectively.

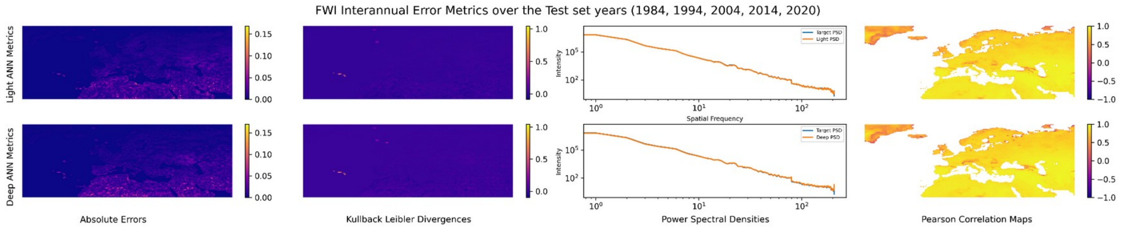


Figure 4.21 *FWI Interannual Error Metrics over Test set years (1984, 1994, 2004, 2014, 2020). On the first row there are Light ANN metrics, on the second row there are Deep ANN metrics. From left to right are represented: Absolute Error, Kullback Leibler Divergence, Power Spectral Density and Pearson Correlation Maps.*

As Figure 4.21 shows, the aforementioned error metrics have been computed over the entire inter-annual Test Dataset. In particular, both AE and KLD have almost always low values, for both the ANNs. It is worth noting that AE shows some peaks and structures in the African region. By the way, it is extremely low, with maximum values around 0.16. KLD divergence, instead, shows in both cases some structures on the left side of the image. On the other hand, Power Spectral Densities are extremely similar for both models. However, both Light and Deep neural networks fail in catching extremely high frequencies, even though the trend is always accurately captured. Finally, the PCI map is extremely significant since it expresses how well the networks have predicted correct values along the time, for each pixel. In fact, in central and southern territories of the EURO-CORDEX domain, the correlation is extremely high, meaning that the networks succeeded in making correct predictions. However, both of them tend to lose accuracy in those regions where fire risk is almost always low such as the Scandinavian peninsula and Greenland.

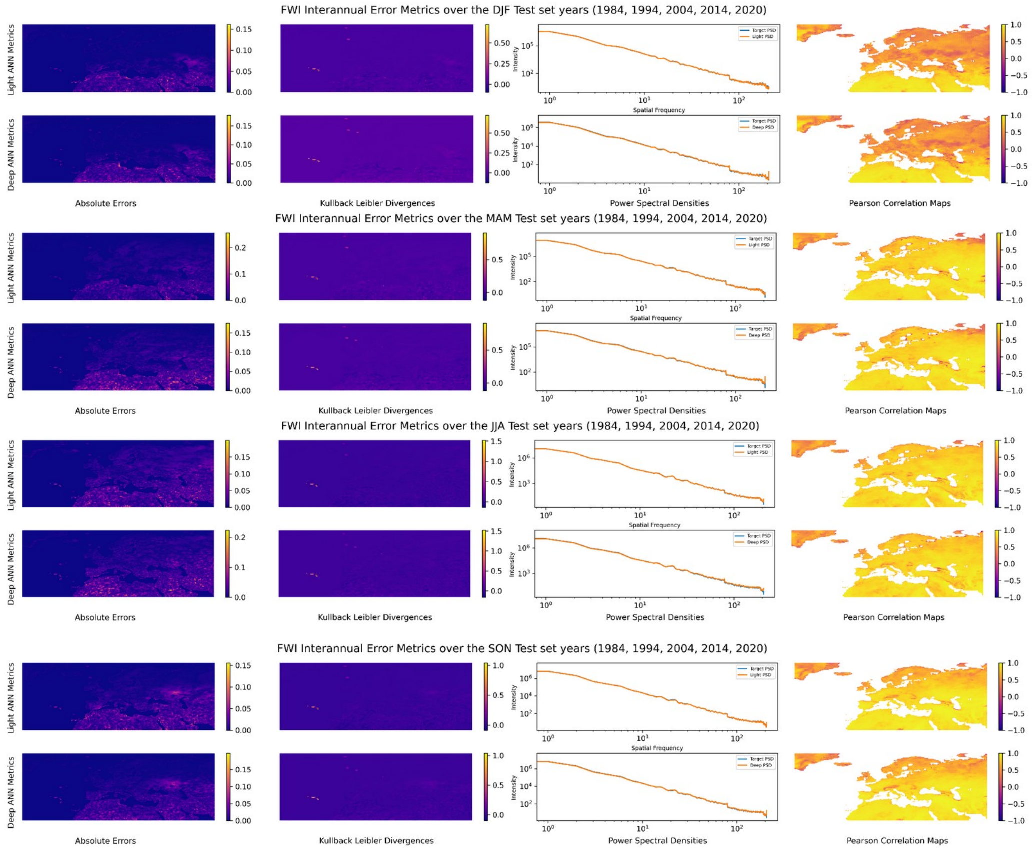


Figure 4.22 *FWI Interannual Average Risk Categories over Test set years (1984, 1994, 2004, 2014, 2020). Ground Truth (left image) compared to LightP2P average prediction (center image) and DeepP2P average prediction (right image).*

Figure 4.22 shows how the two networks behaved during interannual seasonal predictions. The results obtained by both the Machine Learning approaches are extremely good and similar to each other. PSDs are always similar with the target with very little variance along the samples. DJF season shows very little Pearson Correlation along central Europe and Russia for both DeepP2P and LightP2P. This may be related to the fact that the winter season records very low values of FWI, and hence the networks produce more inaccurate results. A very similar behavior is recorded on the SON dataset, during fall seasons.

4.4 Discussion and Conclusion

As stated in the introduction, Fire Weather Index measures the risk of fire ignition and spread on a certain geographical region of interest. For its computation 4 climatic drivers are needed: the temperature at 2m, the cumulative precipitation over the last 24 hours, the relative humidity and the wind speed. In particular, the traditional approach of FWI computation implies that a series of 5 intermediate indices must be evaluated in order to provide a FWI value on a certain day. In addition, some meteorological observations must be accumulated over long time periods.

Since several tables, factors and variables must be taken into account, the traditional approach constitutes a time-consuming and challenging procedure for deterministically computing FWI. The main objective of this thesis work was to exploit a Machine Learning algorithm in order to timely generate accurate and reliable FWI maps. To this extent, two different neural networks have been proposed and a physically consistent training approach has been described and implemented. The first one is a Deep Pix2Pix, with more than 27 million parameters on the generator, while the second one is a Light Pix2Pix, with just 11 million parameters on the U-Net generator.

The results of this work clearly demonstrate that ML overcomes the previously mentioned disadvantages in the computation of FWI maps. In fact, with little error values, both the two ML algorithms, DeepP2P and LightP2P succeed in generating accurate FWI synthetic maps on the EURO-CORDEX domain, by requiring as input just 4 meteorological measurements of a certain day.

Even though the two proposed models produce very low pixel-wise errors, they have some differences. The DeepP2P sometimes is better at generating FWI samples, but it is heavier than the other and it is more challenging to be handled in terms of memory. On the other hand, LightP2P is faster in terms of prediction and training execution time and provides samples which are similar to the deeper one.

In summary, both the networks are much faster than the traditional deterministic FWI computation, since they can generate daily FWI maps in a few seconds by just requiring the daily meteorological drivers, whereas the deterministic FWI computation requires more indices, as well as longer period climatic observations.

4.4.1 Future Works

Even though good results have been achieved, this work can be improved under several points of view. In this chapter will be suggested some future works which may be eventually explored in more detail.

4.4.1.1 Physics Informed Model

Even though the proposed approach already embeds physical constraints into it, those are not the only updates that could be done in that sense. Enclosing physical constraints into the model leads to a Physics Informed Machine Learning (PIML, in the following) approach (Kashinath et al., 2021). In this way, the model itself incorporates constraints which make it a more reliable network. In the literature, many climate models have been enhanced with physical knowledge, obtaining a faster convergence to a local optima which also is physically reliable (Jin-Long Wu et al., 2020). In the following, it will be referred to as normal for networks which are not physically informed. A not physically constrained neural network model may correctly minimize a certain loss function generating results which minimize several metrics taken into account in the model. However, in general, generated results are not physically reliable. This is, obviously, a limitation of these neural networks, which cannot provide reliable values in climate modellations. (Jin-Long Wu et al., 2020), enclosed into their GAN model a statistical constraint, managing to improve correctness of their solutions on one hand and harshly reducing model training on the other. They estimated that their physically informed model was 80% faster than the normal one. Moreover, another advantage of using PIML is, on one hand, the increment of training efficiency, requiring less data to effectively train models. On the other hand, PIML models are more generalizable when opposed to the others.

Besides, in addition to the Pix2Pix generator $L1_{loss}$, it could be added a PI loss term which heavily penalizes values different from the sentinel one of FWI along the sea according to the following equation:

$$PI_{loss} = \lambda_{PI} * M * P$$

Where λ_{PI} is a regularization hyperparameter to enhance the strength of the penalty on bad predictions along the sea, M is the binary sea surface mask and P is the generated fire risk map. It is worth noticing that the product $M * P$ is the Hadamard product between the two matrices.

Furthermore, another loss function has been added to the model in order to improve its consistency as well as the speed of convergence to an optima. The loss involved in this case is the so-called PSD-Loss. (Singh et al., 2019), proposed in their work a PSD-Loss evaluated as the Mean Squared Error (MSE) between the bi-dimensional Power Spectral Density of the ground truth image and the predicted one. In this case, following the same line of (Singh et al., 2019), by using Mean Absolute Error (MAE) between the two PSDs.

4.4.1.2 Additional Input Features

If on one hand the model can be enhanced through PIML, on the other hand it is possible to increase the quality of input data. In fact, the dataset can be augmented by providing other useful drivers which are not present in the original FWI deterministic computation, such as the Normalized Difference Vegetation Index (NDVI), the vegetation distribution, the vegetation quality indices (i.e., EVI), the population displacement along the territory, the CO_2 content in the air, etc…. All these additional datasets should eventually be re-gridded and resized in order to perfectly match the current dataset. Furthermore, information provided by these variables may be useful for a more accurate and reliable training of the ML model.

4.4.1.3 Climate Projection Scenarios

From services like CMIP6, 2020, it is possible to download future climate projections, according to the *Representative Concentration Pathway* (RCP) and *Shared Socioeconomic Pathways* (SSP) scenarios. In short, RCP is the CO_2 concentration trend in future years according to the quantity of greenhouse gases released in the atmosphere. Instead, SSP is the future projection about socio-economic changes taken to face the increment in greenhouse emissions.

It is then possible, after some re-grid preprocessings, to download the 4 drivers used in this thesis, according to a certain projection scenario, and use them to test both DeepP2P and LightP2P and analyze the resulting FWI synthetic maps obtained.

5

A multi-model architecture based on Long Short-Term Memory neural networks for multi-step sea level forecasting

The intensification of extreme events, storm surges and coastal flooding in a climate change scenario increasingly influences human processes, especially in coastal areas where sea-based activities are concentrated. Predicting sea level near the coasts, with a high accuracy and in a reasonable amount of time, becomes a strategic task. Despite the developments of complex numerical codes for high-resolution ocean modeling, the task of making forecasts in areas at the intersection between land and sea remains challenging. In this respect, the use of machine learning techniques can represent an interesting alternative to be investigated and evaluated by numerical modelers.

This article presents the application of the Long Short-Term Memory (LSTM) neural network to the problem of short-term sea level forecasting in the Southern Adriatic Northern Ionian (SANI) domain in the Mediterranean sea. The proposed multi-model architecture based on LSTM networks has been trained to predict mean sea levels three days ahead, for different coastal locations. Predictions were compared with the observation data collected through the tide-gauge devices as well as with the forecasts produced by the Southern Adriatic Northern Ionian Forecasting System (SANIFS) developed at the Euro-Mediterranean Center on Climate Change (CMCC), which provides short-term daily updated forecasts in the Mediterranean basin. Experimental results demonstrate that the multi-model architecture is able to bridge information far in time and to produce predictions with a much higher accuracy than SANIFS forecasts.

This chapter has been published as:

Accarino, G. et al., “A multi-model architecture based on Long Short-Term Memory neural networks for multi-step sea level forecasting”, *Future Generation Computer Systems*, vol. 124, pp. 1-9, 2021. doi: 10.1016/j.future.2021.05.008

5.1 Introduction

The crucial role of sea level prediction, from short- to long-term scale, is nowadays well evident looking at the climate change effects on the sea level in coastal zones (sea level rise, increasing extreme events, storm surges and coastal floodings) (IPCC, 2019; Nicholls and Cazenave, 2010). Accurate sea level forecasts, in areas at the interface between land and ocean, could help implement proper actions in advance to mitigate the environmental, social and economic damages caused by the sea level extremes (Hallegatte et al., 2010). From a numerical perspective, state-of-art ocean models rely on physics equations and processes parameterizations to simulate several ocean-related variables, on very large spatio-temporal domains (global and regional scales (Iovino et al., 2016; Clementi et al., 2019)). Unfortunately, there are many cases (i.e. in near-shore areas, where the most important human and economic activities are concentrated) in which such models are not able to simulate those variables with an acceptable level of accuracy. On the other hand, increasing the resolution to produce results at a finer scale is an even more challenging task. This task is necessary because some phenomena are not manifest at a coarser resolution, thus demanding a high availability of High Performance Computing (HPC) resources and strong mathematical frameworks to resolve processes that manifest at fine scales. The current study addresses the problem of near-shore sea levels predictions using Long Short-Term Memory (LSTM) neural networks (Hochreiter and Schmidhuber, 1997) trained on historical site-specific observation data related to different coastal locations in the Southern Adriatic Northern Ionian (SANI) seas in the Mediterranean basin. The proposed LSTM-based multi-model is able to produce 3-days ahead predictions of sea levels, which have been compared with the hindcast produced by a high-resolution deterministic model based on unstructured-grid approach and designed for coastal zones (Federico et al., 2017). Each

LSTM network within the multi-model architecture uses a fixed number of previous time steps as input, and the right choice of the best number of input time steps is the result of a systematic procedure that is proposed in this work.

5.2 Related work

In the last years, several machine learning approaches were proposed over traditional analysis and forecasting techniques in order to deal with time series related problems. Early developed machine learning techniques are presented in 5.2.1, then more advanced techniques are discussed in 5.2.2.

5.2.1 Early techniques

As reported in (De Gooijer and Hyndman, 2006), a general problem with non-linear models is the curse of complexity and over-parameterization. Authors in (Swanson and White, 1997), suggested that even a single hidden layer feed-forward neural network is a flexible alternative to fixed specification linear models, particularly at forecast horizons greater than one step-ahead. In oceanography, these approaches have been extensively used for predicting ocean related variables such as sea surface temperature (SST), waves, sea level and important processes such as El Niño and monsoon models (Forget et al., 2015; Hsieh, 2009; Mullen et al., 1998; Krasnopolsky, 2002; Thessen, 2016; Wu et al., 2006). Concerning the sea level, which is the focus of this work, Self-Organizing Map (SOM), Genetic Programming (GP) and Artificial Neural Network (ANN) were demonstrated to behave well on the prediction task. In (Ultsch, 2002), a SOM network is used to predict the sea level in Hamburg, comparing the obtained results with the outcomes of 6 different models as well as with observations. The GP algorithm is used in (Ghorbani et al., 2010) to forecast sea level variations, three time steps ahead, for a set of time intervals of 12 h, 24 h, 5 days and 10 days, comparing the results with the outcomes of an ANN model; the employed GP model was trained and validated using the tide gauge measurements at Hillarys Boat Harbor, Western Australia. The previous approach was re-proposed by the same authors for another location in Cocos (Keeling) Islands in the Indian Ocean (Makarynska and Makarynskyy, 2008; Ghorbani et al., 2010). In (Huang et al., 2003), a feed-forward neural network (FFNN), named the Regional Neural Network for Water Level (RNN_WL) was trained to predict long-term water levels in a coastal inlet, based on data provided by a remote station in the South Shore of Long Island, New York. The daily mean sea level heights are estimated in (Sertel et al., 2008), using five different methods: the least squares estimation of sea level model, the multilinear regression (MLR) model, and three ANNs algorithms: Feed forward back propagation (FFBP), radial basis function (RBF), and generalized regression neural network (GRNN). From statistics, they derived that neural networks based approaches can provide reliable results for estimating the daily mean sea level over the classical least squares estimation. An algorithm to remove interneuron connections is introduced in (Leahy et al., 2008), adjusting the weights of the remaining connections in a FFNN, to reduce complexity and to predict river levels 5 h ahead. In (Pashova and Popova, 2011), different ANNs approaches, namely multilayer feed-forward (FF), Cascade-Feed-Forward (CFF), Feed-Forward Time-Delay (FFTD), RBF, Generalized Regression (GR) neural networks and Multiple Linear regression (MLR) methods were compared for the task of predicting sea levels. These ANNs are all suitable for capturing the short and long-term dependencies in the time series, outperforming traditional harmonic analysis techniques. In (Filippo et al., 2012), an ANN is trained by using hourly time series of atmospheric pressure, wind, and harmonically derived tides as input data and hourly time series of measured tides as output data. The network is able to reduce the margin error by almost 50%. In (Karimi et al., 2013), the hourly sea levels were predicted using two different, data driven techniques, namely the adaptive neuro-fuzzy inference system (ANFIS) and ANN. The MLR technique was used for selecting the optimal input combinations (lag times) of hourly sea levels. ANN and ANFIS performed similarly and better, compared with the Auto-Regressive Moving Average (ARMA) model. In (Imani et al., 2018), an ANN-based method named the Extreme Learning Machine (ELM), was employed for predicting sea level variations in Chiayi, Taiwan. Moreover, other methods such as the Relevance Vector Machine (RVM), Support Vector Machine (SVM) and RBF models have been used. Results show that the ELM and RVM models outperformed the other methods. Different gap-filling strategies are provided in (Wenzel and Schröter, 2010), for time series with missing values, mainly based on a forecast network (FCnet) or an equivalent backcast network (BCnet) as well as on the Empirical Orthogonal Function (EOF) reconstruction technique. Once the time series had been reconstructed, a neural network was used to predict regional mean sea level anomalies (regional MSLA) from a set of selected tide gauges. Authors in (Wang and Yuan, 2018), proposed a method for forecasting tide levels of storm surges caused by typhoons by using typhoons parameters as input; then a cubic B-spline curve with knot insertion algorithm is combined with the forecasts to obtain a smoothed tidal level prediction curve. Although classical ANN architectures are able to deal with time series analysis and prediction problems, as reported so far, the task of learning long-term correlations among time steps of a long time series remains challenging.

5.2.2 Advanced techniques: RNN, CNN and LSTM

Recurrent Neural Networks (RNN) can naturally deal with time series, as they can easily capture long-term spatial and/or temporal dependencies, while other architectures such as Convolutional Neural Networks (CNN) can process multidimensional data capturing the spatial correlations. Moreover, RNN and CNN models can be combined together leading to a model that can deal with spatio-temporal data. A combination of Convolutional Neural CNNs and RNNs is proposed in (Braakmann-Folgmann et al., 2017) in order to analyze both spatial and temporal sea level evolution and predict interannual sea level anomalies (SLA). Data from a single tide gauge are used in (Makarynskyy et al., 2004) to train a neural network that has been used to predict hourly sea level variations for the following 24 h up to 10-daily mean sea levels; they also proposed a method to find the best number of previous time steps as input that the network uses to get the highest accuracy on the test set. A similar approach is also proposed in this work. In particular, a special type of RNN called Long Short-Term Memory (LSTM) network is used. Firstly, the LSTM avoids the need to explicitly select a leading time for the sliding window approach reported in (Makarynskyy et al., 2004) (the number of time steps to shift the sliding window forward in the time series); then it avoids some numerical issues related to the vanishing or exploding gradient problem during the training phase, as explained in Section 5.4. So far, LSTM models have been efficiently adopted in several studies with the task of predicting events with relatively long intervals and delays in time series. As an example, in (Zhao et al., 2019) a combination of CNNs and LSTMs is used to create a text-independent speaker verification model whereas in (Yu et al., 2019) a LSTM model with an architectural enhancement based on the Enhanced Forget Gate (EFG) is used in order to forecast wind power with an improvement of 18.3% compared with the other forecasting models. In the remainder of this paper, a short description of the data used is proposed in Section 5.3, whereas a short introduction to the LSTM network follows in Section 5.4. The experimental setup is presented in Section 5.5 along with the results that are provided and discussed in Section 5.6. Finally, Section 5.7 provides conclusions and presents future work.

5.3 Sea level data

Sea level measurements were obtained from the Italian national tide gauge network Rete Mareografica Nazionale, provided by the Italian Institute for Environmental Protection and Research (ISPRA) (RMN, 2021). In particular, the present analysis has been performed on the basis of the five stations in the SANI area (Otranto, Bari, Taranto, Vieste and Crotone) considering a time period of four years (2013–2016). The data has been processed to obtain the daily mean sea level. Missing data was present in some time-series, due to tide gauge unavailability, as well as unknown values for some days. To tackle this problem, a pre-processing step was applied to data before training. The time steps corresponding to the missing values were barely removed, instead of replacing them with an average value (i.e. the monthly mean), basically for two reasons: the lack of long missing values intervals and the robustness of ANNs and especially LSTMs to deal with incomplete time series. The analysis compares data derived from physical model simulations and tide-gauge measurements (observational data) respectively. Since simulation data and observational data are expressed at different reference levels, being the simulation data not initialized by the observational one, a normalization procedure has been necessary to make them comparable. This was basically performed by subtracting the average from each time-series over the investigated period (four years, 2013–2016). For each coastal location considered, the LSTM has been trained with tide-gauge observational data. This data has been split into four non-overlapping subsets, namely 80% for training, 5% for validation, 5% for cold-start and the remaining 10% for testing. The choice of using 80% of the subset for the training phase was not made by chance, as it actually allows training the network with different seasonal trends over a period of about three years, thus minimizing the generalization error that the network made with samples that were not previously seen during the training phase. It should be remarked that the output of the network at a given moment does not only depend on the current input but also on the information about the previous moments that is stored in the network memory. Then, to allow a more accurate prediction, before launching the test phase, a prediction was made by using data in the cold-start subset. This makes it possible to store the information about the context prior to the test set inside the LSTM memory and to exploit it for the actual prediction on the test data.

In order to assess the predictability performance, the sea level forecast of the proposed LSTM network is compared with the hindcast simulation of a physical-based deterministic model, produced for the 2013–2016 time period. The modeling system is SANIFS (Federico et al., 2017), which is a coastal-ocean operational system based on the unstructured grid finite element three-dimensional fully-baroclinic SHYFEM model (Umgiesser et al., 2004), capable of producing both short-term forecasts and retrospective hindcasts. The unstructured-grid approach allows a variable resolution from open sea (≈4.5 km) to coastal waters (≈500 − 50 m). Figure 5.1 shows the model domain (horizontal grid and bathymetry) with overlapping of the five coastal hotspots (Otranto, Bari, Taranto, Vieste and Crotone) where the analysis has been performed, and a detailed view of the sea level map in Otranto coastal zone. The hindcast simulation here produced exploits a downscaling/nesting approach from the regional-scale model for the entire Mediterranean Basin Med-CMEMS (Copernicus — Marine environment monitoring service, (Clementi et al., 2019; CMEMS, 2021)) analysis, which provides initial and boundary condition fields to the nested system. The system is forced at surface by the European Centre for Medium Weather Forecasts (ECMWF) atmospheric analysis. The methodology to simulate the entire 2013–2016 dataset is based for each reference day on a hot re-initialization from the CMEMS parent model (Federico et al., 2017; Rolinski and Umgiesser, 2005; Muis et al., 2016). This means the model simulations have been carried out separately for each reference day, using a spin-up time of three days. The re-initialization procedure from the parent model allows benefiting of the high-quality systematic fields in the Med-CMEMS parent model, which is provided by observing data assimilation (Dobricic and Pinardi, 2008). On the other hand, the spin-up time ensures the development of internal dynamics by the nested model. A 3-day spin-up was considered a reasonable choice by several authors implementing high-resolution models in re-initialized mode on coastal scale (Federico et al., 2017; Cucco et al., 2012; Trotta et al., 2016).

