POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

An Evaluation of Data-Driven Interpretable Methods to Detect Tropical Cyclones

Tesi di Laurea Magistrale in
Computer Science and Engineering
Ingegneria Informatica

Author: **Marco Adriano Ferrero**

Student ID: 977577
Advisor: Prof. Francesco Amigoni
Co-advisors: Prof. Federico Cerutti, Prof. Letizia Tanca, Davide Azzalini
Academic Year: 2023-24

Abstract

Tropical cyclones are atmospheric low-pressure systems that develop at the tropics. They are characterized by powerful winds, heavy rainfall, and storm surges. Due to their geographic extent and destructive power, cyclones can cause extensive environmental damage, especially when they reach coastlines and inhabited areas.

Over the years, the increasing attention to these phenomena has led to the development of systems that can forecast various aspects of cyclones, such as their formation, track, intensification, and related extreme events. Many of the latest techniques rely on Machine Learning algorithms that can automatically learn the relationships between meteorological data and cyclone activity. While these methods are effective in predictions they make, they may be inadequate in explaining and describing their inner behavior, which is often required by human operators.

This thesis aims to provide insight into machine learning methods applied to tropical cyclone forecasting, focusing on the interpretability of their predictions. Specifically, this work analyzes two approaches involving modeling and data analysis. The black-box models are characterized by their opacity concerning the relationship that binds the input to the output, while the white-box models, on the contrary, describe clearly and explicitly the implemented decision process. Some of the black-box models implemented rely on Gradient Boosting (GB) and Long Short-Term Memory Networks (LSTMs), and the Local Interpretable Model-Agnostic Explanations (LIME) provide details on predictions of these methods. White-box models, on the other hand, include Decision Trees and Bayesian Rule Lists, which demonstrate effectiveness in describing the presence of a cyclone and achieving mutually consistent results. The experimental results show that black-box models lack a detailed description of the predictive processes, however, they effectively provide accurate and reliable forecasts. In contrast, white-box models provide rule-based predictions that detail the activity of a TC, although they are not as effective as black-box models at generalizing the forecasting process.

Keywords: Tropical Cyclones, Machine Learning, Deep Learning, Explainable Artificial Intelligence, White-Box, Black-Box.

Abstract in lingua italiana

I cicloni tropicali sono sistemi di bassa pressione atmosferica che si sviluppano ai tropici, caratterizzati da forti venti e da piogge intense. A causa dell'estensione geografica e della loro potenza distruttiva, questi fenomeni possono provocare danni ambientali di grande portata. In particolare quando raggiungono le coste e le aree abitate, i cicloni sono spesso la causa di devastazione e di disagi socio-economici.

Nel corso degli anni, lo studio di questi fenomeni ha portato allo sviluppo di sistemi in grado di prevedere con buona precisione diversi aspetti dei cicloni, come la genesi, lo spostamento, l'intensificazione e gli eventi estremi correlati. Molte delle tecniche più recenti si basano su algoritmi di Machine Learning, in grado di apprendere automaticamente le relazioni che esistono tra i dati meteorologici e l'attività ciclonica. Nonostante questi metodi si dimostrino efficaci in molti contesti, allo stesso tempo risultano inadeguati nel fornire spiegazioni e descrizioni su come essi stessi forniscono le proprie previsioni.

In questo contesto, questo lavoro di tesi vuole fornire un approfondimento sui metodi di Machine Learning che possono essere usati per prevedere in maniera esplicativa i cicloni tropicali. In particolare, il lavoro è svolto contrapponendo due diversi approcci nel campo della modellazione e dell'analisi dei dati. I modelli black-box, che sono caratterizzati dalla loro opacità rispetto alla relazione che lega l'input con l'output, e i modelli white-box che, al contrario, spiegano in modo chiaro ed esplicito il processo decisionale implementato. Alcuni dei modelli black-box sono realizzati tramite Gradient Boosting (GB) e Long Short-Term Memory Networks (LSTMs), e la descrizione delle previsioni prodotte da questi metodi è fornita tramite una tecnica denominata Local Interpretable Model-Agnostic Explanations (LIME). I modelli white-box invece comprendono gli alberi decisionali e le liste di regole Bayesiane. I risultati sperimentali mostrano come i modelli black-box siano adeguati a fornire previsioni precise ma difficilmente interpretabili. Al contrario, i modelli white-box descrivono efficacemente la presenza dei cicloni con regole logiche. Tuttavia non forniscono la stessa abilità predittiva dei modelli black-box.

Parole chiave: Cicloni Tropicali, Machine Learning, Deep Learning, Explainable Artificial Intelligence, White-Box, Black-Box.

Contents

1 | Introduction

Tropical Cyclones (TCs) are some of the most extreme weather phenomena that can occur on Earth, characterized by strong winds, torrential rains, and storm surges. They are defined as intense low-pressure systems that gain energy from warm water near the equator and rotate over tropical oceans. A tropical cyclone is typically between 200 km and 500 km in diameter but can exceed 1000 km. Their impact on coastlines and population centers can be devastating, causing valuable environmental and economic damages. Over the past 50 years, these phenomena have contributed to 1945 environmental disasters that have killed more than 700,000 people and caused more than 1.4 billion in economic damage[1].

The future impact of these phenomena is expected to be even more significant, considering how rapidly climate change contributes to the development of extreme events. Future projections indicate that the greenhouse effect will cause a 2-10% increase in the average intensity of cyclones by 2100. Additionally, it will bring a 20% increase in precipitation rates within 100 km of the storm center [40]. These considerations underline the importance of providing the appropriate tools to predict TCs and to acquire the necessary knowledge to prevent them from affecting human lives.

TCs forecasting aims to detect the occurrence and development of cyclonic activity in a particular area of our planet. Throughout the last century and up to the present day, climatologists and scientists have devised various tools to address this problem. However, due to the extreme complexity and irregularity of these phenomena, it is still difficult to fully resolve some of the open questions regarding the genesis, intensity, track, and extreme events associated with Tropical Cyclones.

In this context, Machine Learning (ML) techniques, which aim to learn relationships and patterns within data, have proven to be highly impactful in contributing to a deeper understanding of the topic. In particular, the use of data collected over more than 50 years by the world's leading weather organizations has enabled the creation of increasingly powerful models capable of improving the predictive skills of forecasting systems.

[1] `https://wmo.int/topics/tropical-cyclone`

Although ML-based solutions can provide highly accurate predictive results, their behavior is often difficult to interpret. Usually, these models are very complex, and their dynamics are not transparent to the users. Even experts who use them may not fully understand the processes and evaluations that bring to a given prediction. This thesis aims to demonstrate the potential contributions of Machine Learning methods in developing *interpretable* systems capable of expressing rules and explanations for tropical cyclone detection.

The experiments in this thesis are based on weather data provided by international organizations. The European Center for Medium-Range Weather Forecast (ECMWF) directly provided to Politecnico di Milano some datasets used in this work to simplify the preprocessing of relevant features. These datasets were previously used in a related work [21]. To further explore the topic and build a more complete collection, some publicly available datasets were also included. The ECMWF's fifth public reanalysis system (ERA5) provided meteorological data describing the oceanic and atmospheric drivers. Additionally, the National Oceanic and Atmospheric Administration (NOAA) provided historical archival data on tropical cyclones with the public International Best Track Archive for Climate Stewardship (IBTrACS).

Several ML models have been implemented and evaluated to determine which techniques are easily interpretable while providing reasonably accurate predictions. The models implemented fall into two categories, grouped according to their inherent ability to provide interpretable predictions. The former are black-box models, which include neural networks and ensemble-based methods, while the latter are white-box models, which include Decision Trees (DTs) and Bayesian Rule Lists (BRLs). According to the experimental results, black-box models effectively provide accurate predictions. However, the interpretability of these methods is limited when dealing with particularly complex models. In contrast, white-box methods provide deep descriptions of the prediction process, although they are limited in their ability to generalize their behavior when high-dimensional data are involved.

The contents of the following chapters in this thesis are:

- Chapter 2 describes the current state of the art on Machine Learning methods applied to TCs forecasting, analyzing the main drivers that influence the problem and the possible solutions implemented in related work.
- Chapter 3 introduces the datasets involved in the experiments and the preprocessing solutions adopted. Additionally, it defines the TCs forecasting as a Machine Learning problem and describes the evaluation metrics used to assess the models'

performance in the following chapters.

- Chapter 4 describes the development and evaluation of the implemented black-box models. In particular, it focuses on a specific interpretability technique that is applied to provide explanations to the predictions produced by these models.
- Chapter 5 focuses on the description and evaluation of white-box models, i.e., models that inherently provide a high degree of interpretability. These models are then compared to see if different algorithms provide similar patterns to identify TCs.
- Chapter 6 summarizes the goals of this thesis, focusing on the open questions and limitations in tropical cyclone detection. In addition, suggestions for possible future work are presented.

2 | State of the Art

Tropical Cyclones (TCs), also known as typhoons or hurricanes, are powerful and potentially destructive meteorological phenomena that produce high winds, heavy rainfall, and storm surge into a low-pressure system [80]. The majority of TCs arise from at low latitudes in both hemispheres, in the so-called tropical zone, where climatic conditions are favorable for the development of these anomalies [61].

The genesis of TCs is characterized by pre-existing atmospheric disturbances originating over warm ocean waters, typically around 26 Celsius degrees, gaining strength from the heat energy of the sea surface. Water from the sea surface evaporates and the rising air warms the disturbance's core by latent heat release and direct heat transfer, lowering atmospheric pressure in the center. This pressure drop leads to increased surface winds, intensifying vapor and heat transfer, reinforcing each other in a positive feedback loop, and contributing to the cyclone's formation and development. The Coriolis effect of the Earth's rotation gives a spin to the developed system, leading to the deflection of winds. As a result, TCs rotate in a counterclockwise direction in the Northern Hemisphere and, on the opposite, rotate in a clockwise verse in the Southern one [80].

The intensity of winds and adverse weather conditions caused by tropical cyclones can have devastating effects on coasts, destroying population centers and causing the death of many people. For these reasons, scholars and researchers have long been devoted to the study and prediction of these phenomena, describing their physical structure and interactions with the atmosphere over the last century. However, the problem of predicting the formation and evolution of a TC remains one of the most complex in meteorology, with many aspects still to be defined [24].

Traditional forecasting technologies, based on numerical models and statistical methods, have made considerable progress. They are often faced with challenges in recording the complex dynamics of TCs, leading to the exploration of Machine Learning as a complementary and potentially transformative approach to increasing predictive accuracy and reliability. Machine Learning has been effective in achieving significant improvements in forecasts for various aspects of TCs such as genesis, intensity, tracking, and disaster

impact forecasts [17].

The availability of several useful techniques to explain Machine Learning predictions and describe the scenario under consideration can make these methods extremely effective in supporting authorities who have to make crucial decisions in the defence of populations affected by these potential disasters [52].

2.1. Tropical Cyclone Drivers and Indices

2.1.1. Local Drivers

Assessing TC activity is a very complex problem, as this phenomenon is influenced by many local or global climate conditions and atmospheric or oceanic variables. *Local drivers* are considered to be all the meteorological variables that describe a specific zone in which the cyclone manifests its activity. Examples of potential drivers to be considered in the analysis of cyclone detection include vorticity, sea surface temperature, surface pressure, and humidity. These variables are closely related to the physical description of this anomaly in the area where its activity occurs.

Studying the interaction of these variables and the relationships that are present in cyclonic activity has over time allowed the development of numerical indices that can describe the characterization of TCs. In particular, the Genesis Potential Index (GPI) and Tropical Cyclone Genesis Index (TCGI) focusing on the early stage of the cyclone activity, aim to quantify the potential genesis of a TC. GPI [26] includes in its formulation absolute vorticity, relative humidity at 700hPa, potential intensity, and the magnitude of the vector shear from 850 to 200 hPa, to define the climatological distribution and seasonal variations of TCs [3]. TCGI [69], in addition to GPI, takes into consideration the Sea Surface Temperature (SST) anomaly concerning the mean tropical SST (30 °N – 30 °S), and the latitude where activity is analyzed, composing an index in the form of a Poisson regression model.

Equations (2.1), and (2.2) show respectively the GPI and the TCGI indicators:

$$GPI = \left|10^5\eta\right|^{3/2} \left(\frac{H}{50}\right)^3 \left(\frac{V_{pot}}{70}\right)^3 (1 + 0.1V_{shear})^{-2}], \tag{2.1}$$

$$TCGI = \exp\left(-4.47 + 0.5\eta + 0.05H + 0.63RSST - 0.17V_{shear} + \log\cos\phi\right). \tag{2.2}$$

where η is the absolute vorticity at 850 hPa, H the relative humidity at 600 hPa, V_{pot} the maximum potential wind speed, $RSST$ the sea surface temperature anomaly, V_{shear} the

wind shear between 200 hPa and 850 hPa, and ϕ the latitude.

Other relevant indices assess the energy and intensity that an already-formed cyclone can reach, to give a description of the destructive potential of this phenomenon. These include the Maximum Potential Index (MPI), which gives an upper bound of the theoretical maximum intensity that a TC can reach in a given environment [25], and the Accumulated Cyclone Energy (ACE), a wind energy index, defined as the sum of the squares of the estimated 6-hourly maximum sustained wind speed [2].

The aforementioned indicators are generally suitable for making predictions based on the climatic conditions to which they have been tuned. However, they may not be very robust when considering future warmer climate scenarios, and their ability to describe this phenomenon may degrade [13]. Additionally, they are not well-suited for detecting interannual variations or for application to different basins [56].

