FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Weather AI

Diogo Maria Rodrigues da Cunha Mariano

Mestrado Integrado em Engenharia Eletrotécnica e de Computadores

Supervisor: Eng. Vitor Teixeira

Second Supervisor: Profa.Dra.Maria Teresa de Andrade

July 31, 2020

Resumo

O objectivo do trabalho exposto foi desenvolver um algoritmo de Machine Learning capaz de emular a variabilidade climática de modo a que fosse possível extrair resultados de previsão.

Este projecto começou com a selecção dos dados a serem utilizados, os dados escolhidos foram os relatórios "Global Surface Summary of the Day" de uma das estações meteorológicas do "Porto", e dados relativos a teleconexões, tais como a North Atlantic Oscillation (NAO), a Arctic Oscillation (AO) e a Pacific Decadal Oscillation (PDO). Após a sua selecção, os dados foram processados e submetidos a técnicas de limpeza e filtragem. Vários datasets tiveram origem a partir destes dados e subsequentemente foram fornecidos aos vários modelos de deep learning desenvolvidos, sendo os mesmos designados: Multi-Channel CNN, Vanilla LSTM, Encoder-Decoder LSTM, CNN-LSTM Encoder-Decoder and ConvLSTM Encoder-Decoder.

Os testes foram realizados por dataset, cada modelo seria avaliado tendo em conta uma configuração de entrada e saída. Os resultados foram avaliados através das seguintes medições estatísticas: Coeficiente de Determinação, R^2 *Score*; Soma de Erros Quadrados, SSE; e o Índice de Persistência, PERS.

Após analisados os resultados, concluiu-se que é possível fazer previsões do estado da atmosfera com técnicas de deep learning utilizando apenas dados do histórico.

Palavras Chave: Previsão climática, Previsão meteorológica, Machine Learning, Deep Learning, Tempo, Teleconexão, GSOD, NAO, AO, PDO

Abstract

The objective of following work was to develop a Machine Learning algorithm capable of emulating climate variability so that it could be possible to extract forecasting results.
This project started with the selection of the data to be utilised throughout the rest of the dissertation, the data chosen was the "Global Surface Summary of the Day" reports from one of "Oporto"'s weather stations, and data regarding teleconnections, such as North Atlantic Oscillation (NAO), Arctic Oscillation (AO) and the Pacific Decadal Oscillation (PDO). Following their selection, the data was processed and subjected to data cleaning and filtering techniques. Several datasets originated from these data and subsequently were provided to the various deep learning models developed, being the same designated as: Multi-Channel CNN, Vanilla LSTM, Encoder-Decoder LSTM, CNN-LSTM Encoder-Decoder and ConvLSTM Encoder-Decoder.
The tests were carried out by dataset, each model would be evaluated taking into account an input and output configuration. The results were evaluated by the following statistical measurements: Coefficient of Determination, R^2 *Score*; Sum of Squared Errors, SSE; and the Persistence Index, PERS.
Upon analysis of the results it was concluded that it is possible to make predictions of the state of the atmosphere with deep learning techniques using only historical data.

Keywords: Climate Prediction, Weather Forecast, Machine Learning, Deep Learning, Weather, Teleconnection, GSOD, NAO, AO, PDO

Acknowledgements

This section enables the opportunity to praise every single person which, directly or indirectly, accompanied and backed the elaboration of this thesis.
Firstly, to my advisor Mr. Vitor Teixeira and my second advisor Prof[a] Maria Teresa de Andrade, I acknowledge all the support and experience bestowed upon me. I also recognise the availability demonstrated for every single "video-conference" that took place during this eventful semester. Without their guidance and knowledged this dissertation would not see the light of day.
To all my colleagues and friends, from Aveiro to Porto, thank you. Thank you for the patience, the company, the strength of will that you gave me and the friendship that I would not trade for anything in this world. Special thanks to the group that usually lives in room 101 of the MIEEC department, the eternal explorers of the question, because without them the student life would not be the same.
And lastly, a special thanks to my family. I'm incredibly grateful for the emotional and overall support I've had all my life. Thanks to my father for showing me and motivating me to the world of technology, to my grandmother for her patience in helping me with my homework, to Pita that has been with my family for so long and has taught me how to be a good and patient man, to my grandfather for being, and continuing to be, my first best friend, and last but certainly most important, thanks to my mother who is responsible for everything in my life and what I have become. I profoundly hope that you will all be proud of me and of what I have been able to do.
This project was a study proposed by Mr. Vitor Teixeira Teixeira through the company "iClimate Adviser".

Diogo Mariano

"No Machine Learning model/algorithm could predict the present year 2020.
So as a matter of justice, don't expect mine to work."

Diogo Mariano

Contents

List of Figures

List of Tables

Abbreviations & Symbols

GSOD	Global Surface Summary of the Day
ECV	Essential Climate Variables
NAO	North Atlantic Oscillation
AO	Arctic Oscillation
PDO	Pacific Decadal Oscillation
WMO	World Meteorological Organization
NWP	Numerical Weather Predictions
GCM	Global Climate Models
RCM	Regional Climate Models
ESDM	Empirical statistical Downscaling models
ML	Machine Learning
ANN	Artificial Neuronal Networks
AR	Auto Regressive
MA	Moving Average
ARMA	Auto Regressive Moving Average
ARIMA	Auto Regressive Integrated Moving Average
K-NN	K-Nearest-Neighbor
SVR	Support Vector Regression
SVM	Support Vector Machine
CART	Classification and Regression Tree
RNN	Recurrent Neural Network
ConvLSTM	Convolutional LSTM
MANN	Modular Artificial Neural Networks
CNN	Convolutional Neural Network
MLP	Multilayer Perceptron Networks
ERNN	ElmanRecurrent Neural Network
RBFN	Radial Basis Function Network
HFM	Hopfield Model
BNN	Bayesian Neural Network
LSTM	Long Short-Term Memory
LCA	Linear Correlation Analysis
SSA	Singular Spectrum Analysis
WA-ANN	Wavelet-Neural Network
RMSE	Root Mean Squared Error
MAE	Mean Absolute Error
PSO-SVM	Particle Swarm Optimization - Support Vector Machine
GP	Gaussian process
MLR	Multiple Linear Regression

NCEI	National Centers for Environmental Information
NOAA	National Oceanic and Atmospheric Administration
NCDC	National Climatic Data Center
NGDC	National Geophysical Data Center
NODC	National Oceanographic Data Center
NCDDC	National CoastalData Development Center
GAW	Global Atmosphere Watch
FTP	File Transfer Protocol
ISH	Integrated Surface Hourly
USAF	Air Force station ID
TEMP	Mean Temperature
DEWP	Mean Dew Point
MIN	Minimum Temperature
MAX	Maximum Temperature
VISIB	Mean Visibility
WDSP	Mean Wind Speed
MXSPD	Maximum Sustained Wind Speed
GUST	Maximum Wind Gust
PRCP	Total Precipitation
SNDP	Snow Depth
SLP	Sea Level Pressure
STP	Mean Station Pressure
ICE	Northern Hemisphere Sea Ice Extent
FRSHTT	Indicators of Fog, Rain or Drizzle, Snow or Ice Pellets, Hail,Thunder, Tornado or Funnel Cloud
Adam	Adaptive MomentEstimation
MSE	Mean Squared Error
ReLU	Rectified Linear Activation Unit
R^2	Coefficient of Determination
SSE	Sum of Squared Errors
PERS	Persistence Index

Chapter 1

Introduction

1.1 Context

The planet Earth is under constant observation by several weather stations, these facilities capture data relative to current weather conditions. Generally, these centres report at 15 min to hourly intervals thus creating huge amounts of data. Since sample intervals are not deterministic the collected data is processed to generate daily weather reports, such as "Global Surface Summary of the Day" reports or commonly known as GSOD. Collecting data samples through long periods of time enables the comprehensive study of weather and climate, thus, allowing for a better understanding of its behaviour and the possibility to formulate more advanced ways to predict meteorological events.

Climate and weather have always been predicted through various models, these are composed of equations that describe the physical and chaotic behaviour of the atmosphere, and as one would expect these need enormous computational power in order to solve and predict the next stages of the atmosphere. And even with the computational power available these models have a maximum resolution of two weeks. Therefore, statistical models, less computationally powerful, were designed to assist the existing models, or even to predict by themselves the next states of the atmosphere. However, these have proven not to be better than traditional models, namely because of the chaotic/non-linear nature of the atmosphere. Even with the presence of non-linear statistical models they suffer from the limitation that it is necessary to establish the non-linearity in order for them to work. But currently the hope of replacing traditional models is high with the appearance of machine learning algorithms, more specifically Deep Learning, which are appreciated for being able to make non-linear predictions without the need to established the non-linearity.

Throughout the course of this thesis an application of climate prediction will be developed using machine learning techniques, specifically Deep Learning.

First the data to be used will be selected. Subsequently the data will be subject to preprocessing techniques in order to clean and filter it. Several deep learning models will be developed to be tested later on with the data. At the end of this project it will be proven that the hope of predicting the

state of the atmosphere through deep learning techniques is possible and that it shows an evolution on the knowledge of the atmosphere and its behaviour.

1.2 Motivation

Climate and weather forecasts influence our daily lives, it can be a key element for business estimates/decisions, and a preventive mechanism for large and dangerous meteorological events that can have societal impact, such as displacement or migration of people. Forecasts are of extreme importance for several sectors of the economy, e.g. agriculture, tourism, insurance losses and even for event companies.

The increasing availability of large amounts of historical data opens the opportunity to generate studies on the long-term evolution of the atmosphere and try to forecast its future behaviour.

1.3 Objectives

The main objective of this dissertation was to develop a Machine Learning algorithm capable of emulating climate variability so that it could be possible to extract forecasting results as good or better than traditional forecasting models being these statistical or conditioned by physical equations of the atmosphere.

Chapter 2

Background

The present chapter will provide a means of contextualising terms, methods and techniques related to the target theme of this project which is Climate Predictions.

Firstly a brief introduction on climate, weather and their differences will be carried out. Afterwards the climatic variability and certain events that lead to it will be presented, such as teleconnections. Concluding with the survey of how weather and climate predictions were originated, existing models, stations and observation centres worldwide responsible for this task.

2.1 Weather and Climate

Weather and climate are meteorological terms although related they are not interchangeable. Weather describes variation of the atmospheric behaviour over short periods of time, whereas, climate describes the weather conditions for a particular location through long periods of time.

2.1.1 Meteorology and Climatology

Climatology and Meteorology are branches of atmospheric science and both take into account the study of atmospheric processes.

Meteorology is the study of weather or atmospheric processes. It is considered to be a branch of atmospheric science which deals with weather phenomena and weather changes over a short timescale [17].

Climatology is the study of atmospheric behaviour and changes in its factors over long periods of time. Responsible for the research of climate: variations, extremes, and the influence on a variety of activities including human health, safety and welfare to support evidence-based decision-making on how to best adapt to a changing climate [17].

The understanding of climate is rooted in observations of the atmosphere, oceans, land surface, the hydrological and carbon cycles and the cryosphere. Utilising weather observations made regularly over a period of time, can provide means to quantify long-term average conditions and gain insight into an area's climate. Climatologists use climate normals — 30-year historical averages of variables — as benchmarks, in order to historically contextualise some climatic events

(e.g. heatwaves). Climate characterisation is influenced by several variables, which are referred to as Essential Climate Variables, or ECV. "An ECV is a physical, chemical or biological variable or a group of linked variables that critically contributes to the characterisation of Earth' s climate."[1].

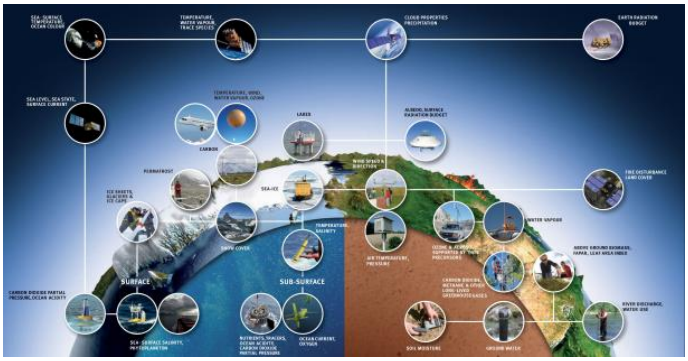


Figure 2.1: Essential Climate Variables, [1]

The proposed study for this dissertation focused mainly on the prediction of climatic variables of the surface of the atmosphere, also considered ECV, the temperature and precipitation.

2.1.2 Atmospheric Variability and Teleconnections

The atmospheric circulation is known to exhibit substantial variability. This variability reflects in weather patterns and circulation systems that occur on many time scales, lasting from a few days (i.e. normal storm system), a few weeks (i.e. mid-winter warm-up or a mid-summer wet period), a few months (i.e. particularly cold winters or hot summers), several years (i.e. abnormal winters for several years in a row) and even centuries (i.e. long-term climate change) [18].

Teleconnection pattern refers to a recurring and persistent, large-scale mode of pressure and circulation anomalies that spans through vast geographical areas. Although these patterns typically last for weeks to months, they can be prominent for several consecutive years, thus reflecting an important part of both inter-annual and inter-decadal variability of the atmospheric circulation. Many of the teleconnection are of planetary-scale nature, and span entire ocean basins and continents. These phenomena reflect large-scale changes in the atmospheric wave, jet stream patterns, temperature, rainfall, storm tracks and jet stream location and intensity over vast areas. Thus, they are often the culprit for abnormal weather patterns occurring simultaneously over seemingly vast distances [18].

Several teleconnection patterns have been identified, however for this study only the following were considered: North Atlantic Oscillation, Arctic Oscillation, Pacific Decadal Oscillation and Sea Ice and Extent.

2.1.2.1 North Atlantic Oscillation

The North Atlantic Oscillation (NAO), the leading recurring mode of variability in the extra-tropical North Atlantic region, describes an opposing pattern of sea level pressure between the Atlantic

subtropical high and the Iceland/Arctic low [19]. The positive phase of the NAO reflects below-normal heights and pressure across the high latitudes of the North Atlantic and above-normal heights and pressure over the central North Atlantic, the eastern United States and western Europe. The negative phase reflects an opposite pattern of height and pressure anomalies over these regions (2.2).

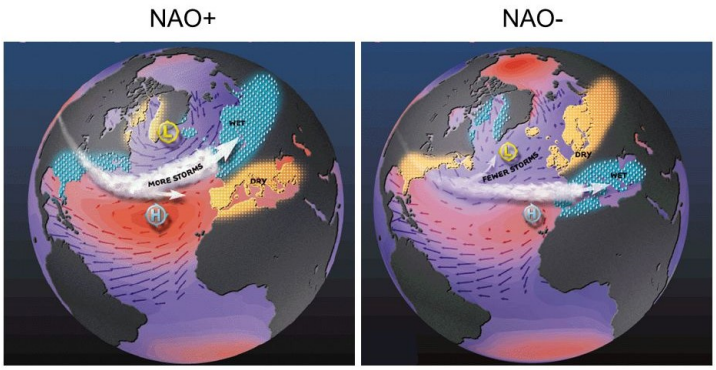


Figure 2.2: NAO: Negative and Positive phase, [20]

Both phases of the NAO are associated with basin-wide changes in the intensity and location of the North Atlantic jet stream and storm track, and in large-scale modulations of the normal patterns of zonal and meridional heat and moisture transport [21], which in turn results in changes in temperature and precipitation patterns often extending from eastern North America to western and central Europe [22].

The NAO index can be measured by the difference between the sea-level pressure of two observational stations located in Iceland and the Azores.

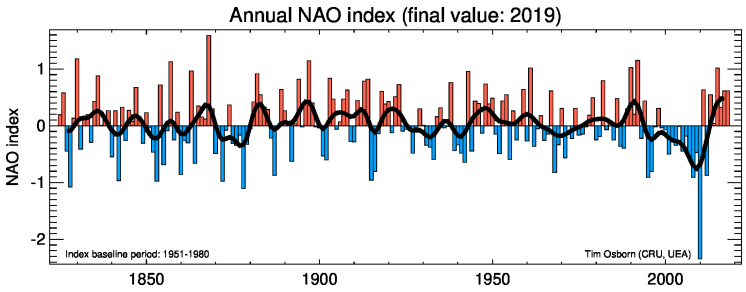


Figure 2.3: Annual NAO Index, [23]

If the state of this phenomena could be predicted in advance then extremely valuable seasonal climate forecasts could be made for Europe. Unfortunately, the NAO is a noisy mid-latitude phenomenon and even the best predictions to date have not been able to capture more than 10% of its year-to-year variation.

2.1.2.2 Arctic Oscillation

The Arctic Oscillation (AO) is a large scale climate pattern characterised by winds circulating counterclockwise around the Arctic. When the AO is presented in its positive phase, a ring of winds circulating around the North Pole acts to confine colder air across polar regions. However, in the negative phase of the AO the belt of winds around the arctic becomes weaker and distorted, which allows for an easier southward entrance of colder, arctic air masses and increased storminess into the mid-latitudes 2.4.

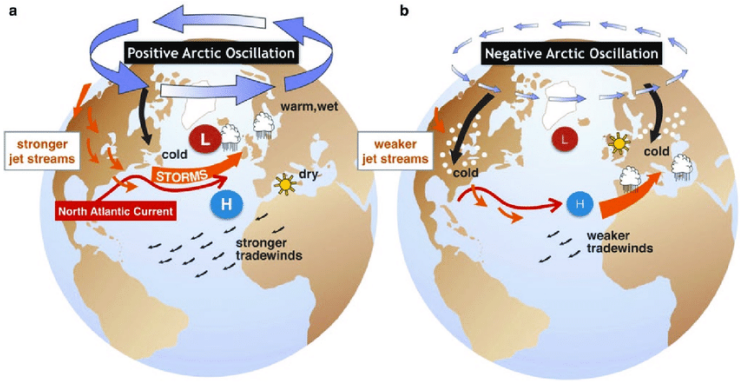


Figure 2.4: AO: Positive (left) and Negative (right) Phases, [2]

The daily AO index is constructed by projecting the daily 1000mb height anomalies pole ward of 20-90°N onto the loading pattern of the AO.

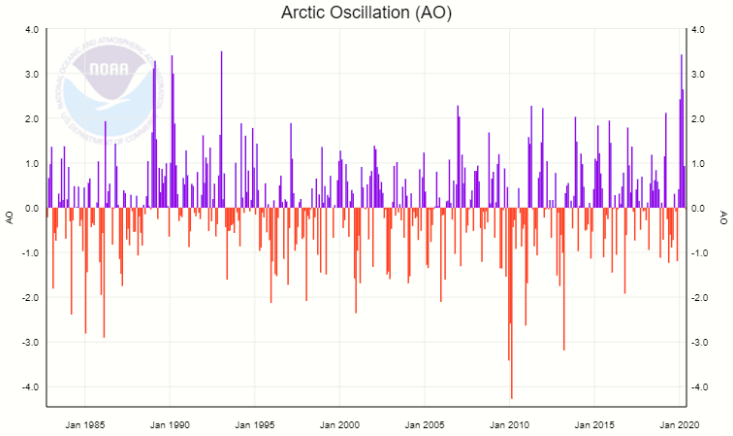


Figure 2.5: AO index, [3]

2.1.2.3 Pacific Decadal Oscillation

The Pacific Decadal Oscillation (PDO) refers to cyclical variations in sea surface temperatures in the Pacific Ocean. The PDO index is defined as the leading principal component of North

Pacific monthly sea surface temperature variability [24]. The PDO index consists of a warm and cool phase which alters upper level atmospheric winds. Shifts in the PDO phase can have significant implications for global climate, affecting Pacific and Atlantic hurricane activity, droughts and flooding around the Pacific basin, the productivity of marine ecosystems, and global land temperature patterns [25].

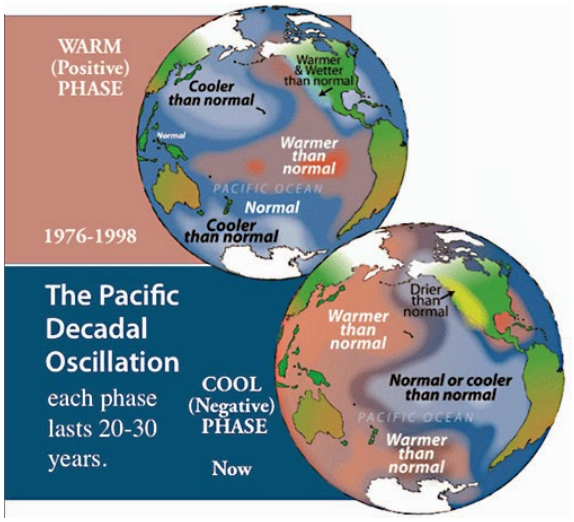


Figure 2.6: PDO, [4]

Global temperatures are tied directly to sea-surface temperatures. When sea-surface temperatures are cool, global climate cools. When sea-surface temperatures are warm, the global climate warms, regardless of any changes in atmospheric CO2. As such during PDO's cold mode, cool sea surface temperatures extend from the equator northward along the coast of North America into the Gulf of Alaska cooling global climate. However, during PDO's warm mode, warm sea surface temperatures extend from the equator northward along the coast of North America into the Gulf of Alaska warming global climate.