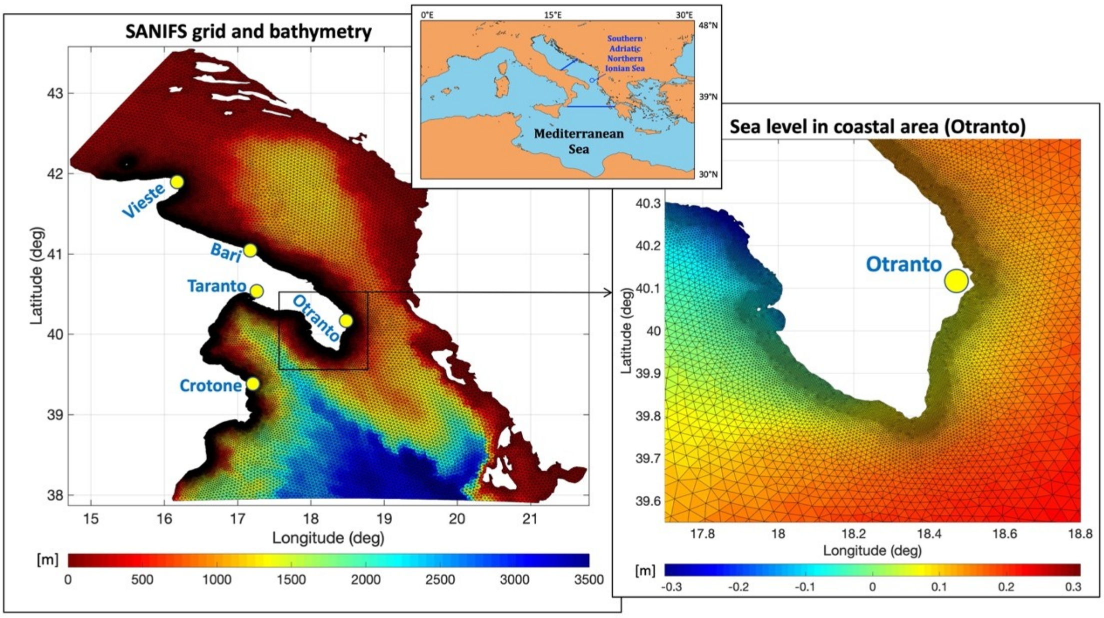


Figure 5.1 *SANIFS model domain (horizontal grid and bathymetry) with overlapping of the five coastal hotspots (Otranto, Bari, Taranto, Vieste and Crotone) where the analysis and a detailed view of sea level map in Otranto coastal zone.*

5.4 Long Short-Term Memory neural networks

Long-Short Term Memory networks, or just LSTMs, are the most effective evolution of the basic model of a Recurrent Neural Network (Rumelhart et al., 1986). A RNN is a class of ANN where connections between units form a directed cyclic graph. They can use feedback connections to store activations related to short-term events. The complexity of the internal architecture of LSTM networks, compared with the RNNs, is justified by the capabilities of such networks to remember information over long periods of time to avoid the vanishing or exploding gradient problem. Recurrent Neural Networks are suited for processing sequential data such as time series as they allow connecting previous information to the present time step. But, as the information to recover is far in time, RNNs become unable to connect this information. During the training phase, when conventional Back-Propagation Through Time (BPTT) is performed, gradients flowing backwards in time of an unrolled network tend to either explode or vanish. The problem refers to the large increase in the norm of the gradient during training. In this situation, weights associated with long-term dependencies receive exponentially larger updates than the short-term ones producing the exploding gradient problem. The opposite behavior produces the vanishing gradient problem, thus long-term components go exponentially fast to norm 0 (Pascanu et al., 2013). Since the weight represents the relevance of a particular information to the present task, the fact that it is given a small value indicates that the RNN is not able to learn anything from it. When the temporal extension of the dependencies increases, the gradient descent method becomes extremely inefficient (Bengio et al., 1994). Although, in theory, RNNs are capable of handling arbitrary length sequences, in practice even very small sequences make the training complicated. It will take a prohibitive amount of time to learn long-term dependencies because they are hidden by short-term ones, or do not work at all. To overcome these problems, the LSTM neural network was introduced, as a new robust architecture (Hochreiter and Schmidhuber, 1997). The main purpose was to capture long-term dependencies without loss of learning of the short-term ones. This is achieved by exploiting an internal architecture that allows a constant error flow through the internal states of special units. To avoid the problems occurring with BPTT in recurrent architectures, the LSTM network enforces constant error flow, by propagating error signals through dedicated recurrent paths named Constant Error Carousels (CECs). The error is stored inside CECs where the activation function is constant and linear, as opposed to non-linear activations such as sigmoid or tanh that shrink the error every time it is back propagated. Moreover, the constant error carousel is part of a memory cell, delimited by input and output gates. Several memory cells are allowed to optimize the memory management composing a memory cell block, in which case the gates are shared by memory cells. The input and output gates are equipped with parameters that the network can learn using a gradient descent algorithm, so the gate units learn to open and close access to cells. The solution proposed in (Gers, 1999) consists in extending the standard architecture by adopting a multiplicative forget layer with a learnable parameter that controls the information stored in the cell state. It decides what information in the cell state is unnecessary and must be dropped. This mitigates the exploding or vanishing gradient problem and stabilizes long-term memory. An overview of the LSTM internal architecture is depicted in Figure 5.2.

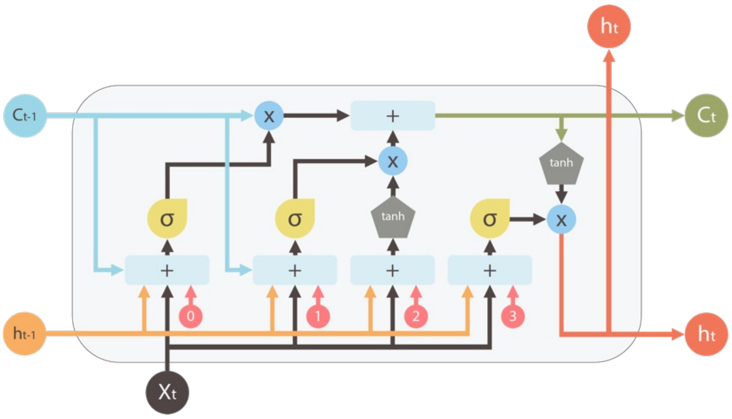


Figure 5.2 *The structure of the Long Short-Term Memory (LSTM) neural network. Source: Reproduced from Yan (Medium, 2016).*

5.5 Experimental setup

The proposed forecasting scheme allows the 3-days ahead daily mean sea levels prediction starting from a fixed number of previous daily mean sea levels as input. The number of input time steps, along with a fine tuning of the network hyperparameters, strongly affects the forecast quality. Inspired by the work in (Makarynskyy et al., 2004), after the network tuning, an analysis to sort out the best number of input time steps is conducted, by checking for local extremes of some interesting metrics calculated on the test set. As opposed to (Makarynskyy et al., 2004), several locations are considered and the procedure slightly changed: during the training phase, since the LSTM internally stores some prior useful information, there was no need to consider a sliding window that shifts with some time increments. In order to train with a supervised learning procedure, the sequence of sea level values is transformed to a sequence of input/output pairs. The neural network has been designed using the Keras (Keras, 2015) module with Tensorflow backend (Abadi et al., 2016). After fine tuning the network hyperparameters with a trial and error procedure, the following architecture has been adopted: one layer composed of 128 neurons, activated by the Parametric Rectified Linear Unit (PReLU), followed by a 3-neuron feed-forward neural network with linear activation to produce three time steps ahead sea level predictions. The network was trained by minimizing the Mean Squared Error (MSE) cost function for 50 epochs with Adam optimizer (Kingma and Ba, 2017). The Learning Rate (LR) was set to 10^{-4} along with ⬆$_1$, ⬆$_2$ and ༄ respectively set to $0.9, 0.999, 10^{-8}$. A batch size of 4 was used. In order to prevent overfitting, the L$_2$ regularization technique was adopted, setting the value of the regularization parameter λ to 10^{-3}.

For each location, the procedure to select the best number of input time steps requires training the same neural network architecture with a different number of previous daily mean sea levels. The choice is not a trivial matter, and this parameter should not be selected arbitrarily. Therefore, the analysis considered from 1 (one day) to 7 (one week) previous input time steps, as evidenced in the following section. Overall, 35 neural networks have been trained. The LSTM performance on the test set was evaluated in terms of the Root Mean Square Error (RMSE) and the coefficient of determination (R^2) (Makarynskyy et al., 2004):

$$RMSE = \sqrt{\frac{\sum_{i=1}^{m} \left(y_i - x_i \right)^2}{m}}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{m} (x_i - y_i)^2}{\sum_{i=1}^{m} (x_i - \bar{x})^2}$$

where x_i is the value observed at the ith time step, y_i is the value predicted by the network or the SANIFS hindcast at the same moment of time, m is the number of time series values, $\bar{x}$ is the mean value of the observations.

5.6 Experimental results

In this section the obtained results are widely discussed starting from the input time step selection analysis. The model selection procedure is reported in Section 5.6.1 whereas the multi-model architecture is introduced in 5.6.2. Finally, the models' comparison is illustrated in Section 5.6.3.

5.6.1 Model selection analysis

Figure 5.3 reports the RMSE and the R^2 score averaged over the three output time steps for seven configurations of input time steps, whereas the color represents a specific coastal location. It is straightforward to see that training the LSTM network with a single previous daily mean sea level leads to the best input-output mapping emulation. In fact, a minimum value of the RMSE is obtained along with the highest R^2 score.

However, it is worth analyzing the 2-input case more in depth. Concerning the ensemble average plot, there is a slight 6.46% increase in RMSE for the 2-input case with respect to the 1-input one, where the R^2 score reflects a similar situation but in a specular manner. As reported in Table 5.1, what was derived is that the 1-input case behaves better when predicting one time step ahead, whereas the 2-input case leads to a slightly better RMSE on the third time step ahead predictions for all the locations. Concerning the 2-steps ahead forecasts, there is no clear pattern related to which number of input time steps is more suited for making lower RMSE predictions for all the locations. In particular, Otranto and Bari required two input time steps for making better predictions on the test set, whereas Crotone, Vieste and Taranto required only one input time step. It is important to note that this is not a trivial result. Generally, in forecasting tasks, the accuracy is likely to be reduced when considering an arbitrary number of output time steps ahead. In this study, the LSTM network with 2-input time steps was demonstrated to be noteworthy, as the RMSE of the third output time step is 3.94% smaller than the RMSE of the 1-input case, on average for all locations. As one considers a higher number of previous time steps, the RMSE increases but, starting from the 5-input case it seems to smoothly decrease. However, the analysis has not been extended over one week as the time series have a daily temporal resolution and the variable to forecast is the daily mean sea level. Cases with six or seven input time steps could still be acceptable as it could happen that the network is learning some weekly trends of input data producing sub-optimal results. Moreover, looking at the ensemble average curve, it is evident that as the number of input time steps increases, the behavior of these networks is independent of the location.

From the previous analysis, the right choice of the number of input time steps depends on how many time steps ahead need to be predicted, according to RMSE values shown in Table 5.1.

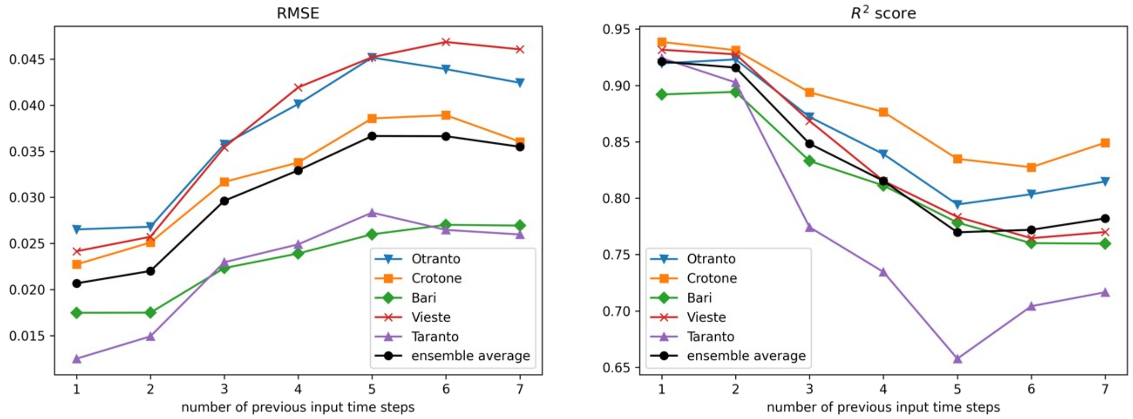


Figure 5.3 *Averaged metrics on the test set over the three output time steps. Each color refers to a specific location. The black line represents the ensemble average of the specific metric over different locations.*

Table 5.1 *The table compares the RMSE (in meters) between 1-step, 2-steps and 3-steps ahead LSTM predictions and observations (first three rows), as well as the RMSE between the SANIFS hindcast and observations (final row). Concerning the LSTM results, each sub-row reports the metrics for the 1 and 2 input cases respectively. The values expressed as a percentage refer to the enhancement in terms of RMSE of the LSTM forecasts with respect to SANIFS hindcast.*

RMSE (in meters) and the RMSE percentage improvement of LSTM forecasts w.r.t. SANIFS hindcast					
Locations	**Otranto**	**Crotone**	**Bari**	**Vieste**	**Taranto**
1-step ahead (SLTM)					
$n_{\square}=1$	**0.0128** **76.3%**	**0.0119** **77.4%**	**0.0115** **68.0%**	**0.0125** **76.8%**	**0.0066** **86.8%**
$n_{\square}=2$	0.0169 68.9%	0.0184 65.1%	0.0128 64.4%	0.0178 67.0%	0.0125 75.0%
2-steps ahead (SLTM)					
$n_{\square}=1$	0.0274 49.5%	**0.0233** **55.8%**	0.0178 50.6%	**0.0257** **52.4%**	**0.0133** **73.4%**
$n_{\square}=2$	**0.0268** **50.6%**	0.0251 52.4%	**0.0174** **51.7%**	0.0258 52.2%	0.0149 70.2%
3-steps ahead (SLTM)					
$n_{\square}=1$	0.0393 27.6%	0.033 37.4%	0.0231 35.8%	0.0343 36.5%	0.0176 64.8%
$n_{\square}=2$	**0.0367** **32.4%**	**0.0317** **39.8%**	**0.0222** **38.3%**	**0.0336** **37.8%**	**0.0173** **65.4%**
SANIFS hindcast	0.0543	0.0527	0.036	0.054	0.05

5.6.2 The multi-model architecture

This leads to the idea of a multi-model architecture that is able to produce the most accurate forecasts for each location. Figure 5.5 reports, for the Otranto location use-case, the proposed architecture which is composed of two independent LSTM networks that are trained in a different manner. The networks, denoted with $n_{\square}=1$ and $n_{\square}=2$, are trained by feeding them with the sea level values of the previous day ($n_{\square}=1$) and the previous two days ($n_{\square}=2$), respectively. Starting from the current time step (denoted as *t*), both networks predict the sea level for the next three days. In order to obtain the best prediction, an output concatenation procedure is performed for each time step. In particular, the first two days ahead (in orange) from the first network, and the third day ahead (in green) from the second network are selected. Unused forecasts are discarded during the final concatenation process (in red). Concerning the other locations, the workflow remains the same except for the final forecasts concatenation step, in which the 2-steps ahead forecast needs to be collected from the $n_{\square}=1$ or $n_{\square}=2$ LSTM blocks output, according to the model selection analysis presented in the previous subsection.

The experimental test-case is to use the aforementioned multi-model to produce forecasts on the out-of-sample data (i.e. the test set). The observation data along with the output of the SANIFS hindcast are compared with the multi-model LSTM-based forecasts. The ability of the LSTM network to generalize even to different coastal sites, leading to a spatial generalization, is still not known and out of the scope of this work.

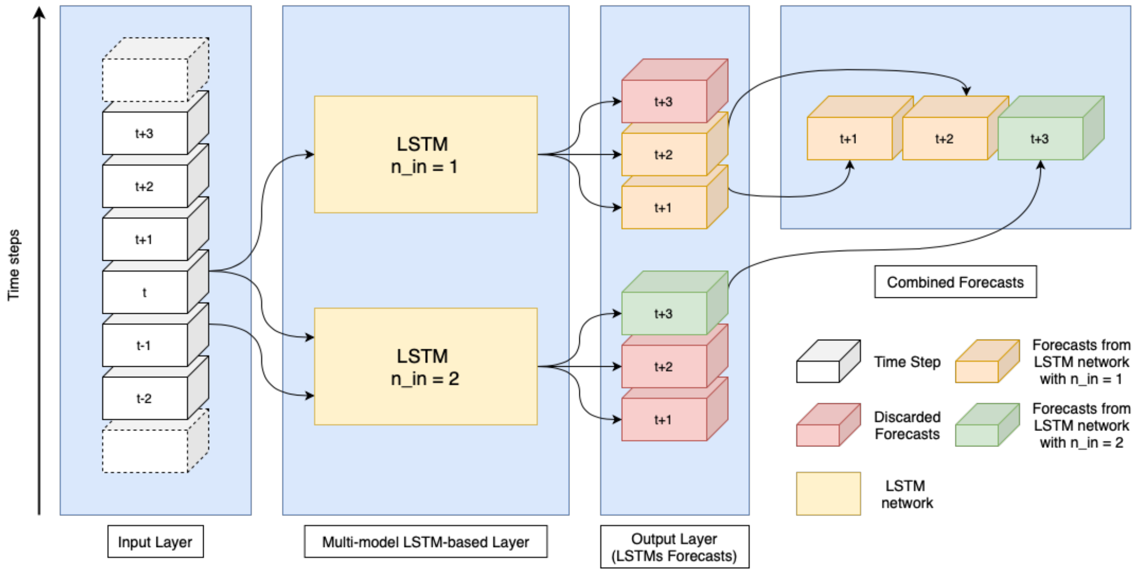
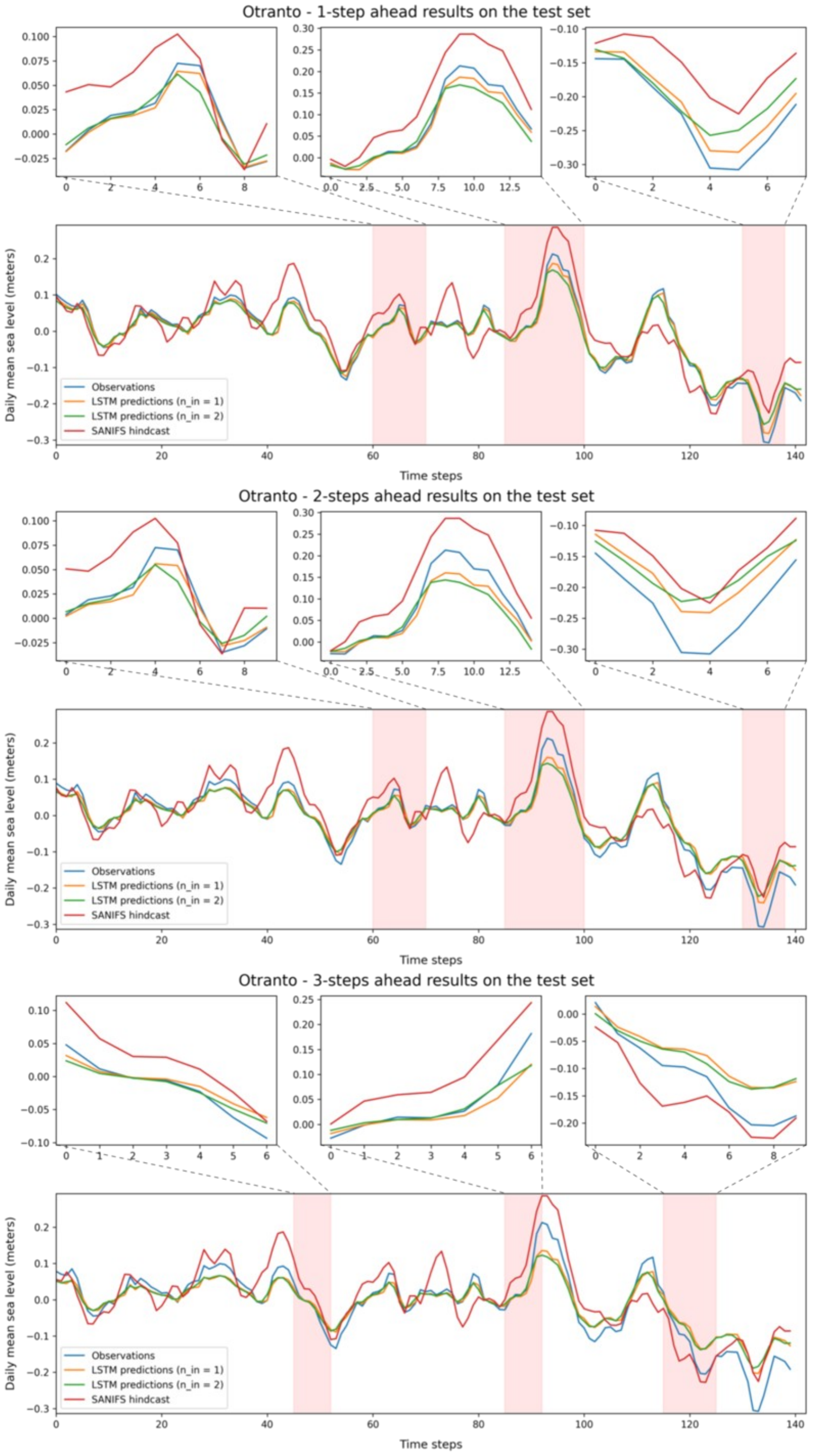


Figure 5.5 *The multi-model system based on LSTM networks as a result of the previous analysis for the Otranto use case. The multi-model layer is composed of two independent LSTM networks trained with one or two input time steps, respectively. Each network outputs three step-ahead predictions which are combined in the last layer, in order to obtain the best accuracy in terms of RMSE for the current location.*

5.6.3 Model comparison

Figure 5.4 reports the results obtained for the Otranto location concerning the $n_{\square}=1$ and $n_{\square}=2$ input cases, respectively. Each subplot refers to a different output time step and contains the LSTM networks forecasts along with the observation data and the SANIFS hindcast. The time steps on the x-axis represent samples from the test set, while the y-axis represents the daily mean sea level (in meters). Plots demonstrate a very close match between LSTM predictions (orange & green), observational data (blue) compared with SANIFS hindcast (red). The forecasts' quality produced by the LSTM decreases with the number of days ahead to predict, but it still performs better than SANIFS in terms of RMSE as summarized in Table 5.1. As depicted in Figure 5.4, the 1-input LSTM case leads to a much fitted curve with respect to observative data, whereas the 2-input case suggests that a slightly smoothed curve could improve the generalization capacity for uncertain and unknown out-of-sample patterns.