2.1.2. Global Drivers

In addition to analyzing these phenomena using local-based indicators, it is important to take into account large-scale variables that might impact TC activity. These *global drivers* include factors that influence climate patterns across the world, particularly in tropical and equatorial zones. Some of the most relevant are:

- Madden-Julian Oscillation (MJO) [81] is one of the most influential tropical intra-seasonal fluctuations (30 to 60 days) and has a significant contribution to extreme events in different basins around the globe. MJO modulates relevant activities that have important influences on the TC lifecycle, such as convective anomalies, vertical wind shear, mid-level moisture, vertical motion, and sea surface pressure [38].
- El Niño–Southern Oscillation (ENSO) is a quasi-periodic anomaly that affects the winds and climate of tropical areas by alternating a heating phase of surface waters (El Niño) with a cooling phase (La Niña). It originates in the tropical Pacific and acts on different climate basins, having an impact on vertical wind shear, humidity, low-level vorticity, and the strength and position of subtropical highs [20, 49, 55].
- Convectively Coupled Equatorial Waves (CCEWs) are atmospheric disturbances that play a crucial role in the tropical climate system. These waves are characterized by the coupling of convective and atmospheric circulation patterns along the equator, partially controlling the tropical rainfall variability [36, 51].

2.2. Machine Learning In TC Detection

Tropical cyclones are exceedingly complex atmospheric anomalies with intricate dynamic processes that are influenced by several meteorological and oceanic elements. Over time, the notion of employing equation-based index has shown various limitations, mostly linked to the difficulties of modeling such large systems, with simple indicators.

Therefore, the use of artificial intelligence-based methods has opened up new possibilities in research on the topic, showing general improvements. These data-driven approaches have enabled scholars to gain predictive performances in the four main open problems related to cyclone forecast: genesis, intensity, tracking, and related weather destructive events [74].

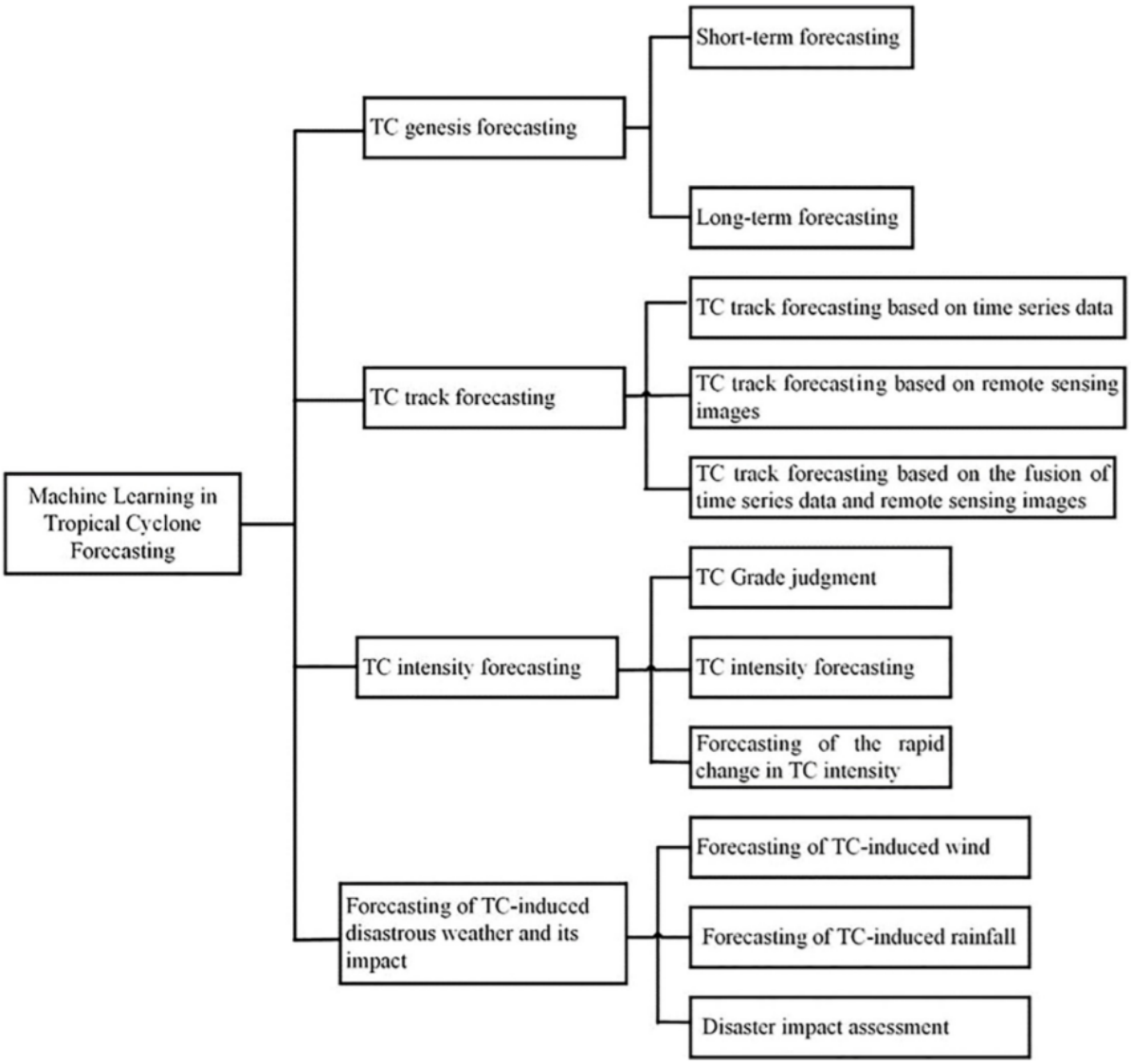


Figure 2.1: TCs forecasting with ML methods [74].

2.2.1. TC Genesis Forecasting

Tropical cyclogenesis is defined as TC development from a pre-existing tropical disturbance. Various statistical and Machine Learning techniques, combined with more traditional model-based methods, allow good prediction of these events, thus, the ability to predict their occurrence is still an open problem.

The problem is mainly analysed following two different approaches, depending on the time window considered:

- Short-term forecasting considers the problem of predicting the emergence of a cyclone from a pre-existing tropical anomaly in the short-term (usually hours or days). The Machine Learning problem turns out to be a supervised classification problem to label samples, whether a cyclone is likely to form in the next few hours or days.
- Long-term forecasting generally refers to the analysis of large-scale data to predict the frequency of cyclone formation over long periods (usually seasons or months). In Machine Learning terms, the problem can be thought of as a regression problem, where the forecast produces a numerical value indicating the rate at which this phenomenon is likely to recur over the interval under consideration.

Various methods have been applied to solve the short-term forcasting problem, ranging from traditional Machine Learning systems that exploit tabular data, to the use of more complex deep learning techniques based on dense or convolutional layers.

According to Zhang et al. [83], one of the simplest methods to approach the problem is through the use of decision trees. Specifically for this work, trees are generated through the C4.5 algorithm, which is based on the concept of information entropy, choosing at each intermediate node the attribute that best separates the dataset into binary classes. Using several atmospheric and oceanic variables as reference datasets, related to the tropical disturbances already present in the North West Pacific basin, this model achieve good performance in classifying samples into the classes "TC formation" and "undeveloped anomalies", with an overall accuracy over the 2004-2010 test set of 81.7 %. This decision tree is a very simple model, with a tree depth equal to five. Its structure facilitates interpretation by clearly defining the variables and their respective threshold values that are relevant in predicting the phenomenon. Effective variables include relative vorticity at 800 hPa, SST, precipitation rate, mean divergence between 1000 and 500 hPa, and air temperature at 300 hPa.

Zhang et al. [82] provide further insight into the topic, analysing how different models behave in predicting the development of a Mesoscale Convective System (MCS) into a

tropical cyclone, evaluating performance at different times and assessing the importance of the variables used. The dataset employed for this work includes 13 environmental variables useful for describing MCSs and TCs. The compared models cover some of the most widely used traditional Machine Learning models, including linear models such as linear logistic regression and naïve Bayes, nonlinear models such as decision trees, k-nearest neighbours, support vector machine and ensemble models such as AdaBoost. Results suggests that AdaBoost is the most robust ML model in this context, achieving a F1-score of 97.2 % and highliting the low-level vorticity and the GPI as the most effective predictors for the problem.

Additional progress was made using deep learning-oriented approaches, as in the work of Matsuoka et al. [53], in which a binary classifier was implemented using Convolutional Neural Networks (CNNs) to categorise 2D data into "developing TCs and their persecutors" or "non-developing depressions" classes. The dataset used in this work includes 20 years of simulated Outgoing Longwave Radiation (OLR) and the model is tested over different basins, seasons and lead times. The model implemented is composed of four convolutional layers with 64x64 input shape attached to a classifier with three fully connected layers. The results achieved with this method show a Probability of Detection (POD) around 80 % and a False Alarm Ratio (FAR) below 50 %. Unlike the techniques described above, the model presented in this paper lacks interpretation due to its deep nature, which does not provide a good description of results.

Regarding the long-term prediction problem, an example of such work is [82], in which the authors applied three different Artificial Neural Networks (ANNs) to obtain a seasonal prediction on the frequency of TCs in the northern Indian Ocean area during the monsoon season. The Multi Layer Perceptrons (MLP) model, in particular, was found to be better than the others, obtaining good results on a test set including data from 2003 to 2013, with root mean-square error (RMSE) values of 0.21 and an agreement index of 96 %.

2.2.2. TC Tracking Forecasting

Tracking the progress of a tropical cyclone is of paramount importance to predict the impact this event may have in the short term if it were to reach the coasts. Early studies were heavily focused on the idea that the cyclone being tracked had very similar motion characteristics to a cyclone previously present in the same area and with similar properties. The first method developed was HURRAN [34], which became operational in the NHC (National Hurricane Center) in 1969. However, its performance proved to be poor compared to climatology and persistence methods developed in the next few

years, such as CLIPER [59], which derived forecast equations using linear regression and produced forecasts up to 72 hours [17].

In recent decades, numerous advances have been made on the topic, taking advantage on the large amount of data that has been accumulated over the years, improving traditional forecasting techniques, and developing new statistical-dynamical models that also employ modern Machine Learning techniques [65].

Due to the complexity and non-linearity of physical processes in tropical cyclones, several models based on nonlinear algorithms have been implemented in different studies, such as artificial neural networks and Support Vector Machines (SVMs). An early approach to the use of neural networks is the work of Zhang [73] in which a fully connected deep model is trained using historical data from the previous 20 years, considering as relevant features only the geographical coordinates of cyclones, and the cyclone position in the next 24 hours as target of the classification problem.

Further progress has been made by adapting models that are better suited to recognize time series dependencies, such as the work of Alemany [6] in which a Recurrent Neural Network (RNN) was implemented with this purpose. Wind speed and surface pressure are considered as additional predictors and a fine grid is employed to reduce typical truncation errors. The results obtained from this study demonstrate competitive results with forecasting systems currently in use, providing up to 120 hours of lead time forecasts.

Among the works based on deep learning models, Deep-Hurricane-Tracker [37] appears to be one of the most interesting and comprehensive in this field. In this study, a deep model based on convolutions and Long Short-Term Memory (LSTM) layers has been proposed to track and predict hurricane trajectories using large-scale climate data, composed of time series and remote sensing images. The peculiarity of this model lies in its ability to learn the spatial distribution using the convolutional part of the model and to capture temporal dependencies by exploiting the LSTM layers. The results of experiments conducted on this model show that at three hours in advance, the prediction error is about 30 km, while increasing the time window, from 6 to 15 hours, the average error is approximately between 140 and 170 km.

A different approach was followed in the work of Tan et al. [68], in which a nonlinear ensemble-based method known as the Gradient Boosting Decision Tree (GBDT) was evaluated and compared to the CLIPER model, showing substantial improvements. Experiments on the model show its ability to predict TC trajectories at three different prediction intervals (24 h, 48 h, and 72 h), with significant improvements compared with the climatology and persistence method, obtaining prediction errors of 138, 264, and 353

km, respectively.

The various studies cited above show that Machine Learning models are suitable to be a guidance for the problem of forecasting the trajectory over the short-term. However, the main difficulties still lay in their ability to accurately predict TC paths over the long-term and in obtaining a model that can be applied to all the different basins where cyclones normally develop.

2.2.3. TC Intensity Forecasting

Forecasting the intensity of tropical cyclones has always been a prime concern in researches on these phenomena, as their power determines the potential risks to which populations may be exposed. The most widely used indicator to assess the intensity of these events is the maximum sustained wind, which corresponds to the wind at 10 metres above sea level, with a distance from the eyewall defined as the maximum wind radius (RMW).

Considering this as the most relevant parameter, several scales classify tropical cyclones into different categories, usually from one to five, defining their hazard index, in each specific basin. Among the best known, the Saffir-Simpson scale is used by the authority that monitors hurricanes in the Atlantic and Central-Eastern Pacific, defining a tropical storm as the event whose winds exceed 17 m/s and tropical cyclones events when winds are above 33 m/s. A complete description of this scale is shown in Table 2.1, along with the corresponding wind speed threshold for each of the TC categories.

Saffir-Simpson Scale

	m/s	**knots**	**mph**	**km/h**
Category 5	≥ 70	≥ 137	≥ 157	≥ 252
Category 4	58-70	113–136	130–156	209–251
Category 3	50–58	96–112	111–129	178–208
Category 2	43–49	83–95	96–110	154–177
Category 1	33–42	64–82	74–95	119–153
Tropical Storm	18–32	34–63	39–73	63–118
Tropical Depression	≤ 17	≤ 33	≤ 38	≤ 62

Table 2.1: 1-minute maximum sustained wind.