2.1.3 Weather Forecast

Weather forecasts are implemented in our society, they are accessible everywhere: on television, computers and even mobile phones. These furnish essential information for everyday life, planting and harvesting crop, selection of routes over land, sea and air, for building roads or infrastructure, for making preparations against impending natural hazards, and for much more.

Weather forecasting is characterised as the act of predicting future weather conditions or an attempt to indicate events that have a high probability of occurrence. It is an application of Science and Technology to predict the state of the atmosphere for a future time to a given location. These require observations of our environment around the clock and around the world. The bulk of those observations are carried out by National Meteorological Services as part of the WMO World

Weather Watch, which networks the observing stations to national, regional and global weather and climate prediction centres 24 hours a day in real-time [26].

2.1.3.1 Numerical Weather Prediction

One of the first attempts to forecast the weather using calculations, i.e. the first attempt to make a Numerical Weather Prediction, was by Lewis Fry Richardson in 1922. A variety of primitive equations were used to calculate, a 6-hour forecast for the state of the atmosphere over two points in central Europe. Unfortunately, the almost non-existent calculation of computational power did not enable an efficient estimation, having been developed again later on when more significant computational power was available [26].

Numerical Weather Prediction, or NWP, targets on taking current observations of the state of the atmosphere and processing these data with computer models to forecast future states. Knowing the current state of the atmosphere is just as important as the numerical computer models processing the data. Current states of the atmosphere serve as input to the numerical computer models through a process known as data assimilation to produce outputs of temperature, precipitation, and hundreds of other meteorological elements from the oceans to the top of the atmosphere.

Since the very first attempt, NWP has made advances due to more and better assimilated observations, higher computing power and progress in our knowledge of dynamics and physics of the atmosphere. Viable NWP systems provide an accurate indication of developing weather events from hours to days ahead. Hense, these are one of the most relevant components of routine and severe weather forecasting and warnings at National Meteorological and Hydrological Services [26].

2.1.4 Climate Predictions

Monitoring shorter-term climate conditions and predicting how climate will change in coming years is critical for sustainable development and is an important component of climate adaptation and climate services. **Climate prediction** is similar to NWP, but the forecasts are for longer periods. Climatic numerical models, global or regional, are used to alter trace atmospheric gases, sea ice and glacier cover, changes in incoming solar radiation, and a host of other parameters.

2.1.4.1 Global Climate Models

Global climate models, or GCM, are mathematical frameworks built on fundamental equations of physics organised using a three dimensional grid over the globe (2.7). They account for the conservation of energy, mass, momentum and how these are exchanged among different components of the climate system. Using these fundamental relationships, GCMs are able to simulate many important aspects of Earth's climate: large-scale patterns of temperature and precipitation, general characteristics of storm tracks and extratropical cyclones, observed changes in global mean temperature and ocean heat content as a result of human emissions [27].

Previously GCMs were designated as “general circulation models” due to including only the physics to simulate the general circulation of the atmosphere. Nowadays, global climate models simulate many aspects of the climate system: atmospheric chemistry and aerosols, land surface interactions including soil and vegetation, land and sea ice, and increasingly even an interactive carbon cycle and/or biogeochemistry [27].

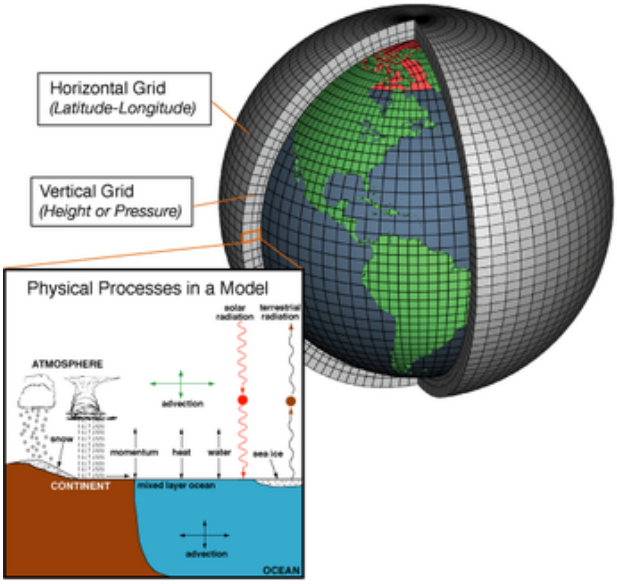


Figure 2.7: GCM, [5]

Thus the WMO designated centres to generate global seasonal forecasts as WMO Global Producing Centres of Long-Range Forecasts (2.8). These form an integral part of the WMO Global Data-Processing and Forecasting System.

Figure 2.8: WMO Global Producing Centres of Long-Range Forecasts, [6]

2.1.4.2 Regional Climate Models

Dynamical downscaling models are often referred to as regional climate models, or RCM, since they include many of the same physical processes that make up a GCM, but simulate these processes at higher spatial resolution over smaller regions (2.9).

At smaller spatial scales, and for specific variables and areas with complex terrain, such as coastlines or mountains, regional climate models have been shown to add value. As model resolution increases, RCMs are also able to explicitly resolve some processes that are parameterized in global models. However, despite the differences in resolution, RCMs are still subject to many of the same types of uncertainty as GCMs [27].

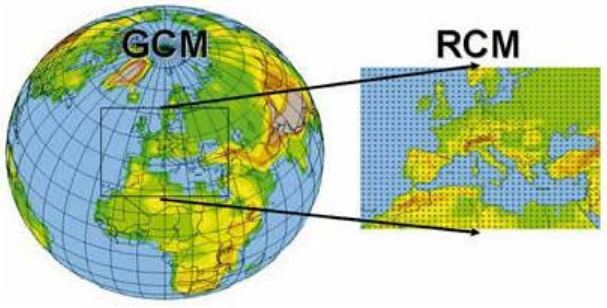


Figure 2.9: Regional Climate Models, [7]

The WMO also designated Regional Climate Centres to produce regional climate products, including long-range forecasts to support regional and national climate activities.

2.1.4.3 Empirical Statistical Downscaling Models

Empirical statistical downscaling models, or ESDM combine GCM output with historical observations to translate large-scale predictors or patterns into high-resolution projections at the scale of observations. The observations used in an ESDM can range from individual weather stations to gridded datasets. As output, ESDMs can generate a range of products, from large grids to analyses optimized for a specific location, variable, or decision-context [27].

ESDMs are limited by the fact that they require observational data as input; the longer and complete the record, the greater the confidence that the ESDM is being trained on a representative sample of climatic conditions for that location [27]. Statistical models are based on the key assumption that the relationship between large-scale weather systems and local climate or the spatial pattern of surface climate will remain stationary over the time horizon of the projections. This assumption may not hold if climate change alters local feedback processes that affect these relationships [27].

2.2 Summary

In the follow-up to this chapter some terms and designations in relation to weather, climate, meteorology were made explicit. The atmospheric variability and its correlation with teleconnection patterns were exposed. Concluding with a brief introduction to weather and weather forecasts. Since the research to be done will focus on climate prediction this chapter is essential for an initial contextualisation.

Chapter 3

Literature Review

3.1 Time Series Forecasting

The study to be carried out throughout this dissertation aims to perform a climatic forecast, which is then considered a time series forecast study.

A time series is a sequence S of historical measurements y_t of an observable variable y at equal time intervals. Time series are studied for several purposes such as the forecasting of the future based on knowledge of the past, the understanding of the phenomenon underlying the measures, or simply a succinct description of the salient features of the series. Forecasting future values of an observed time series plays an important role in nearly all fields of science and engineering, such as economics, finance, business intelligence, meteorology, climatology, telecommunication,power generation, medicine, water resources and environmental science [28, 29].

Weather/Climate forecasting is the application of science and technology to predict the state of the atmosphere for a given location. Weather forecasts are made by collecting quantitative data about the current state of the atmosphere and using scientific understanding of atmospheric processes to project how the atmosphere will evolve. The chaotic nature of the atmosphere, the massive computational power required to solve the equations that describe the atmosphere, error involved in measuring the initial conditions, and an incomplete understanding of atmospheric processes mean that forecasts become less accurate as the difference in current time and the time for which the forecast is being made increases [26].

Accurate prediction of rainfall is crucial for agriculture dependent countries like India, China, Australia, Pakistan, and Iran. Temperature forecasts are used by utility companies to estimate demand over coming days. On an everyday basis, people use weather forecasts to determine what to wear on a given day. Since outdoor activities are severely curtailed by heavy rain, snow and the wind chill, forecasts can be used to plan activities around these events, and to plan ahead [26, 30]. Climate variability leads to increasing risk of weather-related damages that impact virtually all sectors of the economy, from fisheries and agriculture to tourism, even insurance companies [31].

The increasing availability of large amounts of historical data and the need of performing accurate forecasting of future behaviour in several scientific and applied domains demands the

definition of robust and efficient techniques able to infer the stochastic dependency between past and future [28].

But with the discovery of non-linearity in the nature of weather data, the focus has shifted towards the nonlinear prediction of the weather data. Although, there is literature on nonlinear statistics for weather forecasting, most of them require that the nonlinear model be specified before the estimation is done [26].

Research on time series forecasts is widely present in the literature. Hence there are significant methods used to perform these forecasts and these techniques range from traditional and statistical methods to data-driven or Machine Learning (ML) methods. In order to carry out weather or climate forecasts, the methods pointed out in 2 are currently used as well as statistical methods and presently machine learning techniques, especially Artificial Neuronal Networks (ANN), have earned confidence from researchers.

3.2 Statistical Methods

For a long time the forecasting domain has been influenced by linear statistical methods. However it became increasingly clear that linear models are not adapted to many real applications, e.g. climate or weather forecasting which are of non-linear nature. Statistical methods do not generate acceptable results for non-linear processes because statistical methods are developed based on the assumption of linear time series. Therefore, statistical methods cannot clearly identify non-linear pattern and irregularities in weather/climate time series [28, 30].

The most commonly used statistical models for forecasting time series of a climatic nature are Auto Regressive (AR), Moving Average (MA), Auto Regressive Moving Average (ARMA), Auto Regressive Integrated Moving Average (ARIMA), and Multiple Regression. Even so, due to the limitations presented by these methods they are eventually used as reference models to evaluate the performance of machine learning models [32, 30]. Some of these techniques have been employed to predict hydrologic droughts and rainfall/precipitation [33, 32, 30].

Previously it was pointed out the existence of limitations in the statistical models. The AR models regresses against past values of the series. MA models uses past error as an explanatory variable. AR and MA both are suitable for developing models for univariate time series. The AR term only declares the number of linearly correlated lagged observations and is not appropriate for the data having nonlinear relationships. AR and MA can be combined together to form the ARMA model, however it can only be used for stationary time-series data. ARIMA model considers p, d, and q three variables where: p is the number of autoregressive terms, d the number of nonseasonal differences and q the number of lagged forecast errors in the prediction equation. As mentioned above statistical approaches lack the ability to identify nonlinear patterns and irregularity in the time series [32, 30].

With the discovery of non linearity in the nature of weather data, the focus of research has shifted towards nonlinear prediction of the weather data, i.e. researchers are focusing on conducting experiments with nonlinear models. Even though, the present literature has examples of nonlinear

statistical models for the weather forecasting, most of them require that the nonlinear model be specified before the estimation is done [34]. As such, the adoption of machine learning models has increased.

3.3 Machine Learning Methods

In the past two decades, machine learning models have drawn attention and have established themselves as serious contenders to classical statistical models in the forecasting community. These models, also called black-box or data-driven models, are examples of non-parametric nonlinear models which use only historical data to learn the stochastic dependency between the past and the future [28].

Several ML models/algorithms are employed on forecasting applications, and for this case study, it can be found in the literature methods, such as: Artificial Neuronal Networks, or ANN, specifically Deep Learning for long-range prediction of annual rainfall [35, 36, 32, 37, 30], precipitation nowcasting to predict the future rainfall intensity in a local region over a relatively short period of time [38], to predict water resources variables [29], to forecast the daily maximum temperature [34], to forecast daily streamflow [39]. It can also be found method e.g. K-nearest-neighbors, or K-NN, support vector regression, or SVR, Support Vector Machine, or SVM, classification and regression trees model, or CART, even hybrid models such as adaptive neuro-fuzzy inference system.

As expected when referring to models of ANN we are generalising several strands i.e. several variants of the model are present in the literature e.g. recurrent neural network (RNN) [40], convolutional Long-Short Memory (ConvLSTM) [38], modular artificial neural networks (MANN) [36], Convolutional Neural Network (CNN) [41], Multilayer Perceptron Networks (MLP), Elman Recurrent Neural Network (ERNN), Radial Basis Function Network (RBFN) and the Hopfield Model (HFM) [34], Bayesian neural network (BNN) [39].

In [38] it is formulated a precipitation nowcasting as a spatiotemporal sequence forecasting problem and it is proposed a new extension of LSTM designated ConvLSTM to tackle the problem. The ConvLSTM layer preserved the advantages of FC-LSTM (fully connected LSTM) but is also suitable for spatiotemporal data due to its inherent convolutional structure. By incorporating ConvLSTM into a encoding-forecasting structure, an end-to-end trainable model for precipitation nowcasting is built.

In [36] suggests the use of a modular artificial neural network (MANN) coupled with data-preprocessing techniques to improve rainfall predictions from India and China. In order to evaluate MANN's performance, three models, Linear Regression, K-NN and ANN, are used for the purpose of comparison. In the process of model development, model inputs and data-preprocessing techniques are carefully analysed, such methods as: linear correlation analysis (LCA) regarded as an effective and efficient input technique due to its simplicity of computation and comparable capability of forecasting; Singular Spectrum Analysis (SSA) is proved in improving model performance is to strengthen the mapping relation of model input and output by deleting noises in the raw signal.

For the case of the following study [33] it was investigated the ability of data driven models to forecast drought. It proposed and evaluated the use of the Wavelet Transform coupled models. Overall, coupled wavelet-neural network (WA-ANN) models were found to provide better results than the other model types used for the forecasts. Wavelet coupled models were proved to consistently present lower values of RMSE and MAE compared to the other data driven models. Wavelet analysis denoises the time series and subsequently allows the ANN and SVR model to model the main signal without the noise.

The study [42] states that PSO-SVM, Particle Swarm Optimization - Support Vector Machine) algorithm is proven to be an effective method of the rainfall forecast decision. It was established and compared with the traditional mesh optimisation, the Genetic algorithm and the Ant Colony algorithm it was proven through experiment results that the PSO algorithm has a higher accuracy and efficiency.

The following study [43] focuses on the application and evaluation of Classification and Regression Tree (CART) in prediction of seasonal precipitation. The accuracy of the CART model was compared with two commonly used models. The results revealed that the CART produced more accurate fall precipitation values than the other models, and this was also confirmed by spatial bias analysis. The results of the CART, in addition, demonstrated that the predictions accomplished better performance by two of the best climate indices in prediction of fall precipitation at time t by using the climate signals at $t-1$.

In the study [39] several ML models were employed to forecast streamflow at lead times of 1–7 days, models such as Bayesian neural network (BNN), support vector regression (with genetic algorithm used for selecting hyperparameters and kernels) (SVRGA), and Gaussian process (GP). The multiple local minima problem of BNN was alleviated by using the average forecast of an ensemble of BNN models. It was established that the nonlinear models generally outperformed multiple linear regression (MLR), and BNN tended to slightly outperform the other nonlinear models.

The porpuse of the study [44] was to assess whether it is possible to use a simplified reality - in this case the most simple GCM without seasonal cycle - to develop a method that also works on more complex GCMs. We showed that, for the problem of forecasting the model 'weather', this seems to be the case. It was used a deep convolutional auto-encoder architecture from [41]

Chapter 4

Methodology

The chapter will focus on the material and process developed for this thesis. The "Material" section will incorporate the development tools and the data used throughout this project. The "Method" section will narrate and justify the work procedure. The chapter in question is meant to explain the pre-processing methods exposed to the data, the models explored and the way results were acquired and organised.

4.1 Material

The purpose of the following sections will be to exhibit to the reader the material utilised in this project. First and foremost the data adopted throughout the course of this dissertation is going to be presented: GSOD and Teleconnections. Finally, the development platforms explored for the development, analysis and evaluation of the work will be referenced.

4.1.1 Data

The National Centers for Environmental Information (NCEI) was the result of the union of the former information centres belonging to the National Oceanic and Atmospheric Administration (NOAA) — The National Climatic Data Center (NCDC), the National Geophysical Data Center (NGDC) and the National Oceanographic Data Center (NODC) which includes the National Coastal Data Development Center (NCDDC).

NCEI is responsible for hosting and providing access to one of the most significant archives on Earth, with comprehensive oceanic, atmospheric, and geophysical data. Data quality is indisputable as NOAA issued an Information Quality Guidelines to ensure and maximise quality, objectivity, utility and integrity of information which it disseminates, withal, the acquired data is based on data exchanged under the World Meteorological Organization (WMO) World Weather Watch Program according to WMO Resolution 40 (Cg-XII) [45]. NOAA is also associated with one of the six World Data Centres of the WMO Global Atmosphere Watch (GAW) responsible for documenting and archiving atmospheric measurements and associated metadata from measurement stations worldwide and making these data freely available to the scientific community. [46]

All the information handled throughout this project was obtained via FTP, File Transfer Protocol, connection from the NCDC archive of observational data. Although the data is made out of historical observations dating from the 1970s to the present day, it does not prevent it from being exposed to revisions and possible corrections. Therefore taking into consideration the date on which the data was extracted it may have been subjected to modifications. Even if, hypothetically, the data has been altered after the data extraction it does not devalue the work done.

4.1.1.1 GSOD and Teleconnection data

Global Surface Summary of the Day, or GSOD, is derived from the Integrated Surface Hourly (ISH) dataset. The files available online date back almost to the beginning of the 20th century and continue to be updated and reviewed by over 9000 worldwide observational stations. The daily elements included in the dataset are: Mean temperature (.1 Fahrenheit), Mean dew point (.1 Fahrenheit) Mean sea level pressure (.1 mb), Mean station pressure (.1 mb), Mean visibility (.1 miles), Mean wind speed (.1 knots) Maximum sustained wind speed, (.1 knots), Maximum wind gust (.1 knots), Maximum temperature (.1 Fahrenheit), Minimum temperature (.1 Fahrenheit), Precipitation amount (.01 inches), Snow depth (.1 inches) and an Indicator for occurrence of: Fog, Rain or Drizzle, Snow or Ice Pellets, Hail, Thunder, Tornado/Funnel Cloud [47], [48].

Since it was not applicable to make a climate prediction model with all the available stations data, it was decided to transfer data from a single station. Such a facility is located at the coordinates 41º14'52.8"N, 8º40'51.6"W which locates the post responsible for capturing and storing data for the region of OPorto, Portugal (Air Force station ID (USAF) — 085450). The station in question became operational in the early 1930s and continues to store data, however, the available GSOD data dates from 1973 to the present day.

So the GSOD data acquired for the project came from the USAF — 085450 station and dates from 1973 until the day the data was obtained.

For this project Teleconnections data was used as well as GSOD. Teleconnections are spatially and temporally large-scale anomalies that influence the variability of the atmospheric circulation. The anomalies considered for this thesis were the following: Arctic Oscillation (AO) [49], North Atlantic Oscillation (NAO) [50], Pacific Decadal Oscillation (PDO) [51]. This type of information is stored on a monthly basis so each dataset contains monthly values of the anomaly as its variation is negligible on a daily basis. The dataset provided proves that these deviations have been stored since the mid-19th century. PDO dataset dates from 1854 to the day the dataset was obtained, and AO and NAO dataset dates from 1950's.

Lastly, the Northern Hemisphere Sea Ice Extent, ICE cover, was added to the assemble of information [52]. The dataset dates from 1979 to the time it was obtained.

4.1.2 Tensorflow

The development of this project would not be possible without the adoption of the end-to-end open source platform for machine learning — Tensorflow, i.e., an interface for expressing machine

learning algorithms and an implementation for executing such algorithms. The system is flexible and can be used to express a wide variety of algorithms, including training and inference algorithms for deep neural network models, and it has been used for conducting research and for deploying machine learning systems into production across more than a dozen areas of computer science and other fields [53].

The preference for this platform over others, i.g. Pytorch, is because of its comprehensive and flexible ecosystem of tools and libraries dedicated to model ML models, such as Google Colaboratory (or "Colab"), Tensorboard and Keras [54].

The practical part of this dissertation was completely developed with the programming language Python and for the management of the immense amount of data several modules such as Pandas [55], NumPy [56] and Scikit-learn [57] were used.

4.2 Method

In the course of this section, it will become clear the treatment to which the data was subjected, the Deep Learning algorithm/models tested, the procedure for the acquisition of results and how they were subsequently organised and analysed, i.e. how the tests were designed and how the results were evaluated.