Another important point is that, training these networks retaining the same hyperparameters with a different number of input time steps, may not necessarily lead to an acceptable quality, due to the extremely hard job of neural network tuning. One may argue that this is an optimistic analysis; however, this is only a guideline for gaining some insight into the relationship between the number of input time steps and the remaining network hyperparameters. Once this hyperparameter has been selected, a further fine tuning of the architecture could be required.


Otranto - 1-step ahead results on the test set

Otranto - 2-steps ahead results on the test set

Otranto - 3-steps ahead results on the test set

Figure 5.4 Comparison between LSTM forecasts (two LSTM networks trained with (in orange) and (in green) respectively), observation data (in blue) and SANIFS hindcast (in red) for the Otranto location in Southern Italy. Each sub-plot focuses on a particular step ahead in the time series. Moreover, some trends in the plots are highlighted as demonstration that the LSTM trained with is more suitable for predicting the sea level for the next two days, whereas the case is more suitable for predicting the third day ahead.

5.7 Conclusions and future work

This work combines the outcomes of two LSTM neural networks in a multi-model fashion to produce accurate short-term daily mean sea level predictions for different coastal locations. The experiments show that LSTM behaves better than the SANIFS hindcast with respect to observational time-series data. The current study was limited to five locations whereas complex deterministic models are able to cover a huge domain (i.e. regional or global). The results demonstrate that adequately trained LSTMs outperform standard approaches. The spatial generalization capabilities of the network have not been investigated as they were out of the scope of this paper and will be discussed in a future work. Future improvements will be focused on *(i)* the training with high-frequency data (e.g. hourly fields to capture tidal signal); *(ii)* the extension of the forecasting window (e.g. 72 multiple hourly steps); *(iii)* the training of the network with multiple parameters affecting sea level (e.g. sea level pressure and wind fields recorded at tide gauge stations); and *(iv)* the integration of LSTM-based forecasting models into the Ophidia High Performance Data Analytics (HPDA) framework (Fiore et al., 2013; Fiore et al., 2019) to expand the set of timeseries-based functionalities currently offered by the Ophidia system towards HPC and Data Science (Fiore and Aloisio, 2011) capabilities.

Discussion, Future work and Learned lessons

In this work, ML algorithms have been successfully demonstrated to address the same purpose of traditional physics-based approaches, by attaining comparable accuracy metrics, whilst requiring at the same time less execution time, i.e. proved cost effective.

Despite a formal comparison concerning the computational performance against traditional dynamical downscaling approaches has not been performed in the present study, both MSG-GAN-SD and AR-CNN-SD architectures predict four years of the 2-m temperature over the EURO-CORDEX domain in less than a minute. Instead, to the best of my knowledge, traditional downscaling models for the same task are significantly slower by at least one order of magnitude.

Moreover, for the FWI risk map generation task, 1000 maps have been generated by the LightP2P model in 74.85 seconds, with an average time spent for a single instance of 0.074 seconds, and by the DeepP2P model in 162.61 seconds, with an average of 0.162 seconds per instance, by maintaining a comparable level of accuracy with respect to the traditional deterministic algorithms.

These results corroborate the utility of using ML for efficiently predicting Extreme Weather Events directly, or to enhance the spatial resolution of their relevant predictors in a reasonable amount of time, requiring also less computational resources. Moreover, the strength of ML algorithms relies on their generalization capabilities, depending on how well the training procedure has been performed.

The accuracy metrics on the test sets exhibited by the presented algorithms are quite acceptable, in some cases really satisfying. However, the used test sets refer to variables in the past. In order to assess predictive performance in the future, projections data are required, both as predictors and/or as predictands. For example, one might feed an already trained ML-powered downscaling approach with a variable, at a certain resolution, belonging to projection data to predict that variable at a higher resolution in the future. Furthermore, one can feed a ML Tropical Cyclones detection algorithm with projection variables at a future time instant to detect whether a cyclone is present or not. In these cases, the main underlying assumption for future predictions to be as accurate as those of the train and the test set, is that they should exhibit a probability distribution close to those of the train and test sets. Another issue is that, if some strong change does occur in a variable that has not been taken into account into the ML model, yet still having influence in the outcome, then the prediction may not be accurate.

A future work may be trying to assess performance on recent past data that has not been included in the test set because at the time of the presented ML algorithm development, has not been available yet. This should be done in order to assess whether, how and how much the previously assessed algorithms generalization capabilities deteriorate on unseen data not belonging to the train and the test sets.

Another future work for all the developed and presented ML algorithms will consist in assessing their generalization capabilities in space, given that they have been trained on a certain fixed domain, whether they can generalize also to other geographical domains not seen during the training phase. The considered test sets already do contain values different from those of the training set, coming from other years but considering the same geographical domain. It may happen that by focusing on other domains, other phenomena may emerge, whose data the algorithms have not been trained on. At best, this may require only a Transfer Learning to adjust the parameters of the networks in order to take into account the unconsidered probability distribution component exhibited by the new out of sample data. In some cases, a new training should be performed, maybe by considering larger spatial regions or global data. If some kind of normalization/scaling has been performed during the ML algorithm development, and this is the case of all the four presented algorithms, it is possible that the new domain data exhibit different extrema, leading to undefined numerical behavior. In this case, a new conservative normalization should be adopted, for example by not using the extrema of the training set, instead by considering higher values for maxima and lower values for minima.

Another possible drawback is that a ML prediction (i.e. a climate field) though it might appear very good from a perceptual standpoint, under some circumstances may violate, even locally in the space and even if by a little, some physical constraints or laws.
As previously discussed, a future work in this direction may be retraining the model by taking into account the desired physical constraint the outcomes should satisfy.
This can be mainly done through the modification of the loss function, eventually with the addition of proper terms, or even embedding these physical constraints under the form of layers into the network, and then letting it perform a normal training.

The main lesson learned in the Downscaling field yet valid in general is that, by looking at the obtained improvement of the AR-CNN-SD with respect to the MSG-GAN-SD, an architecture tailored for a complex task should not be chosen by looking only at its complexity or deepness, so thinking it will behave better. Clearly the model capacity (the number of the parameters, namely weights and biases, of a NN) should be compared with the number of data points in the training, and the ratio of these two quantities is a good indicator of the algorithm generalization capabilities. More data than parameters means that the algorithm might not please that quantity of variance exhibited by all the numerous data points, leading to underfitting. On the other hand, having more parameters than data means that all the model complexity is used to seal a tight bond with the data, leading to overfitting. However, in some adversarial architectures such as the MSG-GAN-SD, it is not trivial to assess the presence of overfitting. And this is exactly what was worsening these models' performance.
By having switched on a simpler but smarter architecture, which implements some powerful corrections, yet still remaining easily diagnosable about generalization capabilities, the predictive performance incredibly improved, even though the considered variable values in the test set exhibited a strong Climate Change signal, which has not been considered in the training set data instead.

Overall conclusion

In conclusion, the results reported in the present dissertation work demonstrate the effectiveness and value of ML techniques for application to several case studies related to the climate science domain. Specifically, GANs were adopted for generating high-resolution 2m-temperature maps starting from their coarse resolution counterparts at a fraction of the computational cost of actual operational downscaling approaches *(i)*,
cGANs with Pix2Pix-based generators were successfully used for predicting FWI risk maps by taking in input 4 climatic drivers *(ii)*, and Long Short-Term Memory neural networks were exploited to bridge temporal patterns in sea level time series data *(iii)*.
The results presented in this research dissertation, along with a ML State of the Art more oriented on an operative footprint, confirm that the research about the ML algorithms is now more mature and consolidated. Thus, the interest in the climate science domain has begun to step away from standalone applications and explorative case studies, approaching instead real use-cases.
Thus, ML algorithms demonstrated to be cost-effective and require less computational resources with respect to traditional physics-based model. Additionally, they are becoming even more faster to train, due to the voracious improvements that are affecting the new hybrid architectures, i.e. GPUs and TPUs. However, the agnostic use of ML as a black-box tool should be disencouraged, because it may lead to physically inconsistent outcomes and false discoveries. A far more secure and intelligent pathway might be strengthening the link between data-driven and physics-based complementary techniques.

Appendix

Appendix A. The MSG-GAN-SD Architecture

Tables A1 and A2 report the internal structure of the generator and the discriminator within the MSG-GAN-SD architecture, respectively.

Table A2 *Generator Architecture for the MSG-GAN-SD.*

Block	Layer	Activation	Output Shape
0.	Input	–	2 x 4 x 1
	Conv 1 x 1	LReLU	2 x 4 x 480
	Conv 2 x 2	LReLU	2 x 4 x 480
G_0	Conv 1 x 1	Tanh	2 x 4 x 1
1.	Input	–	2 x 4 x 480
	Upsampling (2, 2)	–	4 x 8 x 480
	Conv 3 x 3	LReLU	4 x 8 x 480
	Conv 3 x 3	LReLU	4 x 8 x 480
G_1	Conv 1 x 1	Tanh	4 x 8 x 1
2.	Input	–	4 x 8 x 480
	Upsampling (2, 2)	–	8 x 16 x 480
	Conv 3 x 3	LReLU	8 x 16 x 240
	Conv 3 x 3	LReLU	8 x 16 x 240
G_2	Conv 1 x 1	Tanh	8 x 16 x 1
3.	Input	–	8 x 16 x 240
	Upsampling (2, 2)	–	16 x 32 x 240
	Conv 3 x 3	LReLU	16 x 32 x 120
	Conv 3 x 3	LReLU	16 x 32 x 120
G_3	Conv 1 x 1	Tanh	16 x 32 x 1
4.	Input	–	16 x 32 x 120
	Upsampling (2, 2)	–	32 x 64 x 120
	Conv 3 x 3	LReLU	32 x 64 x 60
	Conv 3 x 3	LReLU	32 x 64 x 60
G_4	Conv 1 x 1	Tanh	32 x 64 x 1
5.	Input	–	32 x 64 x 60
	Upsampling (3, 3)	–	96 x 192 x 60
	Conv 3 x 3	LReLU	96 x 192 x 20
	Conv 3 x 3	LReLU	96 x 192 x 20
G_5	Conv 1 x 1	Tanh	96 x 192 x 1
6.	Input	–	96 x 192 x 20
	Upsampling (5, 5)	–	480 x 960 x 20
	Conv 3 x 3	LReLU	480 x 960 x 4
	Conv 3 x 3	LReLU	480 x 960 x 4
	Conv 3 x 3	Tanh	480 x 960 x 1

Table A3 *Discriminator Architecture for the MSG-GAN-SD.*

Block	Layer	Activation	Output Shape

0.	Input	–	480 x 960 x 1
	Conv 3 x 3	LReLU	480 x 960 x 4
	MiniBatchStd	–	480 x 960 x 5
	Conv 3 x 3	LReLU	480 x 960 x 4
	Conv 3 x 3	LReLU	480 x 960 x 20
	AvgPool (5, 5)	–	96 x 192 x 20
Aux_0	Auxiliary Image	–	96 x 192 x 1
1.	Input	–	96 x 192 x 20
	Concat	–	96 x 192 x 21
	MiniBatchStd	–	96 x 192 x 22
	Conv 3 x 3	LReLU	96 x 192 x 20
	Conv 3 x 3	LReLU	96 x 192 x 60
	AvgPool (3, 3)	–	32 x 64 x 60
Aux_1	Auxiliary Image	–	32 x 64 x 1
2.	Input	–	32 x 64 x 60
	Concat	–	32 x 64 x 61
	MiniBatchStd	–	32 x 64 x 62
	Conv 3 x 3	LReLU	32 x 64 x 60
	Conv 3 x 3	LReLU	32 x 64 x 120
	AvgPool (2, 2)	–	16 x 32 x 120
Aux_2	Auxiliary Image	–	16 x 32 x 1
3.	Input	–	16 x 32 x 120
	Concat	–	16 x 32 x 121
	MiniBatchStd	–	16 x 32 x 122
	Conv 3 x 3	LReLU	16 x 32 x 120
	Conv 3 x 3	LReLU	16 x 32 x 240
	AvgPool (2, 2)	–	8 x 16 x 240
Aux_3	Auxiliary Image	–	8 x 16 x 1
4.	Input	–	8 x 16 x 240
	Concat	–	8 x 16 x 241
	MiniBatchStd	–	8 x 16 x 242
	Conv 3 x 3	LReLU	8 x 16 x 240
	Conv 3 x 3	LReLU	8 x 16 x 480
	AvgPool (2, 2)	–	4 x 8 x 480
Aux_4	Auxiliary Image	–	4 x 8 x 1
5.	Input	–	4 x 8 x 480
	Concat	–	4 x 8 x 481
	MiniBatchStd	–	4 x 8 x 482
	Conv 3 x 3	LReLU	4 x 8 x 480
	Conv 3 x 3	LReLU	4 x 8 x 480
	AvgPool (2, 2)	–	2 x 4 x 480
Aux_5	Auxiliary Image	–	2 x 4 x 1
6.	Input	–	2 x 4 x 480
	Concat	–	2 x 4 x 481
	MiniBatchStd	–	2 x 4 x 482
	Conv 2 x 2	LReLU	2 x 4 x 480
	Conv 2 x 4	LReLU	1 x 1 x 480
	Fully Connected	Linear	1 x 1 x 1

Appendix B. Best Model Selection and Evaluation Results

The best model selection is carried out in two phases. In the first phase, a pool of model candidates is chosen using Equation (2.1). Specifically, for each model, the equation allows selecting the optimal epoch at which both the training error ($MSE_{tr,e}$) and the generalization error ($|MSE_{tr,e} - MSE_{cv,e}|$) are minimum, thus meaning that the training process should be stopped at the e_{best} epoch. Both λ_{tr} and λ_{tr-cv} are real numbers acting as penalty terms whose values can be arbitrarily selected to prioritize the minimization of $MSE_{tr,e}$ or the absolute difference between $MSE_{tr,e}$ and $MSE_{cv,e}$. For the sake of simplicity, in the proposed experiments λ_{tr} and λ_{tr-cv} were set to 1 in order to equally weight all the metrics involved in the computation. In some cases, a high degree of generalization is requested at the expense of the accuracy loss in approximating the real distribution, or vice versa. Once the selection of the best-epoch model was completed, a total of 48 models for the various seasons/DtrainUpdates and months/DtrainUpdates combinations were analyzed. When testing the models on the test set in the second phase, the best model for each season/month is obtained by selecting the discriminator that has the highest value computed using Equation (2.2) among the three discriminator update configurations (DtrainUpdates). In particular, Equation (2.2) is constructed by multiplying the *Accuracy* and the *Perceptivity* factors given by the Expressions (A1) and (A2), respectively.

$$Accuracy = \left(\frac{\lambda_{MSE}}{MSE}\right)\left(\lambda_{PSNR} PSNR\right)\left(\lambda_{SSIM} SSIM\right) \qquad (A1)$$

The Accuracy term measures the quantitative information of the image, such as the colors range and peaks information, in addition to the information related to the overall geometric structure of the image (such as lines, contours, polygons, etc.). It has to be as high as possible to obtain a good image quality. The Perceptivity term relates to the qualitative information of the image and is defined as:

$$Perceptivity = \left(\frac{1}{\lambda_{FID} FID}\right)\left(\frac{1}{\lambda_{LSD} LSD}\right) \tag{A2}$$

where a high perceptivity means that the image has good photorealistic features and that high frequency details have been successfully learnt during the training phase. In fact, LSD and FID should be as low as possible, so that their reciprocal will be high. The λ coefficients are normalizing factors, which act as weights with respect to the various involved terms, enabling the selection of a trade-off between Accuracy and Perceptivity factors, or even between any of their components. Table A3 shows, for each month and season, the number of discriminator updates for each epoch during the training phase (DtrainUpdates) and the epochs resulting from the minimization of Equation (2.1). For example, referring to August, the best monthly model was obtained by applying a single update/epoch of the discriminator (DtrainUpdates = 1), and then gathered at epoch 850 of the training. As shown in Table A3, most of the best models for both the monthly and seasonal training arrangements were obtained with a single update per epoch (DtrainUpdates = 1). Four best monthly models were obtained with two updates per epoch (DtrainUpdates = 2), whereas four best seasonal models and two best monthly models were achieved with three updates per epoch (DtrainUpdates = 3). It can thus be stated that, for this specific use case, there is no evidence of the improvement caused by training D more times than G in a single epoch with relation to the seasonal models. This may be due to the higher number of samples occurring in seasonal models, which can strongly widen the already existing architectural gap (i.e., number of parameters) between D and G. By comparison, the different DtrainUpdates variants in the monthly models are worth considering in order to improve performance, because it is not possible to highlight a clear majority pattern. The following tables report the outputs of the evaluation procedure for the monthly and seasonal models' outcomes, respectively. In Table A4, for each month, the best trained model was tested on the month itself through the test set years (2015–2018). The upward arrows mean that the metrics value has to be as high as possible, whereas the downward arrows mean the opposite. In Table A5 for each season, the best trained model was tested on each month belonging to the corresponding season, during the test set period (2015–2018). The upward arrows mean that the metrics value has to be as high as possible, whereas the downward arrows mean the opposite. All the errors and metrics shown in these tables are calculated on samples normalized in the [−1; 1] interval. Concerning the 5f_{APP} metric, all the normalizing coefficients λ in Equation (2.2) were set to 1, in order to give equal importance to all the involved terms.

Table B1 *Best models properties.*

Training set arrangements	Monthly		Seasonal	
Months	**# D updates**	**Epoch**	**# D updates**	**Epoch**
January	2	950	1	850
February	1	500	1	850
March	1	850	1	1000
April	2	600	1	1000
May	3	750	3	350
June	2	850	1	800
July	1	800	1	800
August	1	850	1	800
September	2	750	3	750
October	1	950	3	750
November	3	600	3	750
December	1	950	1	850

***Table B2** Evaluation results based on monthly models' outcomes.*

Month-based training	**MSE (↓)**	**PSNR (↑)**	**SSIM (↑)**	**FID (↓)**	**LSD (↓)**	**Accuracy (↑)**	**Perceptivity (↑)**	$5f_{APP}$ **(↑)**
January	0.012	17.478	0.811	0.099	8.294	1173.585	1.213	1423.007
February	0.010	18.052	0.834	0.062	8.086	1526.165	1.981	3023.297
March	0.009	18.562	0.842	0.051	8.918	1746.389	2.208	3856.081
April	0.009	18.507	0.839	0.047	9.761	1697.090	2.180	3699.364
May	0.010	18.137	0.799	0.066	9.505	1455.684	1.585	2307.581
June	0.006	20.284	0.831	0.067	9.192	2844.603	1.631	4639.858
July	0.006	20.009	0.812	0.064	9.245	2548.737	1.697	4325.116
August	0.005	20.692	0.836	0.045	7.960	3502.819	2.822	9885.056
September	0.007	19.840	0.894	0.063	7.767	2580.708	2.053	5298.983
October	0.007	20.040	0.905	0.046	8.378	2602.556	2.573	6697.147
November	0.009	19.367	0.877	0.091	7.893	1940.007	1.396	2708.218
December	0.010	18.699	0.853	0.082	8.416	1545.117	1.457	2251.966

***Table B3** Evaluation results based on seasonal models' outcomes.*

Season-based training	**MSE (↓)**	**PSNR (↑)**	**SSIM (↑)**	**FID (↓)**	**LSD (↓)**	**Accuracy (↑)**	**Perceptivity (↑)**	$5f_{APP}$ **(↑)**
January	0.011	17.584	0.817	0.079	8.760	1324.248	1.438	1904.269
February	0.009	18.414	0.843	0.061	8.388	1645.494	1.968	3239.104
March	0.008	17.586	0.797	0.041	9.871	1651.174	2.478	4091.508
April	0.007	18.837	0.873	0.042	10.120	2233.076	2.351	5249.693
May	0.007	20.051	0.924	0.053	9.689	2739.662	1.949	5340.450
June	0.005	20.450	0.829	0.046	9.170	3110.453	2.382	7407.763
July	0.005	21.084	0.873	0.037	8.699	3733.963	3.109	11607.422
August	0.004	21.395	0.877	0.039	7.316	4217.381	3.540	14929.312
September	0.005	21.618	0.958	0.048	7.047	3970.463	2.966	11777.042
October	0.005	20.731	0.908	0.043	7.538	3748.529	3.085	11565.111
November	0.007	18.326	0.825	0.066	7.740	2138.194	1.954	4178.579
December	0.009	18.677	0.852	0.072	8.973	1794.051	1.542	2767.413

Appendix C. The AR-CNN-SD Architecture

Table C1 reports the internal structure of the AR-CNN-SD architecture.

Table C1 Generator Architecture for the AR-GAN-SD.