For over three decades, the Dvorak intensity estimation technique (1970) has been consid-

ered one of the most effective ones in TCs intensity forecasting. This empirical method, using satellite, imagery, and infrared data for its assessment, relies on a comprehensive analysis of cloud patterns, taking into account four key geophysical properties: vorticity, vertical wind shear, convection, and core temperature [72]. This method's significance becomes pronounced in scenarios where direct observational data is limited, especially in remote oceanic regions where collecting local data is extremely challenging. Although this method was considered to be extremely important for many years, a significant downside relies in its strong dependency to subjective biases to deliver accurate forecast findings.

Some of the ML-based works related to the subject are strongly influenced by Dvorak's idea of pattern recognition on satellite data. An example is the work of Pradhan et al. [63], in which a deep convolutional neural network model classifies cyclone images into the Siffer-Simpson scale categories, showing improvements in accuracy and root-mean-square error.

A similar deep learning approach is described in Chen et al. [16], where a hybrid model is proposed, capable of providing a more accurate representation of the spatio-temporal correlations of atmospheric and oceanic variables by combining convolutional and LSTM layers. The published results indicate significant improvements compared to traditional forecasting models and previous statistical and ML methods that are used by many organizations.

Other possibilities are considered with the use of RNNs in the work of [62], using TC intensity and tracking data collected in the Western North Pacific since 1949 to train a recurrent model that aims to estimate the value of maximum wind in the cyclone. This proposed method obtained an error of 5.1 m/s in 24 h prediction, which is better than some dynamical models widely used.

In [19], a traditional Machine Learning technique is implemented, with the construction of a Decision Tree (DT) model to forecast cyclone intensity change. The study highlights the importance of the Ocean Coupling Potential Intensity Index (OC-PI) in the decision tree construction. The OC-PI is an index calculated using pre-tropical cyclone averaged ocean temperatures from the surface down to 100 meters. The DT structure demonstrates how this indicator is strongly relevant to correctly classifying 24-hours tropical cyclone strength fluctuations.

2.2.4. TC Weather Forecasting

A significant aspect in the study of TCs is the ability to forecast their meteorological behaviors and impact on atmospheric anomalies. Specifically, many works have focused

on predicting two of the most dangerous climate threats for economic and life damage: TC-induced wind and TC-induced rainfall.

Wei [76] illustrates how kernel-based Support Vector machines for Regression (SVR) are able to predict wind speeds during cyclonic activities on offshore islands near Taiwan. Several kernel functions were examined, and the Pearson VII universal kernel was indicated to be the most accurate with forecast horizons ranging from one to six hours. While these results are described as promising, they may have a significant bias related to the small basin analyzed, therefore this model should be considered reasonably specific and unlikely to have similar performance on larger basins.

A different approach is instead considered in the work of Snaiki and Wu [67], where a knowledge-enhanced deep learning algorithm is used to simulate the wind field inside TC boundary layers. Considering the prior knowledge of the tropical cyclone boundary-layer wind, based on several state-of-the-art semi-empirical equations, only a small number of training samples are needed to successfully train the model, achieving accurate and efficient results.

Other studies aim to predict rainfall conditions inside a TC event, e.g., Lin and Chen [48] developed a fully connected neural network with two hidden layers as one of the first attempts to assess the problem. The model is evaluated using eight typhoon characteristics and improved with additional spatial rainfall information. Considering these features as inputs, the forecasting model produced reasonable forecast outcomes.

More recent work, such as Young and Liu's [79], has resulted in further progress, as a hybrid model was created by combining a physically-based model (HEC-HMS) with an Artificial Neural Network (ANN). In this study, hourly runoff discharge data from seven heavy rainfall events are collected to train and validate the model. Experiments on the training set indicated that the developed hybrid model outperformed the HEC-HMS model and the single ANN.

2.3. Discussion on Explainability

Artificial intelligence techniques have contributed to significant advances in weather forecasting and are widely used in modern systems. Many Machine Learning techniques are capable of learning complex nonlinear behaviors, which is extremely useful in predicting anomalies such as tropical cyclones. However, the complexity of these models makes it difficult to explain their predictions, which represents a barrier to adopting Machine Learning in a variety of scenarios.

Professionals frequently refer to ensemble techniques or deep models like neural networks as black-boxes due to the difficulty in understanding their real behavior while providing a prediction. Generally, testing models with data is the only way to measure performance quality. However, according to Doshi-Velez and Kim [22], "the problem is that a single metric, such as classification accuracy, is an incomplete description of most real-world tasks" as it does not provide a detailed explanation on how a model generates predictions.

In recent years, a new area of AI called Explainable Artificial Intelligence (XAI) aims to deepen the issue of interpretability in Machine Learning models through techniques that can reveal new aspects of the knowledge gained from the data [8]. There are many scenarios where a detailed explanation and justification of a model's prediction is essential, for example in the medical or meteorological field, where crucial decisions are made by experts and affect other people's lives [66].

Two different approaches can be considered to gain interpretability in an ML system. The first one relies on the use of simple models that are by their nature explainable and provide the necessary transparency to be interpreted by humans, such as simple decision trees or rule lists. The second method involves the application of post-hoc techniques on previously trained black-box models to extract explainable outcomes from the opaque predictions produced [23].

The purpose of this thesis is to investigate this topic in the context of tropical cyclone detection, developing and comparing different Machine Learning models that offer explanations for the forecasts provided.

3 | Problem Definition

Forecasting tropical cyclones is a challenging task that can benefit from various machine-learning techniques. Due to the complex dynamics of cyclone systems, it is difficult to identify consistent patterns across different events. As a result, there is no unique machine-learning technique universally recognized as the most effective for addressing this complexity. Multiple approaches have proven value over the past decades, with varying degrees of accuracy and training efficiency. Assuming the learning problem as the enhancement of forecasting ability by training experience on data, we can consider data collection as a critical factor for performance improvement. Therefore, obtaining sufficient relevant data and selecting appropriate features are essential to correctly address this problem and provide the predictive models with the necessary information.

Supervised learning is the most commonly used Machine Learning paradigm for predictive problems. ML algorithms in this category learn from labeled data, which includes samples associated with a target variable, allowing the model to make predictions on new, unseen data. This thesis proposes TC detection as a supervised classification using models that approximate nonlinear functions to map meteorological data inputs to binary outcomes representing the presence or absence of a TC.

TCs are weather events that can form in various regions of the world, with slight variations and frequency depending on the basin in which they occur [44]. Developing global prediction models for cyclones is a highly complex task due to the challenges posed by their diverse nature in different areas and the vast amount of data required to cover the entire tropical region. Therefore, this study focuses on the South-West Indian Ocean basin, with geographical coordinates of 0°S, 30°S, 30°W, and 90°W, as the target area for forecasts.

This chapter focuses on the datasets used to train predictive models and the strategies adopted to collect, analyze, and process information before making them available to Machine Learning algorithms. The chapter is structured into four sections: Section 3.1 describes the datasets and entities from which the data are obtained. Section 3.2 analyzes the techniques used to represent the data. Section 3.3 explains the choices made for

selecting and processing the data. Finally, Section 3.4 introduces the evaluation metrics used to compare results obtained from models.

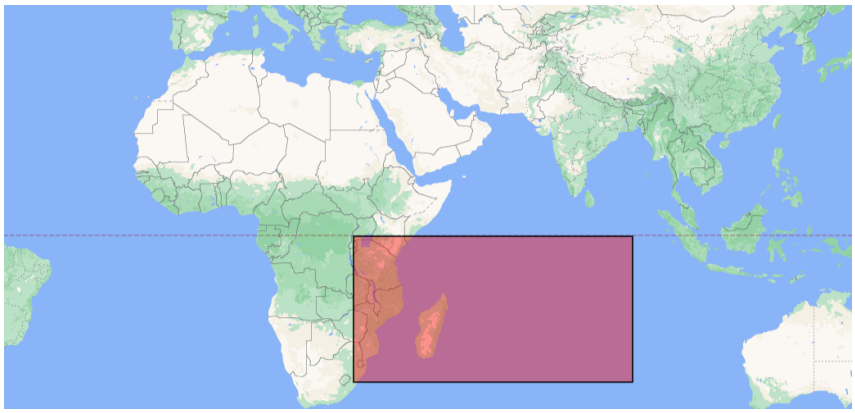


Figure 3.1: South-West Indian Ocean target zone.

3.1. Datasets

In the preliminary work for this thesis, it was crucial to search for data that describe the climatic conditions of tropical cyclones. To achieve this, datasets were sought to represent the local variables and large-scale drivers discussed in the previous chapter clearly and concisely. These datasets are categorized into three groups: data on large-scale climatic drivers, local variables related to the reference area, and data concerning the presence of TC in the target area. The latter, composed of TC historical tracking information, is essential for labeling samples and implementing supervised methods.

The data used in this study were obtained from public datasets provided by major climate research centers, such as the European Center for Medium-Range Weather Forecasts (ECMWF) and the National Centers for Environmental Information (NCEI).

3.1.1. Global Drivers

As described in the previous chapter, large-scale drivers are the factors that influence global climate anomalies. These seasonal or intraseasonal phenomena occur regularly, often increasing the chances of observing TC development. For this study, ECMWF provided datasets representing two well-known anomalies: the Madden-Julian Oscillation (MJO) and the El Niño-Southern Oscillation (ENSO) [21].

The first dataset describes the MJO phenomenon, an intraseasonal (30-90 days) variability in the tropical atmosphere. It consists of large-scale coupled patterns in atmospheric

circulation and deep convection that propagate slowly (5 m/s) eastward across the warm sea surfaces of the Indian and Pacific Oceans [81].

An index is provided to monitor the Madden-Julian Oscillation (MJO), combining data from three different variables: the near-equatorially averaged 850-hPa zonal wind, the 200-hPa zonal wind, and the satellite-observed Outgoing Longwave Radiation (OLR). Empirical Orthogonal Functions (EOFs) are applied to the combined record to identify dominant patterns of variability, and the daily observed data are projected onto the EOFs, resulting in two principal components (PCs) time series. This pair of PC time series making up the index take the name of real-time multivariate MJO series, RMM1 and RMM2 [77].

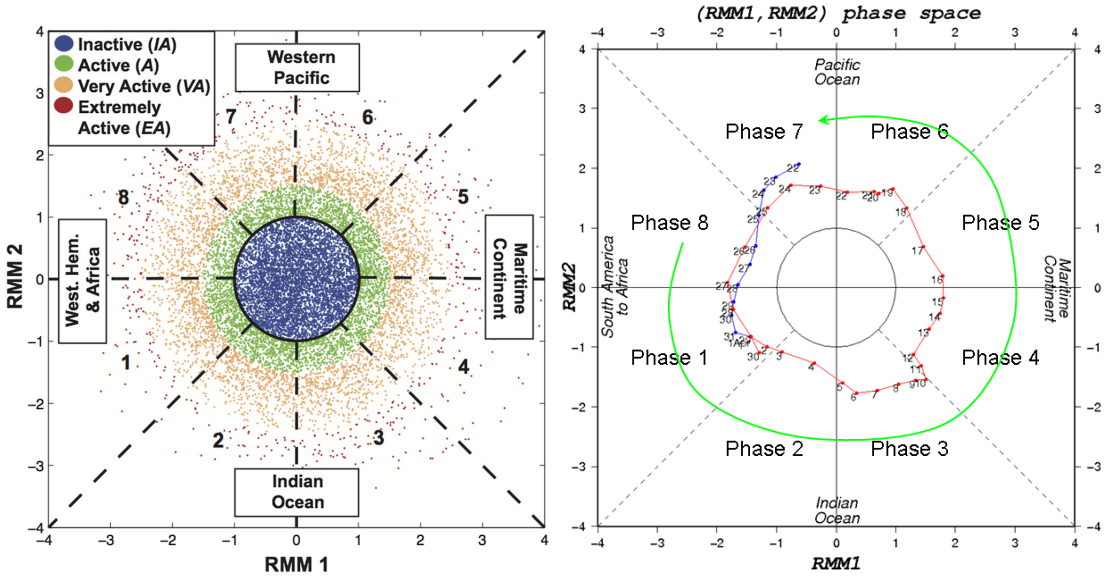


Figure 3.2: Representation of RMM1 and RMM2 indices [1] (right), [43] (left).