Finally, it is important to clarify that throughout this experiment, the variables to be predicted are the following: The monthly mean values of maximum and daily mean temperatures (°C) and the monthly total precipitation (*mm*)

4.2.1 Forecasting Model: Time Series

Firstly, the problem, which is a time series problem, has to be defined, i.e. the variables to be forecasted have to be established and how the forecasts will be performed has to be analysed.

The variables to be forecasted, as above-mentioned (4.2), are the monthly mean values of maximum and daily mean temperatures (°C) and the monthly total precipitation (*mm*).

Time series forecasting involves developing and using a predictive model on data where there is an ordered relationship between observations. Before the development of the project, it is essential to enhance the understanding of the structure of the forecast problem, the structure of the model required, and how to evaluate it. Consequently, certain concepts should be established [16].

4.2.1.1 Inputs and Outputs

Generally, a prediction problem involves using past observations to predict or forecast one or more possible future observations. The objective is to guess about what might happen in the future. When a forecast is required, it is critical to think about the data that will be available to make the forecast and what is the result of the forecast.

- **Input**: Historical data provided to the model in order to make prediction.

- **Output**: Prediction for a future time step beyond the data provided as input.

Since the variables to be forecast are of monthly order what will be represented by input will be the observations of past months and the output the forecast of months to pass.

4.2.1.2 Endogenous and Exogenous

The input data can be subdivided in order to understand its relationship to the output variable. An input variable is endogenous if it is affected by other variables in the system and the output variable depends on it. An input variable is an exogenous variable if it is independent of other variables in the system and the output variable depends upon it. Put simply, endogenous variables are influenced by other variables in the system (including themselves) whereas exogenous variables are independent are considered as outside of the system.

Commonly, a time series forecasting problem has endogenous variables (e.g. the output is a function of some number of prior time steps) and may or may not have exogenous variables. Often, exogenous variables are ignored given the strong focus on the time series [16].

Given the type of time series under study and the features to be forecast, it is clear that most of the data is endogenous, even if exogenous variables are also present.

4.2.1.3 Unstructured and Structured

A series with no pattern might be considered as unstructured, e.i. there is no discernible time-dependent structure. Alternately, a time series may have patterns, e.g. trend or seasonal cycles and be regarded as structured. The modeling process can be simplified by identifying and removing the obvious structures from the data, such as an increasing trend or seasonality [16].

As it is anticipated, being the time series related to observations of meteorological elements some climatic patterns could be identified, such as presented in the figure 4.1.

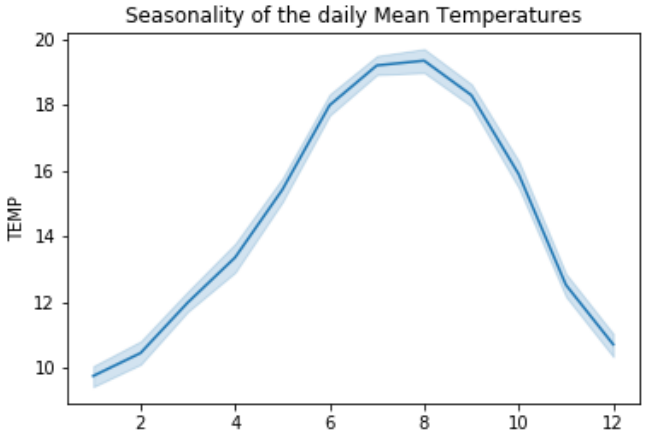


Figure 4.1: Seasonality example

4.2.1.4 Regression and Classification

A time series forecasting problem in which the outcome is to predict one or more future numerical values is a regression type predictive modelling problem. A time series forecasting problem in which the objective is to classify input time series data is a classification type predictive modelling problem.

Concluding that in this case the problem will be of regression.

4.2.1.5 Static and Dynamic

To develop a model and use it frequently to make predictions. Given that the model is not updated or changed between forecasts, it is defined as a static forecasting model. Conversely, new observations may be received prior to making a subsequent forecast that could be used to create a new model or update an existing one. Developing a new or updated model prior to each forecasts is defined as a dynamic model problem [16].

For this project, since deep learning techniques will be employed, the problem will be faced dynamically.

4.2.1.6 Univariate and Multivariate

For univariate data, each sample x_i is described by only one feature. A set with n samples can be represented by $x^j = \{x_1, x_2, \ldots, x_n\}$, where each x_i represents a single value [58]. An univariate time series implies that a single variable/feature is measured over time [16]. Conversely, multivariate data consists of data that has more than one input feature [58], i.e., multiple variables are measured over time [16].

However, when establishing a time series forecast model the number of variables may differ between the inputs and outputs, e.g. the data may not be symmetrical. For example, a model's input may be defined by multiple features, even if its purpose is to predict only one of the variables as output.

- **Univariate and Multivariate Inputs**: One or multiple input variables measured over time.
- **Univariate and Multivariate Outputs**: One or multiple output variables to be predicted.

For this project uni and multivariate models were used to forecast only one variable.

4.2.1.7 Single-step and Multi-step

A forecast problem which requires a prediction of the next time step is termed a one-step forecast model. Whereas a forecast problem that requires a prediction of more than one time step is called a multi-step forecast model.

In the case of this study, the temporal instances to be foreseen are monthly. Tests were carried out with one and more time steps, tests were made in order to provide the value of the following month, the value of the following 6 months or even the values of a whole year.

4.2.1.8 Problem Definition: Conclusion

In conclusion, this section, 4.2.1, served to clarify certain concepts about the dataset that has been studied and certain designations for the models that have been elaborated. It has been stated that the data set is of an endogenous nature (4.2.1.2) and has an organised structure (4.2.1.3). The model was established to be developed for a regression problem (4.2.1.4) and dynamic nature (4.2.1.5). Ultimately, it was stipulated that for the tests the input and output data would be months (4.2.1.1), the models developed would be able to execute tests with uni/multivariate datasets (4.2.1.6) and it would be possible to provide data with one or more time steps (4.2.1.7).

4.2.2 Data Preparation Process

The performance of the application of ML algorithms is linked to the dataset provided, i.e. their quality and state affects the performance of the models. Therefore the datasets (GSOD, Teleconnections, Ice Cover) originally obtained were subjected to cleaning and pre-processing techniques.

Datasets are formed by objects or samples that can represent a physical object, or an abstract notion, however, in the case of GSOD, each object is one day. The attributes or features are the characteristics in which the objects are represented, which, in the case of GSOD, are surface meteorological elements [58].

4.2.2.1 Data Pre-Processing: Formatting, Cleaning and Sampling

The presence of imperfections in the data can result in incorrect statistics and analysis, or even reduce the performance quality of the models. Frequent deficiencies include noisy data (that has errors or different values than expected), inconsistent data, redundant and incomplete data. This type of errors can be caused by problems in the equipment for collecting, transmitting and storing the information, or human error [58].

Considering the GSOD dataset according to the descriptive file (README. file) all samples with attributes presented with the value 9999.9, 999.9 and 99.99 are considered missing values and have therefore been replaced by NaN values.

Since the values presented are in correlation with the imperial system, for a better scientific and global understanding the same have been converted to the metric system. The mean temperature (TEMP), mean dew point (DEWP), minimum temperature (MIN) and maximum temperature (MAX) reported during the day were changed from Fahrenheit, °F, to Celsius, °C, with the following calculation (4.1). The mean visibility (VISIB) for the day presented in miles, *mi*, was altered to kilometers, *km*, as illustrated in (4.2). The mean wind speed (WDSP), maximum sustained wind speed (MXSPD) and maximum wind gust (GUST) reported for the day depicted in knots, *kn*, in which $1kn$ is equal to one nautical mile per hour, was converted to kilometer per hour, $km\backslash h$, following the equation (4.4). Finally, the total precipitation (PRCP) and snow depth (SNDP) reported during the day provided in inches, *in*, was turned to millimeters, *mm*, illustrated in (4.3). The mean sea level pressure (SLP) and the mean station pressure (STD) for the day were not altered and continue with the its original units millibars, *mbar*. ptm

$$^{\circ}C = \left(^{\circ}F - 32\right) \times \frac{5}{9} \tag{4.1}$$

$$1mi = 1,609344km \mapsto d_{km} = d_{mi} \times 1,609344 \tag{4.2}$$

$$1in = 25,4mm \mapsto d_{mm} = d_{in} \times 25,4 \tag{4.3}$$

$$1kn = 1,8520km\backslash h \mapsto km\backslash h = kn \times 1,8520 \tag{4.4}$$

A piece of information is redundant when it is very similar to another of the same dataset, i.e. its attributes have values very similar to the attributes of at least another element. The redundancy of an attribute is related to the correlation with one or more of the attributes in the same dataset. The more correlated the attributes, the greater the degree of redundancy [58]. Since the elimination of redundancies is desirable, a routine has been performed in the dataset which returns the attributes with the highest correlation. The result proves a significant degree of redundancy between SLP and STP attributes. The STP attribute has been deleted as it contains a higher number of incomplete/missing values compared to SLP.

4.2.2.2 Outlier detection and disposal

An outlier is an object that deviates significantly from other objects, as if it were generated by a different mechanism [59] the same are also recognised as aberrant, abnormal or extreme values. Therefore they are points that must be identified and removed. Depending on the nature of the outlier they are classified into [60]:

- **Punctual Outliers**: an observation that deviates from other observations and may be caused by an abnormal measurement error, behaviour or characteristic of the object.
- **Contextual Outliers**: Sometimes abnormal values are not obvious due to the context in which they appear.
- **Collective Outliers**: can also be a sequence of values.

A Boxplot, figure 4.2, also called Box and Whisker diagram, presents a summary of the 1°, 2° (median) and 3° quartiles values, besides the lower and upper limits. The quartiles are one of the many measures to evaluate the distribution of data, they divide the ordered values into quarters. Thus the 1° quartile of a sequence $Q1$ is the value for which 25% of the other values are below it. The use of this type of diagrams facilitates the analysis of data distribution and is a useful tool for the detection of outliers.

The method applied from attribute to attribute for the detection of outliers was precisely the Boxplot technique [61]. This technique is based on the first ($Q1$), third quartile ($Q3$) and the

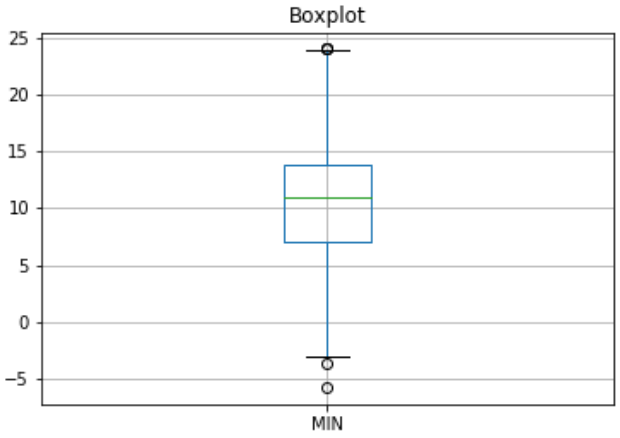


Figure 4.2: Boxplot Example - MIN attribute

interquartile range ($IQR = Q3 - Q1$) of data, it determines that the interval $\left[Q1 - 1,5 * IQR, Q3 + 1,5 * IQR\right]$ contains about $99,3\%$ of the data. Therefore, points outside this range are considered as Moderate Outliers, and points outside the range $\left[Q1 - 3 * IQR, Q3 + 3 * IQR\right]$ are considered Extreme Outliers. Outliers considered extreme were excluded from the datasets.

4.2.2.3 Dimension Reduction

A dataset is considered 'large' either because it contains a high number of objects/samples, or because each object is described by a high number of attributes/features. In general, the performance of a learning algorithm improves with increasing numbers of samples, and decreases with increasing numbers of features. The effect of the very high number of features in algorithms is described by the "Dimensionality Curse" problem.

One way to minimise the impact of the dimensionality problem is to combine, or eliminate, some of the irrelevant attributes. The dataset with the highest number of features is GSOD and, throughout this project, several were considered irrelevant and redundant. We have the case of the STP removed for presenting a high degree of redundancy with the SLP; the feature of indicators which reports an occurrence during the day of (Fog, Rain or Drizzle, Snow or Ice Pellets, Hail, Thunder, Tornado or Funnel Cloud (FRSHTT)) and the Snow depth (SNDP) were considered irrelevant.

4.2.2.4 Feature Engineering

The primary purpose of this project is the elaboration of a computational application in order to make time forecasts of climatic elements using deep learning techniques. These methods achieve better results when they have access to a significant quantity and quality of data.

However, considering the variables to be predicted it is possible and certain that some attributes of the datasets could be irrelevant and may even impair the performance of the models. Perhaps it

would be beneficial to reconstruct datasets for the prediction of specific variables in order to have as much information as possible without suffering a decrease in performance due to the dimensionality problem.

As the features to be foreseen are the monthly mean values of maximum and daily mean temperatures (°C) and the monthly total precipitation (*mm*), at least three datasets were originated to supply the models with. As new datasets were recreated with GSOD and teleconnection features, additional features were developed. These new features are more abstract and based on those already obtained, these are: the number of days per month that had a total precipitation greater than or equal to one, the number corresponding to the month in which a certain observation was made and the season of the year.

	PRCP	1mm	month	season	NAO	AO	PDO
1973-01-31	113.538	10	1	1	-0.46	1.2318	-0.22
1973-02-28	68.8340	9	2	1	0.52	0.7862	-0.59
1973-03-31	76.1200	8	3	1	-0.09	0.53717	-0.89
1973-04-30	27.432	7	4	2	-0.73	-1.1257	-1.4
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

Table 4.1: Example: Dataset to predict Total Precipitation

	MAX	month	season	ICE
1979-01-31	13.839	1	1	15.41
1979-02-28	14.036	2	1	16.18
1979-03-31	13.839	3	1	16.34
1979-04-30	16.567	4	2	15.45
⋮	⋮	⋮	⋮	⋮

Table 4.2: Example: Dataset to predict the Maximum Monthly Temperature

	TEMP	month	season
1974-01-31	8.330	1	1
1974-02-28	9.040	2	1
1974-03-31	11.065	3	1
1974-04-30	12.948	4	2
⋮	⋮	⋮	⋮

Table 4.3: Example: Dataset to predict the Monthly Daily Mean Temperature

The aforementioned examples of the datasets used for the monthly total precipitation forecast (4.1), the maximum monthly temperature forecast (4.2) and the monthly daily mean temperature forecast (4.3)

4.2.3 Deep Learning Models

Under this section it is made explicit which deep learning models have been implemented and tested throughout this thesis.

Modern deep learning provides a powerful framework for supervised learning. By increasing the number of layers and units within a layer, a deep network can produce functions of increasing complexity depending on the data [14]. Deep learning algorithms seek to exploit the unknown structure in the input distribution in order to discover good representations, often at multiple levels, with higher-level learned features defined in terms of lower-level features [62].

In the course of the project, the following models were repeatedly tested:

1. **Multi-channel CNN**.
2. **Vanilla LSTM**.
3. **Encoder-Decoder LSTM**.
4. **CNN-LSTM Encoder-Decoder**.
5. **ConvLSTM Encoder-Decoder**.

4.2.3.1 Multi-channel CNN

Convolutional neural networks, or CNNs, are a specialised kind of neural network for processing data that has a known grid-like topology. It can handle time-series data, which can be thought of as a 1-D grid taking samples at regular time intervals. The designation "convolutional neural network" implies that the network employs a mathematical operation named convolution. CNNs are simply neural networks that use convolution in place of general matrix multiplication in at least one of their layers [14].

The convolutional operation is as stated in 4.5, being x the input, w the kernel and the result defined as feature map

$$s(t) - (x * w)(t) \quad \Rightarrow \quad s(t) - \int x(a) \cdot w(t-a)\,da \tag{4.5}$$

However, regardless of the mathematical of this operation, what motivates the use of it is the following ideas: sparse interactions,parameter sharing and equivariant representations. Moreover, convolution provides a means for working with inputs of variable size [62].

CNNs can be used for uni/multivariate and uni/multi-step time series forecasting because it supports multiple 1D inputs, i.e., it is possible to develop a Multi-Channel model where each input sequence is read as a separate channel [16]. The Multi-Channel CNN will utilise a separate kernel and read each input sequence onto a separate set of filter maps, learning features from each input time series variable.

4.2.3.2 Vanilla LSTM

Recurrent neural networks (RNNs), are a group of neural networks for processing sequential data. A RNN is a neural network specialised on processing a sequence of values $x^1, \ldots, x^\tau$. Such as convolutional networks can readily scale to images with large width and height, and some can process images of variable size, recurrent networks can scale to much longer sequences than would be practical for networks without sequence-based specialisation. RNNs can also process sequences of variable length. RNNs takes advantage of one of the ideas found in machine learning and statistical models: sharing parameters across different parts of a model. Parameter sharing makes it possible to extend and apply the model to examples of different forms, such as in this case different lengths, and generalised across them [14].

The computation in most RNNs can be decomposed under three blocks of parameters and associated transformations: From the input to the hidden state, the previous hidden state to the next hidden state, and to the hidden state to the output (4.3).

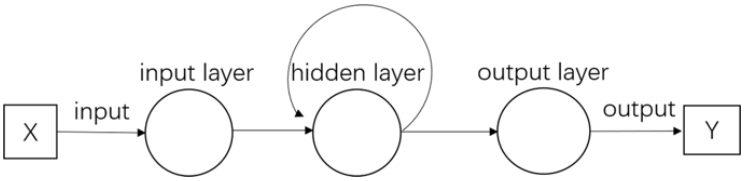


Figure 4.3: RNN Architecture - Example, [8]

In a recurrent neural network, throughout the gradient back-propagation phase, the gradient signal can end up being multiplied by large number of times (as many as the number of time steps) by the weight matrix associated with the connections between the neurons of the recurrent hidden layer. Therefore, it means that the magnitude of weights in the transition matrix can have a strong impact on the learning process [63]. It poses a challenge to learning long-term dependencies in recurrent networks [14].

If the weights in the transition matrix are small (or if the leading eigenvalue of the weight matrix is smaller than 1.0), it can lead to a situation called *vanishing gradients* where the gradient signal gets so insignificant that learning either becomes very slow or stops working altogether. Making it can also more difficult the task of learning long-term dependencies in the data. Conversely, if the weights the matrix are large (or if the leading eigenvalue of the weight matrix is larger than 1.0), it can lead to a situation where the gradient signal is so large that it can cause learning to diverge, referred to as *exploding gradients* [63].

The aforementioned problems are the main motivation behind the gated RNNs models, e.g. Long short-term memory, LSTM, model. The LSTM introduces a new structure cell termed memory cell (4.4). A memory cell is composed of four main elements: an input gate, a neuron with a self-recurrent connection, a forget gate and an output gate. The self-recurrent connection has a weight of 1.0 and ensures that, barring any outside interference, the state of a memory cell can remain constant from one time step to another. The gates serve to modulate the interactions

between the memory cell itself and its environment. The input gate can allow incoming signal to alter the state of the memory cell or block it. On the other hand, the output gate can allow the state of the memory cell to have an effect on other neurons or prevent it. Finally, the forget gate can modulate the memory cell's self-recurrent connection, allowing the cell to remember or forget its previous state, as needed [63].

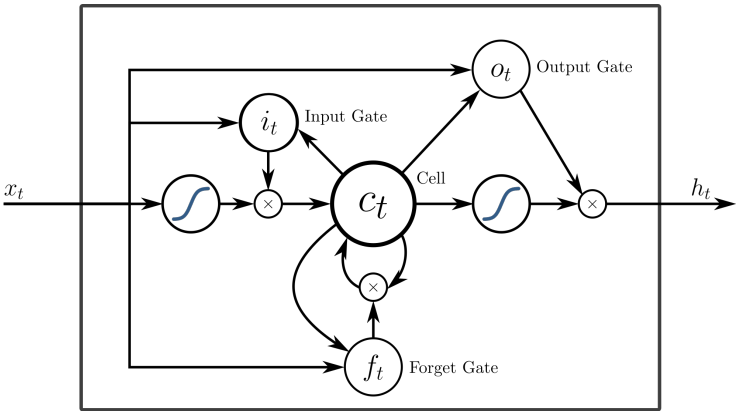


Figure 4.4: LSTM memory cell - Example

Long Short-Term Memory networks (LSTMs) can be applied to time series forecasting. There are many kinds of LSTM models that could be utilised for each specific type of time series forecasting problem. A Vanilla/simple LSTM is an LSTM model that has a single hidden layer of LSTM units, and an output layer used to make a prediction. Key to LSTMs is that it supports sequences. Unlike a CNN that reads across the entire input vector, the LSTM model reads one time step of the sequence at a time and builds up an internal state representation that can be used as a learned context for making a prediction [16].

4.2.3.3 Encoder-Decoder LSTM

A RNN can be trained to map an input sequence to an output sequence which is not necessarily of the same length. This comes up in many applications, such as speech recognition, machine translation and question answering, where the input and output sequences in the training set are generally not of the same length (although their lengths might be related).