ID	Layer	Activation	Output Shape	Params	Connected to
0	Input	–	80 x 160 x 1	_	–
1	Conv 9 x 9	P-ReLU	80 x 160 x 64	5312	0
2	Conv 3 x 3	–	80 x 160 x 64	36928	1
3	BatchNormalization	P-ReLU	80 x 160 x 64	320	2
4	Conv 3 x 3	–	80 x 160 x 64	36928	3
5	BatchNormalization	–	80 x 160 x 64	256	4
6	Add	–	80 x 160 x 64	–	1,5
7	Conv 3 x 3	–	80 x 160 x 64	36928	6
8	BatchNormalization	P-ReLU	80 x 160 x 64	320	7
9	Conv 3 x 3		80 x 160 x 64	36928	8
10	BatchNormalization	–	80 x 160 x 64	256	9
11	Add	–	80 x 160 x 64	–	6,10
12	Conv 3 x 3	–	80 x 160 x 64	36928	11
13	BatchNormalization	–	80 x 160 x 64	320	12
14	Conv 3 x 3	P-ReLU	80 x 160 x 64	36928	13
15	BatchNormalization	–	80 x 160 x 64	256	14
16	Add	–	80 x 160 x 64	–	11,15
17	Conv 3 x 3	–	80 x 160 x 64	36928	16
18	BatchNormalization	–	80 x 160 x 64	320	17
19	Conv 3 x 3	P-ReLU	80 x 160 x 64	36928	18
20	BatchNormalization	–	80 x 160 x 64	256	19
21	Add	–	80 x 160 x 64	–	16,20
22	Conv3 x 3	–	80 x 160 x 64	36928	21
23	BatchNormalization	–	80 x 160 x 64	320	22
24	Conv 3 x 3	P-ReLU	80 x 160 x 64	36928	23
25	BatchNormalization		80 x 160 x 64	256	24
26	Add	–	80 x 160 x 64	–	21,25
27	Conv 3 x 3	–	80 x 160 x 64	36928	26
28	BatchNormalization	–	80 x 160 x 64	320	27
29	Conv 3 x 3	–	80 x 160 x 64	36928	28
30	BatchNormalization	P-ReLU	80 x 160 x 64	256	29
31	Add	–	80 x 160 x 64		26,30

32	Conv 3 x 3	–	80 x 160 x 64	–	31
33	BatchNormalization	–	80 x 160 x 64	36928	32
34	Conv 3 x 3	–	80 x 160 x 64	320	33
35	BatchNormalization	P-ReLU	80 x 160 x 64	36928	34
36	Add	–	80 x 160 x 64	256	31,35
37	Conv 3 x 3	–	80 x 160 x 64	–	36
38	BatchNormalization	–	80 x 160 x 64	36928	37
39	Conv 3 x 3		80 x 160 x 64	320	38
40	BatchNormalization	–	80 x 160 x 64	36928	39
41	Add	P-ReLU	80 x 160 x 64	256	36,40
42	Conv 3 x 3	–	80 x 160 x 64	–	41
43	BatchNormalization	–	80 x 160 x 64	36928	42
44	Conv 3 x 3	–	80 x 160 x 64	320	43
45	BatchNormalization	–	80 x 160 x 64	36928	44
46	Add	P-ReLU	80 x 160 x 64	256	41,45
47	Conv 3 x 3	–	80 x 160 x 64	–	46
48	BatchNormalization	–	80 x 160 x 64	36928	47
49	Conv 3 x 3	–	80 x 160 x 64	320	48
50	BatchNormalization	–	80 x 160 x 64	36928	49
51	Add	P-ReLU	80 x 160 x 64	256	46,50
52	Conv 3 x 3	–	80 x 160 x 64	–	51
53	BatchNormalization	–	80 x 160 x 64	36928	52
54	Conv 3 x 3	–	80 x 160 x 64	320	53
55	BatchNormalization		80 x 160 x 64	36928	54
56	Add	–	80 x 160 x 64	256	51,55
57	Conv 3 x 3	P-ReLU	80 x 160 x 64	–	56
58	BatchNormalization	–	80 x 160 x 64	36928	57
59	Conv 3 x 3	–	80 x 160 x 64	320	58
60	BatchNormalization	–	80 x 160 x 64	36928	59
61	Add	–	80 x 160 x 64	256	56,60
62	Conv 3 x 3	P-ReLU	80 x 160 x 64	–	61
63	BatchNormalization	–	80 x 160 x 64	36928	62
64	Conv 3 x 3	–	80 x 160 x 64	320	63
65	BatchNormalization	–	80 x 160 x 64	36928	64
66	Add	–	80 x 160 x 64	256	61,65
67	Conv 3 x 3	P-ReLU	80 x 160 x 64	–	66
68	BatchNormalization	–	80 x 160 x 64	36928	67
69	Conv 3 x 3	–	80 x 160 x 64	320	68
70	BatchNormalization		80 x 160 x 64		69

		–			
		–			
		P-ReLU		36928	
71	Add	–	80 x 160 x 64	256	66,70
72	Conv 3 x 3	–	80 x 160 x 64	–	71
73	BatchNormalization	–	80 x 160 x 64	36928	72
74	Conv 3 x 3	–	80 x 160 x 64	320	73
75	BatchNormalization	P-ReLU	80 x 160 x 64	36928	74
76	Add	–	80 x 160 x 64	256	71,75
77	Conv 3 x 3	–	80 x 160 x 64	–	76
78	BatchNormalization	–	80 x 160 x 64	36928	77
79	Conv 3 x 3	–	80 x 160 x 64	320	78
80	BatchNormalization	P-ReLU	80 x 160 x 64	36928	79
81	Add	–	80 x 160 x 64	256	76,80
82	Conv 3 x 3	–	80 x 160 x 64	–	81
83	BatchNormalization	–	80 x 160 x 64	36928	82
84	Add	–	80 x 160 x 64	256	1,83
85	Conv 3 x 3	–	80 x 160 x 256	–	84
86	Upsampling (2,2)	–	160 x 320 x 256	147712	85
87	Conv 3 x 3	–	160 x 320 x 256	–	86
88	Upsampling (3,3)	LReLU	480 x 960 x 256	590080	87
89	Conv 9 x 9	–	480 x 960 x 1	–	88
90	Upsampling (6,6)	LReLU	480 x 960 x 1	20737	0
–	Add	Tanh	480 x 960 x 1	–	89,90
		Tanh		–	
		–			

Appendix D. Best Model Selection and Evaluation Results

Table D1 *Best models properties.*

Training set arrangements	Monthly	Seasonal
Months	**Epoch**	**Epoch**
January	350	150
February	450	150
March	300	100
April	400	100
May	300	100
June	350	50
July	300	50
August	300	50
September	400	50
October	250	50
November	150	50
December	500	150

Table D2 *Evaluation results based on monthly models' outcomes.*

Month-based training	MSE (↓)	PSNR (↑)	SSIM (↑)	FID (↓)	LSD (↓)	Accuracy (↑)	Perceptivity (↑)	5f_{APP} (↑)
January	0.020	67.000	0.964	0.473	7.118	3175.721	0.297	942.442
February	0.011	69.813	0.980	0.507	6.969	6166.616	0.283	1746..127
March	0.017	68.913	0.975	0.749	7.152	3846.418	0.187	718.246
April	0.010	70.779	0.979	0.195	7.399	7211.429	0.693	4988.882
May	0.026	71.414	0.978	0.315	7.709	2665.399	0.412	1097.201
June	0.011	72.209	0.980	0.098	7.767	6649.542	1.309	8703.023
July	0.006	72.260	0.978	0.042	7.425	11186.167	3.177	35533.626
August	0.008	71.853	0.977	0.113	6.585	8896.031	1.342	11941.211
September	0.005	73.478	0.983	0.057	5.594	15255.135	3.123	47637.374
October	0.010	70.303	0.974	0.282	6.007	7059.240	0.590	4161.442
November	0.017	67.560	0.964	0.536	6.915	3874.822	0.270	1044.807
December	0.026	66.350	0.968	0.594	6.758	2478.868	0.249	617.956

Table D3 *Evaluation results based on seasonal models' outcomes.*

Season-based training	**MSE (↓)**	**PSNR (↑)**	**SSIM (↑)**	**FID (↓)**	**LSD (↓)**	**Accuracy (↑)**	**Perceptivity (↑)**	$5f_{APP}$ **(↑)**
January	0.010	70.254	0.983	0.395	6.887	6881.912	0.368	2530.789
February	0.010	70.287	0.981	0.382	6.865	7152.318	0.381	2726.729
March	0.034	68.753	0.974	0.644	7.228	1984.958	0.215	426.218
April	0.009	70.527	0.975	0.191	7.500	7384.598	0.699	5163.598
May	0.008	70.947	0.975	0.066	7.815	8170.544	1.948	15915.535
June	0.012	69.270	0.964	0.064	8.081	5455.951	1.923	10489.157
July	0.011	69.546	0.964	0.049	7.773	5857.610	2.602	15238.721
August	0.011	69.651	0.965	0.102	7.099	5988.214	1.385	8293.969
September	0.012	69.403	0.965	0.110	6.185	5711.924	1.468	8382.699
October	0.011	69.780	0.963	0.136	6.487	6210.515	1.134	7042.352
November	0.017	67.654	0.963	0.752	7.045	3749.283	0.189	707.939
December	*0.009*	*70.705*	*0.983*	*0.309*	*6.500*	*7696.617*	*0.497*	*3828.115*

References

Abadi, M. et al. “TensorFlow: A System for Large-Scale Machine Learning”, *In Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI ’16)*, Savannah, GA, USA, 2–4 November, 2016.

Abadi, M. et al., “TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems”, arXiv.org, 2016. [Online]. Available: https://arxiv.org/abs/1603.04467. [Accessed: 20- Dec- 2021].

Ali Ghorbani, M., Khatibi, R., Aytek, A., Makarynskyy, O. and Shiri, J., “Sea water level forecasting using genetic programming and comparing the performance with Artificial Neural Networks”, *Computers & Geosciences*, vol. 36, no. 5, pp. 620-627, 2010. doi: 10.1016/j.cageo.2009.09.014

Anderson, G. J. and Lucas, D. D., “Machine Learning Predictions of a Multiresolution Climate Model Ensemble”, *Geophysical Research Letters*, vol. 45, no. 9, pp. 4273-4280, 2018. doi: 10.1029/2018gl077049

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*, vol. 47, no. 9, 2020. doi: 10.1029/2020gl087776

Arjovsky, M., Chintala, S. and Bottou, L., “Wasserstein GAN”, *arXiv.org*, 2017. [Online]. Available: https://arxiv.org/abs/1701.07875. [Accessed: 22- Dec- 2021].

Baño-Medina, J., Gutiérrez, J., Herrera, S., “Deep Neural Networks for Statistical Downscaling of Climate Change Projections”, In Proceedings of the XVIII Conferencia de la Asociación Española para la Inteligencia Artificial (CAEPIA 2018), Granada, España, 23–26 October 2018; pp. 1419–1424.

Baño-Medina, J., Manzanas, R. and Gutiérrez, J. M., “Configuration and intercomparison of deep learning neural models for statistical downscaling”, *Geoscientific Model Development*, vol. 13, no. 4, pp. 2109-2124, 2020. doi: 10.5194/gmd-13-2109-2020

Barnes, E. A., Hurrell, J. W., Ebert-Uphoff, I., Anderson, C. and Anderson, D., “Viewing Forced Climate Patterns Through an AI Lens”, *Geophysical Research Letters*, vol. 46, no. 22, pp. 13389-13398, 2019. doi: 10.1029/2019gl084944

Barnes, E. A., Toms, B., Hurrell, J. W., Ebert-Uphoff, I., Anderson, C. and Anderson, D., “Indicator Patterns of Forced Change Learned by an Artificial Neural Network”, *Journal of Advances in Modeling Earth Systems*, vol. 12, no. 9, 2020. doi: 10.1029/2020ms002195

Bartkowiak, P., Castelli, M. and Notarnicola, C., “Downscaling Land Surface Temperature from MODIS Dataset with Random Forest Approach over Alpine

Vegetated Areas", *Remote Sensing*, vol. 11, no. 11, p. 1319, 2019. doi: 10.3390/rs11111319

Bauer, P., Dueben, P. D., Hoefler, T., Quintino, T., Schulthess, T. and Wedi, N. P., "The digital revolution of Earth-system science", *Nature Computational Science*, vol. 1, no. 2, pp. 104-113, 2021. doi: 10.1038/s43588-021-00023-0

Bengio, Y., Simard, P. and Frasconi, P., "Learning long-term dependencies with gradient descent is difficult", IEEE Transactions on Neural Networks, vol. 5, no. 2, pp. 157-166, 1994. doi: 10.1109/72.279181

Boer, G. and Yu, B., "Climate sensitivity and response", *Climate Dynamics*, vol. 20, no. 4, pp. 415-429, 2003. doi: 10.1007/s00382-002-0283-3

Boers, N., Goswami, B., Rheinwalt, A., Bookhagen, B., Hoskins, B. and Kurths, J., "Complex networks reveal global pattern of extreme-rainfall teleconnections", *Nature*, vol. 566, no. 7744, pp. 373-377, 2019. doi: 10.1038/s41586-018-0872-x

Braakmann-Folgmann, A., Roscher, R., Wenzel, S., Uebbing, B. and Kusche, J., "Sea Level Anomaly Prediction using Recurrent Neural Networks", *arXiv.org*, 2017. [Online]. Available: https://arxiv.org/abs/1710.07099. [Accessed: 17- Dec- 2021].

Brown, R. D. and Robinson, D. A., "Northern Hemisphere spring snow cover variability and change over 1922–2010 including an assessment of uncertainty", *The Cryosphere*, vol. 5, no. 1, pp. 219-229, 2011. doi: 10.5194/tc-5-219-2011

Brown, T. B. et al., "Language Models are Few-Shot Learners", *arXiv.org*, 2020. [Online]. Available: https://arxiv.org/abs/2005.14165. [Accessed: 01- Dec- 2021].

Caldwell, P., Bretherton, C., Zelinka, M., Klein, S., Santer, B. and Sanderson, B., "Statistical significance of climate sensitivity predictors obtained by data mining", *Geophysical Research Letters*, vol. 41, no. 5, pp. 1803-1808, 2014. doi: 10.1002/2014gl059205

CAMS, 2021a. "Copernicus Atmosphere Monitoring Service (CAMS), Atmosphere Data Store". [Online]. Available: https://ads.atmosphere.copernicus.eu/#!/home. [Accessed: 03- Jan- 2022].

CAMS, 2021b. "Copernicus Atmosphere Monitoring Service (CAMS), CAMS European air quality forecasts". [Online]. Available: https://ads.atmosphere.copernicus.eu/cdsapp#!/dataset/cams-europe-air-quality-forecasts?tab=overview [Accessed: 03- Jan- 2022].

Carman, J. C. et al., "Position paper on high performance computing needs in Earth system prediction". *National Earth System Prediction Capability*. Technical report, 2017. doi: 10.7289/V5862DH3

Cavalieri, D. J. and Parkinson C. L., “Arctic sea ice variability and trends, 1979–2010”, *The Cryosphere*, vol. 6, no. 4, pp. 881-889, 2012. doi: 10.5194/tc-6-881-2012

Cazzolla Gatti, R., Velichevskaya, A., Tateo, A., Amoroso, N. and Monaco, A., “Machine learning reveals that prolonged exposure to air pollution is associated with SARS-CoV-2 mortality and infectivity in Italy”, *Environmental Pollution*, vol. 267, p. 115471, 2020. doi: 10.1016/j.envpol.2020.115471

Chang, Y., Acierto, R., Itaya, T., Akiyuki, K., Tung, C., “A Deep Learning Approach to Downscaling Precipitation and Temperature over Myanmar”, *EGU Gen. Assem. Conf. Abstr*, 4120, 2018.

Chantry, M., Hatfield, S., Dueben, P., Polichtchouk, I. and Palmer, T., “Machine Learning Emulation of Gravity Wave Drag in Numerical Weather Forecasting”, *Journal of Advances in Modeling Earth Systems*, vol. 13, no. 7, 2021. doi: 10.1029/2021ms002477

Chattopadhyay, A., Hassanzadeh, P. and Pasha, S., “Predicting clustered weather patterns: A test case for applications of convolutional neural networks to spatio-temporal climate data”, *Scientific Reports*, vol. 10, no. 1, 2020. doi: 10.1038/s41598-020-57897-9

CINECA, 2021. “Cineca”, *Cineca*, 2021. [Online]. Available: https://www.cineca.it. [Accessed: 22- Dec- 2021].

Clementi, E. et al., “Mediterranean Sea Analysis and Forecast (CMEMS MED-Currents 2016-2019)”, *Earth-prints.org*, 2019. [Online]. Available: https://www.earth-prints.org/bitstream/2122/13246/1/CMEMS-MED-PUM-006-013.pdf. [Accessed: 14- Dec- 2021].

CMEMS, 2021. “Copernicus Marine Environment Monitoring Service”, Marine.copernicus.eu, 2021. [Online]. Available: https://marine.copernicus.eu. [Accessed: 17- Dec- 2021].

Cohen, J. et al., “S2S reboot: An argument for greater inclusion of machine learning in subseasonal to seasonal forecasts”, *WIREs Climate Change*, vol. 10, no. 2, 2018. doi: 10.1002/wcc.567

Comiso, J. C., Parkinson, C. L., Gersten, R. and Stock, L., “Accelerated decline in the Arctic sea ice cover”, *Geophysical Research Letters*, vol. 35, no. 1, 2008. doi: 10.1029/2007gl031972

Copernicus CDS, 2021. Copernicus Climate Data Store, "ERA5-Land monthly averaged data from 1981 to present". [Online]. Available: https://cds.climate.copernicus.eu/cdsapp#!/dataset/reanalysis-era5-land-monthly-means?tab=overview [Accessed on 19*th* July 2021].

Coulibaly, P., “Downscaling daily extreme temperatures with genetic programming”, *Geophysical Research Letters*, vol. 31, no. 16, 2004. doi: 10.1029/2004gl020075

Cucco, A. et al., “Support to oil spill emergencies in the Bonifacio Strait, western Mediterranean”, *Ocean Science*, vol. 8, no. 4, pp. 443-454, 2012. doi: 10.5194/os-8-443-2012

D’Agostino McGowan, L., Greevy, R., “Contextualizing E-values for interpretable sensitivity to unmeasured confounding analyses”, *arXiv preprint arXiv:2011.07030*, 2020.

Dauphin, Y. N., de Vries, H., Chung, J. and Bengio, Y., “RMSProp and equilibrated adaptive learning rates for non-convex optimization”, *arXiv.org*, 2015. [Online]. Available: https://arxiv.org/abs/1502.04390v1. [Accessed: 22- Dec- 2021].

Dawson, M., Olvera, J., Fung, A. and Manry, M., “Inversion of Surface Parameters Using Fast Learning Neural Networks”, *[Proceedings] IGARSS '92 International Geoscience and Remote Sensing Symposium*, 1992. doi: 10.1109/igarss.1992.578294

De Gooijer, J. G. and Hyndman, R. J., “25 years of time series forecasting”, *International Journal of Forecasting*, vol. 22, no. 3, pp. 443-473, 2006. doi: 10.1016/j.ijforecast.2006.01.001

Denton, E., Chintala, S., Szlam, A. and Fergus, R., “Deep generative image models using a Laplacian pyramid of adversarial networks”, *In Proceedings of the 28th International Conference on Neural Information Processing Systems (NIPS'15)*, Montréal, QC, Canada, 7-12 December 2015; Volume 1, pp. 1486-1494.

Dinh, L., Krueger, D. and Bengio, Y., “NICE: Non-linear Independent Components Estimation”, *In Proceedings of the International Conference on Learning Representations (ICLR)*, San Diego, CA, USA, 7-9 May 2015.

Dobricic, S. and Pinardi, N., “An oceanographic three-dimensional variational data assimilation scheme”, *Ocean Modelling*, vol. 22, no. 3-4, pp. 89-105, 2008. doi: 10.1016/j.ocemod.2008.01.004

Dong, C., Loy, C. C., He, K. and Tang, X., “Image Super-Resolution Using Deep Convolutional Networks”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 2, pp. 295-307, 2016. doi: 10.1109/tpami.2015.2439281

Dong, C., Loy, C. C., He, K. and Tang, X., “Learning a Deep Convolutional Network for Image Super-Resolution”, *Computer Vision – ECCV 2014*, pp. 184-199, 2014. doi: 10.1007/978-3-319-10593-2_13

Dongarra, J. et al., “The International Exascale Software Project roadmap”, *The International Journal of High Performance Computing Applications*, vol. 25, no. 1, pp. 3-60, 2011. doi: 10.1177/1094342010391989

Duhan, D. and Pandey, A., “Statistical downscaling of temperature using three techniques in the Tons River basin in Central India”, *Theoretical and Applied Climatology*, vol. 121, no. 3-4, pp. 605-622, 2014. doi: 10.1007/s00704-014-1253-5

Ebert-Uphoff, I. et al., “A vision for the development of benchmarks to bridge geoscience and data science”. *17th International Workshop on Climate Informatics*, 2017.

EEA, 2020a. “European Environment Agency, Air Quality Index”. [Online]. Available: https://www.eea.europa.eu/themes/air/air-quality-index. [Accessed: 03- Jan-2022].

ERA-Interim, 2021a. “ECMWF | ERA Interim, Daily”, *Apps.ecmwf.int*, 2021. [Online]. Available: https://apps.ecmwf.int/datasets/data/interim-full-daily/levtype=sfc/. [Accessed: 21- Dec- 2021].

ERA-Interim, 2021b. “ERA-Interim”, *ECMWF*, 2021. [Online]. Available: https://www.ecmwf.int/en/forecasts/datasets/reanalysis-datasets/era-interim. [Accessed: 21- Dec- 2021].

EURO-CORDEX, 2021. “EURO-CORDEX”, *Euro-cordex.net*, 2021. [Online]. Available: https://euro-cordex.net/index.php.en. [Accessed: 21- Dec- 2021].

Fahey, D., Doherty, S., Hibbard, K., Romanou A. and Taylor, P., “Ch. 2: Physical Drivers of Climate Change. Climate Science Special Report: Fourth National Climate Assessment, Volume I”, 2017. doi: 10.7930/j0513wcr

Federico, I., Pinardi, N., Coppini, G., Oddo, P., Lecci, R. and Mossa, M., “Coastal ocean forecasting with an unstructured grid model in the southern Adriatic and northern Ionian seas”, *Natural Hazards and Earth System Sciences*, vol. 17, no. 1, pp. 45-59, 2017. doi: 10.5194/nhess-17-45-2017

Filippo, A., Rebelo Torres, A., Kjerfve, B. and Monat, A., “Application of Artificial Neural Network (ANN) to improve forecasting of sea level”, *Ocean & Coastal Management*, vol. 55, pp. 101-110, 2012. doi: 10.1016/j.ocecoaman.2011.09.007

Findlay, H. S. and Turley, C., “Ocean acidification and climate change”, *Climate Change*, pp. 251-279, 2021. doi: 10.1016/b978-0-12-821575-3.00013-x

Fiore, S. and Aloisio, G., “Special section: Data management for eScience”, *Future Generation Computer Systems*, vol. 27, no. 3, pp. 290-291, 2011. doi: 10.1016/j.future.2010.08.012

Fiore, S. et al., “Towards High Performance Data Analytics for Climate Change”, *Lecture Notes in Computer Science*, pp. 240-257, 2019. doi: 10.1007/978-3-030-34356-9_20

Fiore, S., D’Anca, A., Palazzo, C., Foster, I., Williams, D. and Aloisio, G., “Ophidia: Toward Big Data Analytics for eScience”, *Procedia Computer Science*, vol. 18, pp. 2376-2385, 2013. doi: 10.1016/j.procs.2013.05.409

Forget, G., Campin, J., Heimbach, P., Hill, C. N., Ponte, R. M. and Wunsch, C., “ECCO version 4: an integrated framework for non-linear inverse modeling and global ocean state estimation”, *Geoscientific Model Development*, vol. 8, no. 10, pp. 3071-3104, 2015. doi: 10.5194/gmd-8-3071-2015

Fowler, H. J., Blenkinsop, S. and Tebaldi, C., “Linking climate change modelling to impacts studies: recent advances in downscaling techniques for hydrological modelling”, *International Journal of Climatology*, vol. 27, no. 12, pp. 1547-1578, 2007. doi: 10.1002/joc.1556

Gagne, D. J., McGovern, A., Haupt, S. E., Sobash, R. A., Williams, J. K. and Xue, M., “Storm-Based Probabilistic Hail Forecasting with Machine Learning Applied to Convection-Allowing Ensembles”, *Weather and Forecasting*, vol. 32, no. 5, pp. 1819-1840, 2017. doi: 10.1175/waf-d-17-0010.1

Gagné, M., Gillett, N. P. and Fyfe, J. C., “Observed and simulated changes in Antarctic sea ice extent over the past 50 years”, *Geophysical Research Letters*, vol. 42, no. 1, pp. 90-95, 2015. doi: 10.1002/2014gl062231

Gao, L., Schulz, K. and Bernhardt, M., “Statistical Downscaling of ERA-Interim Forecast Precipitation Data in Complex Terrain Using LASSO Algorithm”, *Advances in Meteorology*, vol. 2014, pp. 1-16, 2014. doi: 10.1155/2014/472741

Gentine, P., Pritchard, M., Rasp, S., Reinaudi, G. and Yacalis, G., “Could Machine Learning Break the Convection Parameterization Deadlock?”, *Geophysical Research Letters*, vol. 45, no. 11, pp. 5742-5751, 2018. doi: 10.1029/2018gl078202

Gers, F., “Learning to forget: continual prediction with LSTM”, 9th International Conference on Artificial Neural Networks: ICANN '99, 1999. doi: 10.1049/cp:19991218

Gettelman, A. and Rood, R. B., “Demystifying climate models”. *Springer*, 2016.