The MJO dataset includes not only the numerical values of RMM1 and RMM2 but also the amplitude value, which is the square root of the sum of their squares ($Amplitude = \sqrt{RMM1^2 + RMM2^2}$). This value represents the strength of the MJO activity. It is relevant to note that a high amplitude value is often associated with increased precipitation and severe weather conditions, while a low amplitude value generally represents stable conditions. This dataset also includes the variable *Phase*, which indicates the location of the MJO in its cycle. In Figure 3.3 the discrete values of *Phase* from one to eight are plot, providing a geographic representation. These values are useful to approximate the zone of activity of this phenomenon within the tropical belt, covering part of the Indian and Pacific Oceans. Figure 3.2 shows a common representation of this indicator in a phase space fashion, where counterclockwise paths indicate the eastward spread of the MJO over

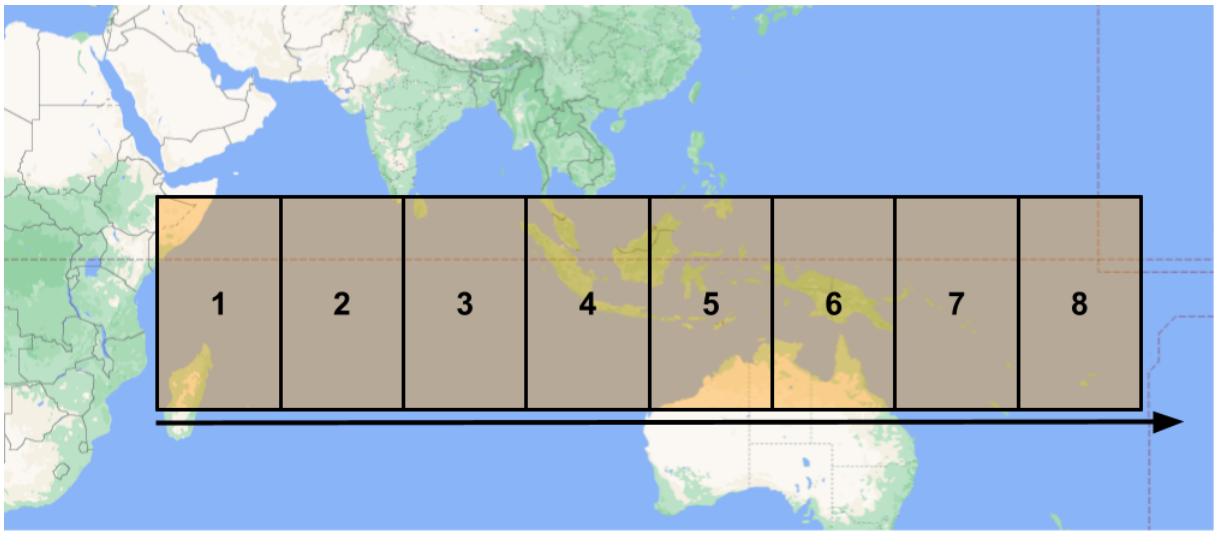


Figure 3.3: RMM phases geographic location.

the tropics, and the distance from the origin, indicates its intensity. The second dataset related to global drivers describes the ENSO anomaly, a quasi-periodic event that alters wind and Sea Surface Temperatures (SSTs) in the Pacific Ocean, causing irregularities in tropical and subtropical areas. These Southern Oscillations are fluctuations in air pressure related to changes in surface temperatures, causing periods of warming waters (El Niño) , and periods of cooling waters (La Niña), recurring from two to seven years [55]. The data used to describe this phenomenon represent SSTs in specific Pacific and Indian Ocean areas. As shown in related works, it is generally accepted that these regions are particularly expressive of the ENSO phenomenon, providing evidences of its occurrence [45]. The variables in this dataset are named *nino12*, *nino34*, *nino3*, *nino4*, *indocW*, and *indocE*, and their geographical location can be seen in Figure 3.4. The numerical values of temperatures are expressed in Celsius degrees, representing the average temperature in each corresponding zone.

Both datasets for MJO and ENSO provide records from 1980 to 2022, creating a rich observational history for data analysis. These samples have a daily frequency, providing 365 values per year for each of the above variables. To capture sequential and temporal dependencies, and to allow models to learn how past observations may influence future developments, it is essential to consider multiple days of observations and manage the data as a multivariate time series.

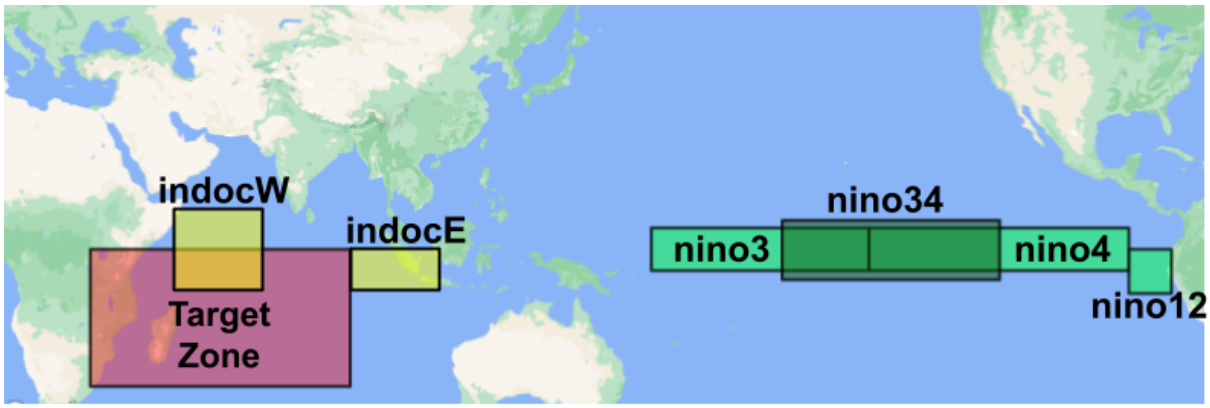


Figure 3.4: ENSO variables.

3.1.2. ECMWF's ERA5

The previous global anomalies data did not yield satisfactory results in terms of predictability and interpretability performances, due to their tendency of describing events that occur with low frequency. Therefore, additional data were necessary to provide a more comprehensive meteorological description of the target area.

In this work, ERA5 was used to retrieve data on local meteorological variables, such as daily weather conditions for wind speed, surface pressure, and precipitation. ERA5, implemented by the Copernicus Climate Change Service (C3S) within the ECMWF, is a highly advanced system that provides data on over 300 meteorological variables at various atmospheric levels [1]. It has one of the highest resolutions among systems of its kind, providing data every 0.25° displacement in latitude or longitude, which corresponds to a distance of approximately 30 km. As a reanalysis system, it provides a detailed representation of the global atmosphere, land surface, and ocean waves from 1950 to the present [32]. It reconstructs a comprehensive set of meteorological data using historical observations from satellites, weather stations, and ocean monitoring systems, applying complex computational models to build a dense representation of information.

After reviewing related works [29], I constructed a dataset using the variables that best describe TC activity. The dataset includes samples from 1980 to 2022 to maintain consistency with previous datasets. The records represent instantaneous values at 00:00 each day and the meteorological drivers considered in this work are surface pressure, temperature, relative vorticity, precipitation, wind speed, and air density. Table 3.1 provides a detailed view of the considered variables and their corresponding atmospheric levels.

[1] `https://climate.copernicus.eu/climate-reanalysis`

#Variables	14
#Points on target zone	$121 \cdot 241 = 29161$
#Time series points	10 (previous days) + 1 (today)
Tot input dimension	$14 \cdot 29161 \cdot 11 =$ **4490794** parameters

Table 3.2: Sample dimension for the ERA5 dataset.

Variables	Atm Levels
Surface Pressure	-
Sea Surface Temperature	-
Temperature	550hPa, 300hPa, 200hPa
Relative Vorticity	850hPa, 550hPa, 250hPa
Air Density	-
Wind Gust Speed	-
Wind Speed	1000hPa, 850hPa, 300hPa
Precipitation	-

Table 3.1: Selected meteoreological variables at different atmospheric levels.

The reference zone for my analysis encompasses an area of over 22 million km^2, with a resolution of 121x241 geographic points provided by ERA5. To obtain a more detailed description of the dataset for each day, I also included the corresponding time series data from the previous 10 days. Table 3.2 illustrates the overall high dimensionality of the original dataset provided by this reanalysis system. Dealing with such a large amount of data requires the implementation of complex models and significant computational power. To provide a more explicit description of this phenomenon through simple models and avoid computationally expensive training, we had to aggregate the data despite its limitations. This resulted in a more manageable dataset, but several approximations were necessary. Sections 3.2.2 and 3.3.1 provide a detailed description of these evaluations through the use of Kernel Density Estimation and feature selection techniques.

3.1.3. IBTrACS

The third dataset included in this work describes the occurrences of TCs in the area of interest. It is provided by the International Best Track Archive for Climate Stewardship (IBTrACS) [39]. IBTrACS is a collaboration between the National Oceanic and Atmospheric Administration (NOAA) and other meteorological organizations worldwide. Its

aim is to provide researchers with a free and open archive of data on the position, intensity, and size of tropical cyclones worldwide. The archive covers a period from the beginning of the previous century to the present day.

IBTrACS [2] is a complex dataset that includes observations from major meteorological institutes to obtain historical tropical cyclone occurrences for relevant basins. A more compact version of this dataset was provided by ECMWF to better address the needs for this work [21]. The dataset named "tc_act_sind" includes daily aggregated data from 1980 to 2022 in the South-West Indian Ocean region. It considers relevant tropical anomalies above the threshold value of 17 m/s. The dataset comprises only two variables that describe the genesis and presence of cyclones on each day:

- *S.IndGen* is an integer value that represents the number of cyclones occurring on a specific day of the year.
- *S.IndAll* is a float value that represents the number of cyclones present on a specific day and their corresponding duration within the day.

The variable *S.IndAll* provides information on a 6-hour basis. Therefore, a value of 0.25 indicates the presence of a cyclone during a quarter of the corresponding day. To improve labeling for ML-supervised classifications, both the *S.IndAll* and *S.IndGen* variables are binarized. This involves considering all observations with values greater than zero as positive samples. For instance, if a cyclone has occurred for the entire day or a portion of it, that sample is considered positively labeled. This approach is preferred over using float values because it provides a clear definition of the two classes, referring to the presence or absence of cyclone activity.

The two variables are typically correlated, as the formation of a cyclone corresponds to its presence in the area in the following hours or days. However, some samples may represent cyclones that developed outside the target zone and were recorded in the *S.IndAll* variable without a previous record of formation in the *S.IndGen* variable. The opposite condition, representing a cyclone generation inside the target zone and its movement outside the following day more rarely occurs. Less than a dozen samples indicate anomalies forming at low latitudes and leaving the target zone within hours. These events are typically of low intensity, representing subtropical depressions or storms. High-intensity cyclones do not usually form inside the target zone and exit within the next 24 hours. These occasional inconsistencies may cause anomalies in the detection process of this phenomenon and should be considered when analyzing the results. Further analysis of the temporal distribution and location occurrences of tropical cyclones can be found in the following

[2] https://www.ncei.noaa.gov/products/international-best-track-archive

section.

3.2. Data Analysis

Data analysis is an essential initial step in ML modeling. It helps to understand data relationships, identify patterns and trends, assess data quality and cleanliness, and determine feature importance. This section outlines the data analyses conducted to reduce dataset dimensionality and the resulting considerations to support the implementation of predictive models.

3.2.1. Tropical Cyclones Distribution

TCs are seasonal events that occur at specific times of the year with varying frequency in different basins. According to Mavume et al. [54], the South-West Indian Ocean basin is characterized by a maximum occurrence of cyclones during the months from November to April, which is considered the cyclone season in this area, with more than 85 % of the cyclones occurring during this period. Additionally, there has been an overall intensification of cyclones in recent decades, which appears to be related to the increase in mean sea surface temperature and to climate change effects.

The statistical analyses conducted on the acquired data reveal a consistent trend, identifying the period of highest anomaly frequency between November and April, with a peak in January-February. Figure 3.5 displays the seasonal distribution of the variables *S.IndAll* and *S.IndGen*, confirming the observed tendency and highlighting the significance of seasonality in TC forecasting. Considering the time of the year is crucial in my forecasting problem, enabling us to exclude samples that fall outside the known period of TC activity. To achieve this, I introduced a new variable representing the day of the year of each sample, in addition to the variables obtained from the previous datasets.

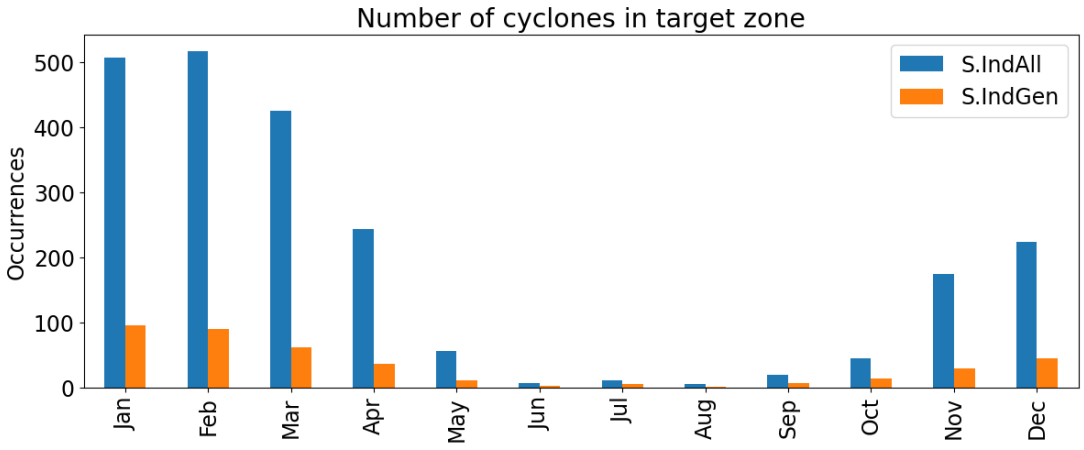


Figure 3.5: Distribution of TCs presence and genesis over months.

The location of cyclones in the reference target area was analyzed by plotting data from the IBTrACS archive. The archive includes the location and intensity of cyclones, which allowed for a geographical analysis of the phenomenon in the target zone. Figure 3.6 illustrates how these anomalies have weak activities near the equator, as expected. Therefore, when selecting local variables, it is advisable to consider data from the central latitudes of the tropical belt as more relevant than those from the lower latitudes, considering this sub-region as the most interesting for frequent and high-intensity events.

As previously stated, a significant number of anomalies recorded in the dataset under the variable *S.IndAll* were not generated in the target area. Table 3.3 shows that approximately one quarter of the 393 anomalies to be predicted were not generated within the area of interest and had a shorter duration than those that occurred solely within this area. As already mentioned, this discrepancy presents challenges in creating a completely homogeneous dataset to predict accurately the genesis events. To address this issue, I have generalized the problem to a binary classification on the presence of cyclones in the target area, using *S.IndAll* as the target variable to label samples with binary values that represent one of the two classes "detection of a TC" and "no anomaly".

Time duration of TCs (in days)

	TC generated in target zone	TC non-generated in target zone
count	298	95
mean	5.89	4.95
std	4.21	3.99
min	1.00	1.00
25%	2.25	2.00
50%	5.00	3.00
75%	8.00	7.00
max	25.00	17.00

Table 3.3: Statistics of tropical cyclones generated and non generated inside the target zone.