The presented model is an allegedly update of the vanilla LSTM 4.5. The model means that the output will not be a vector sequence directly. Alternately, the model will be comprised of two sub models, the encoder to read and encode the input sequence, and the decoder that will read the encoded input sequence and make a one-step prediction for each element in the output sequence. Additionally a LSTM model is used in the decoder, allowing it to both know what was predicted for the prior day in the sequence and accumulate internal state while outputting the sequence. For multivariate model forecasting it will be provided each one-dimensional time series to the model as a separate sequence of input. The LSTM will in turn create an internal representation of each input sequence that will together be interpreted by the decoder [16].

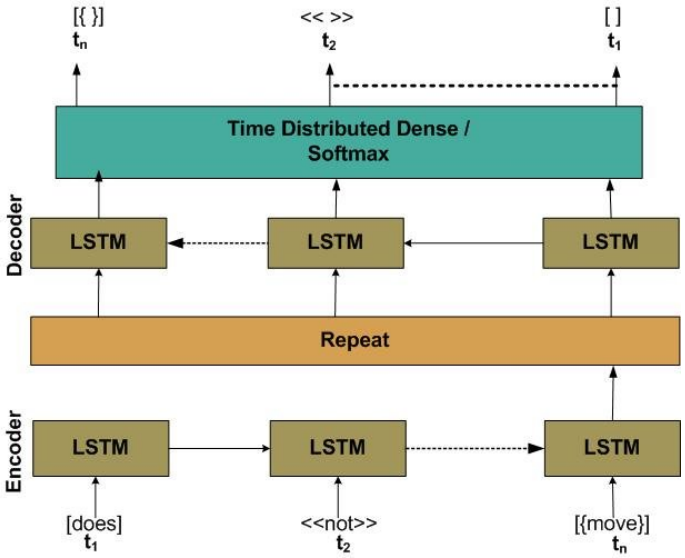


Figure 4.5: Encoder-Decoder LSTM network - Example [9]

4.2.3.4 CNN-LSTM Encoder-Decoder

CNN can be very effective at extracting and learning features from one-dimensional sequence data e.g. time series data 4.6. A CNN model can be used in a hybrid model with an LSTM backend where the CNN is capable of automatically understand the sequence input and learn its salient features, while these can then be interpreted by an LSTM decoder. This hybrid model is defined as CNN-LSTM Encoder-Decoder [16].

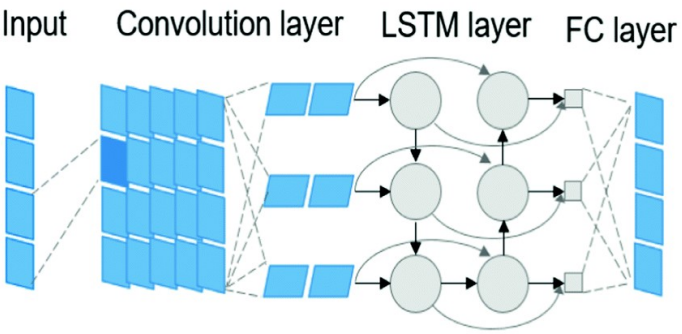


Figure 4.6: CNN-LSTM Encoder-Decoder network - Example [10]

4.2.3.5 ConvLSTM Encoder-Decoder

A extension of the CNN-LSTM approach is to perform the convolutions of the CNN as part of the LSTM for each time step. This combination is called a Convolutional LSTM (ConvLSTM, 4.7) and, such as the CNN-LSTM, it is also used for spatiotemporal data. Dissimilar to previous

models the ConvLSTM is using convolutions directly as part of reading input into the LSTM units themselves [16].

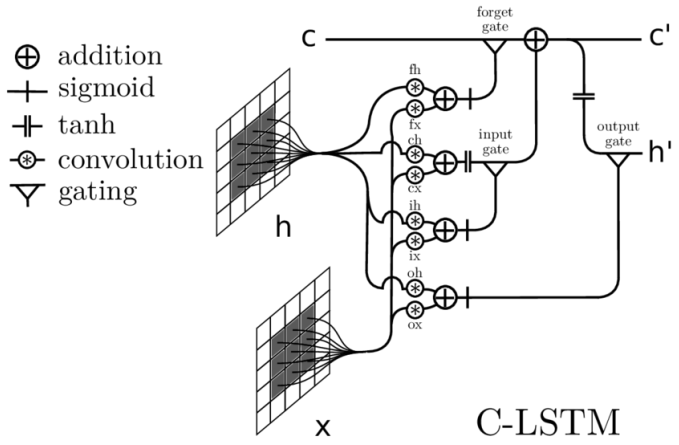


Figure 4.7: ConvLSTM unit - Example [11]

4.2.3.6 Common Considerations

Ultimately, the following clarifies and justifies the common implementations imposed on the above architectures, such as the loss function, the optimisation algorithm, the metrics chosen to examine the performance of the algorithms, the activation function for each unit and even the batch/epoch number. However, it is imperative to state that the choice of certain parameters justifies the choice of others.

↪ **Gradient Descent Optimisation algorithm and the number/size of epoch/batch**

Gradient descent is a neural network optimisation algorithm. Without exception every state-of-the-art Deep Learning library contains implementations of algorithms to optimise gradient descent (e.g. keras). Gradient descent is a form of minimising an objective function $J(\theta)$ parameterized by a model's parameters $\theta \in R^d$ by updating the parameters in the opposite direction of the gradient of the objective function $\nabla_\theta J(\theta)$. The learning rate η determines the size of the steps we take to reach a local minimum [64].

Three variations of gradient descent exist, it differ in how much data it is used to compute the gradient of the objective function. Depending on the amount of data, a trade-off is made between the accuracy of the parameter update and the time it takes to perform an update [64].

Prior to describing each variant and discussing the advantages and disadvantages the definition of sample, epoch and batch is made explicit as it has a direct correlation with the variants, further explained.

A **sample** is considered as a single row of data in a dataset. The dataset includes inputs fed into the algorithm and an output compared to the prediction to calculate an error, in order for the algorithm to evaluate itself.

The **batch size**, or batch, is a hyper-parameter that defines the number of samples to work through before updating the internal model parameters.

Finally, the number of **epochs**, or simply epoch, is a hyper-parameter which defines the number of times that a certain algorithm will work through the entire training dataset. One epoch signifies that each sample in the training dataset has had an opportunity to update the internal model parameters. An epoch is comprised of one or more batches.

The number of epochs has no limit, as long as it is represented by an integer, however, the batch size will have to vary between one and the maximum number (including) of samples present in the training dataset.

Subsequently, the three variants are presented.

1. **Batch Gradient Descent** $\Rightarrow$ $Batch\ Size = Size\ of\ Training\ Dataset$

   Computes the gradient of the cost function w.r.t. to the parameters θ for the entire training dataset, as illustrated in (4.6).

   $$\theta = \theta - \eta \cdot \nabla_\theta J(\theta) \tag{4.6}$$

   As it is necessary to calculate the gradient for the whole dataset in order to execute just a single update, batch gradient descent can be slow and is intractable for datasets that won't fit in memory. However, it is guaranteed to converge to the global minimum for convex error surfaces and to a local minimum for non-convex surfaces [64].

2. **Stochastic Gradient Descent, SGD** $\Rightarrow$ $Batch\ Size = one$

   Performs a parameter update for each training example x^i and label y^i, referenced on (4.7):

   $$\theta = \theta - \eta \cdot \nabla_\theta J(\theta; x^i; y^i) \tag{4.7}$$

   Conversely to the redundant computations Batch Gradient Descent performs, as it recomputes gradients for similar examples before each parameter update. SGD clears the redundancy by performing one update at a time, therefore it is faster. SGD's fluctuation enables it to jump to new and potentially better local minima, or, however, it could hinder convergence to the exact minimum, as SGD could keep overshooting [64].

3. **Mini-Batch Gradient Descent** $\Rightarrow$ $one < Batch\ Size < Size\ of\ Training\ Dataset$

   Performs an update for every mini-batch of n training examples, (4.8).

   $$\theta = \theta - \eta \cdot \nabla_\theta J(\theta; x^{(i:i+n)}; y^{(i:i+n)}) \tag{4.8}$$

   The presented variant reduces the variance of the parameter updates, which can lead to more stable convergence and can make use of highly optimised matrix optimisations common to

state-of-the-art deep learning libraries that make computing the gradient w.r.t. a mini-batch very efficient. However, does not guarantee good convergence [64].

Throughout the implementation of the various NN models, a Mini-Batch strategy was applied in which, preferably, the number of samples for each batch is the same.

To optimise the gradient several challenges must be analysed, such as, choosing a proper learning rate, dragging out learning rate schedules in order to adjust the learning rate during training by reducing the η according to a predefined schedule or when the change in objective between epochs falls below a threshold (problem being the schedule or threshold predefined), avoid getting trapped in the numerous sub-optimal local minima.

The Gradient Descent optimisation algorithm chosen was, the default, Adaptive Moment Estimation (Adam). It computes adaptive learning rates for each parameter, stores an exponentially decaying average of past squared gradients v_t and keeps an exponentially decaying average of past gradients m_t. The decaying averages of past and past squared gradients m_t (4.9) and v_t (4.10), are, respectively, computed:

$$m_t = \beta_1 \cdot m_{t-1} + (1-\beta_1) \cdot g_t \tag{4.9}$$

$$v_t = \beta_2 \cdot v_{t-1} + (1-\beta_2) \cdot g_t^2 \tag{4.10}$$

m_t and v_t are estimates of the first moment (mean) and the second moment (uncentered variance) of the gradients respectively. Since both m_t and v_t are initialised as zero, the Adam algorithm is biased towards zero, such a phenomenon was indicated by the authors of the algorithm. In order to correct the bias problem it was computed a bias-corrected first, (4.11) and second moment estimates, (4.12).

$$\hat{m}_t = \frac{m_t}{1-\beta_1^t} \tag{4.11}$$

$$\hat{v}_t = \frac{v_t}{1-\beta_2^t} \tag{4.12}$$

Finally the Adam update rule goes as follows (4.13).

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{\hat{v}_t} + \varepsilon} \cdot \hat{m}_t \tag{4.13}$$

Being the default values of β_1, β_2, η and ε specified in the tensorflow documentation.

↪ Loss Function

The loss function (cost function) is a crucial ingredient in all optimising problems, such as forecasting [65]. In regression cases, the error of the hypothesis $\hat{f}$ can be calculated by the distance between the known value, y_i, and the value predicted by the model, $\hat{f}(x_i)$ [58]. The loss function employed is the Mean Squared Error ("MSE", 4.14):

$$MSE(\hat{f}) = \frac{1}{n} \cdot \sum_{i=1}^{n} \left(y_i - \hat{f}(x_i)\right)^2 \tag{4.14}$$

Squaring the forecast error forces it to be positive and has the effect of putting more weight on large errors. Very large or outlier errors drag the mean of the squared forecast errors out resulting in a larger mean squared error score. Essentially, this loss function penalises the performance of models that make large and incorrect forecasts.

↪ Activation Function

A NN is composed of layers of nodes and learns to map examples of inputs to outputs. For a given node, the inputs are multiplied by the weights in a node and summed. This value is referred to as the summed activation of the node. The summed activation is then transformed via an **activation function** and defines the specific output or "activation" of the node, 4.15).

$$Y = Activation\ \ Function\left(\sum \left(weights \cdot input + bias\right)\right) \tag{4.15}$$

Essentially the activation function controls the activation and deactivation of each unit outputting a value dependent on the input and function, it could perform linear/simple transformations or nonlinear transformations depending on the appropriate function[66].

The activation function adopted for each unit of the developed models was the rectified linear activation unit, or "ReLU" (4.8).

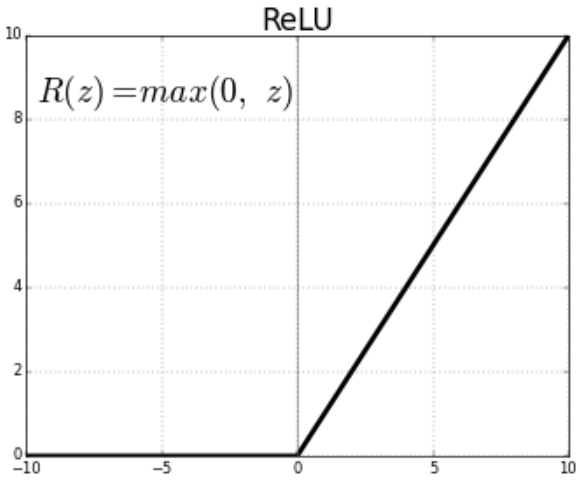


Figure 4.8: ReLU Activation Function [12]

"ReLU" is linear for values greater than zero, it has a lot of properties of linear activation functions when training a neural network using backpropagation. Nevertheless, it is a nonlinear function as negative values are always output as zero. However, even if the function is not differentiable at $z = 0$ theoretically invalidating it for gradient-based learning algorithm, in practice gradient descent still performs well enough for the models with "ReLU" to be used for machine learning tasks.

↪ Visualisation tool and Performance Metrics

The performance of the models described above was evaluated by the use of Root Mean Squared Error ("RMSE", 4.16) metrics and the learning evolution was diagnosed with the Tensorboard visualisation tool.

$$RMSE\left(\hat{f}\right) = \sqrt{MSE\left(\hat{f}\right)} \tag{4.16}$$

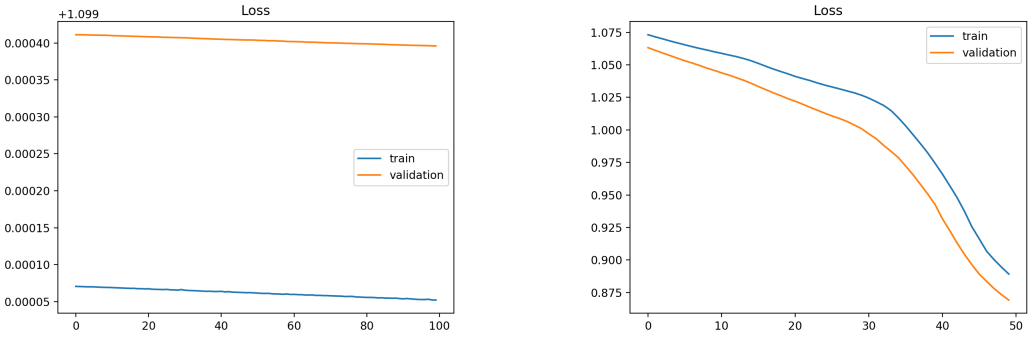
Tensorboard offers a means to examine the learning curves of each model. A learning curve is a plot of model learning performance over epoch. Examining learning curves while the models are training can be used to diagnose problems with learning, e.g. underfit or overfit problems and if the training and validation datasets are suitably representative.

The shape and dynamics of a learning curve is utilised to diagnose the behaviour of a ML model and it could suggest configuration changes to improve learning and/or performance. There are two common challenges, which need to be controlled, that can be easily analysed though learning curves, such problems are **underfitting** and its counterpart **overfitting**.

→ Underfit

Underfitting occurs when the model is not able to obtain a sufficiently low error value on the training set, or simply refers to a model that cannot learn the training dataset [14].

An underfit model can be identified only from the learning curve of the training loss. It may show a flat line or noisy values of significant high loss, indicating that the model is unable to learn the training dataset (4.9a), howerver some examples of underfitting may indicate that the model could be capable of further learning and improvement and that the training process was halted prematurely (4.9b) [13].



(a) Extreme Underfitting

(b) Controllable Underfitting

Figure 4.9: Underfit Examples [13]

→ Overfit

Overfitting occurs when the gap between the training error and test error is too large, i.e. the generalisation error [14]. Put simple, overfit refers to a model that has learned the training

dataset too well, including statistical noise or random fluctuations in the training dataset 4.10. The more specialised the model becomes to training data, the less it can generalise to new data, resulting in an increase in generalisation error. This increase is measured by the performance of the model on the validation dataset [13].

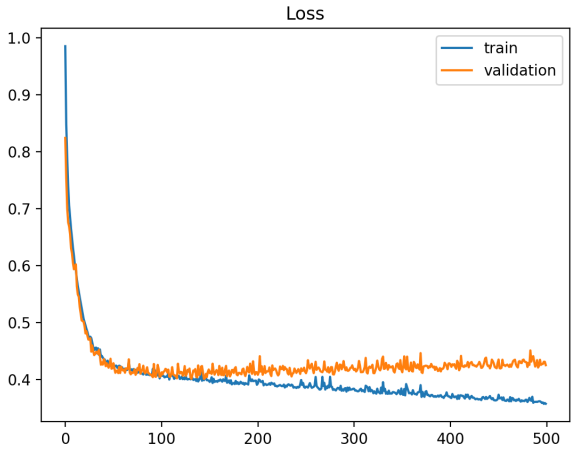


Figure 4.10: Overfit Example [13]

We can control whether a model is more likely to overfit or underfit by modifying its **capacity**. Informally, a model's capacity is its ability to fit a wide variety of functions. Models with low capacity may struggle to fit the training set. Models with high capacity can overfit by memorizing properties of the training set that do not serve them well on the test set [14]. As such, the best performance is found with the encounter of the middle ground between the capacity and the error.

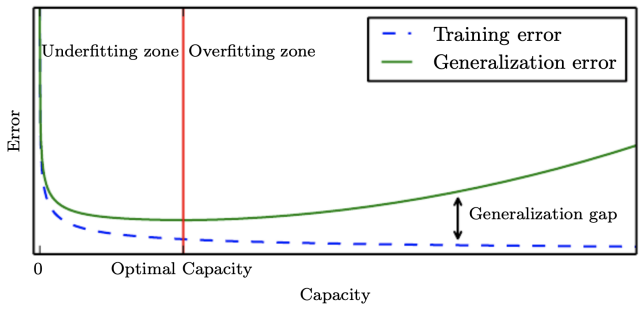


Figure 4.11: Capacity over Error relationship [14]

Yet this subject can be examined as a Bias-Variance trade-off. As the prediction error for any ML algorithm can be divided into three types of error: Bias, Variance and Irreducible error.

As for the irreducible error, it is the error introduced from the chosen framing of the problem and may be caused by factors, such as unknown variables. As such it cannot be reduced.

Bias is the simplifying assumption executed by a model to make the target function easier to learn. Generally, linear algorithms have a high bias making them fast to learn and easier to understand but generally less flexible. In turn, they have lower predictive performance on complex problems that fail to meet the simplifying assumptions of the algorithms bias. Then a model that presents low/high bias suggests less/more assumptions about the form of the target function, respectively [67].

Variance is the amount that the estimate of the target function will change if different training data was used. The function is estimated from the training data by a ML algorithm, so it is expected to have some variance. Ideally, it should not change significantly from one training dataset to another [67]. Therefore a model which presents low/high variance suggests small/large, respectively, changes to the estimate of the target function with changes to the training dataset.

The ultimate objective of any supervised ML algorithm is to achieve low bias and low variance, in order to achieve good prediction performance. In order to diminish the bias the complexity/capacity of the model has to be augmented, however, while complexity increases, it is accompanied by rising variance, (4.17). There is a trade-off at play, noted by the graph 4.12, between Bias and Variance, as such an achievable objective is to search for a 'sweet spot', (4.17), between the complexity and the error, which is equivalent to saying that it is necessary to find a meeting point between the bias and the variance or between capacity and error.

$$\frac{dBias}{dComplexity} = -\frac{dVariance}{dComplexity} \tag{4.17}$$

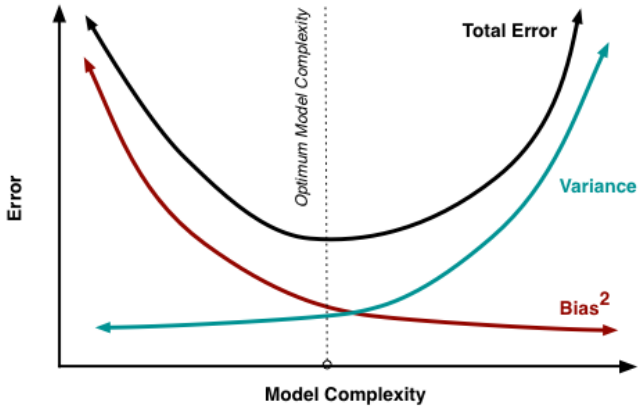


Figure 4.12: Bias/Variance and error relationship [15]

4.2.4 Model Evaluation Test Harness: Walk-Forward Validation

The objective of a test harness is to consistently evaluate candidate models against a fair representation of the problem. The outcome of testing multiple algorithms against the harness will be an estimation of how a variety of models perform on the problem against a chosen performance

measure. It must be robust and trustworthy in the results it provides, in order to focus on evaluating different algorithms and learn about the problem [16].

For this project a test harness scheme was developed and applied for the above models. This allowed for a huge variety of tests to be elaborated and the results subsequently analysed.

The developed test harness can be subdivided into four stages being:

1. Split the dataset into a Train and Test set.
2. Fit a candidate model on the training dataset.
3. Elaborate predictions on the test set using **Walk-Forward Validation**.
4. Saving the result for future analysis.

Finally, after finalising a platform to test each model, the optimum configuration for each forecasting algorithm will also be evaluated. Configuration of an ML algorithm represents the adjustment of hyper-parameters, such as: the number of epochs, batch size, number of hidden layers, number of nodes per layer and even the number of inputs and outputs. However, in this case, the configuration will only concern the number of input and output. Simply, it will be explored which number of months to provide (input) and which number of months to forecast (output) configuration that can actually provide the best forecast data. The settings will be evaluated with a grid search.