Gettelman, A. et al., “Machine Learning the Warm Rain Process”, *Journal of Advances in Modeling Earth Systems*, vol. 13, no. 2, 2021. doi: 10.1029/2020ms002268

Ghorbani, M., Makarynskyy, O., Shiri, J. and Makarynska, D., “Genetic Programming for Sea Level Predictions in an Island Environment”, *The International Journal of Ocean and Climate Systems*, vol. 1, no. 1, pp. 27-35, 2010. doi: 10.1260/1759-3131.1.1.27

Gil, Y. et al., “Intelligent systems for geosciences”, *Communications of the ACM*, vol. 62, no. 1, pp. 76-84, 2018. doi: 10.1145/3192335

Gillett, N. P., “Testing the linearity of the response to combined greenhouse gas and sulfate aerosol forcing”, *Geophysical Research Letters*, vol. 31, no. 14, 2004. doi: 10.1029/2004gl020111

Goly, A., Teegavarapu, R. S. V. and Mondal, A., “Development and Evaluation of Statistical Downscaling Models for Monthly Precipitation”, *Earth Interactions*, vol. 18, no. 18, pp. 1-28, 2014. doi: 10.1175/ei-d-14-0024.1

Goodfellow, I. et al., “Generative adversarial networks”, *Communications of the ACM*, vol. 63, no. 11, pp. 139-144, 2020. doi: 10.1145/3422622

Goosse, H., Barriat, P. Y., Lefebvre, W., Loutre, M. F. and Zunz V., “Introduction to Clime Dynamics and Climate Modeling”. *Cambridge University Press*, 2010.

Grabherr, G., Gottfried, M. and Pauli, H., “Climate Change Impacts in Alpine Environments”, *Geography Compass*, vol. 4, no. 8, pp. 1133-1153, 2010. doi: 10.1111/j.1749-8198.2010.00356.x

Groenke, B., Madaus, L. and Monteleoni, C., “ClimAlign: Unsupervised statistical downscaling of climate variables via normalizing flows”, *Proceedings of the 10th International Conference on Climate Informatics*, 2020. doi: 10.1145/3429309.3429318

Groisman, P., Karl, T., and Knight, R., “Observed Impact of Snow Cover on the Heat Balance and the Rise of Continental Spring Temperatures”, *Science*, vol. 263, no. 5144, pp. 198-200, 1994. doi: 10.1126/science.263.5144.198

Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., Courville, A., “Improved Training of Wasserstein GANs”, *Adv. Neural Inf. Process. Syst. 2017*, 30, 5767–5777, 2017.

Hagerty, A. and Rubinov, I., “Global AI Ethics: A Review of the Social Impacts and Ethical Implications of Artificial Intelligence”, *arXiv.org*, 2019. [Online]. Available: https://arxiv.org/abs/1907.07892. [Accessed: 01- Dec- 2021].

Hallegatte, S. et al., “Assessing climate change impacts, sea level rise and storm surge risk in port cities: a case study on Copenhagen”, *Climatic Change*, vol. 104, no. 1, pp. 113-137, 2010. doi: 10.1007/s10584-010-9978-3

Ham, Y., Kim, J. and Luo, J., “Deep learning for multi-year ENSO forecasts”, *Nature*, vol. 573, no. 7775, pp. 568-572, 2019. doi: 10.1038/s41586-019-1559-7

Hanna, E. et al., “Ice-sheet mass balance and climate change”, *Nature*, vol. 498, no. 7452, pp. 51-59, 2013. doi: 10.1038/nature12238

Harrould-Kolieb, E. and Herr, D., “Ocean acidification and climate change: synergies and challenges of addressing both under the UNFCCC”, *Climate Policy*, vol. 12, no. 3, pp. 378-389, 2012. doi: 10.1080/14693062.2012.620788

Hayhoe, K. et al., “Ch. 4: Climate Models, Scenarios, and Projections. Climate Science Special Report: Fourth National Climate Assessment, Volume I”, 2017. doi: 10.7930/j0wh2n54

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B. and Hochreiter, S., “GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium”, *Adv. Neural Inf. Process. Syst.*, 30, 6626-6637, 2017.

Hochreiter, S. and Schmidhuber, J., “Long Short-Term Memory”, *Neural Computation*, vol. 9, no. 8, pp. 1735-1780, 1997. doi: 10.1162/neco.1997.9.8.1735

Höhlein, K., Kern, M., Hewson, T. and Westermann, R., “A comparative study of convolutional neural network models for wind field downscaling”, *Meteorological Applications*, vol. 27, no. 6, 2020. doi: 10.1002/met.1961

Hourdin, F. et al., “The Art and Science of Climate Model Tuning”, *Bulletin of the American Meteorological Society*, vol. 98, no. 3, pp. 589-602, 2017. doi: 10.1175/bams-d-15-00135.1

Hsieh, W. W. and Tang, B., “Applying Neural Network Models to Prediction and Data Analysis in Meteorology and Oceanography”, *Bulletin of the American Meteorological Society*, vol. 79, no. 9, pp. 1855-1870, 1998. doi: 10.1175/1520-0477(1998)079<1855:annmtp>2.0.co;2

Hsieh, W. W., *Machine learning methods in the environmental sciences*. Cambridge: Cambridge University Press, 2009.

Huang, W., Murray, C., Kraus, N. and Rosati, J., “Development of a regional neural network for coastal water level predictions”, *Ocean Engineering*, vol. 30, no. 17, pp. 2275-2295, 2003. doi: 10.1016/s0029-8018(03)00083-0

Huang, X., “Deep-learning based climate downscaling using the super-resolution method: a case study over the western US”, 2020. doi: 10.5194/gmd-2020-214

Hwang, J., Orenstein, P., Cohen, J., Pfeiffer, K. and Mackey, L., “Improving Subseasonal Forecasting in the Western U.S. with Machine Learning”, *arXiv.org*, 2019. [Online]. Available: https://arxiv.org/abs/1809.07394. [Accessed: 13- Dec- 2021].

Il Sole 24 ore, 2021a. "Indice del tempo libero", [Online]. Available: https://lab24.ilsole24ore.com/indice-tempolibero/indexT.php [Accessed on 19*th* July 2021].

Il Sole 24 ore, 2021b. "La mappa dei redditi nelle città - Il Sole 24 Ore". [Online]. Available: https://lab24.ilsole24ore.com/mappaRedditi/redditiTabelle.html [Accessed on 19*th* July 2021].

Il Sole 24 ore, 2021c. "Indice della salute", [Online]. Available: https://lab24.ilsole24ore.com/indice-della-salute/indexT.php. [Accessed on 19*th* July 2021].

Il Sole 24 ore, 2021d. "Indice del clima", [Online]. Available: https://lab24.ilsole24ore.com/indice-del-clima/indexT.php [Accessed on 19*th* July 2021].

Imani, M., Kao, H., Lan, W. and Kuo, C., “Daily sea level prediction at Chiayi coast, Taiwan using extreme learning machine and relevance vector machine”, *Global and Planetary Change*, vol. 161, pp. 211-221, 2018. doi: 10.1016/j.gloplacha.2017.12.018

Iovino, D., Masina, S., Storto, A., Cipollone, A. and Stepanov, V. N., “A 1/16° eddying simulation of the global NEMO sea-ice–ocean system”, *Geoscientific Model Development*, vol. 9, no. 8, pp. 2665-2684, 2016. doi: 10.5194/gmd-9-2665-2016

IPCC, 2013. Climate Change 2013: The Physical Science Basis. Contribution of Working Group I to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change [Stocker, T.F., D. Qin, G.-K. Plattner, M. Tignor, S.K. Allen, J. Boschung, A. Nauels, Y. Xia, V. Bex and P.M. Midgley (eds.)]. *Cambridge University Press*, Cambridge, United Kingdom and New York, NY, USA, Chapter 12, pp. 1054, Figure 12.5.

IPCC, 2019. “IPCC Special Report on the Ocean and Cryosphere in a Changing Climate*”* [H.-O. Pörtner, D.C. Roberts, V. Masson-Delmotte, P. Zhai, M. Tignor, E. Poloczanska, K. Mintenbeck, A. Alegría, M. Nicolai, A. Okem, J. Petzold, B. Rama, N.M. Weyer (eds.)]. In press.

John, D., “Artificial Intelligence in Geoscience and Remote Sensing”, *Geoscience and Remote Sensing New Achievements*, 2010. doi: 10.5772/9104

Johnson, J., Alahi, A. and Fei-Fei, L., “Perceptual Losses for Real-Time Style Transfer and Super-Resolution”, *Computer Vision – ECCV 2016*, pp. 694-711, 2016. doi: 10.1007/978-3-319-46475-6_43

Jones, A., Haywood, J. M. and Boucher, O., “Aerosol forcing, climate response and climate sensitivity in the Hadley Centre climate model”, *Journal of Geophysical Research*, vol. 112, no. 20, 2007. doi: 10.1029/2007jd008688

Kadow, C., Hall, D. M. and Ulbrich, U. “Artificial intelligence reconstructs missing climate information”, *Nature Geoscience*, vol. 13, no. 6, pp. 408-413, 2020. doi: 10.1038/s41561-020-0582-5

Karimi, S., Kisi, O., Shiri, J. and Makarynskyy, O., “Neuro-fuzzy and neural network techniques for forecasting sea level in Darwin Harbor, Australia”, *Computers & Geosciences*, vol. 52, pp. 50-59, 2013. doi: 10.1016/j.cageo.2012.09.015

Karnewar, A. and Wang, O., “MSG-GAN: Multi-Scale Gradients for Generative Adversarial Networks”, *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. doi: 10.1109/cvpr42600.2020.00782

Karniadakis, G., Kevrekidis, I., Lu, L., Perdikaris, P., Wang, S. and Yang, L., “Physics-informed machine learning”, *Nature Reviews Physics*, vol. 3, no. 6, pp. 422-440, 2021. doi: 10.1038/s42254-021-00314-5

Karras, T., Aila, T., Laine, S. and Lehtinen, J., “Progressive Growing of GANs for Improved Quality, Stability, and Variation”, *arXiv.org*, 2018. [Online]. Available: https://arxiv.org/abs/1710.10196. [Accessed: 21- Dec- 2021].

Keras, 2015. “GitHub - keras-team/keras: Deep Learning for humans”, GitHub, 2015. [Online]. Available: https://github.com/keras-team/keras. [Accessed: 20- Dec-2021].

KERAS, 2021. K. Team, “Keras: the Python deep learning API”, *Keras.io*, 2021. [Online]. Available: https://keras.io. [Accessed: 22- Dec- 2021].

Kern, M., Höhlein, K., Hewson, T. and Westermann, R., “Towards Operational Downscaling of Low Resolution Wind Fields using Neural Networks”, 2020. doi: 10.5194/egusphere-egu2020-5447

Kim, H., Choi, M., Lim, B. and Mu Lee, K., “Task-Aware Image Downscaling”, *Computer Vision – ECCV 2018*, pp. 419-434, 2018. doi: 10.1007/978-3-030-01225-0_25

Kim, J., Lee, J. K. and Lee, K. M., “Accurate Image Super-Resolution Using Very Deep Convolutional Networks”, *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. doi: 10.1109/cvpr.2016.182

Kingma, D. and Ba, J., “Adam: A Method for Stochastic Optimization”, arXiv.org, 2017. [Online]. Available: https://arxiv.org/abs/1412.6980. [Accessed: 20- Dec-2021].

Kinney, P. L., “Interactions of Climate Change, Air Pollution, and Human Health”, *Current Environmental Health Reports*, vol. 5, no. 1, pp. 179-186, 2018. doi: 10.1007/s40572-018-0188-x

Knutti, R., Stocker, T. F., Joos, F. and Plattner, G., “Probabilistic climate change projections using neural networks”, *Climate Dynamics*, vol. 21, no. 3-4, pp. 257-272, 2003. doi: 10.1007/s00382-003-0345-1

Kopp, R. E. et al., “Evolving Understanding of Antarctic Ice-Sheet Physics and Ambiguity in Probabilistic Sea-Level Projections”, *Earth's Future*, vol. 5, no. 12, pp. 1217-1233, 2017. doi: 10.1002/2017ef000663

Krasnopolsky, V. M., “Neural Network Applications to Solve Forward and Inverse Problems in Atmospheric and Oceanic Satellite Remote Sensing”, *Artificial Intelligence Methods in the Environmental Sciences*, pp. 191-205, 2002. doi: 10.1007/978-1-4020-9119-3_9

Kurth, T., et al., “Exascale deep learning for climate analytics. In *Proceedings of the International Conference for High Performance Computing, Networking,* *Storage, and Analysis"*, SC '18, pages 51:1–51:12, Piscataway, NJ, USA, 2018, IEEE Press.

Kussul, N., Lavreniuk, M., Skakun, S. and Shelestov, A., "Deep Learning Classification of Land Cover and Crop Types Using Remote Sensing Data", *IEEE Geoscience and Remote Sensing Letters*, vol. 14, no. 5, pp. 778-782, 2017. doi: 10.1109/lgrs.2017.2681128

Lacis A. A., Schmidt, G. A., Rind D. and Ruedy R. A., "Atmospheric CO_2 : Principal Control Knob Governing Earth's Temperature", *Science*, vol. 330, no. 6002, pp. 356-359, 2010. doi: 10.1126/science.1190653

Lakshmanan, V. and Smith, T., "An Objective Method of Evaluating and Devising Storm-Tracking Algorithms", *Weather and Forecasting*, vol. 25, no. 2, pp. 701-709, 2010. doi: 10.1175/2009waf2222330.1

Lary, D. J., Alavi, A. H., Gandomi, A. H. and Walker A. L., "Machine learning in geosciences and remote sensing", *Geoscience Frontiers*, vol. 7, no. 1, pp. 3-10, 2016. doi: 10.1016/j.gsf.2015.07.003

Leahy, P., Kiely, G. and Corcoran, G., "Structural optimisation and input selection of an artificial neural network for river level prediction", *Journal of Hydrology*, vol. 355, no. 1-4, pp. 192-201, 2008. doi: 10.1016/j.jhydrol.2008.03.017

Ledig, C. et al., "Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network", *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. doi: 10.1109/cvpr.2017.19

Lehmann, J., Coumou, D. and Frieler, K., "Increased record-breaking precipitation events under global warming", *Climatic Change*, vol. 132, no. 4, pp. 501-515, 2015. doi: 10.1007/s10584-015-1434-y

Leinonen, J., Nerini, D. and Berne, A., "Stochastic Super-Resolution for Downscaling Time-Evolving Atmospheric Fields With a Generative Adversarial Network", *IEEE Transactions on Geoscience and Remote Sensing*, vol. 59, no. 9, pp. 7211-7223, 2021. doi: 10.1109/tgrs.2020.3032790

Levitus, S., Antonov, J., Boyer, T., Baranova, O., Garcia, H., Locarnini, R., Mishonov, A., Reagan, J., Seidov, D., Yarosh, E., Zweng, M., "NCEI ocean heat content, temperature anomalies, salinity anomalies, thermosteric sea level anomalies, halosteric sea level anomalies, and total steric sea level anomalies from 1955 to present calculated from in situ oceanographic subsurface profile data (NCEI Accession 0164586)". Version 4.4. *NOAA National Centers for Environmental Information*, 2017. Dataset. doi: 10.7289/V53F4MVP

Li, W., Ni, L., Li, Z., Duan, S. and Wu, H., "Evaluation of Machine Learning Algorithms in Spatial Downscaling of MODIS Land Surface Temperature", *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 12, no. 7, pp. 2299-2307, 2019. doi: 10.1109/jstars.2019.2896923

Li, X., Li, Z., Huang, W. and Zhou, P., "Performance of statistical and machine learning ensembles for daily temperature downscaling", *Theoretical and Applied Climatology*, vol. 140, no. 1-2, pp. 571-588, 2020. doi: 10.1007/s00704-020-03098-3

Liu, Y., et al., "Application of deep convolutional neural networks for detecting extreme weather in climate datasets", *International Conference on Advances in Big Data Analytics*, 2016.

Liu, Y., Yang, Y., Jing, W. and Yue, X., "Comparison of Different Machine Learning Approaches for Monthly Satellite-Based Soil Moisture Downscaling over Northeast China", *Remote Sensing*, vol. 10, no. 2, p. 31, 2017. doi: 10.3390/rs10010031

Lockwood, M., "Solar Influence on Global and Regional Climates", *Surveys in Geophysics*, vol. 33, no. 3-4, pp. 503-534, 2012. doi: 10.1007/s10712-012-9181-3

Loeb, N. G., Kato, S. and Wielicki, B. A., "Defining Top-of-the-Atmosphere Flux Reference Level for Earth Radiation Budget Studies", *Journal of Climate*, vol. 15, no. 22, pp. 3301-3309, 2002. doi: 10.1175/1520-0442(2002)015<3301:dtotaf>2.0.co;2

Loh, E., "Medicine and the rise of the robots: a qualitative review of recent advances of artificial intelligence in health", *BMJ Leader*, vol. 2, no. 2, pp. 59-63, 2018. doi: 10.1136/leader-2018-000071

Lu, Z., Hunt, B. R. and Ott, E., "Attractor reconstruction by machine learning", *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 28, no. 6, p. 061104, 2018. doi: 10.1063/1.5039508

Mahajan, S., Evans, K. J., Hack, J. J. and Truesdale, J. E., "Linearity of Climate Response to Increases in Black Carbon Aerosols", *Journal of Climate*, vol. 26, no. 20, pp. 8223-8237, 2013. doi: 10.1175/jcli-d-12-00715.1

Makarynska, D. and Makarynskyy, O., "Predicting sea-level variations at the Cocos (Keeling) Islands with artificial neural networks", *Computers & Geosciences*, vol. 34, no. 12, pp. 1910-1917, 2008. doi: 10.1016/j.cageo.2007.12.004

Makarynskyy, O., Makarynska, D., Kuhn, M. and Featherstone, W., "Predicting sea level variations with artificial neural networks at Hillarys Boat Harbour, Western Australia", *Estuarine, Coastal and Shelf Science*, vol. 61, no. 2, pp. 351-360, 2004. doi: 10.1016/j.ecss.2004.06.004

Maraun, D. and Widmann, M., "Statistical Downscaling and Bias Correction for Climate Research", *Cambridge University Press*, Cambridge, UK, 2018.

MARCONI100, 2021. "MARCONI100 | SCAI", *Hpc.cineca.it*, 2021. [Online]. Available: https://www.hpc.cineca.it/hardware/marconi100. [Accessed: 22- Dec-2021].

Marotzke, J. and Forster, P. M., “Forcing, feedback and internal variability in global temperature trends”, *Nature*, vol. 517, no. 7536, pp. 565-570, 2015. doi: 10.1038/nature14117

Marshall, S., “Glacier retreat crosses a line”, Science, vol. 345, no. 6199, pp. 872-872, 2014. doi: 10.1126/science.1258584

Masui, T. et al., “An emission pathway for stabilization at 6 Wm−2 radiative forcing”, *Climatic Change*, vol. 109, no. 1-2, pp. 59-76, 2011. doi: 10.1007/s10584-011-0150-5

Matthews, H. D., Weaver, A. J., Meissner, K. J., Gillett, N. P. and Eby, M., “Natural and anthropogenic climate change: incorporating historical land cover change, vegetation dynamics and the global carbon cycle”, *Climate Dynamics*, vol. 22, no. 5, pp. 461-479, 2004. doi: 10.1007/s00382-004-0392-2

McGovern, A. et al., “Using Artificial Intelligence to Improve Real-Time Decision-Making for High-Impact Weather”, *Bulletin of the American Meteorological Society*, vol. 98, no. 10, pp. 2073-2090, 2017. doi: 10.1175/bams-d-16-0123.1

McQuade, S., and Monteleoni, C., “Global Climate Model Tracking Using Geospatial Neighborhoods”, *Proceedings of the AAAI Conference on Artificial Intelligence*, *26*(1), 335-341, 2012. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/8178. [Accessed: 2- Dic- 2021]

Medium, 2016. “Understanding LSTM and its diagrams”, Medium, 2016. [Online]. Available: https://blog.mlreview.com/understanding-lstm-and-its-diagrams-37e2f46f1714. [Accessed: 20- Dec- 2021].

Meehl, G. A., Washington, W. M., Ammann, C. M., Arblaster, J. M., Wigley, T. M. L. and Tebaldi, C., “Combinations of Natural and Anthropogenic Forcings in Twentieth-Century Climate”, *Journal of Climate*, vol. 17, no. 19, pp. 3721-3727, 2004. doi: 10.1175/1520-0442(2004)017<3721:conaaf>2.0.co;2

Melillo, J. M., Richmond, T. C. and Yohe, G. W., “Climate Change Impacts in the United States: The Third National Climate Assessment”, U.S. Global Change Research Program, 841 pp, 2014. doi: 10.7930/j0z31wj2

Mendes, D. and Marengo, J. A., “Temporal downscaling: a comparison between artificial neural network and autocorrelation techniques over the Amazon Basin in present and future climate change scenarios”, *Theoretical and Applied Climatology*, vol. 100, no. 3-4, pp. 413-421, 2009. doi: 10.1007/s00704-009-0193-y

Miao, Q., Pan, B., Wang, H., Hsu, K. and Sorooshian, S., “Improving Monsoon Precipitation Prediction Using Combined Convolutional and Long Short Term Memory Neural Network”, *Water*, vol. 11, no. 5, p. 977, 2019. doi: 10.3390/w11050977

Miller, D. M., Kaminsky, E. J. and Rana, S., “Neural network classification of remote-sensing data”, *Computers & Geosciences*, vol. 21, no. 3, pp. 377-386, 1995. doi: 10.1016/0098-3004(94)00082-6

Min, X. et al., “Spatially Downscaling IMERG at Daily Scale Using Machine Learning Approaches Over Zhejiang, Southeastern China”, *Frontiers in Earth Science*, vol. 8, 2020. doi: 10.3389/feart.2020.00146

Misra, S., Sarkar, S. and Mitra, P., “Statistical downscaling of precipitation using long short-term memory recurrent neural networks”, *Theoretical and Applied Climatology*, vol. 134, no. 3-4, pp. 1179-1196, 2017. doi: 10.1007/s00704-017-2307-2

Monteleoni, C. et al., “Climate informatic”. In T. Yu, N. Chawla, and S. Simoff, editors, *Computational Intelligent Data Analysis for Sustainable Development; Data Mining and Knowledge Discovery Series*, chapter 4, pages 81–126. CRC Press, Taylor & Francis Group, 2013.