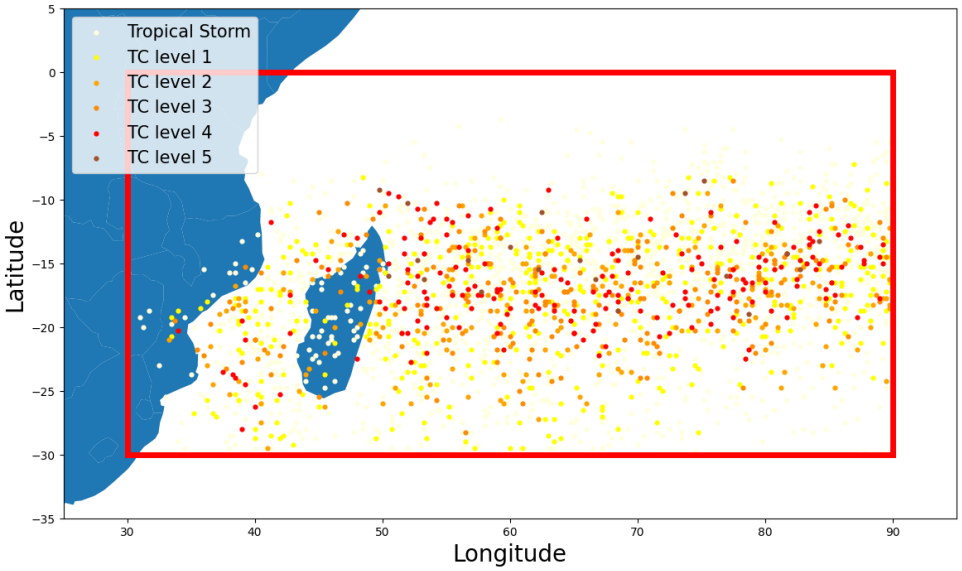


Figure 3.6: Distribution of TCs in the target area according to the Saffir-Simpson scale.

3.2.2. Kernel Density Estimation

As stated in Section 3.1.2, the ERA5 reanalysis system provides high-resolution data for the target area. This large data dimensionality represents a challenge when training simple and low-computational models. Therefore, a preliminary analysis of the dataset was necessary to determine how to approximate the data while reducing its dimensionality. Three different approaches were initially proposed:

- Considering only a small number of points within the target area.

- Considering only geographic points with more extreme weather conditions, e.g., maximum wind gusts.
- Segmenting the target area into a few sub-areas and evaluating the mean and standard deviation for each of them.

The first two approaches are ineffective in obtaining a fair representation of the original dataset. The first method omits relevant information, such as considering geographical areas where a cyclone is potentially active. The second one introduces a bias related to prior knowledge of the most expressive locations. The third approach was found to be the most consistent with the initial dataset representation. However, the main issue in this case is associated with the decision of how to segment the zone of interest, in particular the choice of the number of zones and their arrangement.

To address this issue, I applied Kernel Density Estimation (KDE) [75], a technique for estimating the Probability Density Function (PDF). This method is applied to estimate how the meteorological variables are distributed over the target area. PDFs are evaluated on different area segmentations, to find the best solution that could be a fair compromise between an approximation of the original ERA5 data distribution and a reasonable reduction of its dimensionality.

KDE is a non-parametric method based on a kernel function, a bell-shaped function used to represent the influence of each data point in its neighborhood, and a bandwidth that controls the smoothness of this function. Wider kernel functions provide a smooth approximation, but this may lead to the omission of some data points with extreme behavior, while narrower functions provide a more jagged representation, with more expression of detail.

Figure 3.7 shows the dataset segmentation process. The target rectangle was divided into four zones, starting with a minimum division of two per each dimension, and increasing by one the division for each dimension. The process is carried out for each subsequent segmentation until a division of 36 zones was achieved. This approach helped to understand the extent to which further subdivisions could lead to a more detailed representation of the data.

This analysis indicates that the incremental segmentation offers greater detail up to a 16-zone division, with no significant additional insights gained from increasing the resolution to 25 or 36 zones. Therefore, the optimal solution for this work is to divide the target zone into 16 zones, achieving a reasonable balance between dimensionality and quality of the representation. Each zone provides the mean and standard deviation of the variables,

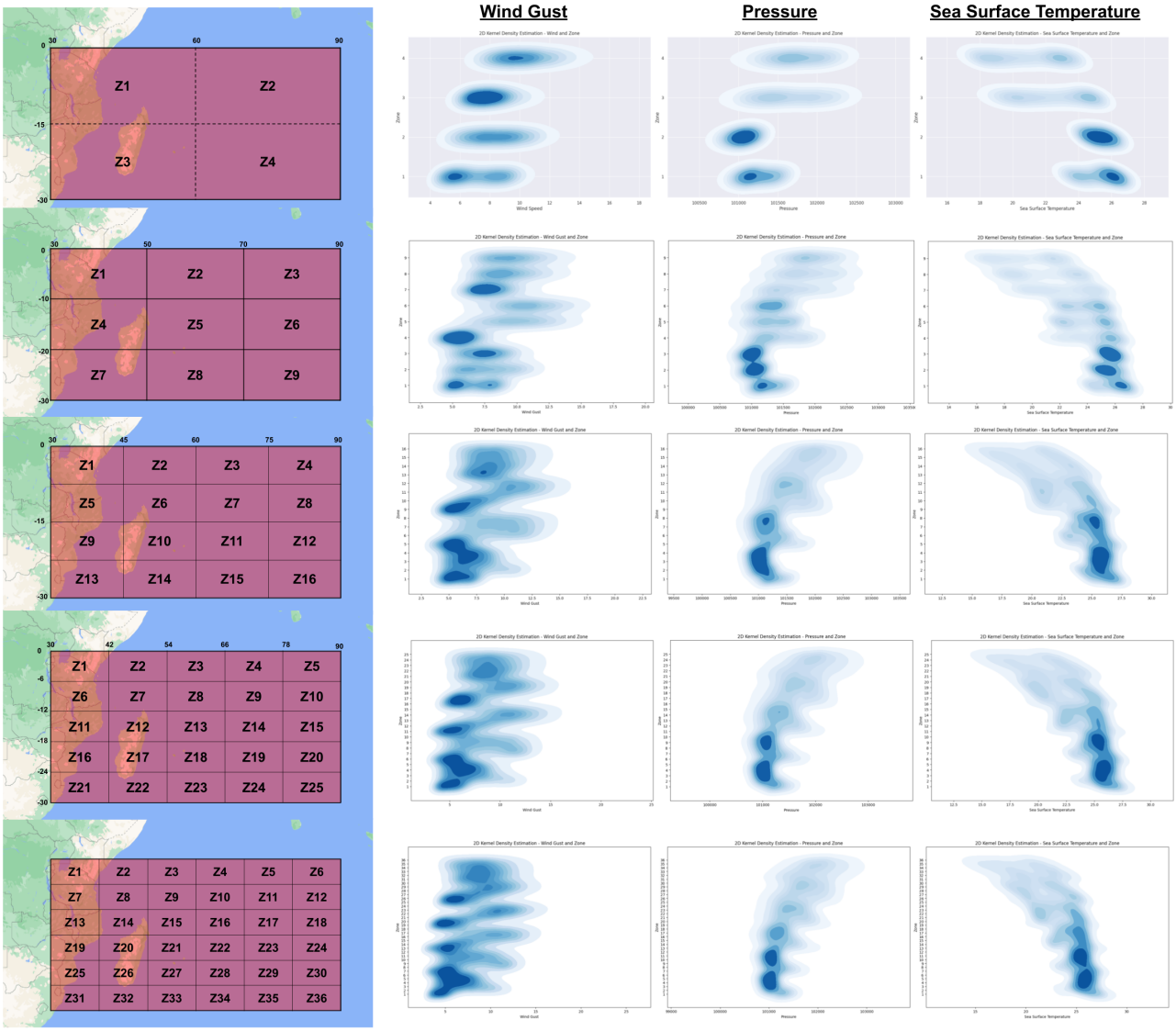


Figure 3.7: Kernel Density Estimation for three variables with different segmentation.

resulting in a good approximation of the starting dataset with a significant reduction in its dimensionality. This updated representation has considerably decreased the number of variables for each day of the year, simplifying the training process and reducing the complexity of the models implemented.

3.3. Data Preparation

The following paragraphs describe the two most relevant techniques implemented to optimize the data quality of my dataset. The goal is to delete irrelevant features that could introduce noise in the training phase and to optimize the strongly unbalanced dataset.

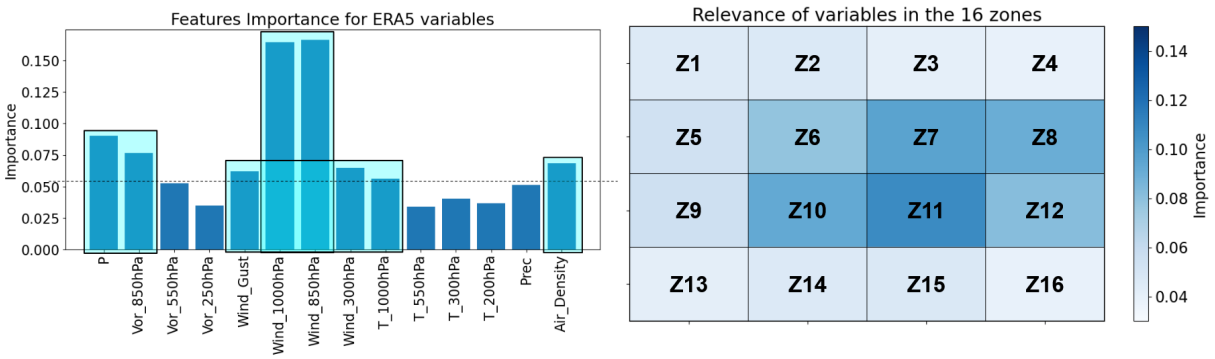


Figure 3.8: Meteorological variables and 16-zones importance.

3.3.1. Feature Selection

Feature selection is a critical stage in the Machine Learning modeling process, aiming to address some of the traditional ML issues related to high dimensionality [47]. It enhances performance by removing redundant or unnecessary information beforehand. Dimensionality reduction simplifies models, making them easier to train and interpret, and reduces problems such as overfitting while saving computational resources. Several approaches can be used for feature selection techniques. Many of these methods rely on statistical properties, such as correlation or variance, to assess the importance of the association between features and their related targets. Wrapper methods and embedded methods are two possible implementations for feature selection. Wrapper methods add or remove features to evaluate model performance, while embedded methods incorporate feature selection into the training process, such as L1 regularization [14].

For my work, I implemented a tree-based feature selector. This method uses tree-based estimator construction to compute feature importance based on impurity. During tree construction, each feature is used to make decisions that partition the data. The importance of each feature is related to the Gini impurity, which defines the grade of impurity or disorder of a set of data points. A Gini impurity value of zero corresponds to a pure set, where all elements belong to the same class, while a value of one indicates a set with maximum impurity, where elements are evenly distributed across classes.

Figure 3.8 displays the overall relevance of the meteorological variables considered in this work and their respective importance in each of the 16 zones considered. The variables with the highest importance are wind speed, relative vorticity, surface pressure, and air density, while the most significant zones are located in the central tropical belt, between 7.5°S and 22.5°S, and further offshore in the Indian Ocean.

Furthermore, it is found that the mean and standard deviation of the variables for each

Time duration of TCs (in days)

	ERA5 dataset	16 zones	6 zones	Z11
#Variables	14	14	8	8
#Points on target zone	29161	32	12	2(avg+std)
#Time series points	10+1	10+1	3+1	1 (today)
Tot input dimension	**4,490,794**	**4,928**	**384**	**16**

Table 3.4: Dimensionality reduction with feature selection.

zone have the same weight, so they will always be considered in subsequent model implementations. Table 3.4 illustrates the significant reduction in data dimensionality resulting from various selection scenarios. These options allowed us to consider a larger or smaller selection of features, depending on each model's ability to handle a larger or smaller number of variables. I kept this analysis as a reference for my considerations on model implementation in the next chapters.

3.3.2. Data Rebalancing

In classification tasks, an imbalanced dataset refers to a set of labeled samples where the classes are not equally represented. This problem is particularly prevalent in binary classifications, where a majority class has significantly more samples than a minority class. In the TC classification of this thesis, the problem is that only about one-seventh of the days between 1980 and 2022 represent the positive class associated with the presence of a TC on that day. This creates an imbalance in data, with the majority class dominating. As a result, biased models may be produced during training, causing them to favor the dominant class over the minority one. This phenomenon can have significant implications in various fields, particularly in the detection of extreme weather events, where the objective is to predict specific events that occur with low frequency.