4.2.4.1 Time Series data preparation: Dataset Split, Sliding Window and Normalisation

For the elaboration of the tests of this project each dataset was divided in a training and test dataset. The training set is used to train each of the models and the test set will be to examine their performance. For the test set the year 2019 was chosen, that is, each dataset exposed to this test method is taken from it twelve final months corresponding to 2019. Recalling that each dataset sample is interspersed with a monthly interval.

After setting the training dataset it is necessary to process the same, in order to convert what is in time series format into a two-dimensional supervised learning format. The sliding window or lag method is the basis for how any time series dataset can be turned into a supervised learning format 4.4. This technique utilises previous time steps as input variables and employs the next time step as the output variable. The number of prior time steps is called the window width or size of the lag, and even if it does not have a designation the number of time steps ahead to be forecasted can also be varied.

Therefore, a sliding window can be created for many variants of the problem, as it can be recreated using one or more variables, i.e. uni/multivariate, and also where the amount of timesteps to be predicted can vary from single to several, i.e. uni/multistep.

For the aforementioned models modifying the data to the supervised learning format is not sufficient, it will be necessary to make a shape change due to the *input_shape* parameter present in the first hidden layer of all algorithms. As such for the *CNN Multichannel*, *Vanilla LSTM*,

TIME	MEASURE
1	100
2	110
3	108
4	115
⋮	⋮

→

X	y
–	100
100	110
110	108
108	115
115	⋮

Table 4.4: Sliding Window Example [16]

LSTM Encoder – Decoder and the *CNN – LSTM Encoder – Decoder* must be organised in a three dimensional way. The dimensions can be designated as:

- **Samples**: Each sequence (or row). A batch is comprised of one or several samples.
- **Time Steps**: One time step is a single point of observation in a sample. A single sample is comprised of multiple time steps.
- **Features**: One feature is a single observation at a time step. One time step is comprised of one or multiple features.

The desired input data structure is often summarised using following notation: $[samples, timesteps, features]$ 4.5 [16].

X1	X2	y
10	15	25
20	25	45
30	35	65
40	45	85
⋮	⋮	⋮

→

(7,3,2)	(7,1)
[[10 15] [20 25] [30 35]]	65
[[20 25]	
⋮	⋮

Table 4.5: Input data Structure Example [16]

However, as regards to the model *ConvLSTM Encoder – Decoder* the data structure is done differently according to the following notation: $[samples, timesteps, rows, columns, features]$, [16], this is required because this algorithm was developed for reading two-dimensional spatial-temporal data being essential to be adapted for use with time series, one dimensional, data. The input data is split into subsequences where each subsequence has a fixed number of time steps, although we must also specify the number of rows in each subsequence, which in this case is fixed at 1 for 1D data.

Thus, the way to acquire results had to be adapted, for example, if the dataset consisted of data with several attributes the number of features had to match the number of attributes.

Several tests were performed, however, while the routine of acquiring results with more than one feature, i.e. multivariate, became imperative that the data be processed in order to normalise them, following the the calculation (4.18). Normalisation is a rescaling of the data from the original range so that all values are within the range of 0 and 1 or any other range, such as -1 and 1 which

was the one applied in this project. It is useful, and even required in some machine learning algorithms when your time series data has input values with differing scales [68].

$$Scaled\_Value = \left(\frac{x - min(x)}{max(x) - min(x)} \right) \cdot (new\_max - new\_min) + new\_min \tag{4.18}$$

4.2.4.2 Walk-Forward Validation

The Walk-forward validation approach is distinguished by making a forecast for each observation in the test dataset one at a time and after each forecast is made for a time step in the test dataset, the true observation for the forecast is added to the test dataset and made available to the model. Simpler models can be refit with the observation prior to making the subsequent prediction. However complex models, such as NN, are not refit given the computational cost. Nevertheless, the true observation for the time step can then be used as part of the input for making the prediction on the next time step [16].

1. Starting at the beginning of the time series, the minimum number of samples in the window is used to train a model.
2. The model makes a prediction for the next time step.
3. The prediction is stored for further evaluation against the known value.
4. The window is expanded to include the known value and the process is repeated.

Since this technique involves moving along the time series one-time step at a time, it is often called Walk-Forward Testing or Walk-Forward Validation.

4.2.4.3 Grid Search

This dissertation incorporated a grid search strategy in order to explore the models and evaluate various input and output configurations. Essentially, two lists were prepared, one with the input number and one with the output number, and they were evaluated cyclically in the models already presented.

The output list is fixed, that is, it is the same for all models. It lists the number of months that must be foreseen in each timestep. The list is as follows $[1,3,4,6,12]$, it means that certain configurations will have to predict the following month, or the following three months, four months, half a year and up to the total year. However, the list of inputs is not fixed, i.e., it is different given the model and the way the data is organised to be trained. Differently to the output list, the input list refers to the number of months provided to each algorithm in order to predict another quantity. An example of a list provided to the model *Vanilla LSTM* is as follows $[3,6,9,12,18,24]$, which means that the algorithm is given from three months to two years of data to provide a forecast.

Therefore the grid search performed all the configuration examples provided by the lists and stored the results for further analysis. It is imperative to inform that each configuration was repeated

five times due to the stochastic nature of the models. Which means that, given the same model configuration and the same training dataset, a different internal set of weights will result each time the model is trained that will in turn have performance vary.

4.3 Summary

For the development of this thesis, GSOD and teleconnection data, such as AO, NAO, PDO and ICE cover, provided by NCEI were used. It was also essential the access to open source resources for the project development, such as Tensorflow, Pandas, Numpy and Sklearn.

After the variables to be forecast are established: The monthly mean values of maximum and daily mean temperatures (°C) and the monthly total precipitation (*mm*), the proposed problem to be solved was exposed to analysis. Some concepts and designations related to the data and models have been clarified, it is known that the dataset is of an endogenous nature and presents an organised structure and the model, or models, developed attempts to resolve a regression in a dynamic order, they would be capable of performing forecasts with uni/multivariate datasets and resolute predictions for one or more time steps (uni/multistep). Ultimately the input/output given and extracted from the model would be months.

The data was subjected to a preparatory treatment in which the data was exposed to cleaning techniques, unit conversion and identification for disposal of outliers. Subsequently the datasets to be tested by the algorithms were created, as it would not be beneficial to put the GSOD or the teleconnections dataset in its entirety. Taking advantage of the dataset creation, new features were also imagined, such as the seasons, the month of each observation and the number of days that rained more or than 1*mm*. The datasets were purposely created with fewer features thus providing a way to combat the problem known as the "Dimensionality Curse".

In total, five deep learning models have been created for this project, being: Multi-Channel CNN, Vanilla LSTM, Encoder-Decoder LSTM, CNN-LSTM Encoder-Decoder and ConvLSTM Encoder-Decoder.

After discussing the preparation of the data and the models drawn up, it remains to be explained how the tests were carried out.

Prior to initiating any test it is necessary to establish two parameters: the dataset (depending on which variable to predict) and the assignment of an input/output configuration (i.e. how much information (previous months) is available to the algorithm in order to predict information (months to predict). The dataset is then divided into two datasets, training and validation. The training dataset is exposed to a process that normalises and transforms the data, which is presented in a time-series format, into a supervised learning format supported for either model. Following the preparation of the data the model being tested is trained. Once the training is complete, the model is required to predict the months of 2019 for later evaluation, and this process is done by walk forward validation.

Chapter 5

Results

The variables to be predicted were already specified, in the previous chapter 4, as being: the monthly mean values of maximum and daily mean temperatures (°C) and the monthly total precipitation (*mm*). However, even though the main objective is to research the best model, from the ones displayed in 4.2.3, to predict the above variables in terms of accuracy and performance. It was also imperative to analyse the following topics: the behaviour displayed by the models in relation to the data that would be provided, i.e. to investigate how the overall performance of the models evolves by varying the datasets provided; and what input/output configuration shows the best results, i.e. for this case study, how many months could be predicted.

5.1 Process and Evaluation

This section will describe the test preparation process and also the various post-processing evaluation formulas that the tests were exposed to.

Firstly the dataset to be provided is defined, it is prepared taking into account which variable the model will predict. For each test a new dataset is designed in which the most recent one contains more or different information than the prior, e.g. to predict the monthly mean of daily mean temperature, TEMP, the first test can be run with a dataset containing only the information of the daily mean temperature average, while for the second test the dataset will contain information of the daily mean temperature average and the month to which it corresponds 5.1.

	TEMP		TEMP	MON		TEMP	ICE
1979-01-31	13.839	→	13.839	1	→	13.839	15.41
1979-02-28	14.036		14.036	2		14.036	16.18
1979-03-31	13.839		13.839	3		13.839	16.34
⋮	⋮		⋮	⋮		⋮	⋮

Table 5.1: Dataset for TEMP tests - Examples

Once a dataset is determined for testing it is performed by each model, discussed in 4.2.3, by several input/output settings. The same configuration is executed five times to ascertain whether the

configuration in question makes consistent or inconsistent predictions. Subsequently, once the test is completely carried out, the resulting data is organised into tables that correspond to its in/out setting 5.2.

IN-12-OUT-4	Repeat 0	Repeat 1	Repeat 2	Repeat 3	Repeat 4
	⋮	⋮	⋮	⋮	⋮

Table 5.2: Organised table correponding with a 24 to 4 in/out setting - Example

Finally the resulting data will be exposed to post-processing methods in order to quantify the performance of each in/out setting from model-to-model. The following three statistical measures were explored in order to evaluate the models.

1. **Coefficient of Determination,** R^2 *Score*

   The coefficient of determination measures the degree of association among the observed and predicted values. The calculation of the $R^2 Score$ is illustrated in (5.1).

   $$R^2 = \frac{\sum_{i=1}^{N} \left(y_i - \hat{y}_i\right)^2}{\sum_{i=1}^{N} \left(y_i - \bar{y}\right)^2} \tag{5.1}$$

   Where $\bar{y}$ is the mean taken over N data points, y_i is the observed value, whereas the $\hat{y}_i$ is the forecasted value.

2. **Sum of Squared Errors, SSE**

   Since the R^2 *Score* measures an estimate of the relationship between movements of a dependent variable based on an independent variable's movements it is not enough to distinguish between a good or a bad model. SSE is the measure of discrepancy between the observed and forecasted data. A small SSE implies a tight fit of the model to the data. The computation of SSE is depicted in (5.2).

   $$SSE = \sum_{i=1}^{N} \left(y_i - \hat{y}_i\right)^2 \tag{5.2}$$

3. **Persistence Index, PERS**

   The PERS formula is illustrated in (5.3).

   $$PERS = 1 - \frac{SSE}{SSE_{reference}} \tag{5.3}$$

   A value of PERS smaller or equal to 0 indicates that the model under evaluation performs worse or no better than the reference. A PERS value of 1 is obtained when the model under study provides exact estimates of observed values [33].

5.1.1 Climate Normals

Climate Normals are three-decade averages of climatological variables including temperature and precipitation. Climate normals are used for two principal purposes: 1) As a benchmark against which recent or current observations can be compared, including providing a basis for many anomaly based climate datasets; 2) Used, as a prediction of the conditions most likely to be experienced in a given location.

In order to evaluate the results, the average value of the last three-decades was used as a reference, e.g. if the variable to be predicted is the monthly total precipitation, PRCP, and the year to be forecasted is 2019 the values resulting from these tests will have as reference the average value of the last three-decades corresponding from the year 1989 to 2018 5.3.

	PRCP
1	140.94
2	89.07
3	98.50
4	86.57
5	82.28
6	27.93
7	13.00
8	26.59
9	61.00
10	133.10
11	155.43
12	145.19

Table 5.3: Mean of the last 30 years (1989-2018), PRCP

Thus the reference point varies depending on the year that is proposed to be foreseen, i.e. if the year to be foreseen by the models is 2005 the reference value for the last 30 years (1974-2004).

5.2 Temperature Results

In the presenting section the results of the monthly mean values of maximum and daily mean temperatures will be presented. It is imperative to note that a significant part of the results correspond to the 2019 forecast. Subsequently and for specific models, simulations were carried out in order to forecast different years such as 1990, 1991 and 2018.

5.2.1 Monthly Mean of daily Mean Temperature, TEMP

The tests for the TEMP variable prediction were performed in a manner described in 5.1. Therefore, four different datasets were elaborated keeping present in each one the TEMP feature. So the first dataset is simple and consists of only the TEMP variable; the second contains the TEMP variable and the corresponding month; the third instead of containing information about the month contains

information about ICE cover; and finally the fourth dataset consists of the TEMP variable, ICE cover and the season corresponding to each sample.

The obtained results correspond to attempts to predict the year 2019, so the reference values will be the following 5.4.

	2019	**Average (1989-2018)**
1	9.23	9.96
2	11.30	10.52
3	12.99	12.43
4	13.08	13.60
5	17.14	15.91
6	16.32	18.33
7	18.95	19.46
8	18.79	19.73
9	19.26	18.42
10	15.75	16.40
11	12.75	12.72
12	11.59	10.92

Table 5.4: Observed values of TEMP - 2019 and three-decadal average

The aforementioned reference values make it possible to establish the reference values for the R^2 *Score* and the SSE, the same being: $R^2 \approx 0.932$ and $SSE \approx 9.984$. Henceforth the results of each dataset for the TEMP forecast will be demonstrated in the appendix of this work in A.1. It is important to note that the results correspond only to the best in/out configurations for each model and for the statistical evaluation measures (R^2, SSE and $PERS$) only the average value corresponding to the five repetitions performed will be presented.

Starting with the results of the dataset that only contains the TEMP variable, presented in the table A.1. The results from the best configuration/model of the TEMP dataset is displayed right below the former referenced table in A.1.

Subsequently the results corresponding to the dataset containing information on TEMP and the month are represented in the table A.2. The best configuration/model of the TEMP and month dataset is displayed below the table in A.2.

The results of the third dataset, which includes the TEMP variable with the ICE cover variable, are demonstrated in the chart A.3. The best configuration/model of the TEMP and Ice cover dataset is shown below the referenced chart in A.3.

Finally the results of the fourth dataset, which includes the TEMP variable in conjunction with the ICE cover and season variable, are presented in the table A.4. The best configuration/model of the TEMP, Ice cover and season dataset is exhibited right after the former referenced chart in A.4.

For the particular case of the test with the dataset corresponding to TEMP and month, two more simulations were performed, with the best model, in order to predict the year 1991 and 2018 with the purpose of evaluating the generalisation capacity of the model and analyse its behaviour when assigned the task of forecasting a different year, the result are presented on the table A.5. As can be

expected since the task imposed on these simulations is to predict a different year from 2019 the reference and assessment values are differing: for 1991 the $R^2 \approx 0.938$ and $SSE \approx 13.081$; as for the year 2018 $R^2 \approx 0.945$ and $SSE \approx 9.223$. With the best forecast for each year presented below the results in A.5.

To end the exposure of the TEMP variable results a summary will be displayed taking into account the best results for each dataset. A table with the reference values for TEMP is displayed first in 5.5, followed by the results summary 5.6.

Referenced Values - TEMP		
Year	R^2	*SSE*
2019	0.932	9.984
2018	0.945	13.081
1991	0.938	9.223

Table 5.5: Reference Values (Climate Normals) of TEMP

Dataset	Model	IN/OUT Setting	R^2	*SSE*	*PERS*
TEMP	*ConvLSTM Enc – Dec*	IN-24/OUT-3	0.934	9.343	0.064
TEMP+MON	*ConvLSTM Enc – Dec*	IN-24/OUT-6	0.938	10.795	-0.081
TEMP+ICE	*ConvLSTM Enc – Dec*	IN-24/OUT-3	0.933	9.318	0.067
TEMP+SEASON+ICE	*ConvLSTM Enc – Dec*	IN-24/OUT-3	0.930	10.179	-0.020

Table 5.6: Summary of the results of TEMP

5.2.2 Monthly Mean of the Maximum daily Temperature, MAX

Proceeding to the results of the monthly mean of the maximum daily temperature, MAX. The reference values for this variable are $R^2 \approx 0.868$ and $SSE \approx 23.211$. As previously mentioned, the results available correspond only to the most effective in/out configuration evaluated with the aforementioned measures. The results are available at the appendix of this dissertation A.1.

The only dataset used to predict the MAX is constituted with the MAX variable itself, the month and season corresponding to each sample and the ICE cover. With the results available in the board A.6, subsequently the best model is represented below the referenced table in A.6. Similar to what was done for the TEMP and month dataset, this dataset will also be subjected to two simulations, with the best model, in order to forecast the year 1990 and 2018. As previously the reference values are changed according to the year to be foreseen: for 1990 $R^2 \approx 0.850$ and $SSE \approx 41.425$; and for 2018 $R^2 \approx 0.881$ and $SSE \approx 29.931$. The results are available at A.7, A.7.

To terminate the exposure of the MAX variable results a summary will be displayed taking into account the best results for each dataset. A table with the reference values for MAX is displayed first in 5.7, followed by the results summary 5.8.

Referenced Values - MAX		
Year	R^2	*SSE*
2019	0.868	23.211
2018	0.881	29.931
1990	0.850	41.425

Table 5.7: Reference Values (Climate Normals) of MAX

Dataset	Model	IN/OUT Setting	R^2	*SSE*	*PERS*
MAX+MON+SEA+ICE	$CNN-LSTM\ Enc-Dec$	IN-24/OUT-3	0.871	24.503	-0.056

Table 5.8: Summary of the results of MAX

5.3 Precipitation Results

Throughout the following section the results of the total monthly precipitation, (PRCP), will be reported. As before, a significant portion of the data to be demonstrated relates to the year 2019. However, for some specific cases simulations have also been made in order to forecast the years 2005 and 2018.

As previously done, several datasets have been prepared with the purpose of predicting the PRCP for each month of the year 2019. For this case the reference values are the following: $R^2 \approx 0.471$ and $SSE \approx 59468.493$.

The first dataset to be presented is the one containing only the PRCP variable, showing the following behaviour in the table A.8. Being the best model represented in the couple of images A.8.

The second dataset to be tested is the one containing the PRCP variable with the corresponding month. Its behaviour can be analysed in the board A.9, as the best performance model behaviour in the couple images A.9. As for other future datasets, this dataset has been exposed to two more simulations in order to forecast the year 2005 and 2018. The reference values for the year 2005 is $R^2 \approx 0.377$ and $SSE \approx 58329.63$; for 2018 the values are $R^2 \approx 0.487$ and $SSE \approx 44231.82$. The result of both simulations can be analyzed in the table A.10 and displayed the images A.10.

The third datatset in the line comprises the PRCP variable, the month and the number of days per month on which the total precipitation value was greater than or equal to $1mm$. Its behaviour can be evaluated on the chart A.11 and the best accurate model for 2019 can be examined in the images A.11. For this dataset, tests were also carried out in order to predict the years 2005 and 2018 and the following results were obtained presented on the table A.12 being the best models of both years represented in the images A.12.

Afterwards the dataset which is incorporated with the variable PRCP, 1mm, month and season is put to the test and can be examined on the chart A.13, while the best performance model can be visualised here A.13. As with the previous dataset this one was also exposed to tests in order to forecast the years 2005 and 2018. The performance for each year can be analysed on the chart A.14 being the best models of both years shown in A.14.

The following dataset was the first interaction that history data had with data from a teleconnection. This contained only the PRCP variable and NAO index data. Its performance evaluated in the table A.15, while the best performing model can be examined in A.15.

Finally, the dataset consisting of the variable PRCP, 1mm, season and NAO is evaluated. Its performance can be analyzed on the chart A.16, and the best performing model available in A.16.

As previously done a summary of the former PRCP results will be displayed taking into account the best results for each dataset. A table with the reference values for PRCP is displayed first in 5.9, followed by the results summary 5.10.

Referenced Values - PRCP		
Year	R^2	*SSE*
2019	0.471	59468.493
2018	0.487	44231.82
2005	0.377	58329.63

Table 5.9: Reference Values (Climate Normals) of PRCP

Dataset	Model	IN/OUT Setting	R^2	*SSE*	*PERS*
PRCP	*Vanilla LSTM*	IN-3/OUT-3	0.619	49109.29	0.174
PRCP+MON	*Enc – Dec LSTM*	IN-9/OUT-3	0.545	54573.18	0.082
PRCP+1*mm*+MON	*Enc – Dec LSTM*	IN-9/OUT-3	0.561	52350.68	0.120
PRCP+1*mm*+MON+SEA	*ConvLSTM Enc – Dec*	IN-6/OUT-3	0.611	54390.8	0.085
PRCP+NAO	*Enc – Dec LSTM*	IN-6/OUT-12	0.451	74188.25	-0.248
PRCP+1*mm*+SEA+NAO	*CNN – LSTM Enc – Dec*	IN-24/OUT-6	0.454	59418.54	0

Table 5.10: Summary of the results of PRCP

5.4 Summary

Throughout this chapter it was made explicit how the results were acquired, how they were evaluated in terms of performance and accuracy, ending with the disclosure of the results in relation to temperature and precipitation. The results are separated by dateset, analysed from model to model being traversed by a series of configurations in/out and evaluated by the coefficient of determination, or R^2 *Score*, the sum of squared errors, or SSE, and the persistence index, or PERS.