Monteleoni, C., Schmidt, G. A., Saroha, S. and Asplund, E., “Tracking climate models”, *Statistical Analysis and Data Mining: The ASA Data Science Journal*, vol. 4, no. 4, pp. 372-392, 2011. doi: 10.1002/sam.10126

Moss, R. H. et al., “The next generation of scenarios for climate change research and assessment”, *Nature*, vol. 463, no. 7282, pp. 747-756, 2010. doi: 10.1038/nature08823

Mouatadid, S., Easterbrook, S. and Erler, A. R., “A Machine Learning Approach to Non-uniform Spatial Downscaling of Climate Variables”, *2017 IEEE International Conference on Data Mining Workshops (ICDMW)*, 2017. doi: 10.1109/icdmw.2017.49

Muis, S., Verlaan, M., Winsemius, H. C., Aerts, J. C. and Ward, P. J., “A global reanalysis of storm surges and extreme sea levels”, *Nature Communications*, vol. 7, no. 1, 2016. doi: 10.1038/ncomms11969

Mukkavilli, S. K., “EnviroNet: ImageNet for environment”, In *18th Conference on Artificial and Com- putational Intelligence and its Applications to the Environmental Sciences*. American Meteorological Society 99th Annual Meeting, 2019.

Mullen, S. L., Schmitz, J. T. and Rennó, N. O., “Intraseasonal Variability of the Summer Monsoon over Southeast Arizona”, *Monthly Weather Review*, vol. 126, no. 11, pp. 3016-3035, 1998. doi: 10.1175/1520-0493(1998)126<3016:ivotsm>2.0.co;2

NASA, 2017, NASA, “NOAA Data Show 2016 Warmest Year on Record Globally”, 2017. [Online]. Available: https://www.giss.nasa.gov/research/news/20170118/. [Accessed: 9- Nov- 2021]

NASA, 2021. “Climate Change Evidence: How Do We Know?”, *Climate Change: Vital Signs of the Planet*, 2021. [Online]. Available: https://climate.nasa.gov/evidence/. [Accessed: 26- Nov- 2021]

Nerem, R. S., Beckley, B. D., Fasullo, J. T., Hamlington, B. D., Masters, D. and Mitchum, G. T., “Climate-change–driven accelerated sea-level rise detected in the altimeter era”, *Proceedings of the National Academy of Sciences*, vol. 115, no. 9, pp. 2022-2025, 2018. doi: 10.1073/pnas.1717312115

NetCDF, 2021. “Unidata | NetCDF”, *Unidata.ucar.edu*, 2021. [Online]. Available: https://www.unidata.ucar.edu/software/netcdf/. [Accessed: 21- Dec- 2021].

Neumann, P. et al., “Assessing the scales in numerical weather and climate predictions: will exascale be the rescue?”, *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 377, no. 2142, p. 20180148, 2019. doi: 10.1098/rsta.2018.0148

Nicholls, R. J. and Cazenave, A., “Sea-Level Rise and Its Impact on Coastal Zones”, *Science*, vol. 328, no. 5985, pp. 1517-1520, 2010. doi: 10.1126/science.1185782

NOAA, 2021, “How do we know the Earth's climate is warming?”, Global Climate Change Indicators, National Oceanic and Atmospheric Administration, 2021. [Online]. Available: https://www.ncdc.noaa.gov/monitoring-references/faq/indicators.php#warming-climate. [Accessed: 9- Nov- 2021]

Nosratabadi, S. et al., “Data Science in Economics: Comprehensive Review of Advanced Machine Learning and Deep Learning Methods”, *Mathematics*, vol. 8, no. 10, p. 1799, 2020. doi: 10.3390/math8101799

Nowack, P., Braesicke, P., Haigh, J., Abraham, N., L., Pyle, J. and Voulgarakis, A., “Using machine learning to build temperature-based ozone parameterizations for climate sensitivity simulations”, *Environmental Research Letters*, vol. 13, no. 10, p. 104016, 2018. doi: 10.1088/1748-9326/aae2be

OECD, 2021. Organization for Economic Co-operation and Development (OECD), "OECD Statistics". [Online]. Available: https://stats.oecd.org [Accessed on 19*th* July 2021].

Oerlemans, J., *Glaciers and climate change*. Lisse: A.A. Balkema, 2001, ISBN 9789026518133

Orru, H., Ebi, K. L. and Forsberg, B., “The Interplay of Climate Change and Air Pollution on Health”, *Current Environmental Health Reports*, vol. 4, no. 4, pp. 504-513, 2017. doi: 10.1007/s40572-017-0168-6

Pan, B., Hsu, K., AghaKouchak, A. and Sorooshian, S., “Improving Precipitation Estimation Using Convolutional Neural Network”, *Water Resources Research*, vol. 55, no. 3, pp. 2301-2321, 2019. doi: 10.1029/2018wr024090

Park, D., Kim, J. and Chun, S. Y., “Down-Scaling with Learned Kernels in Multi-Scale Deep Neural Networks for Non-Uniform Single Image Deblurring”, *arXiv.org*, 2019. [Online]. Available: https://arxiv.org/abs/1903.10157. [Accessed: 21-Dec- 2021].

Pascanu, R., Mikolov, T., and Bengio, Y., “On the difficulty of training recurrent neural networks”, In *Proceedings of the 30th International Conference on International Conference on Machine Learning*, vol. 28 (ICML'13), JMLR.org, III–1310–III–1318, 2013.

Pashova, L. and Popova, S., “Daily sea level forecast at tide gauge Burgas, Bulgaria using artificial neural networks”, *Journal of Sea Research*, vol. 66, no. 2, pp. 154-161, 2011. doi: 10.1016/j.seares.2011.05.012

Peng, S., Piao, S., Ciais, P., Friedlingstein, P., Zhou, L. and Wang, T., “Change in snow phenology and its potential feedback to temperature in the Northern Hemisphere over the last three decades”, *Environmental Research Letters*, vol. 8, no. 1, p. 014008, 2013. doi: 10.1088/1748-9326/8/1/014008

Perc, M., Ozer, M. and Hojnik, J., “Social and juristic challenges of artificial intelligence”, *Palgrave Communications*, vol. 5, no. 1, 2019. doi: 10.1057/s41599-019-0278-x

Priestley, R. K., Heine, Z. and Milfont, T. L., “Public understanding of climate change-related sea-level rise”, *PLOS ONE*, vol. 16, no. 7, p. e0254348, 2021. doi: 10.1371/journal.pone.0254348

Racah, E. et al., “ExtremeWeather: A large-scale climate dataset for semi-supervised detection, localization, and understanding of extreme weather events”, in *Advances in Neural Information Processing Systems 30*, pages 3402–3413, 2017.

Ramachandran, P., Zoph, B. and Le, Q. V., “Searching for Activation Functions”, *arXiv.org*, 2017. [Online]. Available: https://arxiv.org/abs/1710.05941. [Accessed: 02- Dec- 2021].

Rasp, S., Pritchard, M. and Gentine, P., “Deep learning to represent subgrid processes in climate models”, *Proceedings of the National Academy of Sciences*, vol. 115, no. 39, pp. 9684-9689, 2018. doi: 10.1073/pnas.1810286115

Ravindra, K., Rattan, P., Mor, S. and Aggarwal, A. N., “Generalized additive models: Building evidence of air pollution, climate change and human health”, *Environment International*, vol. 132, p. 104987, 2019. doi: 10.1016/j.envint.2019.104987

Read, S., Jia, X., Willard, J., Appling, A. P., Zwart, J. A., Oliver, S. K., Karpatne, A., Hansen, G. J. A., Hanson, P. C., Watkins, W., et al. “Process-guided deep learning predictions of lake water temperature”. WRR, 2019.

Reichstein, M. et al., “Deep learning and process understanding for data-driven Earth system science”, *Nature*, vol. 566, no. 7743, pp. 195-204, 2019. doi: 10.1038/s41586-019-0912-1

Riahi, K. et al., “RCP 8.5—A scenario of comparatively high greenhouse gas emissions”, *Climatic Change*, vol. 109, no. 1-2, pp. 33-57, 2011. doi: 10.1007/s10584-011-0149-y

Riccò, M., Ranzieri, S., Balzarini, F., Bragazzi, N. and Corradi, M., “SARS-CoV-2 infection and air pollutants: Correlation or causation?”, *Science of The Total Environment*, vol. 734, p. 139489, 2020. doi: 10.1016/j.scitotenv.2020.139489

Ritto, T. G. and Rochinha, F. A., “Digital twin, physics-based model, and machine learning applied to damage detection in structures”. arXiv 2020, arXiv:2005.14360.

RMN, 2021. “Rete Mareografica Nazionale”, Mareografico.it, 2021. [Online]. Available: https://mareografico.it/?session=0S186518459284B83C90899065&syslng=ita&sysmen=-1&sysind=-1&syssub=-1&sysfnt=0. [Accessed: 17- Dec- 2021].

Robinson, D. A., Hall, D. K. and Mote, T. L., “MEaSUREs Northern Hemisphere Terrestrial Snow Cover Extent Daily 25km EASE-Grid 2.0, Version 1”, NASA National Snow and Ice Data Center Distributed Active Archive Center, Boulder, Colorado USA, 2014. doi: https://doi.org/10.5067/MEASURES/CRYOSPHERE/nsidc-0530.001. [Accessed: 11- Nov- 2021].

Robock, A., “Volcanic eruptions and climate”, *Reviews of Geophysics*, vol. 38, no. 2, pp. 191-219, 2000. doi: 10.1029/1998rg000054

Rocha Rodrigues, E., Oliveira, I., Cunha, R. and Netto, M., “DeepDownscale: A Deep Learning Strategy for High-Resolution Weather Forecast”, *2018 IEEE 14th International Conference on e-Science (e-Science)*, 2018. doi: 10.1109/escience.2018.00130

Roe, G., Baker, M., and Herla, F., “Centennial glacier retreat as categorical evidence of regional climate change”, *Nature Geoscience*, vol. 10, no. 2, pp. 95-99, 2016. doi: 10.1038/ngeo2863

Rolinski, S. and Umgiesser, G., “Modelling short-term dynamics of suspended particulate matter in Venice Lagoon, Italy”, *Estuarine, Coastal and Shelf Science*, vol. 63, no. 4, pp. 561-576, 2005. doi: 10.1016/j.ecss.2005.01.002

Rolnick, D. et al., “Tackling Climate Change with Machine Learning”, *arXiv.org*, 2019. [Online]. Available: https://arxiv.org/abs/1906.05433. [Accessed: 01- Dec-2021].

Romm, J., “CLIMATE CHANGE, What everyone needs to know”, 2nd ed. *OXFORD University Press*, 2018

Rumelhart, D. E., Hinton, G. E. and Williams, R. J., "Learning representations by back-propagating errors", *Nature*, vol. 323, no. 6088, pp. 533-536, 1986. doi: 10.1038/323533a0

Sabine, C. et al., "The Oceanic Sink for Anthropogenic CO_2", *Science*, vol. 305, no. 5682, pp. 367-371, 2004. doi: 10.1126/science.1097403

Sachindra, D. A. and Kanae, S., "Machine learning for downscaling: the use of parallel multiple populations in genetic programming", *Stochastic Environmental Research and Risk Assessment*, vol. 33, no. 8-9, pp. 1497-1533, 2019. doi: 10.1007/s00477-019-01721-y

Sachindra, D. A., Huang, F., Barton, A. and Perera, B., "Least square support vector and multi-linear regression for statistically downscaling general circulation model outputs to catchment streamflows", *International Journal of Climatology*, vol. 33, no. 5, pp. 1087-1106, 2012. doi: 10.1002/joc.3493

Sachindra, D., Ahmed, K., Rashid, M. M., Shahid, S. and Perera, B., "Statistical downscaling of precipitation using machine learning techniques", *Atmospheric Research*, vol. 212, pp. 240-258, 2018. doi: 10.1016/j.atmosres.2018.05.022

Salcedo-Sanz, S. et al., "Machine learning information fusion in Earth observation: A comprehensive review of methods, applications and data sources", *Information Fusion*, vol. 63, pp. 256-272, 2020. doi: 10.1016/j.inffus.2020.07.004

Salimi, A. H., Masoompour Samakosh, J., Sharifi, E., Hassanvand, M. R., Noori, A. and von Rautenkranz, H., "Optimized Artificial Neural Networks-Based Methods for Statistical Downscaling of Gridded Precipitation Data", *Water*, vol. 11, no. 8, p. 1653, 2019. doi: 10.3390/w11081653

Scafetta, N., "Distribution of the SARS-CoV-2 Pandemic and Its Monthly Forecast Based on Seasonal Climate Patterns", *International Journal of Environmental Research and Public Health*, vol. 17, no. 10, p. 3493, 2020. doi: 10.3390/ijerph17103493

Sekiyama, T. T., "Statistical Downscaling of Temperature Distributions from the Synoptic Scale to the Mesoscale Using Deep Convolutional Neural Networks", *arXiv.org*, 2020. [Online]. Available: https://arxiv.org/abs/2007.10839. [Accessed: 21- Dec- 2021].

Serpico, S., Bruzzone, L. and Roli, F., "An experimental comparison of neural and statistical non-parametric algorithms for supervised classification of remote-sensing images", *Pattern Recognition Letters*, vol. 17, no. 13, pp. 1331-1341, 1996. doi: 10.1016/s0167-8655(96)00090-6

Sertel, E., Cigizoglu, H. K. and Sanli, D. U., "Estimating Daily Mean Sea Level Heights Using Artificial Neural Networks", *Journal of Coastal Research*, vol. 243, pp. 727-734, 2008. doi: 10.2112/06-742.1

Sharifi, E., Saghafian, B. and Steinacker, R., “Downscaling Satellite Precipitation Estimates With Multiple Linear Regression, Artificial Neural Networks, and Spline Interpolation Techniques”, *Journal of Geophysical Research: Atmospheres*, vol. 124, no. 2, pp. 789-805, 2019. doi: 10.1029/2018jd028795

Shi, X., “Enabling Smart Dynamical Downscaling of Extreme Precipitation Events With Machine Learning”, *Geophysical Research Letters*, vol. 47, no. 19, 2020. doi: 10.1029/2020gl090309

Shiogama, H., Stone, D. A., Nagashima, T., Nozawa, T. and Emori, S., “On the linear additivity of climate forcing-response relationships at global and continental scales”, *International Journal of Climatology*, vol. 33, no. 11, pp. 2542-2550, 2012. doi: 10.1002/joc.3607

Shukla, J., “Predictability in the Midst of Chaos: A Scientific Basis for Climate Forecasting”, *Science*, vol. 282, no. 5389, pp. 728-731, 1998. doi: 10.1126/science.282.5389.728

Singh, A., Albert, A. and White, B., “Downscaling numerical weather models with gans”, In *Proceedings of the 9th International Conference on Climate Informatics 2019*, Paris, France, 2-4 October, 2019; pp. 275-278.

Smith, P. et al., “Biophysical and economic limits to negative CO2 emissions”, *Nature Climate Change*, vol. 6, no. 1, pp. 42-50, 2015. doi: 10.1038/nclimate2870

Spickett, J., Brown, H. and Rumchev, K., “Climate Change and Air Quality: The Potential Impact on Health”, *Asia Pacific Journal of Public Health*, vol. 23, no. 2, pp. 37S-45S, 2011. doi: 10.1177/1010539511398114

Stott, P., “How climate change affects extreme weather events”, *Science*, vol. 352, no. 6293, pp. 1517-1518, 2016. doi: 10.1126/science.aaf7271

Strobach, E. and Bel, G., “Improvement of climate predictions and reduction of their uncertainties using learning algorithms”, *Atmospheric Chemistry and Physics*, vol. 15, no. 15, pp. 8631-8641, 2015. doi: 10.5194/acp-15-8631-2015

Sun, L. and Lan, Y., “Statistical downscaling of daily temperature and precipitation over China using deep learning neural models: Localization and comparison with other methods”, *International Journal of Climatology*, vol. 41, no. 2, pp. 1128-1147, 2020. doi: 10.1002/joc.6769

Swain, D. L., Singh, D., Touma, D. and Diffenbaugh, N. S., “Attributing Extreme Events to Climate Change: A New Frontier in a Warming World”, *One Earth*, vol. 2, no. 6, pp. 522-527, 2020. doi: 10.1016/j.oneear.2020.05.011

Swanson, N. R. and White, H., “Forecasting economic time series using flexible versus fixed specification and linear versus nonlinear econometric models”, *International Journal of Forecasting*, vol. 13, no. 4, pp. 439-461, 1997. doi: 10.1016/s0169-2070(97)00030-7

TDT, 2021. “Distributed training with TensorFlow | TensorFlow Core”, *TensorFlow*, 2021. [Online]. Available: https://www.tensorflow.org/guide/distributed_training. [Accessed: 22- Dec-2021].

Tebaldi, C. and Knutti, R., “The use of the multi-model ensemble in probabilistic climate projections”, *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 365, no. 1857, pp. 2053-2075, 2007. doi: 10.1098/rsta.2007.2076

Tebaldi, C. et al., “Extreme sea levels at different global warming levels”, *Nature Climate Change*, vol. 11, no. 9, pp. 746-751, 2021. doi: 10.1038/s41558-021-01127-1

Thessen, A., “Adoption of Machine Learning Techniques in Ecology and Earth Science”, *One Ecosystem*, vol. 1, p. e8621, 2016. doi: 10.3897/oneeco.1.e8621

Thomson, A. M. et al., “RCP4.5: a pathway for stabilization of radiative forcing by 2100”, *Climatic Change*, vol. 109, no. 1-2, pp. 77-94, 2011. doi: 10.1007/s10584-011-0151-4

Tol, R., “The Economic Impacts of Climate Change”, *Review of Environmental Economics and Policy*, vol. 12, no. 1, pp. 4-25, 2018. doi: 10.1093/reep/rex027

Topol, E., “High-performance medicine: the convergence of human and artificial intelligence”, *Nature Medicine*, vol. 25, no. 1, pp. 44-56, 2019. doi: 10.1038/s41591-018-0300-7

Tran Anh, D., Van, S. P., Dang, T. D. and Hoang, L. P., “Downscaling rainfall using deep learning long short-term memory and feedforward neural network”, *International Journal of Climatology*, vol. 39, no. 10, pp. 4170-4188, 2019. doi: 10.1002/joc.6066

Tran Anh, Q. and Taniguchi, K., “Coupling dynamical and statistical downscaling for high-resolution rainfall forecasting: case study of the Red River Delta, Vietnam”, *Progress in Earth and Planetary Science*, vol. 5, no. 1, 2018. doi: 10.1186/s40645-018-0185-6

Trenberth, K. E. and Fasullo, J. T., “An apparent hiatus in global warming?”, *Earth's Future*, vol. 1, no. 1, pp. 19-32, 2013. doi: 10.1002/2013ef000165

Trenberth, K. E., “Has there been a hiatus?”, *Science*, vol. 349, no. 6249, pp. 691-692, 2015. doi: 10.1126/science.aac9225

Trotta, F. et al., “A Structured and Unstructured grid Relocatable ocean platform for Forecasting (SURF)”, *Deep Sea Research Part II: Topical Studies in Oceanography*, vol. 133, pp. 54-75, 2016. doi: 10.1016/j.dsr2.2016.05.004

Ultsch, A., “Self-organizing feature maps predicting sea levels”, *Information Sciences*, vol. 144, no. 1-4, pp. 91-125, 2002. doi: 10.1016/s0020-0255(02)00203-7

Umgiesser, G., Canu, D. M., Cucco, A. and Solidoro, C., “A finite element model for the Venice Lagoon. Development, set up, calibration and validation”, *Journal of Marine Systems*, vol. 51, no. 1-4, pp. 123-145, 2004. doi: 10.1016/j.jmarsys.2004.05.009

UNFCCC, 2015. Paris Agreement. 25 pp., United Nations Framework Convention on Climate Change. [Online]. Available: http://unfccc.int/files/essential_background/convention/application/pdf/english_paris_agreement.pdf. [Accessed: 12- Nov- 2021]

van der Geest, K. et al., “The Impacts of Climate Change on Ecosystem Services and Resulting Losses and Damages to People and Society”, *Loss and Damage from Climate Change*, pp. 221-236, 2018. doi: 10.1007/978-3-319-72026-5_9

van Vuuren, D. P. et al., “RCP2.6: exploring the possibility to keep global mean temperature increase below 2°C”, *Climatic Change*, vol. 109, no. 1-2, pp. 95-116, 2011. doi: 10.1007/s10584-011-0152-3

Vandal, T., Kodra, E. and Ganguly, A. R., “Intercomparison of machine learning methods for statistical downscaling: the case of daily and extreme precipitation”, *Theoretical and Applied Climatology*, vol. 137, no. 1-2, pp. 557-570, 2018b. doi: 10.1007/s00704-018-2613-3

Vandal, T., Kodra, E., Ganguly, S., Michaelis, A., Nemani, R. and Ganguly, A., “DeepSD”, Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2017. doi: 10.1145/3097983.3098004

Vandal, T., Kodra, E., Ganguly, S., Michaelis, A., Nemani, R. and Ganguly, A., “Generating High Resolution Climate Change Projections through Single Image Super-Resolution: An Abridged Version”, *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*, 2018a. doi: 10.24963/ijcai.2018/759

VanderWeele, T. and Ding, P., “Sensitivity analysis in observational research: introducing the E-Value”, *Annals of Internal Medicine*, vol. 167, no. 4, p. 268, 2017. doi: 10.7326/m16-2607

Velicogna, I. et al., “Continuity of Ice Sheet Mass Loss in Greenland and Antarctica From the GRACE and GRACE Follow-On Missions”, *Geophysical Research Letters*, vol. 47, no. 8, 2020. doi: 10.1029/2020gl08729

von Schuckmann, K. et al., “Heat stored in the Earth system: where does the energy go?”, *Earth System Science Data*, vol. 12, no. 3, pp. 2013-2041, 2020. doi: 10.5194/essd-12-2013-2020

Voulodimos, A., Doulamis, N., Doulamis, A. and Protopapadakis, E., “Deep Learning for Computer Vision: A Brief Review”, *Computational Intelligence and Neuroscience*, vol. 2018, pp. 1-13, 2018. doi: 10.1155/2018/7068349

Wang, W. and Yuan, H., “A Tidal Level Prediction Approach Based on BP Neural Network and Cubic B-Spline Curve with Knot Insertion Algorithm”, *Mathematical Problems in Engineering*, vol. 2018, pp. 1-9, 2018. doi: 10.1155/2018/9835079

Wang, X. et al., “ESRGAN: Enhanced Super-Resolution Generative Adversarial Networks”, *Lecture Notes in Computer Science*, pp. 63-79, 2019. doi: 10.1007/978-3-030-11021-5_5

Wang, Z., Bovik, A., Sheikh, H. and Simoncelli, E., “Image Quality Assessment: From Error Visibility to Structural Similarity”, *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600-612, 2004. doi: 10.1109/tip.2003.819861

Wang, Z., Chen, J. and Hoi, S. C. H., “Deep Learning for Image Super-Resolution: A Survey”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 10, pp. 3365-3387, 2021. doi: 10.1109/tpami.2020.2982166

Wenzel, M. and Schröter, J., “Reconstruction of regional mean sea level anomalies from tide gauges using neural networks”, *Journal of Geophysical Research*, vol. 115, no. 8, 2010. doi: 10.1029/2009jc005630

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*, vol. 11, no. 8, pp. 2680-2693, 2019. doi: 10.1029/2019ms001705

Weyn, J. A., Durran, D. R. and Caruana, R., “Improving Data-Driven Global Weather Prediction Using Deep Convolutional Neural Networks on a Cubed Sphere”, *Journal of Advances in Modeling Earth Systems*, vol. 12, no. 9, 2020. doi: 10.1029/2020ms002109

Wheeler, T. and von Braun, J., “Climate Change Impacts on Global Food Security”, *Science*, vol. 341, no. 6145, pp. 508-513, 2013. doi: 10.1126/science.1239402

WHO, 2005. “Air Quality Guidelines for Particulate Matter, Ozone, Nitrogen Dioxide and Sulfur Dioxide. Summary of Risk Assessment”. [Online]. Available: https://apps.who.int/iris/bitstream/handle/10665/69477/WHO_SDE_PHE_OEH_06.02_eng.pdf?sequence1⁄41. [Accessed: 03- Jan- 2022].