In the field of Machine Learning, there are several solutions available to improve the predictive performance when dealing with imbalanced data. Generally, these techniques are based on the following approaches [31]:

- Sampling methods, to modify the original dataset and achieve a balanced distribution of examples by adding or removing samples.
- Cost-sensitive methods, that are learning techniques taking into account the cost of misclassified samples during the learning process.
- Kernel-based methods, that use kernel functions to project data points into a dif-

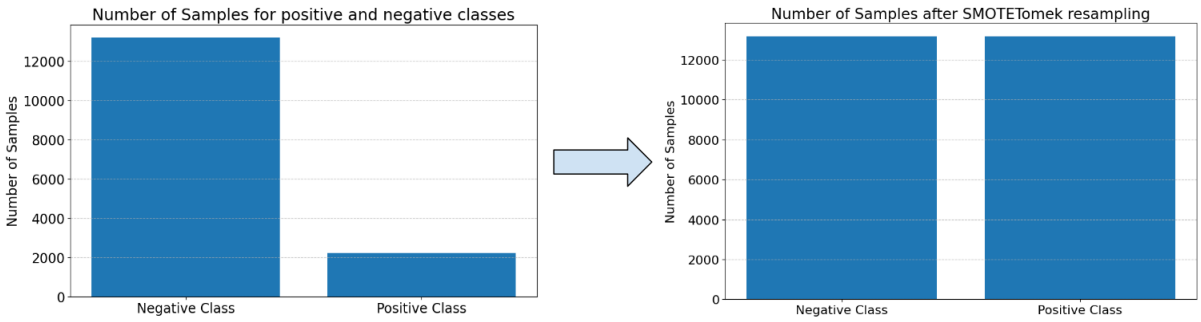


Figure 3.9: SMOTE + Tomek Links technique for data rebalancing.

ferent dimensional space where the distribution of classes becomes more balanced.

For my work, I chose to rely on the first of the three approaches mentioned above, using various sampling techniques to redistribute the original data. However, I found that reducing the samples in the majority class through traditional random or informed under-sampling techniques did not improve the performance of the implemented models. Using these methods that reduce the amount of information available to feed the models leads to a significant decrease in predictive results.

A technique with an opposite and more sophisticated focus was to balance the dataset using the Synthetic Minority Over-sampling Technique (SMOTE), an oversampling method that allows to generate synthetic samples of the minority class [15]. In this study, the technique was used in combination with under-sampling techniques such as Edited Nearest Neighbor (ENN) [78] or Tomek Links [70]. These methods identify and remove data points from the majority class that are too close to the minority class, thus producing cleaner decision boundaries and reducing noise. However, poor results were obtained in both cases, leading to the conclusion that none of the attempted techniques to balance the original dataset could bring significant improvements. For this reason, in the following chapters, I will present the implemented models and their respective performance evaluations using the original dataset as the reference, without employing any balancing techniques.

3.4. Evaluation Metrics

Evaluating the quality of forecasting models is essential to compare different techniques objectively and determine their ability to provide accurate predictions. In my work, model evaluations must consider two fundamental aspects:

- Quality of predictions to evaluate the forecasting ability of the implemented models.

- Clear and understandable explanations to describe the prediction that was made.

In Machine Learning, predictions are evaluated using various metrics to assess their quality. These metrics are numerical values that quantify a model's ability to learn a behavior from the data. Binary classification models use metrics based on predictions made against a test set, specifically considering the number of correct and incorrect predictions. Therefore, the following framework defines the four possible outcomes when classifying a sample from the test set:

- True Positive (**TP**), when both the predicted and actual samples are positive (TC presence).
- False Positive (**FP**), when the predicted sample is negative (TC absence) and the actual sample is positive (TC presence).
- True Negative (**TN**), when both the predicted and actual samples are negative (TC absence).
- False Negative (**FN**), when the predicted sample is positive (TC presence) and the actual sample is negative (TC absence).

By evaluating these parameters, more comprehensive metrics can be obtained to describe the behavior of the model. Accuracy is a widely used metric that describes the number of correct instances among all predictions. It is defined as:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}. \tag{3.1}$$

However, it is important to note that this metric may not be robust when testing an anomaly detection model. Due to the highly unbalanced datasets used in this scenario, positive samples are infrequent, such as in the case of TC occurrence in a calendar year. Therefore, a model that strongly predicts negative classes but weakly predicts positive samples may have high accuracy, even though its actual ability to solve the problem is limited.

Additional evaluation metrics are necessary in our scenario to provide a more robust and consistent assessment of the goodness of fit for the developed models. The considered metrics include precision, recall, F1-score, and False Alarm Ratio (FAR). Precision measures the quality by providing the rate of true positive classified samples over all the positive predictions:

$$Precision = \frac{TP}{TP + FP}. \tag{3.2}$$

Recall gives a measure of quantity by providing the rate of true positive classified samples over all the positive labeled ones:

$$Recall = \frac{TP}{TP + FN}. \tag{3.3}$$

Finally, F1-score is a metric that combines precision and recall, while FAR represents the rate of incorrect extreme event detection:

$$F1_{score} = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall}, \tag{3.4}$$

$$FAR = \frac{FP}{TP + FP}. \tag{3.5}$$

This work evaluates both the quality of the predictions and the interpretability of the developed models. To achieve this, various explainability techniques are used to derive rules that define the classification into two classes. This enables a comparison of how each model can provide conditions for cyclone detection and evaluates if two models are similar to each other, comparing variables included in the rules and the threshold values related to TC activities.

4 | Black-Box Models

The success of Machine Learning techniques in learning data relationships has led to their use in increasingly complex scenarios, improving the ability of data-driven models to solve a wide range of predictability problems. However, concerns have been raised about their use in critical circumstances. Over the years, there has been an increasing need for models performing well in solving traditional supervised tasks while also providing results that can be easily understood by users. This scenario is represented in the ML field by the dualism between black-box models, which are so-called because the complex logic and internal structure is unknown to the users, and white-box models, capable of providing inherent transparency and interpretability to support predictive processes [50]. The former tend to favor better results on prediction metrics, such as accuracy or precision, at the expense of the ability to show their actual behavior. In contrast, the latter generally have a weaker predictive ability on complex problems but provide clear descriptions of the process.

The ultimate goal of this thesis is to obtain detailed explanations of the tropical cyclones forecasts, optimizing predictions while also describing how these processes successfully perform. A system with these characteristics can be useful to provide concrete support for the delicate decision-making processes that occur during the development of TC activities.

This chapter describes the black-box models implemented for TCs detection, analyzing their main features concerning the data previously analyzed. This includes methods based on neural networks and ensemble models, the complex nature of which makes it very difficult for users to fully understand the patterns and relationships that enable predictions from the data. For this reason, the second part of this chapter focuses on the explainability techniques that can be adopted when dealing with opaque systems, to obtain rules that describe the dependencies between classifications and input values.

Section 4.3 exposes a comparison between the predictive capabilities over various time horizons of the different strategies adopted. Additionally, a quantitative analysis of the relevant features of the datasets involved in the predictive process is provided. This is done using one of the most common techniques to exploit understandable predictions in black-box models.

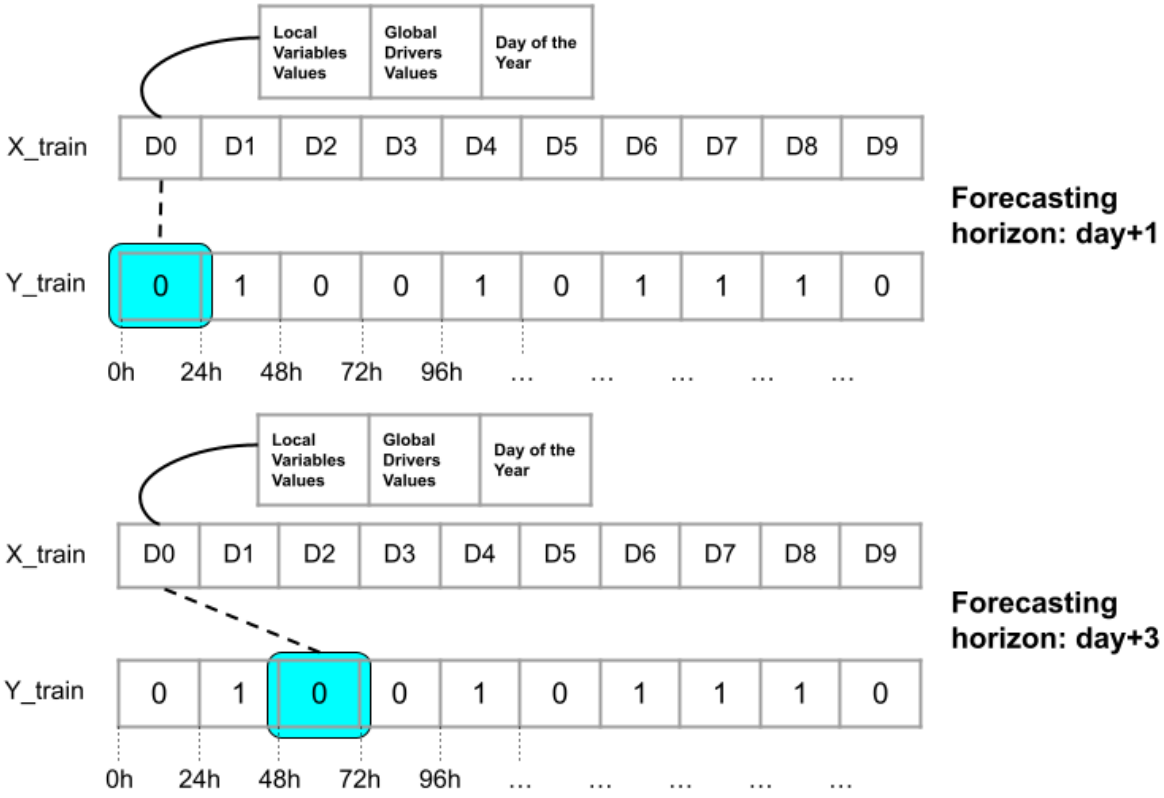


Figure 4.1: Target shifting for samples with forecasting horizon day+3.

4.1. Models Implementation

As already described in Chapter 3, the forecasting problem is addressed as a binary classification problem using supervised learning techniques. The black-box models selected have the peculiar ability to handle high-dimensional data, making them an optimal solution to learn from the complete dataset of this thesis. Adopting these models on TC forecast allows us to achieve high performance measures on predictive metrics. Additionally, this evaluation considers predictions with varying time horizons, ranging from one to ten days after the date corresponding to the input data. In order to create a dataset that can be used for multiple forecasts with varying time horizons, it is necessary to shift the target vector by one position for each additional day that requires forecasting. An example of data pre-processing to obtain multiple instances of the same dataset for different horizon forecasts is displayed in Figure 4.1.

To conduct black-box training and evaluation, all the variables from the datasets are included without applying any pre-processing technique such as under-sampling or features selection. In addition, the behavior of the local and global drivers are described by time series consisting of the samples representing the days before the reference date. The length

	Input Shape
Global Drivers Timeseries	30 (timesteps) · 9 (variables)
Local Drivers Timeseries	10 (timesteps) · 38 (variables) · 16 (zones)
Local Drivers Today	1 (timestep) · 38 (variables) · 16 (zones)
Day of the Year	1 (variable)
Total Input Dimension	6959

Table 4.1: Input shape for black-box models.

of these time series was evaluated by analyzing their impact on the final predictive ability. The time series describing the global drivers are more effective when up to thirty days were considered. In contrast, local drivers seems to contribute up to 10 days before the forecast. The samples associated with each of the days under consideration are useful to describe the trends of these variables in the time windows preceding the forecast date. In table 4.1, the dimensions of input data are reported, specifying the number of timesteps, variables, and zones that contribute to the total shape.

In order to properly perform the model training and evaluation phases, it is necessary to divide the source dataset into subsets to be used at different stages of modeling. In fact, the inclusion of the same data in the training and testing sets would be counterproductive, making the evaluation process ineffective and potentially incorrect. The main goal of the classifiers implemented is to be able to generalize to new unseen data, which would not be possible if the same samples are present in both sets. In this case, the trained models would reduce their behavior to memorize specific instances, rather than learning relationships and patterns in the data. In several cases, in addition to the need of using one set to test the overall predictive performance of the model, it is essential to have a third set to support the training process, in an additional phase called validation. Validation consists in a process in which the trained model is evaluated at each iteration of the training phase, also called epochs. This evaluation allows to monitor the progress of a model and its tendency towards phenomena such as overfitting or underfitting. The former represents the undesirable situation in which a model performs very well on predicting training data, but is unable to generalize on new unseen examples. The latter occurs when the model is not complex enough to capture patterns in the data and is not able to learn any relevant relationship to exploit supervised tasks.

Table 4.2 shows the division in the cases where a model is trained by including the validation phase or not. In both situations, the models are evaluated using the samples for the years 2012-2021 as a reference, whose time length is sufficient to include a relevant number of TCs to evaluate the model's robustness. This partition into test and training

Datasets with validation and testing sets

	Sets Splitting
Training + Test	Training Set: 1980 - 2011 Test Set: 2012 - 2021
Training + Validation + Test	Training Set: 1980 - 2001 Validation Set: 2002 - 2011 Test Set: 2012 - 2021

Table 4.2: Training - Validation - Test splitting.

sets is also considered in the following implementation of white-box models, described in Chapter 5, to provide a common test set to evaluate the overall performance in both solutions.

4.1.1. Gradient Boosting Decision Trees

Gradient Boosting (GB) is a Machine Learning technique that belongs to the category of ensemble methods, which combine different models to improve predictive performance. Specifically, boosting is a solution that takes advantage in building and evaluating models sequentially, to correct the errors of the previous model with each new one. Typically, these boosting algorithms use decision trees as base learners, although it is not necessary to rely on them. Any weak predictor that allows optimization of the gradient of the loss function, which must be differentiable, can be considered as a valuable candidate. These methods are widely used to build high-performance predictors and provide several options for solving classification and regression tasks.

One of the earliest boosting algorithms implemented is AdaBoost [27], a still highly successful method that adaptively updates the parameters of weak predictors. The boosted model implemented in AdaBoost uses a sum of weak learners' predictions to obtain the outcome. At each iteration of the training process, it is assigned a weight to each sample in the training set. These weights are equal to the prediction error on that specific sample. Higher weights are assigned to misclassified examples, forcing estimators to improve their predictive ability to learn from previous mistakes. This iterative process repeates until a stopping criterion is met, usually specified by the fixed number of weak estimators generated in the training phase or by a fixed number of maximum iterations.