Chapter 6

Discussion

In the course of this chapter, commentary on the results presented in 5.2 and 5.3, available at A.1 and A.2 will be presented.

6.1 Temperature Results

Prior to starting the discussion on the values obtained regarding the monthly mean of daily mean temperature, TEMP, and maximum daily temperature, MAX, it is essential to recall the reference values of each one for the year 2019, being for the TEMP: $R^2 \approx 0.932$ and $SSE \approx 9.984$, whereas for the MAX: $R^2 \approx 0.868$ and $SSE \approx 23.211$.

Following a brief analysis of the results, it should first be noted that, clearly, only the indication of R^2 is not sufficient to discriminate between a good and a bad model, since, even if, for certain examples the R^2 value is acceptable compared to the reference value, the same cannot be stated for the value of SSE and PERS which measures the discrepancy between the forecasted and reference data, while the R^2 measures the degree of association among the reference and predicted values.

The exposed datasets have been prepared with precision. The variables selected to accompany the variable to be predicted (TEMP and MAX) were not assigned randomly. These were exposed to an analysis method that mathematically quantified the degree of correlation that they exhibited with the variable to be predicted, and were also selected according to statements in the literature. So it would only be natural to assume that the results from these dateset were to be acceptable, however, this is not necessarily the case for all instances. There are cases where the opposite occurs, that is, the performance of the models decreases.

As more information is made available, the models show a general improvement in performance, it is sufficient to analyse the differences between the results presents on the tables A.1 and A.2. Nevertheless, it is reported that the inclusion of a larger amount of information worsens the performance being examples of this presented on the charts A.3 and A.4. The reason why these examples performed poorly in relation to the benchmarks is not clear when examining each one individually, so two questions arise: was the limit of information reached, i.e. has the maximum dimensionality allowed by the models been reached and is that the reason of the declining

performance ("dimensionality curse"); or are some of the allegedly correlated variables the source of the problem, i.e, is the inclusion of some of these variables, e.g. the ICE cover, benefiting the performance of the models, or is it simply providing sources of noise.

By paying more attention to cases represented on the boards A.2 and A.3 it becomes clear that the cause of general model performance discrepancy is due to the ICE cover variable. The issue of dimensionality is not applicable in these example, since both datasets have the same dimension containing both two features. Considering that the supposed main cause for poor performance of certain models is due to the fact of the ICE cover variable presence in the dataset clarifies the performance presented on the charts A.4 and A.6.

Even with the diversity of results made available it became clear that with the exclusive use of history it was possible to develop some instances that exceed the reference value, which is the average of the last three decades, for 2019. Such instances can be analysed in the tables A.2 and surprisingly in the chart A.3.

In order to analyse the generalisation capacity and better evaluate the accuracy of the models, two more simulations were performed in order to predict two different years. The results of these simulations can be viewed on the charts A.5 and A.6. The results indicated above refer to the year: 1990, 1991 and 2018; and with different variables to be predicted: TEMP and MAX; then the reference values are: for TEMP, 1991 $R^2 \approx 0.938$ and $SSE \approx 13.081$; for TEMP, 2018 $R^2 \approx 0.945$ and $SSE \approx 9.223$; for MAX, 1990 $R^2 \approx 0.850$ and $SSE \approx 41.425$; and for MAX, 2018 $R^2 \approx 0.881$ and $SSE \approx 29.931$. When analysing both cases it becomes clear that the models predict relatively well the year 1990 and 1991, however for 2018 forecast both models show a poor performance. It is necessary, however, to point out that for these simulations the selected models were those that quantitatively presented a better performance but nothing prevented that another model already trained could not impose better results.

6.2 Precipitation Results

As previously conducted before initiating the discussion on the total monthly precipitation, PRCP, results it is necessary to recall the reference values for the year 2019: $R^2 \approx 0.471$ and $SSE \approx 59468.493$.

Exactly as previously verified, by quickly analysing the results it can be said that the value of R^2 is still not enough to distinguish a good from a bad model.

It can also be verified, as previously mentioned, as more information is made available the overall accuracy and performance of the models increases. This can be assessed by analysing the following charts A.8, A.9, A.11 and A.13.

However, occasionally the increase in information does not benefit the performance of the models and can even deteriorate it. It depends on the variables that are arranged for the model. These occurrences can be witnessed in the tables A.15 and A.16. In both occurrences it is evident that the variable NAO degrades the accuracy and performance of the models.

As previously, it became clear that with the exclusive use of history it was possible to develop models that exceed the reference value for 2019. Such instances are presented on the board A.8, A.9, A.11 and A.13.

Since the results for the 2019 forecast were quite acceptable, simulations were also prepared for each of the above mentioned dateset to forecast 2005 and 2018. The reference values for 2005 and 2018 are as follows: 2005, $R^2 \approx 0.377$ and $SSE \approx 58329.63$ and for 2018, $R^2 \approx 0.487$ and $SSE \approx 44231.82$. The results of the simulations are available at the table A.10, A.12 and A.14. When analysing the three available examples, it is found that, exactly as before for TEMP and MAX, the forecast for 2018 is substantially weaker, even so the forecast for 2005 for each of the examples presents acceptable results.

Chapter 7

Conclusion

Considering the enormous influence that weather forecasts have on the life of each individual and their enormous impact on countries where the economy is based on agriculture and tourism, it is crucial that these forecasts are as accurate as possible. Therefore the study of the atmosphere is something that will continue to be developed as more data is made available and with its evolution more and better forecasting models are going to be created.

The main objective of this dissertation was to develop a Machine Learning algorithm capable of emulating climate variability so that it could be possible to extract forecasting results as good or better than traditional forecasting models being these statistical or conditioned by physical equations of the atmosphere. A further objective of this thesis is to contribute to the continuous study of the atmosphere and to provide aspects that may assist in the development of predictive models.

This project was started with the selection of data to be utilised throughout this thesis, these were the GSOD reports and data regarding teleconnections. Following their selection, the data was processed and several datasets were prepared. These datasets would later be provided to the various deep learning models developed, being the same designated as: Multi-Channel CNN, Vanilla LSTM, Encoder-Decoder LSTM, CNN-LSTM Encoder-Decoder and ConvLSTM Encoder-Decoder. The tests were carried out by dataset, each model would be evaluated taking into account the in/out configuration imposed on to it. The results were evaluated by the following statistical measurements: Coefficient of Determination, R^2 *Score*; Sum of Squared Errors, SSE; and the Persistence Index, PERS.

Following the analysis of the results, the following conclusions were established.

It has been witnessed that the numerical values of teleconnections cannot be used in isolation because instead of improving the performance of the models they do exact opposite. This means that indeces such as NAO do not benefit this mode of forecasting, i.e. even if it is proven in the literature that NAO improves the performance of global climate models (models of global circulation) the same does not apply for the type of forecasting elaborated throughout this project.

Furthermore, it is imperative to point out that the method of behaviour of this type of models is not linear. In other words, for certain variables that proved to be correlated with the predicted

variable, these did not show improvements in the models' performance and, in some cases, could make them even worse. This was the case for the ICE cover variable.

However, in conclusion, it has been proven that it is possible to make predictions of the state of the atmosphere with deep learning techniques using only historical data.

7.1 Future Work

Upon completion of this study, several possibilities for future work were considered such as:

1. Development of more tests with different datasets, because it has been proven that deep learning models can find correlations with several variables.
2. Perform tests covering more years and perform tests with data from different locations.
3. Cover a larger area so as not to be dependent on data from the same station and have access to more data.
4. Optimisation of models already created from an optimisation algorithm, such as Bayesian
5. Possible development of other architectures for the models, such as the adoption of "GRU" cell instead of "LSTM" and "Swish" activation function instead of "ReLU"
6. Applications of more complex pre-processing methods, e.g. wavelet decomposition.

Appendix A

Results

A number of graphics corresponding to the results which did not prove necessary to be included in the main text will be available in an attachment

A.1 Temperature Results

A.1.1 Monthly Mean of daily Mean Temperature, TEMP

TEMP				
Model	IN/OUT Setting	R^2	*SSE*	*PERS*
Multi – channel CNN	IN-12/OUT-3	0.911	13.713	-0.374
Vanilla LSTM	IN-9/OUT-6	0.913	13.638	-0.366
Encoder – Decoder LSTM	IN-6/OUT-4	0.920	24.364	-1.440
ConvLSTM Encoder – Decoder	IN-24/OUT-3	0.934	9.343	0.064

Table A.1: Results from the TEMP dataset

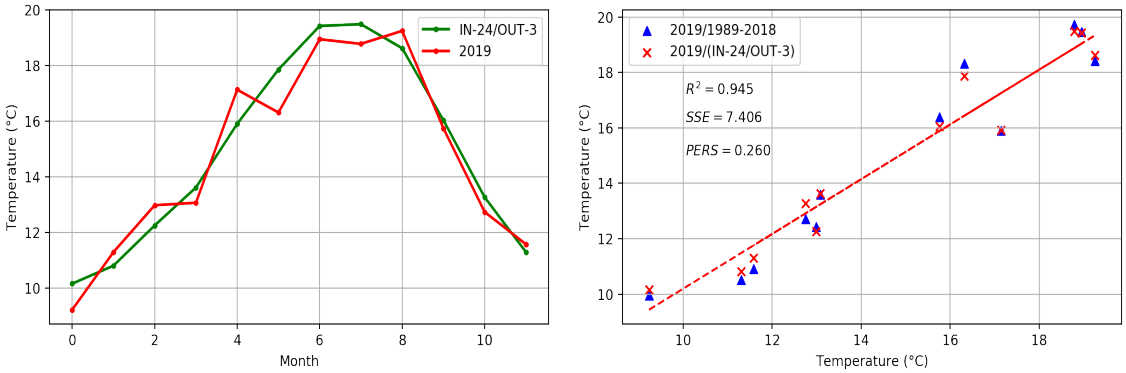


Figure A.1: Result of In-24/Out-3, *ConvLSTM Encoder – Decoder*, TEMP

TEMP + Month				
Model	IN/OUT Setting	R^2	*SSE*	*PERS*
Multi – channel CNN	IN-24/OUT-6	0.932	10.821	-0.084
Vanilla LSTM	IN-9/OUT-12	0.918	14.338	-0.436
Encoder – Decoder LSTM	IN-12/OUT-4	0.934	11.772	-0.179
CNN – LSTM Encoder – Decoder	IN-24/OUT-1	0.923	11.968	-0.199
ConvLSTM Encoder – Decoder	IN-24/OUT-6	0.938	10.795	-0.081

Table A.2: Results from the TEMP + Month dataset

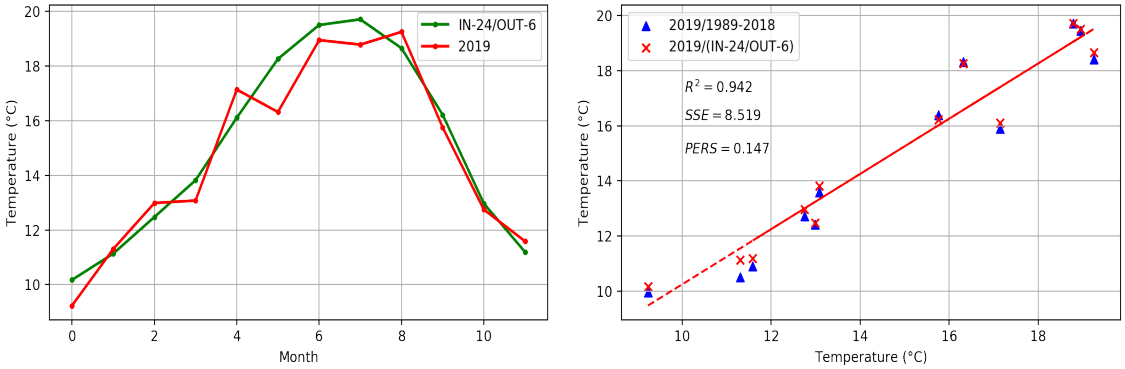


Figure A.2: Result of In-24/Out-6, *ConvLSTM Encoder – Decoder*, TEMP + Month

TEMP + ICE				
Model	IN/OUT Setting	R^2	*SSE*	*PERS*
Multi – channel CNN	IN-24/OUT-1	0.909	19.231	-0.926
Vanilla LSTM	IN-18/OUT-12	0.917	11.942	-0.196
Encoder – Decoder LSTM	IN-9/OUT-12	0.908	13.550	-0.357
CNN – LSTM Encoder – Decoder	IN-24/OUT-12	0.929	13.814	-0.384
ConvLSTM Encoder – Decoder	IN-24/OUT-3	0.933	9.318	0.067

Table A.3: Results from the TEMP + ICE cover dataset

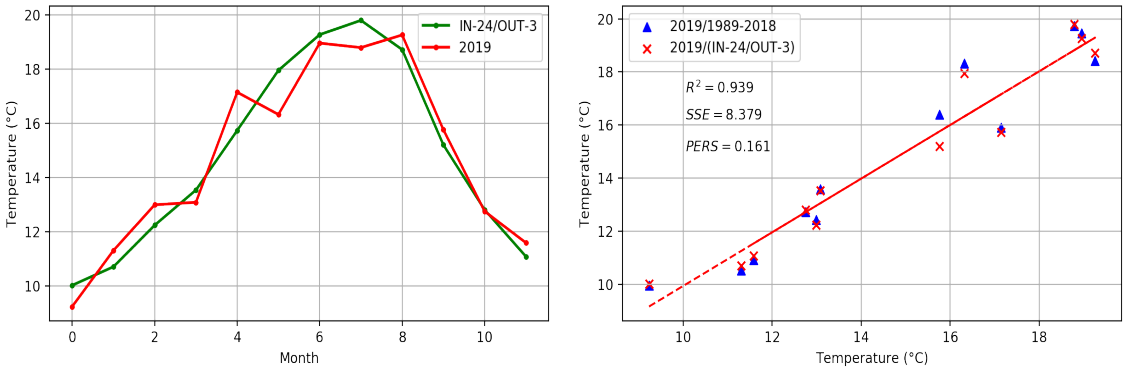


Figure A.3: Result of In-24/Out-3, *ConvLSTM Encoder – Decoder*, TEMP + ICE cover

TEMP + SEASON + ICE				
Model	IN/OUT Setting	R^2	*SSE*	*PERS*
Multi – channel CNN	IN-18/OUT-6	0.923	14.916	-0.494
Vanilla LSTM	IN-6/OUT-12	0.920	12.553	-0.257
Encoder – Decoder LSTM	IN-3/OUT-12	0.921	12.822	-0.284
CNN – LSTM Encoder – Decoder	IN-24/OUT-3	0.924	13.340	-0.336
ConvLSTM Encoder – Decoder	IN-24/OUT-3	0.930	10.179	-0.020

Table A.4: Results from the TEMP + Season + ICE cover dataset

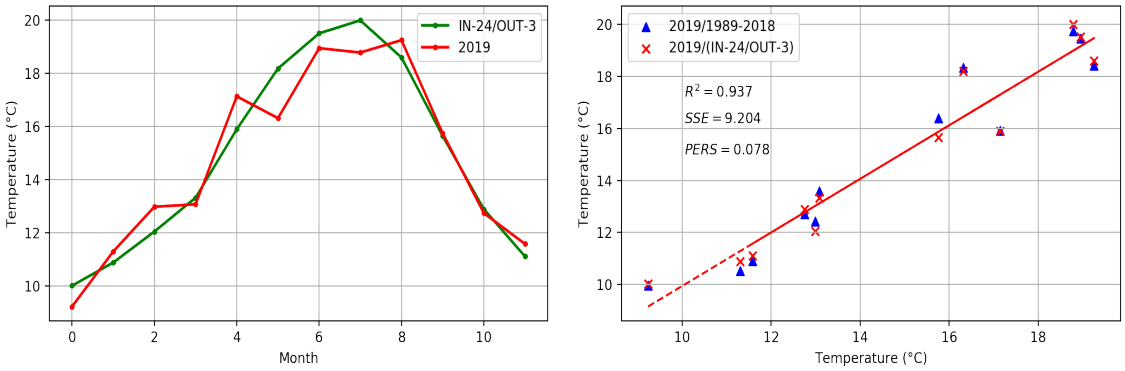


Figure A.4: Result of In-24/Out-3, *ConvLSTM Encoder – Decoder*, TEMP + Season + ICE

TEMP + Month					
	Model	IN/OUT Setting	R^2	SSE	PERS
1991	*ConvLSTM Encoder – Decoder*	IN-24/OUT-6	0.949	10.852	0.170
2018	*ConvLSTM Encoder – Decoder*	IN-24/OUT-6	0.935	10.982	-0.191

Table A.5: Result of the year 1991 and 2018, TEMP + Month

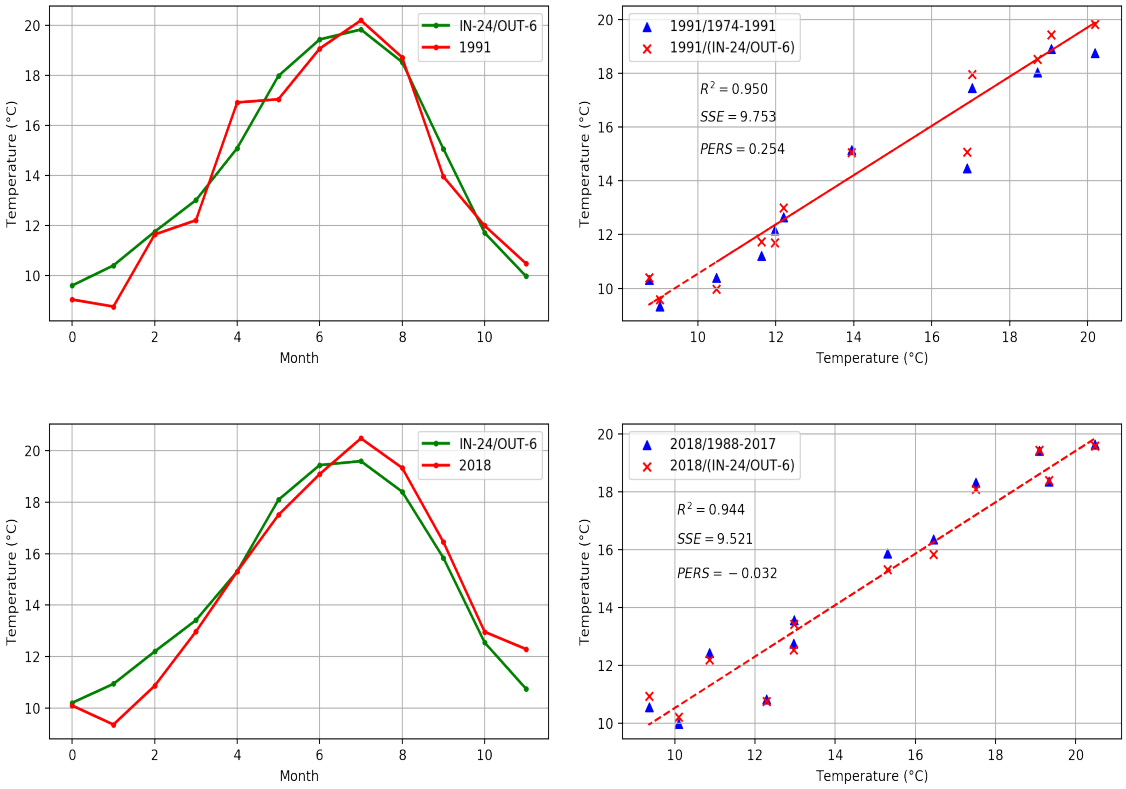


Figure A.5: Result of In-24/Out-6, *ConvLSTM Encoder – Decoder*, TEMP + Month. 1991(top) and 2018(below)

A.1.2 Monthly Mean of the Maximum daily Temperature, MAX

MAX + Month + SEASON + ICE				
Model	IN/OUT Setting	R^2	*SSE*	*PERS*
Multi – channel CNN	IN-9/OUT-3	0.860	32.116	-0.384
Vanilla LSTM	IN-12/OUT-6	0.820	29.868	-0.287
Encoder – Decoder LSTM	IN-9/OUT-6	0.801	32.894	-0.417
CNN – LSTM Encoder – Decoder	IN-24/OUT-3	0.871	24.503	-0.056
ConvLSTM Encoder – Decoder	IN-24/OUT-4	0.835	25.007	-0.077