WHO, 2020a. “WHO Director-General's remarks at the media briefing on 2019-nCoV on 11 February 2020”. [Online]. Available: https://www.who.int/dg/speeches/detail/who-director-general-s-remarks-at-the-media-briefing-on-2019-ncov-on-11-february-2020 [Accessed: 03- Jan- 2022].

Wilby, R., Charles, S., Zorita, E., Timbal, B., Whetton, P., Mearns, L, “Guidelines for Use of Climate Scenarios Developed from Statistical Downscaling Methods”, Supporting Material to the IPCC, pp. 3–21, 2004 [Online]. Available: https://www.narccap.ucar.edu/ [Accessed: 21- Dec- 2021].

Willard, J. D. , Jia, X., Xu, S., Steinbach, M., and Kumar, V.. “Integrating Physics-Based Modeling with Machine Learning: A Survey”. arXiv 2022, arXiv:2003:04919v6.

WMO, 2021, “Climate”, World Meteorological Organization, 2021. [Online]. Available: https://public.wmo.int/en/our-mandate/climate. [Accessed: 30- Sep-2021]

Wood, A. W., Leung, L. R., Sridhar, V. and Lettenmaier, D. P., “Hydrologic Implications of Dynamical and Statistical Approaches to Downscaling Climate Model Outputs”, *Climatic Change*, vol. 62, no. 1-3, pp. 189-216, 2004. doi: 10.1023/b:clim.0000013685.99609.9e

Wu, A., Hsieh, W. W. and Tang, B., “Neural network forecasts of the tropical Pacific sea surface temperatures”, *Neural Networks*, vol. 19, no. 2, pp. 145-154, 2006. doi: 10.1016/j.neunet.2006.01.004

Wuebbles, D. et al., “Ch. 1: Our Globally Changing Climate. Climate Science Special Report: Fourth National Climate Assessment, Volume I”, 2017. doi: 10.7930/j08s4n35

Xu, R., Chen, N., Chen, Y. and Chen, Z., “Downscaling and Projection of Multi-CMIP5 Precipitation Using Machine Learning Methods in the Upper Han River Basin”, *Advances in Meteorology*, vol. 2020, pp. 1-17, 2020. doi: 10.1155/2020/8680436

Yan, J., Mu, L., Wang, L., Ranjan, R. and Zomaya, A. Y., “Temporal Convolutional Networks for the Advance Prediction of ENSO”, *Scientific Reports*, vol. 10, no. 1, 2020. doi: 10.1038/s41598-020-65070-5

Yang, W., Zhang, X., Tian, Y., Wang, W., Xue, J. and Liao, Q., “Deep Learning for Single Image Super-Resolution: A Brief Review”, *IEEE Transactions on Multimedia*, vol. 21, no. 12, pp. 3106-3121, 2019. doi: 10.1109/tmm.2019.2919431

Yu, R. et al., “LSTM-EFG for wind power forecasting based on sequential correlation features”, *Future Generation Computer Systems*, vol. 93, pp. 33-42, 2019. doi: 10.1016/j.future.2018.09.054

Zhao, Z. et al., “A lighten CNN-LSTM model for speaker verification on embedded devices”, *Future Generation Computer Systems*, vol. 100, pp. 751-758, 2019. doi: 10.1016/j.future.2019.05.057

Hirabayashi, Yukiko; Mahendran, Roobavannan; Koirala, Sujan; Konoshima, Lisako; Yamazaki, Dai; Watanabe, Satoshi; Kim, Hyungjun; Kanae, Shinjiro (2013). "Global flood risk under climate change". Nature Climate Change. 3 (9): 816–821. Bibcode:2013NatCC...3..816H. doi:10.1038/nclimate1911. ISSN 1758-6798.

Dunne, Daisy (14 July 2020). "Explainer: How climate change is affecting wildfires around the world". Carbon Brief. Retrieved 17 February 2022.

Wunderling, N., Willeit, M., Donges, J.F. et al. “Global warming due to loss of large ice masses and Arctic summer sea ice”. Nat Commun 11, 5177 (2020). https://doi.org/10.1038/s41467-020-18934-3

Marx, Werner & Haunschild, Robin & Bornmann, Lutz. (2021). “Heat waves: a hot topic in climate change research”. Theoretical and Applied Climatology. 146. 10.1007/s00704-021-03758-y.

Fowler Hayley J., Wasko Conrad and Prein Andreas F. 2021 “Intensification of short-duration rainfall extremes and implications for flood risk: current state of the art and future directions” Phil. Trans. R. Soc. A.3792019054120190541 http://doi.org/10.1098/rsta.2019.0541

Trenberth, Kevin E. (2022). “The Changing Flow of Energy Through the Climate System” (1 ed.). Cambridge University Press. doi:10.1017/9781108979030. ISBN 978-1-108-97903-0. S2CID 247134757.

IPCC, 2019: “Summary for Policymakers. In: IPCC Special Report on the Ocean and Cryosphere in a Changing Climate” [H.-O. Pörtner, D.C. Roberts, V. Masson-Delmotte, P. Zhai, M. Tignor, E. Poloczanska, K. Mintenbeck, A. Alegría, M. Nicolai, A. Okem, J. Petzold, B. Rama, N.M. Weyer (eds.)]. Cambridge University Press, Cambridge, UK and New York, NY, USA. https://doi.org/10.1017/9781009157964.001.

Hoegh-Guldberg, O.; Jacob, D.; Taylor, M.; Bindi, M.; et al. (2018). "Chapter 3: Impacts of 1.5ºC Global Warming on Natural and Human Systems" (PDF). IPCC SR15 2018. p. 179. Archived (PDF) from the original on 15 November 2019. Retrieved 15 December 2019.

Doney, Scott C.; Busch, D. Shallin; Cooley, Sarah R.; Kroeker, Kristy J. (17 October 2020). "The Impacts of Ocean Acidification on Marine Ecosystems and Reliant Human Communities". Annual Review of Environment and Resources. 45 (1): 83–112. doi:10.1146/annurev-environ-012320-083019. ISSN 1543-5938

Douville, H., K. Raghavan, J. Renwick, R.P. Allan, P.A. Arias, M. Barlow, R. Cerezo-Mota, A. Cherchi, T.Y. Gan, J. Gergis, D. Jiang, A. Khan, W. Pokam Mba, D. Rosenfeld, J. Tierney, and O. Zolina, 2021: “Chapter 8: Water Cycle Changes. In Climate Change 2021: The Physical Science Basis. Contribution of Working Group I to the Sixth Assessment Report of the Intergovernmental Panel on Climate Change”. [Masson-Delmotte, V., P. Zhai, A. Pirani, S.L. Connors, C. Péan, S. Berger, N. Caud, Y. Chen, L. Goldfarb, M.I. Gomis, M. Huang, K. Leitzell, E. Lonnoy, J.B.R. Matthews, T.K. Maycock, T. Waterfield, O. Yelekçi, R. Yu, and B. Zhou (eds.)]. Cambridge University Press, Cambridge, United Kingdom and New York, NY, USA, pp. 1055–1210, doi:10.1017/9781009157896.010

IPCC (2021). Masson-Delmotte, V.; Zhai, P.; Pirani, A.; Connors, S. L.; et al. (eds.). “Climate Change 2021: The Physical Science Basis (PDF). Contribution of Working Group I to the Sixth Assessment Report of the Intergovernmental Panel on Climate Change”. Cambridge University Press (In Press).

Diffenbaugh, Noah S.; Burke, Marshall (2019). "Global warming has increased global economic inequality". Proceedings of the National Academy of Sciences. 116 (20): 9808–9813. Bibcode:2019PNAS..116.9808D. doi:10.1073/pnas.1816020116. ISSN 0027-8424. PMC 6525504. PMID 31010922.

Rocque RJ, Beaudoin C, Ndjaboue R, et al “Health effects of climate change: an overview of systematic reviews”, BMJ Open 2021;11:e046333. doi: 10.1136/bmjopen-2020-046333

Seneviratne, S., Nicholls, N., Easterling, D., Goodess, C., Kanae, S., Kossin, J., Luo, Y., Marengo, J., McInnes, K., Rahimi, M., Reichstein, M., 2012. “Changes in Climate Extremes and Their Impacts on the Natural Physical Environment”. IPCC.

KanimozhiSelvi, C.S.; Sowmiya, G. “Prediction of Extreme Weather Events using Machine Learning Technique”. International Journal of Applied Engineering Research ISSN 0973-4562 Volume 14, Number 4. 2019, pp. 925–929

Norouzi, A., Habibi, H., Nazari, B., Noh, S.J., Seo, D.J., Zhang, Y., 2019. “Toward parsimonious modeling of frequency of areal runoff from heavy-to-extreme precipitation in large urban areas under changing conditions: a derived moment approach”. Stoch. Env. Res. Risk A. 33 (7), 1263–1281.

Stott, P., 2016. “How climate change affects extreme weather events”. Science 352 (6293), 1517–1518.

Boo, K.O., Kwon, W.T., Baek, H.J., 2006. “Change of extreme events of temperature and precipitation over Korea using regional projection of future climate change”. Geophys. Res. Lett. 33 (1)

"Pubs.GISS: Sun and Hansen 2003: Climate simulations for 1951-2050 with a coupled atmosphere-ocean model". pubs.giss.nasa.gov. 2003. Retrieved 25 August 2015.

Alemohammad, Hamed & Kolassa, Jana & Prigent, Catherine & Aires, Filipe & Gentine, Pierre. (2018). “Global Downscaling of Remotely-Sensed Soil Moisture using Neural Networks”. Hydrology and Earth System Sciences Discussions. 1-19. 10.5194/hess-2017-680.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. “PyTorch: an imperative style, high-performance deep learning library”. Proceedings of the 33rd International Conference on Neural

Information Processing Systems. Curran Associates Inc., Red Hook, NY, USA, Article 721, 8026–8037.

Norman P. Jouppi, Cliff Young, Nishant Patil, David Patterson, Gaurav Agrawal, Raminder Bajwa, Sarah Bates, Suresh Bhatia, Nan Boden, Al Borchers, Rick Boyle, Pierre-luc Cantin, Clifford Chao, Chris Clark, Jeremy Coriell, Mike Daley, Matt Dau, Jeffrey Dean, Ben Gelb, Tara Vazir Ghaemmaghami, Rajendra Gottipati, William Gulland, Robert Hagmann, C. Richard Ho, Doug Hogberg, John Hu, Robert Hundt, Dan Hurt, Julian Ibarz, Aaron Jaffey, Alek Jaworski, Alexander Kaplan, Harshit Khaitan, Daniel Killebrew, Andy Koch, Naveen Kumar, Steve Lacy, James Laudon, James Law, Diemthu Le, Chris Leary, Zhuyuan Liu, Kyle Lucke, Alan Lundin, Gordon MacKean, Adriana Maggiore, Maire Mahony, Kieran Miller, Rahul Nagarajan, Ravi Narayanaswami, Ray Ni, Kathy Nix, Thomas Norrie, Mark Omernick, Narayana Penukonda, Andy Phelps, Jonathan Ross, Matt Ross, Amir Salek, Emad Samadiani, Chris Severn, Gregory Sizikov, Matthew Snelham, Jed Souter, Dan Steinberg, Andy Swing, Mercedes Tan, Gregory Thorson, Bo Tian, Horia Toma, Erick Tuttle, Vijay Vasudevan, Richard Walter, Walter Wang, Eric Wilcox, and Doe Hyun Yoon. 2017. "In-Datacenter Performance Analysis of a Tensor Processing Unit". In Proceedings of the 44th Annual International Symposium on Computer Architecture (ISCA '17). Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3079856.3080246

Norman P. Jouppi, Cliff Young, Nishant Patil, David Patterson, Gaurav Agrawal, Raminder Bajwa, Sarah Bates, Suresh Bhatia, Nan Boden, Al Borchers, Rick Boyle, Pierre-luc Cantin, Clifford Chao, Chris Clark, Jeremy Coriell, Mike Daley, Matt Dau, Jeffrey Dean, Ben Gelb, Tara Vazir Ghaemmaghami, Rajendra Gottipati, William Gulland, Robert Hagmann, C. Richard Ho, Doug Hogberg, John Hu, Robert Hundt, Dan Hurt, Julian Ibarz, Aaron Jaffey, Alek Jaworski, Alexander Kaplan, Harshit Khaitan, Daniel Killebrew, Andy Koch, Naveen Kumar, Steve Lacy, James Laudon, James Law, Diemthu Le, Chris Leary, Zhuyuan Liu, Kyle Lucke, Alan Lundin, Gordon MacKean, Adriana Maggiore, Maire Mahony, Kieran Miller, Rahul Nagarajan, Ravi Narayanaswami, Ray Ni, Kathy Nix, Thomas Norrie, Mark Omernick, Narayana Penukonda, Andy Phelps, Jonathan Ross, Matt Ross, Amir Salek, Emad Samadiani, Chris Severn, Gregory Sizikov, Matthew Snelham, Jed Souter, Dan Steinberg, Andy Swing, Mercedes Tan, Gregory Thorson, Bo Tian, Horia Toma, Erick Tuttle, Vijay Vasudevan, Richard Walter, Walter Wang, Eric Wilcox, and Doe Hyun Yoon. 2017. "In-Datacenter Performance Analysis of a Tensor Processing Unit". SIGARCH Comput. Archit. News 45, 2 (May 2017), 1–12. https://doi.org/10.1145/3140659.3080246

Rudin, C. "Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead". Nat Mach Intell 1, 206–215 (2019). https://doi.org/10.1038/s42256-019-0048-x

M. Raissi, P. Perdikaris, G.E. Karniadakis, "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations", Journal of Computational Physics, Volume 378, 2019, Pages 686-707, ISSN 0021-9991, https://doi.org/10.1016/j.jcp.2018.10.045.

Gohel, Prashant et al. "Explainable AI: current status and future directions." ArXiv abs/2107.07045 (2021)

Ari Yair Barrera-Animas, Lukumon O. Oyedele, Muhammad Bilal, Taofeek Dolapo Akinosho, Juan Manuel Davila Delgado, Lukman Adewale Akanbi, "Rainfall prediction: A comparative analysis of modern machine learning algorithms for time-series forecasting", Machine Learning with Applications, Volume 7, 2022, 100204, ISSN 2666-8270, https://doi.org/10.1016/j.mlwa.2021.100204.

Papacharalampous, G., Tyralis, H. & Koutsoyiannis, D. "Univariate Time Series Forecasting of Temperature and Precipitation with a Focus on Machine Learning Algorithms: a Multiple-Case Study from Greece". Water Resour Manage 32, 5207–5239 (2018). https://doi.org/10.1007/s11269-018-2155-6

Cifuentes, J.; Marulanda, G.; Bello, A.; Reneses, J. "Air Temperature Forecasting Using Machine Learning Techniques: A Review". Energies 2020, 13, 4215. https://doi.org/10.3390/en13164215

Stefan Wolff, Fearghal O'Donncha, Bei Chen, "Statistical and machine learning ensemble modelling to forecast sea surface temperature", Journal of Marine Systems, Volume 208, 2020, 103347, ISSN 0924-7963, https://doi.org/10.1016/j.jmarsys.2020.103347.

Khabat KHOSRAVI, Ali GOLKARIAN, Rahim BARZEGAR, Mohammad T. AALAMI, Salim HRDDAM, Ebrahim OMIDVAR, Saskia D. KEESSTRA, Manuel LÓPEZ-VICENTE, "Multi-step-ahead soil temperature forecasting at multiple-depth based on meteorological data: Integrating resampling algorithms and machine learning models", Pedosphere, 2022, ISSN 1002-0160, https://doi.org/10.1016/j.pedsph.2022.06.056.

Qian, Q. F., Jia, X. J., & Lin, H. (2020). "Machine learning models for the seasonal forecast of winter surface air temperature in North America. Earth and Space Science", 7, e2020EA001140. https://doi.org/10.1029/2020EA001140

Alizamir M, Kisi O, Ahmed AN, Mert C, Fai CM, Kim S, et al. (2020) "Advanced machine learning model for better prediction accuracy of soil temperature at different depths". PLoS ONE 15(4): e0231055. https://doi.org/10.1371/journal.pone.0231055

Renfei He, Limao Zhang, Alvin Wei Ze Chew, "Modeling and predicting rainfall time series using seasonal-trend decomposition and machine learning", Knowledge-Based Systems, Volume 251, 2022, 109125, ISSN 0950-7051, https://doi.org/10.1016/j.knosys.2022.109125.

C. Z. Basha, N. Bhavana, P. Bhavya and S. V, "Rainfall Prediction using Machine Learning & Deep Learning Techniques," 2020 International Conference on

Electronics and Sustainable Communication Systems (ICESC), 2020, pp. 92-97, doi: 10.1109/ICESC48915.2020.9155896.

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

Mansfield, L.A., Nowack, P.J., Kasoar, M. et al. “Predicting global patterns of long-term climate change from short-term simulations using machine learning”. npj Clim Atmos Sci 3, 44 (2020). https://doi.org/10.1038/s41612-020-00148-5

Zahura, F. T., Goodall, J. L., Sadler, J. M., Shen, Y., Morsy, M. M., & Behl, M. (2020). “Training machine learning surrogate models from a high-fidelity physics-based model: Application for real-time street-scale flood prediction in an urban coastal community”. Water Resources Research, 56, e2019WR027038. https://doi.org/10.1029/2019WR027038

Jhajharia, Kavita, and Pratistha Mathur. "Machine Learning Approaches to Predict Crop Yield Using Integrated Satellite and Climate Data." International Journal of Ambient Computing and Intelligence (IJACI) 13.1 (2022): 1-17.

Yaping Cai, Kaiyu Guan, David Lobell, Andries B. Potgieter, Shaowen Wang, Jian Peng, Tianfang Xu, Senthold Asseng, Yongguang Zhang, Liangzhi You, Bin Peng, “Integrating satellite and climate data to predict wheat yield in Australia using machine learning approaches”, Agricultural and Forest Meteorology, Volume 274, 2019, Pages 144-159, ISSN 0168-1923, https://doi.org/10.1016/j.agrformet.2019.03.010.

Elisa Kamir, François Waldner, Zvi Hochman, “Estimating wheat yields in Australia using climate records, satellite image time series and machine learning methods”, ISPRS Journal of Photogrammetry and Remote Sensing, Volume 160, 2020, Pages 124-135, ISSN 0924-2716, https://doi.org/10.1016/j.isprsjprs.2019.11.008.

Diego Gómez, Pablo Salvador, Julia Sanz, José Luis Casanova, “Modelling wheat yield with antecedent information, satellite and climate data using machine learning methods in Mexico”, Agricultural and Forest Meteorology, Volume 300, 2021, 108317, ISSN 0168-1923, https://doi.org/10.1016/j.agrformet.2020.108317.

Weimo Zhou, Yujie Liu, Syed Tahir Ata-Ul-Karim, Quansheng Ge, Xing Li, Jingfeng Xiao, “Integrating climate and satellite remote sensing data for predicting county-level wheat yield in China using machine learning methods”, International Journal of Applied Earth Observation and Geoinformation, Volume 111, 2022, 102861, ISSN 1569-8432, https://doi.org/10.1016/j.jag.2022.102861.

Bouras, E.h.; Jarlan, L.; Er-Raki, S.; Balaghi, R.; Amazirh, A.; Richard, B.; Khabba, S. “Cereal Yield Forecasting with Satellite Drought-Based Indices, Weather Data and Regional Climate Indices Using Machine Learning in Morocco”. Remote Sens. 2021, 13, 3101. https://doi.org/10.3390/rs13163101

Deen Dayal, Praveen K. Gupta & Ashish Pandey (2021) “Streamflow estimation using satellite-retrieved water fluxes and machine learning technique over monsoon-dominated catchments of India”, Hydrological Sciences Journal, 66:4, 656-671, DOI: 10.1080/02626667.2021.1889557

L. Cornejo-Bueno, C. Casanova-Mateo, J. Sanz-Justo, S. Salcedo-Sanz, “Machine learning regressors for solar radiation estimation from satellite data”, Solar Energy, Volume 183, 2019, Pages 768-775, ISSN 0038-092X, https://doi.org/10.1016/j.solener.2019.03.079.

Schneider, R.; Vicedo-Cabrera, A.M.; Sera, F.; Masselot, P.; Stafoggia, M.; de Hoogh, K.; Kloog, I.; Reis, S.; Vieno, M.; Gasparrini, A. “A Satellite-Based Spatio-Temporal Machine Learning Model to Reconstruct Daily PM2.5 Concentrations across Great Britain”. Remote Sens. 2020, 12, 3803. https://doi.org/10.3390/rs12223803

Rojo Hernández, J. D., Mesa, Ó. J., & Lall, U. (2020). “ENSO Dynamics, Trends, and Prediction Using Machine Learning, Weather and Forecasting”, 35(5), 2061-2081. Retrieved Nov 2, 2022, from https://journals.ametsoc.org/view/journals/wefo/35/5/wafD200031.xml

Pal, M., Maity, R., Ratnam, J.V. et al. “Long-lead Prediction of ENSO Modoki Index using Machine Learning algorithms”. Sci Rep 10, 365 (2020). https://doi.org/10.1038/s41598-019-57183-3

Lima, C.H.R., Lall, U., Jebara, T., Barnston, A.G. (2015). “Machine Learning Methods for ENSO Analysis and Prediction”. In: Lakshmanan, V., Gilleland, E., McGovern, A., Tingley, M. (eds) Machine Learning and Data Mining Approaches to Climate Science. Springer, Cham. https://doi.org/10.1007/978-3-319-17220-0_2

Dijkstra HA, Petersik P, Hernández-García E and López C (2019) “The Application of Machine Learning Techniques to Improve El Niño Prediction Skill”. Front. Phys. 7:153. doi: 10.3389/fphy.2019.00153

Nooteboom, P. D., Feng, Q. Y., López, C., Hernández-García, E., and Dijkstra, H. A.: “Using network theory and machine learning to predict El Niño”, Earth Syst. Dynam., 9, 969–983, https://doi.org/10.5194/esd-9-969-2018, 2018.

Maher, N., Tabarin, T. P., and Milinski, S.: “Combining machine learning and SMILEs to classify, better understand, and project changes in ENSO events”, Earth Syst. Dynam., 13, 1289–1304, https://doi.org/10.5194/esd-13-1289-2022, 2022.

Chen, N., Gilani, F., & Harlim, J. (2021). “A Bayesian machine learning algorithm for predicting ENSO using short observational time series”. Geophysical Research Letters, 48, e2021GL093704. https://doi.org/10.1029/2021GL093704

M. Gupta, H. Kodamana and S. Sandeep, "Prediction of ENSO Beyond Spring Predictability Barrier Using Deep Convolutional LSTM Networks," in IEEE

Geoscience and Remote Sensing Letters, vol. 19, pp. 1-5, 2022, Art no. 1501205, doi: 10.1109/LGRS.2020.3032353.