A further improvement of this algorithm was proposed by Friedman [28], who introduced the concept of gradient descent in updating the weights. Unlike AdaBoost, the parameters of the base learners are updated following the negative gradient vector of the loss function

computed in the previous iteration. In this way, GB can identify the most difficult samples to predict based on the largest residuals obtained from the previous step.

Several implementations of this method have been proposed over the years, providing improvements in training speed and model accuracy [9]. Among them, Extreme Gradient Boosting (XGBoost) is a GB framework that provides a particularly suitable implementation for its use in TCs forecasting. Its design provides support for scalability to large datasets, good efficiency in the learning process, and robustness to overfitting [18]. Experiments on the dataset were performed using this implementation and found an improvement over more classical versions of this algorithm, such as the one provided by the scikit-learn library.

A key aspect of optimally adapting this system to the modeling of TCs forecasting, is the definition of hyperparameters. These are important to define the structure and complexity of the model, such as the maximum depth or the maximum number of leaves in the trees involved as weak learners. They define properties of the training process and can be used to tune the regularization parameters. These parameters include the learning rate, which has a value between zero and one and is useful for scaling the contribution of each weak learner, and the number of estimators, which specifies how many weak learners make up the final model. The optimal configuration of these parameters is a problem closely related to the outcome of the overall classification performance, and for this reason it must be addressed with the right approach. A thorough knowledge of the ML models implemented is essential, but there are also several techniques that provide support for proper hyperparameters optimization.

The current state of the art on this topic is Optuna [5], an optimization framework capable of efficiently exploring the space of hyperparameters. The Optuna library provides a simple method for defining the parameter search space and for searching and pruning it, to obtain a configuration that can optimize an objective function. In this work, the hyperparameter search study was defined to optimize the recall value, to obtain values that maximize the number of true positives, over the number of cyclones actually present in the test set. The search process consists of a loop in which at each iteration, new values of parameters are defined and the objective function is evaluated. This iterative process terminates when a stopping criterion is reached, such as a maximum number of cycles or a maximum execution time. Each iteration is an approximation of the complete training processes, allowing us to obtain the optimal values for the hyperparameters of a XGBoost model that best fit on the reference dataset. The results of this optimization are shown in Table 4.3, where the best numerical values of the parameters on which the search is conducted can be seen. The base learner decision trees resulting from this process have a

Hyperparameters	Values
Learning Rate	0.1
Max Depth	41
Max Leaves	102
Number of Estimators	223
Booster Type	"gbtree"

Table 4.3: Hyperparameters for the best model obtained with Optuna.

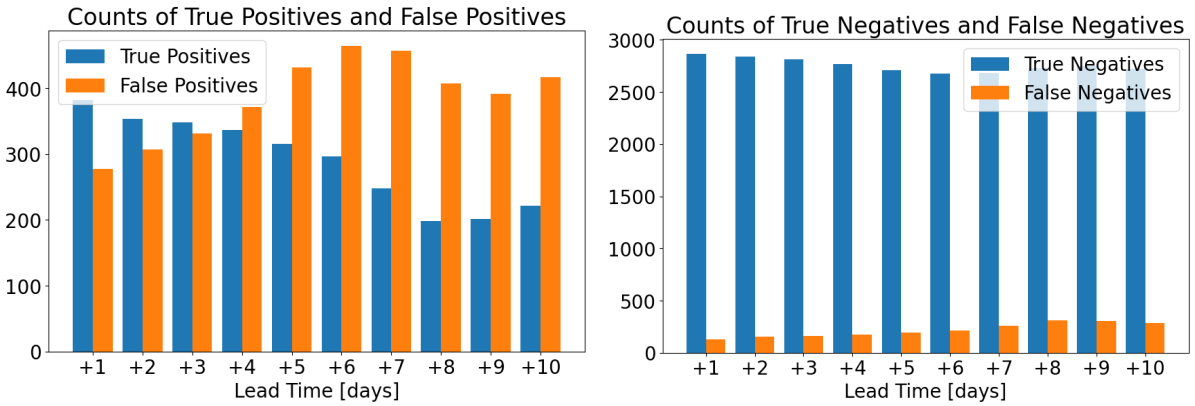


Figure 4.2: Classifications on the test set 2011-2021 for the XGBoost model.

maximum depth of 41 and a maximum number of leaves of 102. 223 estimators make up the complete ensemble structure. Additionally, XGBoost gives different options of booster type. In this case, the "gbtree" specifies that the booster technique should build decision trees as weak learners.

The classification performance of this model can be seen in Figures 4.2 and 4.3. The number of correctly classified positive examples is higher than the number of false positives only in the first three forecast horizons and is related to the results in predicting at 24

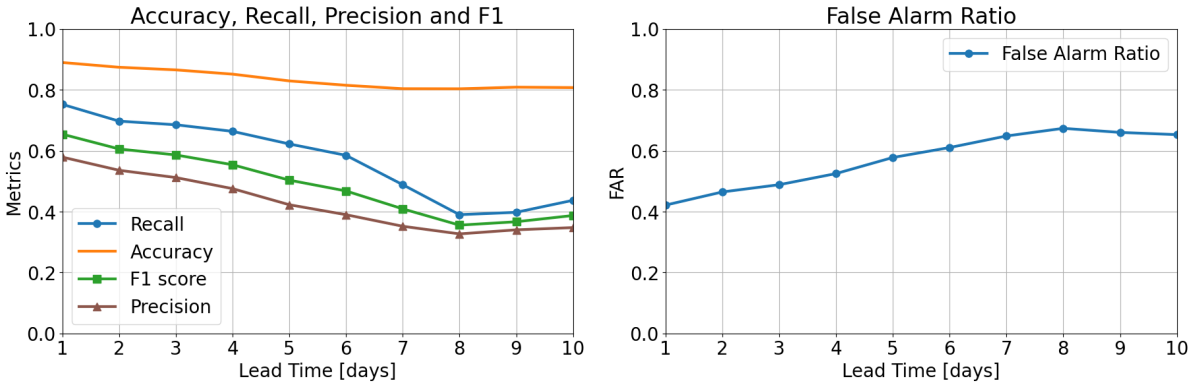


Figure 4.3: Classification metrics and false alarm rate for the XGBoost model.

h, 48 h, and 72 h. In contrast, the model can predict the negative class with great skill, as a consequence of the strong imbalance of the two classes in the initial dataset. The performance metrics provide a deeper evaluation of the overall behavior of this predictor. In particular, the False Alarm Rate is generally high for all the predictions over different horizons, assuming at least four false alarms for every ten positive outcomes computed by the model. At the same time, the ability to detect a TC presence within 48 h is high enough since the recall value is slightly lower than 0.8. However, it becomes impossible to obtain relevant results for extended forecast horizons.

4.1.2. Long Short-Term Memory Networks

After the first ensemble approach presented in the previous section, a deep learning oriented model was considered to better capture dependencies in global and local time series. Long Short-Term Memory networks (LSTMs) [33] are special Recurrent Neural Networks (RNNs) widely implemented to learn temporal dependencies within data. Their great ability to detect long-term patterns in time series has led to great success, allowing them to be used to solve numerous Machine Learning problems, such as time series classification and forecasting.

LSTM precursors are vanilla RNNs, a type of artificial neural networks based on very simple cells that use the outputs of previous computations as current inputs. Unlike these models, LSTMs have a more articulated architecture that consists of complex units with multiple interactive layers. Figure 4.4 shows the basic structure of a single LSTM cell, highlighting the main components that have the following functionalities:

- Cell State represents the long-term memory of each cell. Its content results from the interaction of gates, to add or remove information.
- Forget Gate determines the amount of information of the previous state to be removed.
- Input Gate combines information from the previous hidden state and the current input and adds it to the cell state.
- Output Gate controls the amount of information to expose to the following cells.

The architecture of this cell is key to the success of LSTM-based models, as it allows to memorize data with long-term temporal dependencies, which is crucial for solving those tasks that require learning from long time series. To obtain a network that is complex enough to perform our classification task, it is necessary to stack several layers consisting of LSTM base units. The construction of the network results in the architecture shown in

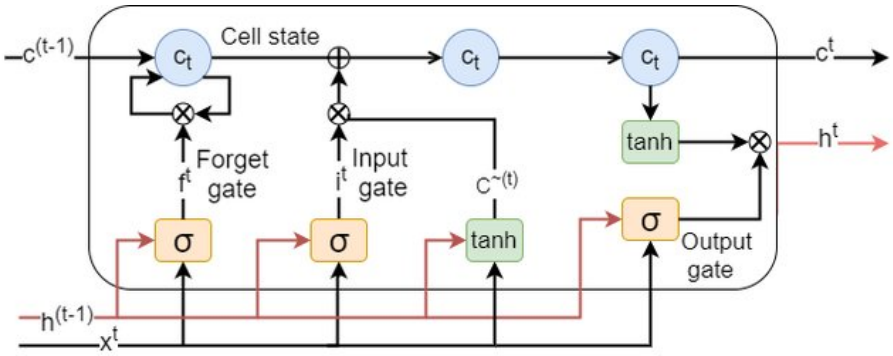


Figure 4.4: Long Short-Term Memory cell architecture [35].

Figure 4.5. It consists of two blocks working in parallel. Each of them is used to process the two different time series, which have a different number of samples, i.e., 10 steps for the ERA5 data and 30 steps for the MJO and ENSO data. Once the two series pass through these blocks, their outputs are concatenated and then used as input for a classifier based on dense layers. The number of units for both LSTM and dense layers are reported in Figure 4.5. The last layer consists of a single neuron that outputs the probability of classifying each sample in the positive class via a sigmoid activation function.

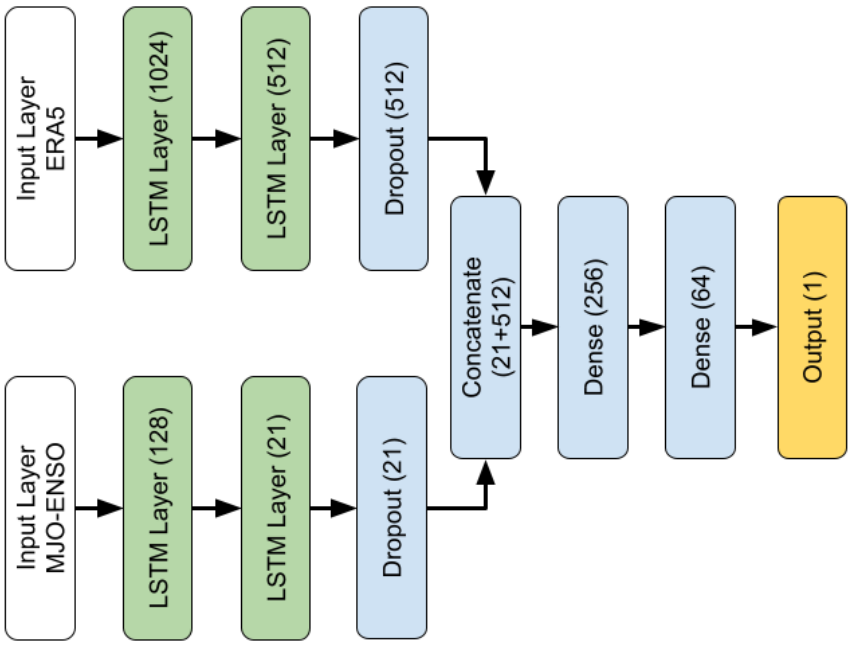


Figure 4.5: Long Short-Term Memory network architecture.

Ten models are trained with this architecture, for each one of the time horizons considered. The results obtained are shown in Figures 4.6 and 4.7. As in the previous analysis on the

XGBoost results, LSTMs still have a very high false alarm rate value, causing the model to make positive classifications even when the current target is not. However, unlike the previous model, the overall performance metrics for positive examples classification show a wider gap between precision and recall. This trend is mainly due to the ability of the LSTM model to predict the positive class with a higher rate than the previous GB model. This behavior increases the likelihood of classifying positive samples, but consequently decreases the precision related to the predictions that are made. Again, it is important to note that beyond a certain time horizon, predictions with this model lose quality and prove not to be robust enough.

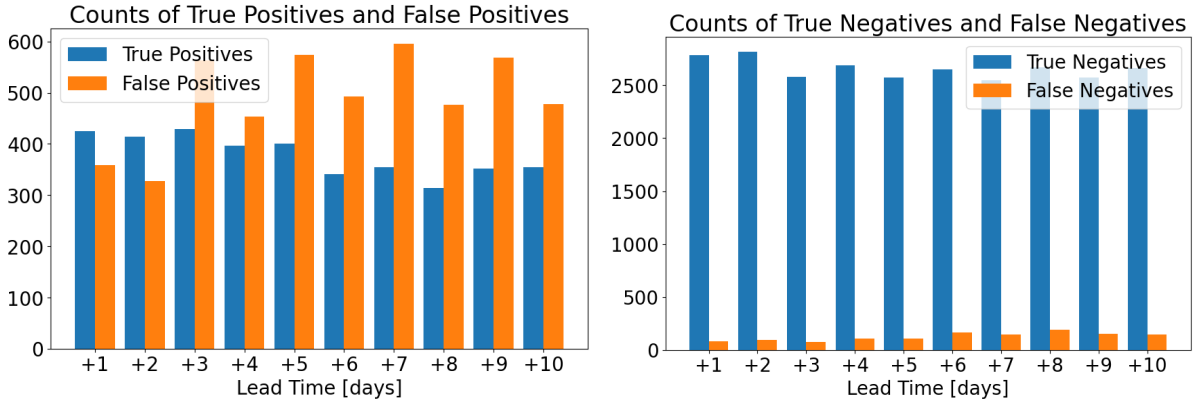


Figure 4.6: Classifications on the test set 2011-2021 for the LSTM network.