Table A.6: Results from the MAX + Month + Season + ICE cover dataset

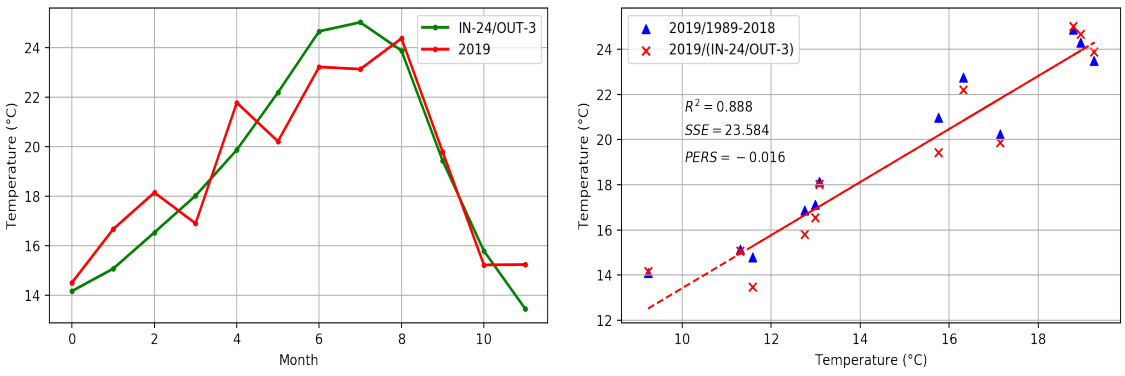


Figure A.6: Result of In-24/Out-3, *CNN – LSTM Encoder – Decoder*, TEMP + Month + Season + ICE

MAX+MON+SEA+ICE					
	Model	IN/OUT Setting	R^2	SSE	PERS
1990	$CNN - LSTM\ Enc - Dec$	IN-24/OUT-3	0.929	37.732	0.089
2018	$CNN - LSTM\ Enc - Dec$	IN-24/OUT-3	0.810	40.914	-0.367

Table A.7: Result of the year 1990 and 2018, MAX + Month + SEASON + ICE

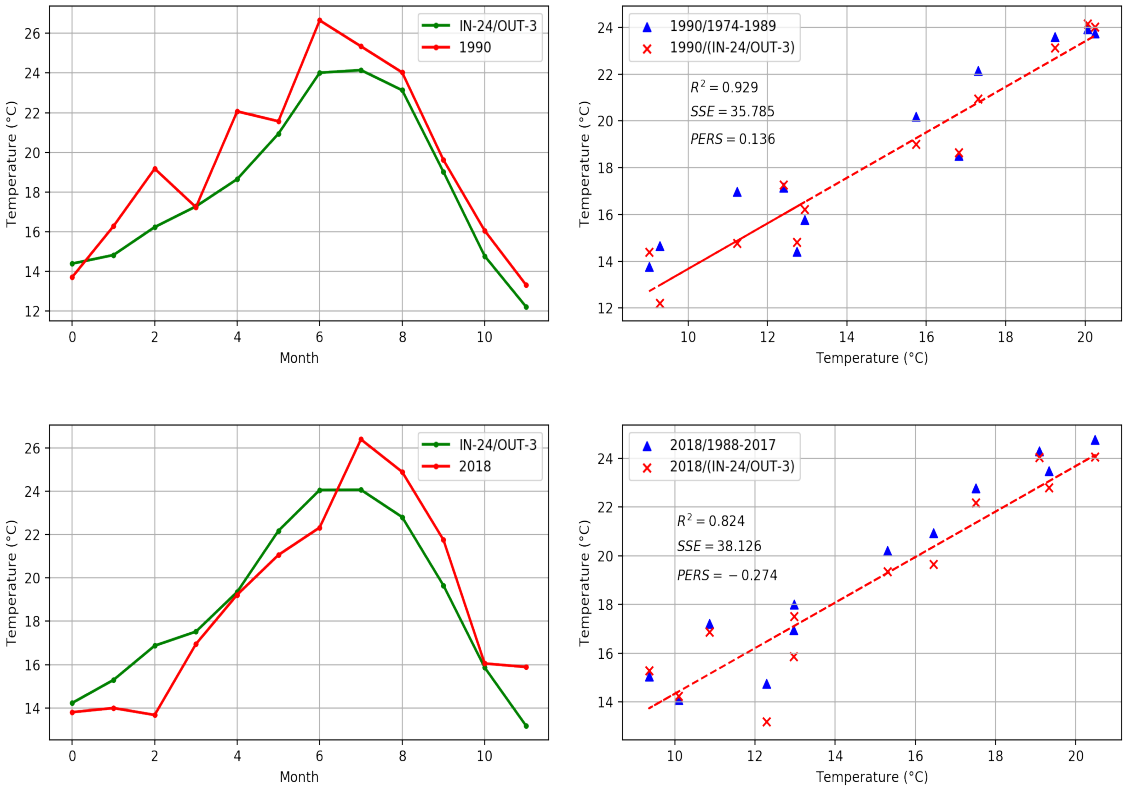


Figure A.7: Result of In-24/Out-3, $CNN - LSTM\ Encoder - Decoder$, MAX + Month + SEASON + ICE. 1990(top) and 2018(below)

A.2 Precipitation Results

PRCP				
Model	IN/OUT Setting	R^2	*SSE*	*PERS*
Multi – channel CNN	IN-9/OUT-3	0.242	191634.8	-2.222
Vanilla LSTM	IN-3/OUT-3	0.619	49109.29	0.174
Encoder – Decoder LSTM	IN-6/OUT-4	0.482	58475.11	0.017
ConvLSTM Encoder – Decoder	IN-18/OUT-4	0.293	189463.5	-2.186

Table A.8: Results from the PRCP dataset

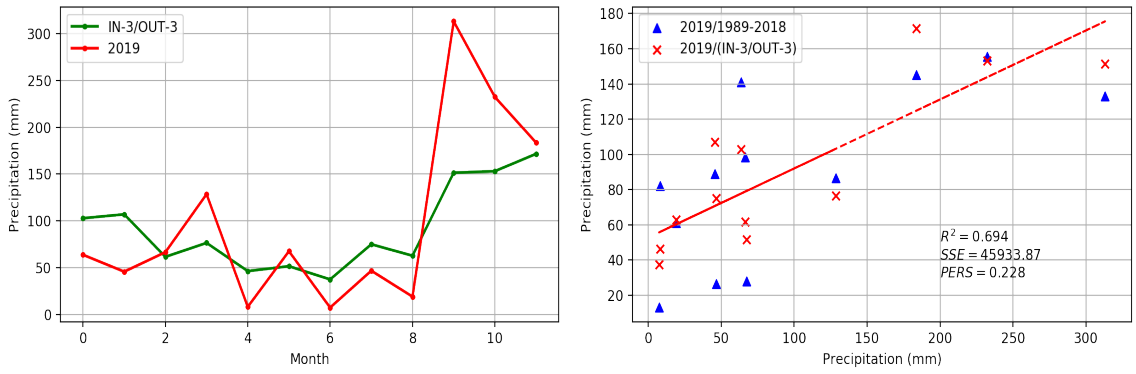


Figure A.8: Result of In-3/Out-3, *Vanilla LSTM*, PRCP

PRCP + Month				
Model	IN/OUT Setting	R^2	*SSE*	*PERS*
Multi – channel CNN	IN-24/OUT-12	0.393	66062.79	-0.111
Vanilla LSTM	IN-9/OUT-3	0.486	58475.5	0.017
Encoder – Decoder LSTM	IN-9/OUT-3	0.545	54573.18	0.082
CNN – LSTM Encoder – Decoder	IN-24/OUT-3	0.485	61231.71	-0.030
ConvLSTM Encoder – Decoder	IN-24/OUT-3	0.507	56115.9	0.056

Table A.9: Results from the PRCP + Month dataset

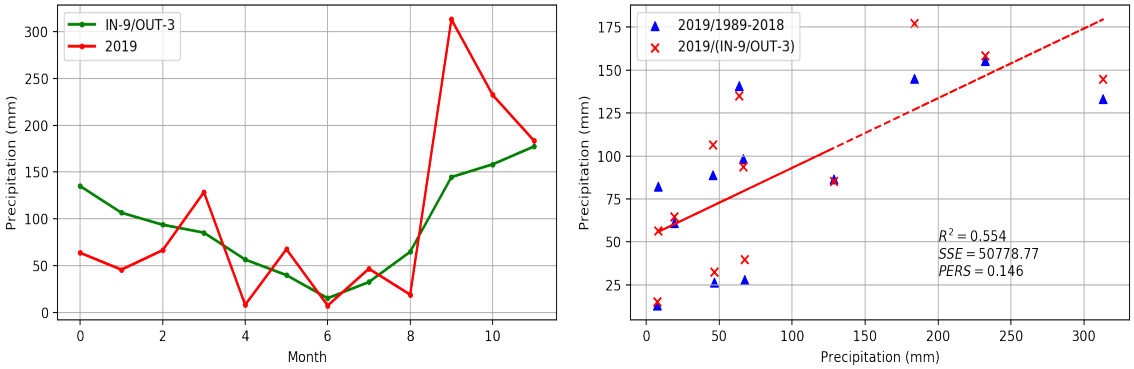


Figure A.9: Result of In-9/Out-3, *Encoder – Decoder LSTM*, PRCP + Month

PRCP + Month					
	Model	IN/OUT Setting	R^2	SSE	PERS
2005	*Encoder – Decoder LSTM*	IN-9/OUT-3	0.491	43104.71	0.261
2018	*Encoder – Decoder LSTM*	IN-9/OUT-3	0.460	56011.65	-0.266

Table A.10: Result of the year 2005 and 2018, PRCP + Month

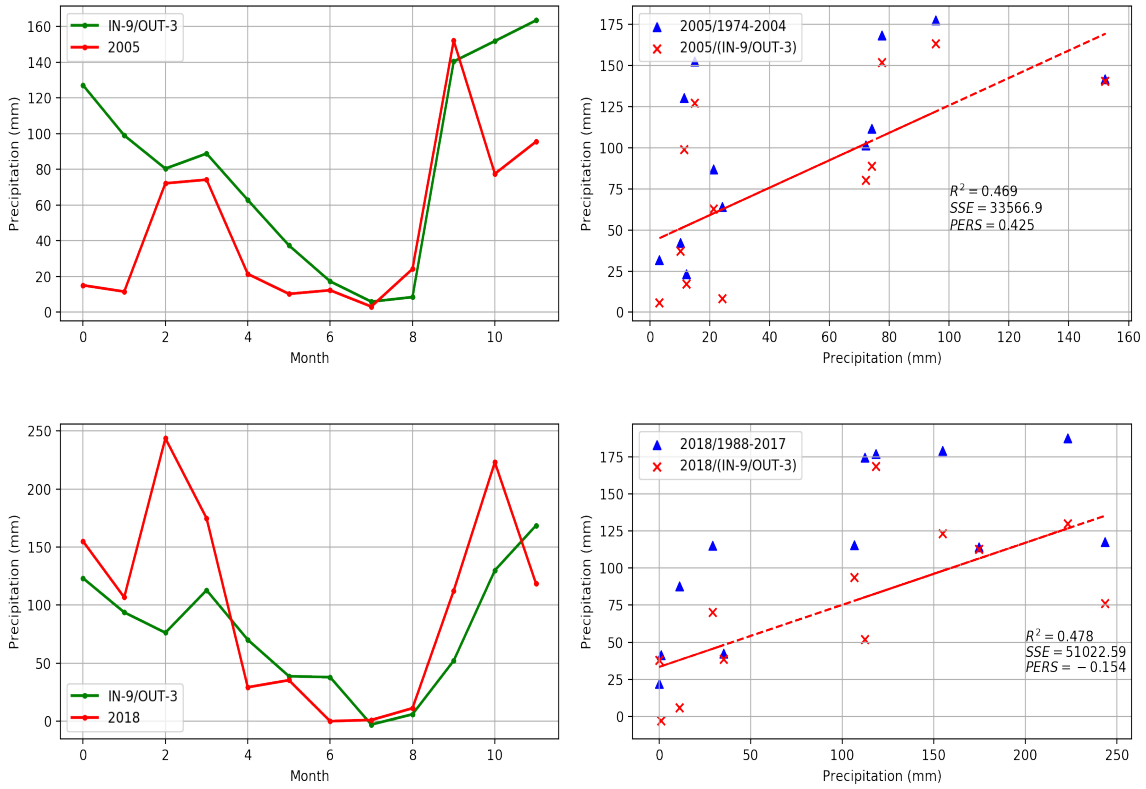


Figure A.10: Result of In-9/Out-3, *Encoder – Decoder LSTM*, PRCP + Month. 2005(top) and 2018(below)

PRCP + 1*mm* + Month				
Model	IN/OUT Setting	R^2	*SSE*	*PERS*
Multi – channel CNN	IN-24/OUT-6	0.389	67951.55	-0.143
Vanilla LSTM	IN-6/OUT-6	0.486	56203.79	0.055
Encoder – Decoder LSTM	IN-9/OUT-3	0.561	52350.68	0.120
CNN – LSTM Encoder – Decoder	IN-6/OUT-6	0.552	50555.25	0.150
ConvLSTM Encoder – Decoder	IN-9/OUT-3	0.539	52753.79	0.113

Table A.11: Results from the PRCP + 1*mm* + Month dataset

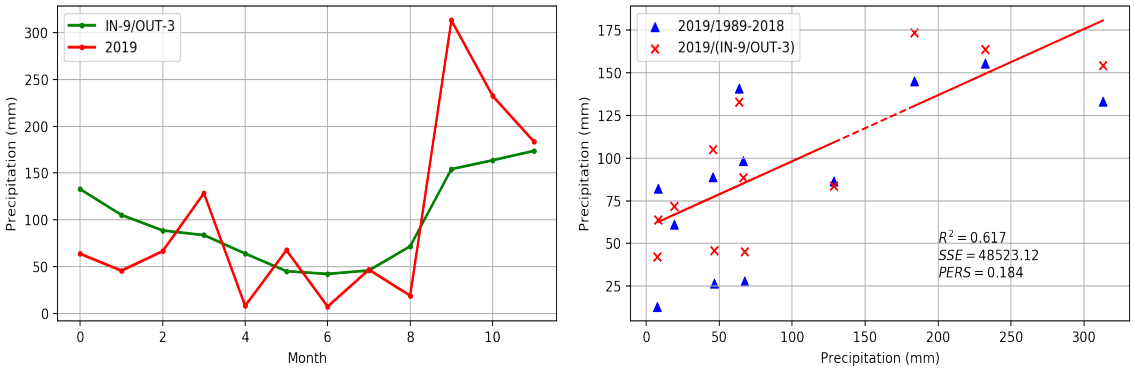


Figure A.11: Result of In-9/Out-3, *Encoder – Decoder LSTM*, PRCP + 1*mm* + Month

PRCP+1*mm*+Month					
	Model	IN/OUT Setting	R^2	SSE	PERS
2005	*Encoder – Decoder LSTM*	IN-9/OUT-3	0.479	33889.58	0.419
2018	*Encoder – Decoder LSTM*	IN-9/OUT-3	0.432	56546.07	-0.278

Table A.12: Result of the year 2005 and 2018, PRCP + 1*mm* + Month

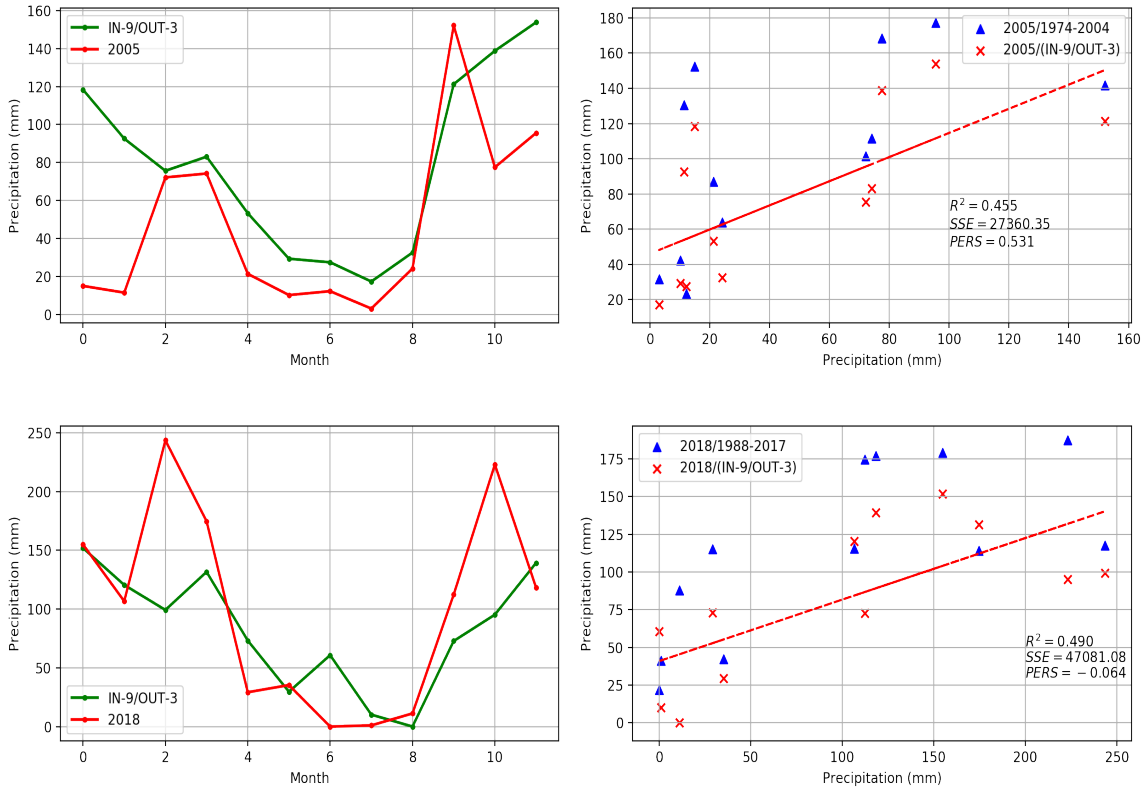


Figure A.12: Result of In-9/Out-3, *Encoder – Decoder LSTM*, PRCP + 1*mm* + Month. 2005(top) and 2018(below)

PRCP+1*mm*+MON+SEA				
Model	IN/OUT Setting	R^2	*SSE*	*PERS*
Multi – channel CNN	IN-24/OUT-3	0.583	47113.02	0.208
Vanilla LSTM	IN-24/OUT-3	0.492	66366.3	-0.116
Encoder – Decoder LSTM	IN-12/OUT-3	0.564	58601.65	0.015
CNN – LSTM Encoder – Decoder	IN-24/OUT-3	0.536	52236.02	0.122
ConvLSTM Encoder – Decoder	IN-6/OUT-3	0.611	54390.8	0.085

Table A.13: Results from the PRCP + 1*mm* + Month + Season dataset

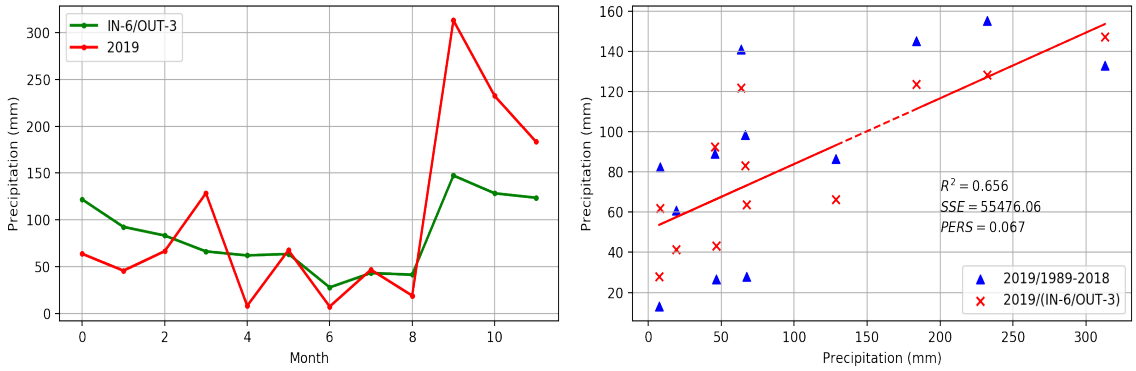


Figure A.13: Result of In-6/Out-3, *ConvLSTM Encoder – Decoder*, PRCP + 1*mm* + Month + Season

PRCP+1*mm*+MON+SEA					
	Model	IN/OUT Setting	R^2	SSE	PERS
2005	*ConvLSTM Encoder – Decoder*	IN-6/OUT-3	0.447	50517.05	0.134
2018	*ConvLSTM Encoder – Decoder*	IN-6/OUT-3	0.452	49478.85	-0.119