He, D., Lin, P., Liu, H., Ding, L., Jiang, J. (2019). “DLENSO: A Deep Learning ENSO Forecasting Model”. In: Nayak, A., Sharma, A. (eds) PRICAI 2019: Trends in Artificial Intelligence. PRICAI 2019. Lecture Notes in Computer Science(), vol 11671. Springer, Cham. https://doi.org/10.1007/978-3-030-29911-8_2

B. Mu, C. Peng, S. Yuan and L. Chen, "ENSO Forecasting over Multiple Time Horizons Using ConvLSTM Network and Rolling Mechanism," 2019 International Joint Conference on Neural Networks (IJCNN), 2019, pp. 1-8, doi: 10.1109/IJCNN.2019.8851967.

Songyan Zhu, Robert Clement, Jon McCalmont, Christian A. Davies, Timothy Hill, “Stable gap-filling for longer eddy covariance data gaps: A globally validated machine-learning approach for carbon dioxide, water, and energy fluxes”, Agricultural and Forest Meteorology, Volume 314, 2022, 108777, ISSN 0168-1923, https://doi.org/10.1016/j.agrformet.2021.108777.

Sarafanov, M.; Kazakov, E.; Nikitin, N.O.; Kalyuzhnaya, A.V. “A Machine Learning Approach for Remote Sensing Data Gap-Filling with Open-Source Implementation: An Example Regarding Land Surface Temperature, Surface Albedo and NDVI”. Remote Sens. 2020, 12, 3865. https://doi.org/10.3390/rs12233865

Pedro Arriagada, Bruno Karelovic, Oscar Link, “Automatic gap-filling of daily streamflow time series in data-scarce regions using a machine learning algorithm”, Journal of Hydrology, Volume 598, 2021, 126454, ISSN 0022-1694, https://doi.org/10.1016/j.jhydrol.2021.126454. (https://www.sciencedirect.com/science/article/pii/S0022169421005011)

Olaf Menzer, Wendy Meiring, Phaedon C. Kyriakidis, Joseph P. McFadden, “Annual sums of carbon dioxide exchange over a heterogeneous urban landscape through machine learning based gap-filling”, Atmospheric Environment, Volume 101, 2015, Pages 312-327, ISSN 1352-2310, https://doi.org/10.1016/j.atmosenv.2014.11.006.

Sun, H.; Xu, Q. “Evaluating Machine Learning and Geostatistical Methods for Spatial Gap-Filling of Monthly ESA CCI Soil Moisture in China”. Remote Sens. 2021, 13, 2848. https://doi.org/10.3390/rs13142848

C. Tong, H. Wang, R. Magagi, K. Goïta and K. Wang, "Spatial Gap-Filling of SMAP Soil Moisture Pixels Over Tibetan Plateau via Machine Learning Versus Geostatistics," in IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 14, pp. 9899-9912, 2021, doi: 10.1109/JSTARS.2021.3112623.

Bellido-Jiménez, J.A.; Gualda, J.E.; García-Marín, A.P. “Assessing Machine Learning Models for Gap Filling Daily Rainfall Series in a Semiarid Region of Spain”. Atmosphere 2021, 12, 1158. https://doi.org/10.3390/atmos12091158

Kim, Y, Johnson, MS, Knox, SH, et al. “Gap-filling approaches for eddy covariance methane fluxes: A comparison of three machine learning algorithms and a traditional method with principal component analysis”. Glob Change Biol. 2020; 26: 1499– 1518. https://doi.org/10.1111/gcb.14845

Khan, M.S.; Jeon, S.B.; Jeong, M.-H. “Gap-Filling Eddy Covariance Latent Heat Flux: Inter-Comparison of Four Machine Learning Model Predictions and Uncertainties in Forest Ecosystem”. Remote Sens. 2021, 13, 4976. https://doi.org/10.3390/rs13244976

Tzanis, C.G.; Alimissis, A.; Koutsogiannis, I. “Addressing Missing Environmental Data via a Machine Learning Scheme”. Atmosphere 2021, 12, 499. https://doi.org/10.3390/atmos12040499

Aoxuan Chen, Jin Yang, Yan He, Qiangqiang Yuan, Zhengqiang Li, Liye Zhu, “High spatiotemporal resolution estimation of AOD from Himawari-8 using an ensemble machine learning gap-filling method”, Science of The Total Environment, Volume 857, Part 3, 2023, 159673, ISSN 0048-9697, https://doi.org/10.1016/j.scitotenv.2022.159673.

K. Bala, D. K. Choubey and S. Paul, "Soft computing and data mining techniques for thunderstorms and lightning prediction: A survey," 2017 International conference of Electronics, Communication and Aerospace Technology (ICECA), 2017, pp. 42-46, doi: 10.1109/ICECA.2017.8203729.

Chakrabarty, Himadri, and Sonia Bhattacharya. "Application of K-nearest neighbor technique to predict severe thunderstorms." International Journal of Computer Applications 110.10 (2015): 1-4.

Lagerquist, R., McGovern, A., Homeyer, C. R., Gagne II, D. J., & Smith, T. (2020). “Deep Learning on Three-Dimensional Multiscale Data for Next-Hour Tornado Prediction”, Monthly Weather Review, 148(7), 2837-2861. Retrieved Nov 2, 2022, from https://journals.ametsoc.org/view/journals/mwre/148/7/mwrD190372.xml

Kamangir, H, Collins, W, Tissot, P, King, SA. “A deep-learning model to predict thunderstorms within 400 km2 South Texas domains”. Meteorol Appl. 2020; 27:e1905. https://doi.org/10.1002/met.1905

Trafalis, T.B., Adrianto, I., Richman, M.B. (2007). “Active Learning with Support Vector Machines for Tornado Prediction”. In: Shi, Y., van Albada, G.D., Dongarra, J., Sloot, P.M.A. (eds) Computational Science – ICCS 2007. ICCS 2007. Lecture Notes in Computer Science, vol 4487. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-540-72584-8_148

Aleskerov, Fuad and Demin, Sergey and Richman, Michael B. and Shvydun, Sergey and Trafalis, Theodore B. and Yakuba, Vyacheslav. "Constructing an Efficient Machine Learning Model for Tornado Prediction" International Journal of Information Technology & Decision Making 2020 19.05 1177-1187 https://doi.org/10.1142/S0219622020500261

Coffer, B., Kubacki, M., Wen, Y., Zhang, T., Barajas, C.A. and Gobbert, M.K. (2021), “Machine Learning with Feature Importance Analysis for Tornado Prediction from Environmental Sounding Data”. Proc. Appl. Math. Mech., 20: e202000112. https://doi.org/10.1002/pamm.202000112

Indra Adrianto, Theodore B. Trafalis & Valliappa Lakshmanan (2009) “Support vector machines for spatiotemporal tornado prediction”, International Journal of General Systems, 38:7, 759-776, DOI: 10.1080/03081070601068629

Michael P. McGuire, Todd W. Moore, “Prediction of tornado days in the United States with deep convolutional neural networks”, Computers & Geosciences, Volume 159, 2022, 104990, ISSN 0098-3004, https://doi.org/10.1016/j.cageo.2021.104990.

Adrianto, Indra, Michael B. Richman, and Theodore B. Trafalis. "Machine learning techniques for imbalanced data: an application for tornado detection." Proceedings of the international conference on artificial neural networks in engineering. 2010.

Burke, A., Snook, N., Gagne II, D. J., McCorkle, S., & McGovern, A. (2020). “Calibration of Machine Learning–Based Probabilistic Hail Predictions for Operational Forecasting”, Weather and Forecasting, 35(1), 149-168. Retrieved Nov 2, 2022, from https://journals.ametsoc.org/view/journals/wefo/35/1/waf-d-19-0105.1.xml

Bartosz Czernecki, Mateusz Taszarek, Michał Marosz, Marek Półrolniczak, Leszek Kolendowicz, Andrzej Wyszogrodzki, Jan Szturc, “Application of machine learning to large hail prediction - The importance of radar reflectivity, lightning occurrence and convective parameters derived from ERA5, Atmospheric Research”, Volume 227, 2019, Pages 249-262, ISSN 0169-8095, https://doi.org/10.1016/j.atmosres.2019.05.010.

Gagne, D. J., II, McGovern, A., Haupt, S. E., Sobash, R. A., Williams, J. K., & Xue, M. (2017). “Storm-Based Probabilistic Hail Forecasting with Machine Learning Applied to Convection-Allowing Ensembles”, Weather and Forecasting, 32(5), 1819-1840. Retrieved Nov 2, 2022, from https://journals.ametsoc.org/view/journals/wefo/32/5/waf-d-17-0010_1.xml

Gagne II, D. J., McGovern, A., Brotzge, J., Coniglio, M., Correia Jr, J., & Xue, M. (2015, March). “Day-ahead hail prediction integrating machine learning with storm-scale numerical weather models”. In Twenty-Seventh IAAI Conference.

M. Pullman, I. Gurung, M. Maskey, R. Ramachandran and S. A. Christopher, "Applying Deep Learning to Hail Detection: A Case Study," in IEEE Transactions on Geoscience and Remote Sensing, vol. 57, no. 12, pp. 10218-10225, Dec. 2019, doi: 10.1109/TGRS.2019.2931944.

Zhang, Y., Ji, Z., Xue, B. et al. “A Novel Fusion Forecast Model for Hail Weather in Plateau Areas Based on Machine Learning”. J Meteorol Res 35, 896–910 (2021). https://doi.org/10.1007/s13351-021-1021-2

Pulukool, F.; Li, L.; Liu, C. “Using Deep Learning and Machine Learning Methods to Diagnose Hailstorms in Large-Scale Thermodynamic Environments”. Sustainability 2020, 12, 10499. https://doi.org/10.3390/su122410499

Han Yao, Xiaodong Li, Huaji Pang, Lifang Sheng, Wencai Wang, “Application of random forest algorithm in hail forecasting over Shandong Peninsula”, Atmospheric Research, Volume 244, 2020, 105093, ISSN 0169-8095, https://doi.org/10.1016/j.atmosres.2020.105093.

Apostolakis, A., Girtsou, S., Giannopoulos, G., Bartsotas, N. S., & Kontoes, C. (2022). “Estimating Next Day’s Forest Fire Risk via a Complete Machine Learning Methodology”. Remote Sensing, 14(5), 1222.

Barbero, R.; Abaoglou, J. T.; Larkin, N. K.; Kolden, C. A.; Stocks, B. 2015. “Climate change presents increased potential for very large fires in the contiguous United States”. International Journal of Wildland Fire. 24(7):892–899.

Fire Effects on Plants and Animals. “How fire affects plants and animals”, on NSW Environment. Available at: hps://www.environment.nsw.gov.au/topics/fire/plants-animals-fire

Fire Effects on the Environment. “Fire Effects on the Environment”, in the U.S. Forest Fires. Available at: hps://www.fs.usda.gov/pnw/page/fire-effects-environment

Fire Effects on the Soil. “Fire Effects on the Soil”, on Northern Arizona University. Available at: https://www2.nau.edu/~gaud/bio300w/frsl.html

Fire risk will increase by 2050, 2022. “Wildfires likely to increase by a third by 2050, warns UN” on The Guardian amended on 25 February 2022. Available at https://www.theguardian.com/environment/2022/feb/23/climate-crisis-driving-increase-in-wildfires-across-globe-says-report-ae

Global Warming and Extreme Wildfires, 2022. “Global warming and land use change to drive more extreme wildfires”, on BBC, by Ma McGrath, hps://www.bbc.com/news/science-environment-60483431

Global Wildfire Crisis, 2022. “Climate Scientists Warn of a ‘Global Wildfire Crisis’” in the New York Times, by Raimond Zhong. Updated on 22 February 2022. Available at https://www.nytimes.com/2022/02/23/climate/climate-change-un-wildfire-report.html

Will global warming produce more frequent and more intense wildfires?. “Will global warming produce more frequent and more intense wildfires?” on the USGS website. Available at https://www.usgs.gov/faqs/will-global-warming-produce-more-frequent-and-more-intense-wildfires

I. N. Kosović and D. Škurić Kuraži, "Machine Learning Approach in Fire Risk Estimation," 2021 International Conference on Software, Telecommunications and Computer Networks (SoftCOM), 2021, pp. 1-6, doi: 10.23919/SoftCOM52868.2021.9559064.

N. Omar, A. Al-zebari and A. Sengur, "Deep Learning Approach to Predict Forest Fires Using Meteorological Measurements," 2021 2nd International Informatics and Software Engineering Conference (IISEC), 2021, pp. 1-4, doi: 10.1109/IISEC54230.2021.9672446.

Ricotta C., Guglietta D. Chapter 10 - “Indici di rischio di incendio boschivo”. Available at https://www.vigilfuoco.it/aspx/isaViewDoc.aspx?id=21&t=2

Rosadi, Dedi & Arisanty, Deasy & Agustina, Dina. (2022). “PREDICTION OF FOREST FIRE USING NEURAL NETWORKS WITH BACKPROPAGATION LEARNING AND EXTREME LEARNING MACHINE APPROACH USING METEOROLOGICAL AND WEATHER INDEX VARIABLES”. MEDIA STATISTIKA. 14. 118-124. 10.14710/medstat.14.2.118-124.

Safi, Youssef & Bouroumi, A.. (2013). “Prediction of forest fires using Artificial neural networks”. Applied Mathematical Sciences. 7. 271-286. 10.12988/ams.2013.13025.

Isola, Phillip and Zhu, Jun-Yan and Zhou, Tinghui and Efros, Alexei A. “Image-to-Image Translation with Conditional Adversarial Networks”. arXiv, 2016. Available at: https://arxiv.org/abs/1611.07004

About Copernicus. “About Copernicus”, on the Copernicus website. Available at: https://www.copernicus.eu/en/about-copernicus

ERA5 pressure levels dataset, 2021. “ERA5 hourly data on pressure levels from 1979 to present”, on Copernicus Climate Data Store. Last updated on 2021-12-03. Available at: https://cds.climate.copernicus.eu/cdsapp#!/dataset/reanalysis-era5-pressure-levels?tab=overview

ERA5 single levels dataset, 2021. “ERA5 hourly data on single levels from 1979 to present”, on Copernicus Climate Data Store. Last updated on 2021-12-03. Available at: https://cds.climate.copernicus.eu/cdsapp#!/dataset/reanalysis-era5-single-levels?tab=overview

Mirza, Mehdi and Osindero, Simon. “Conditional Generative Adversarial Nets”. arXiv, 2014. Available at: https://arxiv.org/abs/1411.1784

Ronneberger, Olaf and Fischer, Philipp and Brox, Thomas. “U-Net: Convolutional Networks for Biomedical Image Segmentation”. arXiv, 2015. Available at: https://arxiv.org/abs/1505.04597

L1 and L2 functions, 2019. “What Are L1 and L2 Loss Functions?”, on AfterAcademy. Last updated on 2 August 2019. Available at: https://afteracademy.com/blog/what-are-l1-and-l2-loss-functions

Sigmoid, 2019. “Sigmoid Activation and Binary Crossentropy —A Less Than Perfect Match?”, on Towards Data Science. Last updated on February 21, 2019.

Available at: https://towardsdatascience.com/sigmoid-activation-and-binary-crossentropy-a-less-than-perfect-match-b801e130e31

Pix2pix Implementation, 2022. “pix2pix: Image-to-image translation with a conditional GAN” on Tensorflow. Available at https://www.tensorflow.org/tutorials/generative/pix2pix. Last updated on 2022-01-26.

Activation Functions Explained, 2017. “Activation Functions in Neural Networks”, on Towards Data Science. Last updated on September 6, 2017. Available at: https://towardsdatascience.com/activation-functions-neural-networks-1cbd9f8d91d6

Manepalli, A., Albert, A., Rhoades, A., Feldman, D., & Jones, A. D. (2019, December). “Emulating numeric hydroclimate models with physics-informed cGANs”. In *AGU fall meeting*.

Ordered Enqueuer doc, 2022. “tf.keras.utils.OrderedEnqueuer”, on Tensorflow website. Last updated: 2022-02-03. Available at: https://www.tensorflow.org/api_docs/python/tf/keras/utils/OrderedEnqueuer

TF Dataset doc, 2022. “tf.data.Dataset”, on Tensorflow website. Last updated: 2022-03-26. Available at: https://www.tensorflow.org/api_docs/python/tf/data/Dataset

Mirrored Strategy doc, 2022. “tf.distribute.MirroredStrategy”, on Tensorflow website. Last updated: 2022-02-03. Available at: https://www.tensorflow.org/api_docs/python/tf/distribute/MirroredStrategy

FWI Classification, Climate Adapt. “Fire Weather Index - Monthly Mean, 1979-2020”, on Climate ADAPT. Available at: https://climate-adapt.eea.europa.eu/metadata/indicators/fire-weather-index-monthly-mean-1979-2019#:~:text=The%20fire%20weather%20index%20can,is%20between%2021.3%20and%2038.0

Jensen Shannon Divergence computation, 2020. “Measuring the statistical similarity between two samples using Jensen-Shannon and Kullback-Leibler divergences”, on Medium. Last updated on February 28, 2020. Available at: https://medium.com/datalab-log/measuring-the-statistical-similarity-between-two-samples-using-jensen-shannon-and-kullback-leibler-8d05af514b15

Kullback Leibler Divergence computation, 2019. “How to Calculate the KL Divergence for Machine Learning”, on Machine Learning Mastery. Last updated on October 18, 2019. Available at https://machinelearningmastery.com/divergence-between-probability-distributions/

Kashinath K., Mustafa M., Albert A., Wu J-L., Jiang C., Esmaeilzadeh S., Azizzadenesheli K., Wang R., Chattopadhyay A., Singh A., Manepalli A., Chirila D., Yu R., Walters R., White B., Xiao H., Tchelepi H. A., Marcus P., Anandkumar A., Hassanzadeh P. and Prabhat. 2021. “Physics-informed machine learning: case studies for weather and climate modelling”. Phil. Trans. R. Soc.

A.3792020009320200093. Available at: https://royalsocietypublishing.org/doi/10.1098/rsta.2020.0093

Jin-Long Wu and Karthik Kashinath and Adrian Albert and Dragos Chirila and Prabhat and Heng Xiao. “Enforcing statistical constraints in generative adversarial networks for modeling chaotic dynamical systems”. Published on April, 2020. On Journal of Computational Physics https://arxiv.org/abs/1905.06841

Singh et al., 2019. Singh, Alok Pratap. “Numerical Weather Model Super-Resolution.” (2019). https://ml4physicalsciences.github.io/2019/files/NeurIPS_ML4PS_2019_75.pdf

Xu, Y. “Characteristics and Psychological Response in the Post-disaster Era”. Logist. Chin. Gov. 2020, pp. 68–69, doi: 10.3969/j.issn.1007-4775.2020.06.031

Emanuel, K. (2003). “Tropical cyclones”. Annual Review of Earth and Planetary Sciences, 31(1):75-104.

Riehl, H. (1950). “A model of hurricane formation”. Journal of Applied Physics, 21(9):917-925.

Kleinschmidt, E. (1951). “Grundlagen einer Theorieder tropischen Zyklonen”. Archives for Meteorology Geophysics and Bioclimatology Series A Meteorology and Atmopsheric Physics, 4(1):53-72.

Roy, C. and Kovordányi, R. (2012). “Tropical cyclone track forecasting techniques|a review”. Atmospheric Research, 104-105:40-69.

Debsarma, S. (2001). “Cyclone and its warning system in bangladesh”. National Disaster Reduction Day.

Tong, B., Sun, X., Fu, J., He, Y., and Chan, P.: “Identification of tropical cyclones via deep convolutional neural network based on satellite cloud images”, Atmos. Meas. Tech., 15, 1829–1848, https://doi.org/10.5194/amt-15-1829-2022, 2022.

Nair, A., Srujan, K. S., Kulkarni, S. R., Alwadhi, K., Jain, N., Kodamana, H., Sandeep, S., John, V. O. (2022). “A deep learning framework for the detection of tropical cyclones from satellite images”. IEEE Geoscience and Remote Sensing Letters, vol. 19, pp. 1-5., Art no. 1004405, doi: 10.1109/LGRS.2021.3131638.

Matsuoka, D., Nakano, M., Sugiyama, D. et al. “Deep learning approach for detecting tropical cyclones and their precursors in the simulation by a cloud-resolving global nonhydrostatic atmospheric model”. Prog Earth Planet Sci 5, 80 (2018). https://doi.org/10.1186/s40645-018-0245-y

Gardoll, S. and Boucher, O.: “Classification of tropical cyclone containing images using a convolutional neural network: performance and sensitivity to the learning dataset”, Geosci. Model Dev., 15, 7051–7073, https://doi.org/10.5194/gmd-15-7051-2022, 2022.

Pang, S.; Xie, P.; Xu, D.; Meng, F.; Tao, X.; Li, B.; Li, Y.; Song, T. “NDFTC: A New Detection Framework of Tropical Cyclones from Meteorological Satellite Images with Deep Transfer Learning”. Remote Sens. 2021, 13, 1860. https://doi.org/10.3390/rs13091860

Fang, W.; Xue, Q.; Shen, L.; Sheng, V.S. “Survey on the Application of Deep Learning in Extreme Weather Prediction”. Atmosphere 2021, 12, 661, doi: 10.3390/atmos12060661

Rüttgers et al., 2018a. Rüttgers, M., Lee, S., and You, D. (2018a). “Prediction of typhoon tracks using a generative adversarial network with observational and meteorological data”.

Rüttgers et al., 2018b. Rüttgers, M., Lee, S., Jeon, S. *et al.* “Prediction of a typhoon track using a generative adversarial network and satellite images”. *Sci Rep* **9**, 6057 (2019). https://doi.org/10.1038/s41598-019-42339-y

Mudigonda, M., Kim, S., Mahesh, A., Kahou, S., Kashinath, K., Williams, D., Michalski, V., O’Brien, T., and Prabhat, M. (2017). “Segmenting and tracking extreme climate events using neural networks”. In Deep Learning for Physical Sciences (DLPS) Workshop, held with 31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, California, United States.

Giffard-Roisin, S., Yang, M., Charpiat, G., Kégl, B., and Monteleoni, C. (2018). “Fused deep learning for hurricane track forecast from reanalysis data”. In 8th International Workshop on Climate Informatics, Boulder, Colorado, United States.

Giffard-Roisin, S., Yang, M., Charpiat, G., Kumler Bonfanti, C., Kégl, B., and Monteleoni, C. (2020). “Tropical cyclone track forecasting using fused deep learning from aligned reanalysis data”. Frontiers in Big Data, 3.

Matsuoka, D., Nakano, M., Sugiyama, D., and Uchida, S. (2018). “Deep learning approach for detecting tropical cyclones and their precursors in the simulation by a cloud-resolving global nonhydrostatic atmospheric model”. Progress in Earth and Planetary Science, 5(1):80.

Chaudhuri, S., Dutta, D., Goswami, S., and Middey, A. (2015). “Track and intensity forecast of tropical cyclones over the north indian ocean with multilayer feed forward neural nets”. Meteorological Applications, 22(3):563–575.

Alemany, S., Beltran, J., Perez, A., and Ganzfried, S. (2018). “Predicting hurricane trajectories using a recurrent neural network”. Proceedings of the AAAI Conference on Artificial Intelligence, 33.

Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." arXiv preprint arXiv:1409.1556 (2014).