Table A.14: Result of the year 2005 and 2018, PRCP + 1*mm* + Season

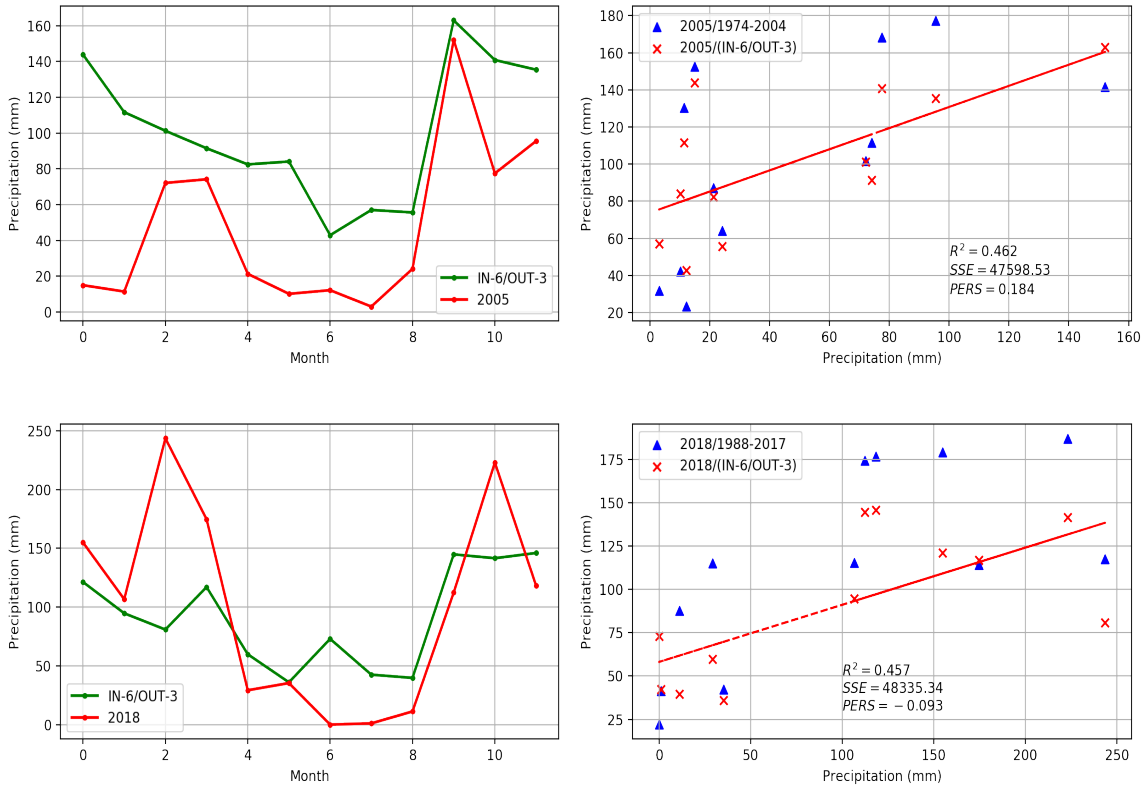


Figure A.14: Result of In-6/Out-3, *ConvLSTM Encoder – Decoder*, PRCP + 1*mm* + Month + Season. 2005(top) and 2018(below)

PRCP+NAO				
Model	IN/OUT Setting	R^2	*SSE*	*PERS*
Multi – channel CNN	IN-6/OUT-1	0.183	90301.06	-0.518
Vanilla LSTM	IN-12/OUT-1	0.312	78622.05	-0.322
Encoder – Decoder LSTM	IN-6/OUT-12	0.451	74188.25	-0.248
CNN – LSTM Encoder – Decoder	IN-12/OUT-12	0.450	81715.07	-0.374
ConvLSTM Encoder – Decoder	IN-8/OUT-3	0.333	156644	-1.634

Table A.15: Results from the PRCP + NAO dataset

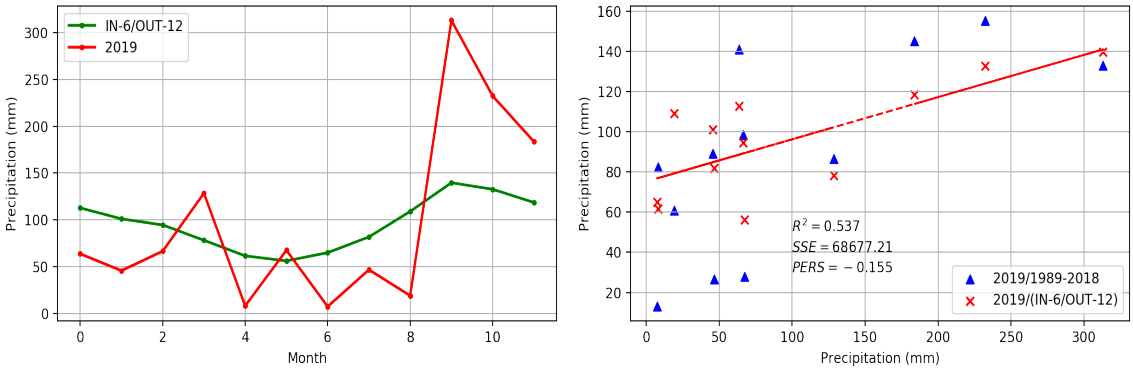


Figure A.15: Result of In-6/Out-12, *Encoder – Decoder LSTM*, PRCP + NAO

PRCP+1*mm*+SEA+NAO				
Model	IN/OUT Setting	R^2	*SSE*	*PERS*
Vanilla LSTM	IN-6/OUT-3	0.403	71074.42	-0.195
Encoder – Decoder LSTM	IN-12/OUT-3	0.427	67472.63	-0.135
CNN – LSTM Encoder – Decoder	IN-24/OUT-6	0.454	59418.54	0
ConvLSTM Encoder – Decoder	IN-24/OUT-3	0.429	66348.26	-0.116

Table A.16: Results from the PRCP + 1*mm* + Season + NAO dataset

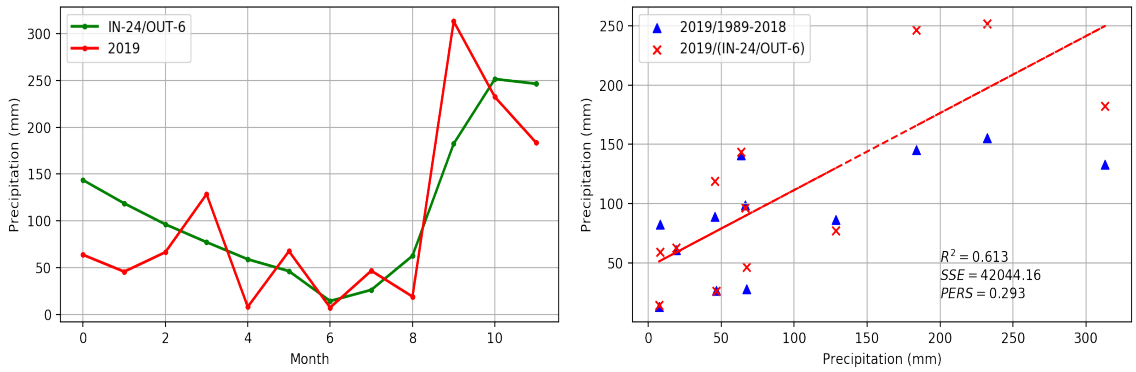


Figure A.16: Result of In-24/Out-6, *CNN – LSTM Encoder – Decoder*, PRCP + 1*mm* + Season + NAO

References

[1] WMO webteam, Feb 2019. Accessed: 2020-06-30. URL: https://public.wmo.int/en/programmes/global-climate-observing-system/essential-climate-variables.

[2] Camila Campos and Myriel Horn. *The Physical System of the Arctic Ocean and Subarctic Seas in a Changing Climate: Proceedings of the 2017 conference for YOUng MARine RESearchers in Kiel, Germany*, pages 25–40. 01 2018. doi:10.1007/978-3-319-93284-2_3.

[3] Arctic oscillation (ao), Jul 2020. Accessed: 2020-06-30. URL: https://www.ncdc.noaa.gov/teleconnections/ao/.

[4] The Mound of Sound, Jan 1970. Accessed: 2020-06-30. URL: https://the-mound-of-sound.blogspot.com/2015/03/coming-soon-great-warming-spurt.html.

[5] Climate models: Noaa climate.gov. Accessed: 2020-06-30. URL: https://www.climate.gov/maps-data/primer/climate-models.

[6] Xianli Zhu, Authors Clements, Jeremy Haggar, Alicia Quezada, and Juan Torres. *Technologies for Climate Change Adaptation – Agriculture Sector*. 01 2011.

[7] Kevin Hanley, Michael Cronin, and Paul Brady. Multiple event survival analysis on the risk of arterial oxygen desaturation during conscious dental sedation. 05 2015.

[8] Jiale Liu and Xinqi Gong. Attention mechanism enhanced LSTM with residual architecture and its application for protein-protein interaction residue pairs prediction. *BMC Bioinformatics*, 2019. doi:10.1186/s12859-019-3199-1.

[9] Dipesh Gautam, Nabin Maharjan, Rajendra Banjade, Jimba Lasang, Vasile Tamang, and Rus. Long short term memory based models for negation handling in tutorial dialogues. May 2018. doi:10.13140/RG.2.2.26250.36804.

[10] Tae Young Kim and Sung Bae Cho. Predicting residential energy consumption using CNN-LSTM neural networks. *Energy*, 2019. doi:10.1016/j.energy.2019.05.230.

[11] Marijn Stollenga. *Advances in Humanoid Control and Perception*. PhD thesis, Faculty of Informatics of the Università della Svizzera Italiana, 05 2016.

[12] Kanchan Sarkar. Relu:not a differentiable function: "why used in gradient based optimization?", May 2018. Accessed: 2020-06-17. URL: https://medium.com/@kanchansarkar/relu-not-a-differentiable-function-why-used-in-gradient-based-optimization-

[13] Jason Brownlee. How to use learning curves to diagnose machine learning model performance, Aug 2019. Accessed: 2020-06-17. URL: https://machinelearningmastery.com/learning-curves-for-diagnosing-machine-learning-model-performance/.

[14] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. http://www.deeplearningbook.org.

[15] Scott Fortmann-Roe. Bias and variance, Jun 2012. Accessed: 2020-06-17. URL: http://scott.fortmann-roe.com/docs/BiasVariance.html.

[16] Jason Brownlee. *Deep Learning for Time Series Forecasting: Predict the Future with MLPs, CNNs and LSTMs in Python*. Machine Learning Mastery, 2018, 1.4 edition, 2018.

[17] Caleb Strom. Difference between climatology and meteorology, Apr 2019. Accessed: 2020-06-17. URL: http://www.differencebetween.net/science/difference-between-climatology-and-meteorology/.

[18] Climate Prediction Center Internet Team. Introduction, May 2008. Accessed: 2020-06-30. URL: https://www.cpc.ncep.noaa.gov/data/teledoc/teleintro.shtml.

[19] J. Perlwitz, T. Knutson, J.p. Kossin, and A.n. Legrande. Ch. 5: Large-scale circulation and climate variability. climate science special report: Fourth national climate assessment, volume i. 2017. doi:10.7930/j0rv0kvq.

[20] Wes McKinney. Data Structures for Statistical Computing in Python. In Stéfan van der Walt and Jarrod Millman, editors, *Proceedings of the 9th Python in Science Conference*, pages 56 – 61, 2010. doi:10.25080/Majora-92bf1922-00a.

[21] James W. Hurrell. Decadal trends in the North Atlantic oscillation: Regional temperatures and precipitation. *Science (80-.).*, 269(5224):676–679, 1995. doi:10.1126/science.269.5224.676.

[22] Climate Prediction Center Internet Team. North atlantic oscillation (nao), Jan 2012. Accessed: 2020-06-30. URL: https://www.cpc.ncep.noaa.gov/data/teledoc/nao.shtml.

[23] Tim Osborn. Annual nao index (final value: 2019), 2017. Accessed: 2020-06-30. URL: https://crudata.uea.ac.uk/cru/data/nao/viz.htm.

[24] D.J. Easterbrook. Chapter 21 - using patterns of recurring climate cycles to predict future climate changes. In Don J. Easterbrook, editor, *Evidence-Based Climate Science (Second Edition)*, pages 395 – 411. Elsevier, second edition edition, 2016. URL: http://www.sciencedirect.com/science/article/pii/B9780128045886000215, doi:https://doi.org/10.1016/B978-0-12-804588-6.00021-5.

[25] NC State University. Global patterns: Pacific decadal oscillation. Accessed: 2020-06-30. URL: https://climate.ncsu.edu/climate/patterns/pdo.

[26] Weather, Jun 2018. Accessed: 2020-06-30. URL: https://public.wmo.int/en/our-mandate/weather.

[27] K. Hayhoe, J. Edmonds, R.e. Kopp, A.n. Legrande, B.m. Sanderson, M.f. Wehner, and D.j. Wuebbles. Ch. 4: Climate models, scenarios, and projections. climate science special report: Fourth national climate assessment, volume i. 2017. doi:10.7930/j0wh2n54.

[28] Gianluca Bontempi, Souhaib Ben Taieb, and Yann Aël Le Borgne. Machine learning strategies for time series forecasting. In *Lect. Notes Bus. Inf. Process.*, volume 138 LNBIP, pages 62–77, 2013. doi:10.1007/978-3-642-36318-4_3.

[29] Holger R. Maier and Graeme C. Dandy. Neural networks for the prediction and forecasting of water resources variables: A review of modelling issues and applications. *Environ. Model. Softw.*, 15(1):101–124, 2000. doi:10.1016/S1364-8152(99)00007-9.

[30] Mohini P. Darji, Vipul K. Dabhi, and Harshadkumar B. Prajapati. Rainfall forecasting using neural network: A survey. In *Conf. Proceeding - 2015 Int. Conf. Adv. Comput. Eng. Appl. ICACEA 2015*, 2015. doi:10.1109/ICACEA.2015.7164782.

[31] Vyacheslav Lyubchich, Nathaniel K. Newlands, Azar Ghahari, Tahir Mahdi, and Yulia R. Gel. Insurance risk assessment in the face of climate change: Integrating data science and statistics. *Wiley Interdiscip. Rev. Comput. Stat.*, 11(4):e1462, jul 2019. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/wics.1462, doi:10.1002/wics.1462.

[32] Sarah Afshin, Hedayat Fahmi, Amin Alizadeh, Hussein Sedghi, and Fereidoon Kaveh. Long term rainfall forecasting by integrated artificial neural network-fuzzy logic-wavelet model in karoon basin. *Sci. Res. Essays*, 6(6):1200–1208, 2011. doi:10.5897/SRE10.448.

[33] A. Belayneh, J. Adamowski, B. Khalil, and B. Ozga-Zielinski. Long-term SPI drought forecasting in the Awash River Basin in Ethiopia using wavelet neural networks and wavelet support vector regression models. *J. Hydrol.*, 508:418–429, 2014. doi:10.1016/j.jhydrol.2013.10.052.

[34] Kumar Abhishek, M.P. Singh, Saswata Ghosh, and Abhishek Anand. Weather Forecasting Model using Artificial Neural Network. *Procedia Technol.*, 4:311–318, jan 2012. URL: https://www.sciencedirect.com/science/article/pii/S221201731200326X, doi:10.1016/j.protcy.2012.05.047.

[35] P. Goswami and Srividya. A novel neural network design for long range prediction of rainfall pattern. *Curr. Sci.*, 70(6):447–457, 1996.

[36] C. L. Wu, K. W. Chau, and C. Fan. Prediction of rainfall time series using modular artificial neural networks coupled with data-preprocessing techniques. *J. Hydrol.*, 389(1-2):146–167, 2010. doi:10.1016/j.jhydrol.2010.05.040.

[37] Mithila Parmar, Aakash and Mistree, Kinjal and Sompura. Machine Learning Techniques For Rainfall Prediction : A Review. *Int. Conf. Innov. Inf. Embed. Commun. Syst.*, 2017.

[38] Xingjian Shi, Zhourong Chen, and Hao Wang. Convolutional LSTM Network. *Nips*, 2015-Janua:2–3, 2015. URL: http://papers.nips.cc/paper/5955-convolutional-lstm-network-a-machine-learning-approach-for-precipitati
arXiv:1506.04214, doi:[].

[39] Kabir Rasouli, William W. Hsieh, and Alex J. Cannon. Daily streamflow forecasting by machine learning methods with weather and climate inputs. *J. Hydrol.*, 414-415:284–293, 2012. doi:10.1016/j.jhydrol.2011.10.039.

[40] I. Ching Chen and Shueh Cheng Hu. Realizing specific weather forecast through machine learning enabled prediction model. In *ACM Int. Conf. Proceeding Ser.*, pages 71–74, 2019. doi:10.1145/3341069.3341084.

[41] Sebastian Scher and Gabriele Messori. Predicting weather forecast uncertainty with machine learning. *Q. J. R. Meteorol. Soc.*, 144(717):2830–2841, oct 2018. URL: https://www.engineeringvillage.com/share/document.url?mid=inspec{_}54e65a8116b52182912M7d5d10178163167{&}database=ins, doi:10.1002/qj.3410.

[42] Jinglin Du, Yayun Liu, Yanan Yu, and Weilan Yan. A Prediction of Precipitation Data Based on Support Vector Machine and Particle Swarm Optimization (PSO-SVM) Algorithms. *Algorithms*, 10(2):57, may 2017. URL: http://www.mdpi.com/1999-4893/10/2/57, doi:10.3390/a10020057.

[43] Bahram Choubin, Gholamreza Zehtabian, Ali Azareh, Elham Rafiei-Sardooi, Farzaneh Sajedi-Hosseini, and Özgür Kişi. Precipitation forecasting using classification and regression trees (CART) model: a comparative study of different approaches. *Environ. Earth Sci.*, 77(8), 2018. doi:10.1007/s12665-018-7498-z.

[44] Sebastian Scher and Gabriele Messori. Weather and climate forecasting with neural networks: using GCMs with different complexity as study-ground. *Geosci. Model Dev. Discuss.*, 2019. doi:10.5194/gmd-2019-53.

[45] WMO.INT Webteam. World meteorological organization. Accessed: 2020-06-17. URL: https://www.wmo.int/pages/prog/www/ois/Operational_Information/Publications/Congress/Cg_XII/res40_en.html.

[46] Data rescue and archives, Jun 2018. Accessed: 2020-06-15. URL: https://public.wmo.int/en/our-mandate/what-we-do/observations/data-rescue-and-archives.

[47] National Climatic Data Center, Noaa, and Department of Commerce. Global surface summary of the day - gsod, Apr 2019. Accessed: 2020-06-15. URL: https://data.nodc.noaa.gov/cgi-bin/iso?id=gov.noaa.ncdc:C00516#.

[48] Global surface summary of the day - gsod, Feb 2019. Accessed: 2020-06-15. URL: https://catalog.data.gov/dataset/global-surface-summary-of-the-day-gsod.

[49] Arctic oscillation (ao), May 2020. Accessed: 2020-06-15. URL: https://www.ncdc.noaa.gov/teleconnections/ao/.

[50] North atlantic oscillation (nao), May 2020. Accessed: 2020-06-15. URL: https://www.ncdc.noaa.gov/teleconnections/nao/.

[51] Pacific decadal oscillation (pdo), May 2020. Accessed: 2020-06-15. URL: https://www.ncdc.noaa.gov/teleconnections/pdo/.

[52] Sea ice and snow cover extent, May 2020. Accessed: 2020-06-15. URL: https://www.ncdc.noaa.gov/snow-and-ice/extent/.

[53] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol

Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org. URL: https://www.tensorflow.org/.

[54] François Chollet et al. Keras. https://keras.io, 2015.

[55] The pandas development team. pandas-dev/pandas: Pandas, Feb 2020. URL: https://doi.org/10.5281/zenodo.3509134, doi:10.5281/zenodo.3509134.

[56] Travis E. Oliphant. NumPy: A guide to NumPy. USA: Trelgol Publishing, 2006. URL: http://www.numpy.org/.

[57] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[58] João Gama, André Ponce de Leon Carvalho, Katti Faceli, Ana Carolina Lorena, and Márcia Oliveira. *Extração de Conhecimento de Dados Data Mining*. Edições Sílabo, Lda, Third edition, Sep 2017.

[59] D. M. Hawkins. *Identifications of Outliers*. Chapman and Hall, 1980.

[60] C. C. Aggarwal. *Outlier Analysis*. Springer, 2017.

[61] J. W. Tukey. *Exploratory Data Analysis*. Addison-Wesley, 2017.

[62] Yoshua Bengio. Deep Learning of Representations for Unsupervised and Transfer Learning. In *JMLR Work. Conf. Proc.*, 2011.

[63] Pierre Luc Carrier and Kyunghyun Cho. Lstm networks for sentiment analysis¶, Jun 2018. Accessed: 2020-06-17. URL: http://deeplearning.net/tutorial/lstm.html.

[64] Sebastian Ruder. An overview of gradient descent optimization algorithms, 2016. arXiv:1609.04747.

[65] TH Lee. Loss Functions in Time Series Forecasting. *Univ. Calif.*, 2007.

[66] Dinesh Kumawat. 7 types of activation functions in neural network, Aug 2019. Accessed: 2020-06-17. URL: https://www.analyticssteps.com/blogs/7-types-activation-functions-neural-network.

[67] Jason Brownlee. Gentle introduction to the bias-variance trade-off in machine learning, Oct 2019. Accessed: 2020-06-17. URL: https://machinelearningmastery.com/gentle-introduction-to-the-bias-variance-trade-off-in-machine-learning/.

[68] Jason Brownlee. How to normalize and standardize time series data in python, Aug 2019. Accessed: 2020-06-17. URL: https://machinelearningmastery.com/normalize-standardize-time-series-data-python/.