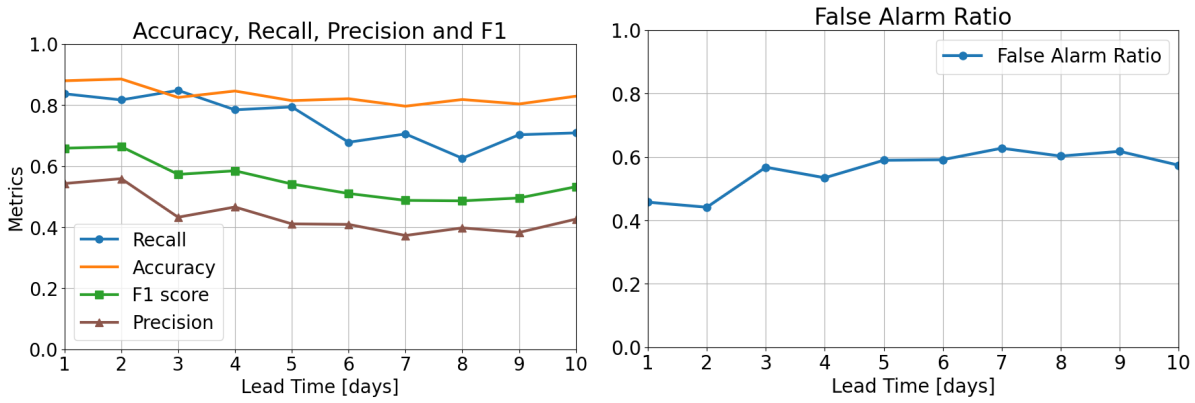
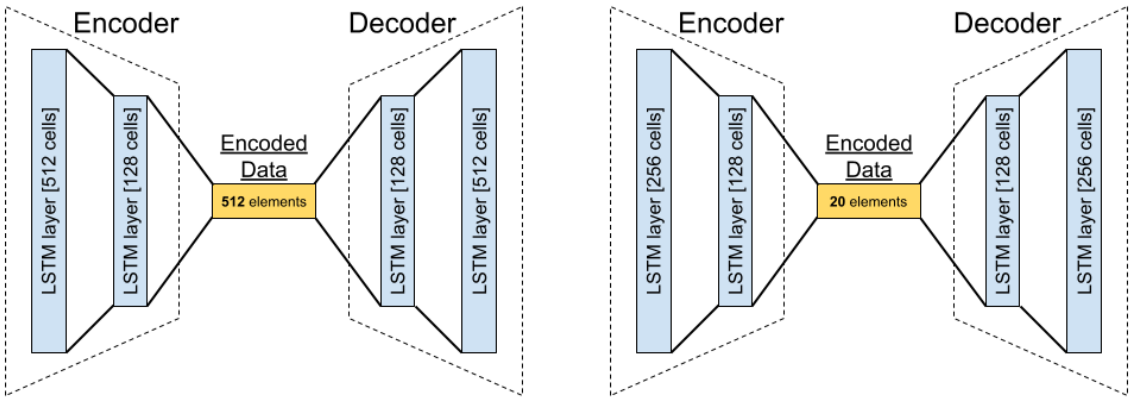


Figure 4.7: Classification metrics and false alarm rate for the LSTM network.

4.1.3. Autoencoders + XGBoost Model

The third black-box model implemented in this thesis results from the combination of the methods described in Section 4.1.1 and Section 4.1.2. The idea behind this solution exploits LSTM layers to process the time series, obtaining a compressed representation of



(a) Local Drivers Autoencoder (b) Global Drivers Autoencoder

Figure 4.8: Autoencoders Structure.

them. In addition, an ensemble method via XGBoost implementation combines multiple base models to produce a classifier that is complex enought to address the problem. The hybrid model produced can benefit from both architectures and improve performance on some of the predictive metrics considered.

As described in Section 3.1, the high dimensionality of the source dataset turns out to be one of the most challenging problems in capturing the intrinsic relationships among the data provided. In this thesis, an autoencoder model proved to be particularly effective in implementing a dimensionality reduction, in terms of computing effort and quality of the results obtained. Autoencoders are neural networks that can solve various problems, including dimensionality or noise reduction, feature extraction, and generative modeling. They are unsupervised methods that learn from unlabeled examples. These models aim to learn an identity function by minimizing a loss function, which, in this case, is expressed as the mean square error between the input sample and its reconstructed version. The process consists of two stages: compressing the original data to a latent representation and then decoding it to reconstruct the input provided. The model's structure is symmetrical, allowing for the reduction and expansion of data, composed of two sub-networks: the encoder and the decoder [42].

In this thesis, autoencoders are implied to decrease the dimensionality of the input time series that describe the ERA5 local drivers of the 10 days before the forecast and the time series that describe the global drivers of the 30 days before the forecast. The purpose is to compress these data into a proper latent representation using the encoder part of two pre-trained autoencoders that have learned the identity function of these time series. The layers employed for this purpose are based on the LSTM cells described in Section 4.1.2,

and their size was chosen after evaluating the correct trade-off between size reduction and loss minimization.

Figure 4.8 shows the structure of the two autoencoders, including the dimensions of the hidden layers, the number of cells in each layer, and the length of the latent vectors. The architecture of both autoencoders is designed to maintain the same ratio between the size of the input data and the length of the vector in which they are compressed. The ratio has a value of approximately 13 and is constant in both autoencoders to maintain the same dimensionality of the original data. This ratio is sufficient to reduce the input time series into a compressed version. The architectures demonstrate good reconstruction behavior, as evidenced by the mean square error between the input and output.

The input for the XGBoost model consists of the compressed representations of the previous days time series and data from the current day local variables. GB model is trained with labeled data, as described in Section 4.1.1, and generates future predictions at different time horizons. Hyperparameters values are obtained with the previously mentioned Optuna technique, that provides an approximation of the optimal solution in the related hyperparameter space. Figure 4.9 illustrates the combination of the two models to propose a hybrid solution for the problem.

Figures 4.10 and 4.11 show the performance evaluations. The negative class (TC absence) is predicted with large success, and there is a significant improvement in predicting the positive class (TC presence) compared to previous methods. This hybrid solution is highly effective in forecasting for the first three time horizons (24 hours, 48 hours, and 72 hours). Considering these three time horizons, Recall and Precision metrics are above 0.7 and False Alarm Rate is below 0.3. In addition, this method exhibits a behavior similar to that of traditional weather forecasting models, having high forecast reliability in the immediate future, which degrades as the forecast time window widens.

4.2. Post-Hoc Explainability Techniques

The use of black-box models in Machine Learning has significantly increased in recent decades. This trend poses the need for techniques to provide transparency to the predictive processes behind the complexities of these models. Post-hoc methods can provide support in terms of explanation for the predictions that black-boxes produce. The term *post-hoc* refers to the fact that these algorithms are applied after the models have passed the training phase, at inference time. This solution allows the predictive properties of black-box methods to remain intact and the explainability algorithms to be applied on different systems in an agnostic manner [57].

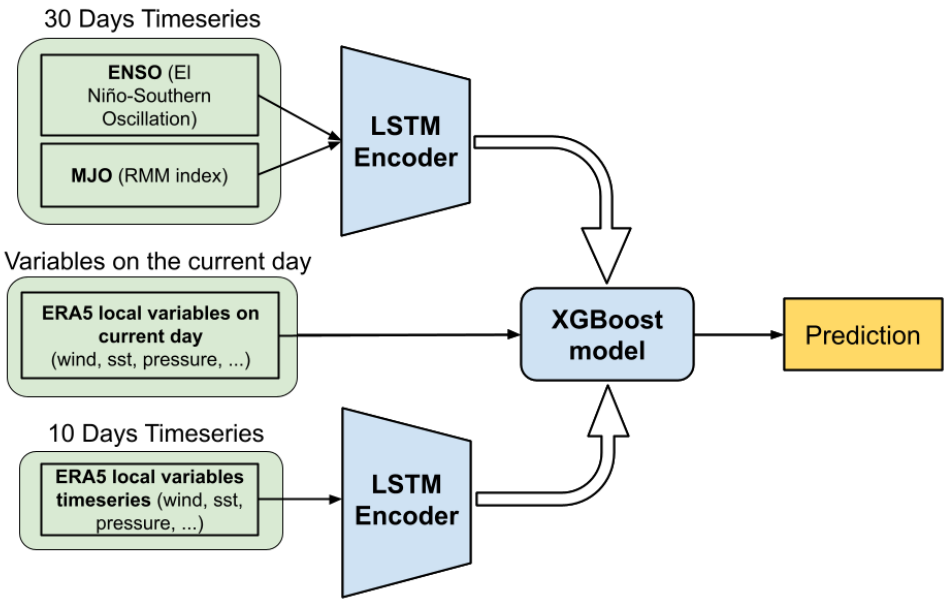


Figure 4.9: Hybrid model architecture.

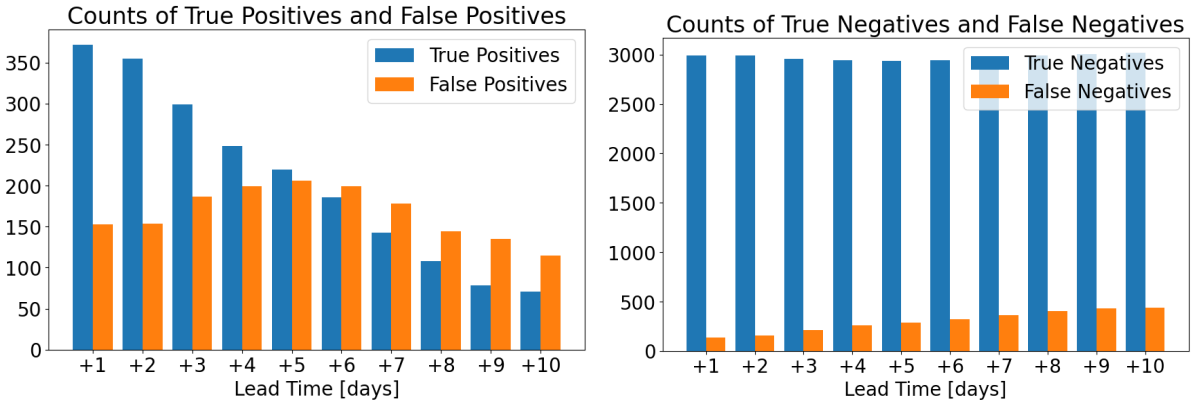


Figure 4.10: Classifications on the test set 2011-2021 for the Autoencoder+XGB model.

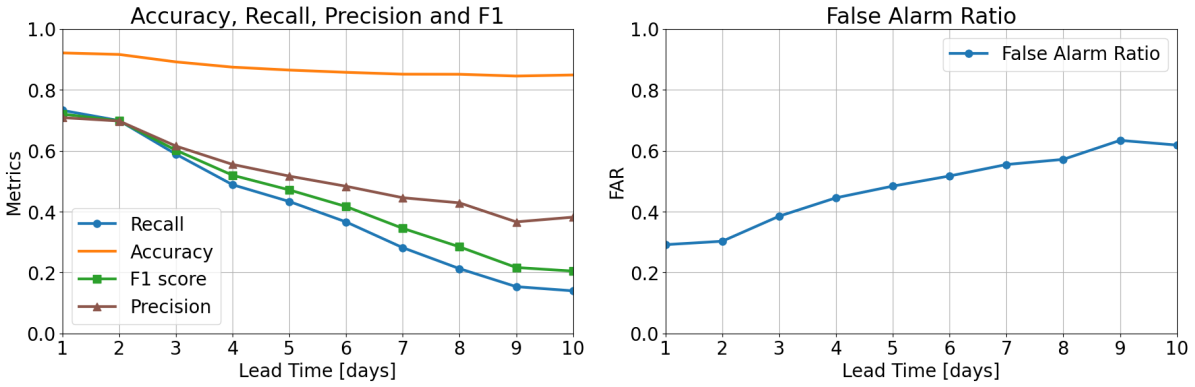


Figure 4.11: Classification metrics and false alarm rate for the Autoencoder+XGB model.

In this context, various solutions are proposed to describe the model's average behavior or provide explanations for each individual prediction. Post-hoc models generally provide a list of rules, weighted by their relevance, to give a qualitative understanding of the predictive process [71]. For the forecasting problem of this thesis, post-hoc outcomes can be useful for determining critical values above which cyclones are likely to occur or evaluating the relevance of variables and related rules in addressing the problem as a whole.

4.2.1. LIME for Explainable Predictions

Local Interpretable Model-agnostic Explanations (LIME) is a widely used explanation method for black-boxes models. It was developed by Ribiero et al. [64] and provides interpretability to any classifier by learning a model that is locally close to the prediction made.

LIME is based on two key concepts that defines the main properties of this technique:

- *Local* refers to the search space involved in the explainability process. LIME relies on the analysis of the behavior of the model in a neighborhood of a specific instance-prediction. This approach is the opposite of global XAI techniques which aim to provide explanations for the general behavior of a model.
- *Model-agnostic* refers to the ability of this technique to be applied to any predictive model, treating it as a black-box, without the need of understanding the learning processes or algorithms involved.

LIME's ability to explain each predictive instance is achieved by approximating black-box predictive behavior to simpler, and usually linear, models. To achieve this, a perturbation is applied to a selected instance of data, either through random sampling or by adding noise. The corresponding predictions of the black-box model are then obtained for the entire set of perturbed data. Finally, a simple model, such as a linear regressor, is trained using the perturbated data of the selected instance and the corresponding black-box outputs. The obtained simple model is an approximation of the black-box behavior in the specific neighborhood of the data point initially selected. It results in a more easily interpretable predictor that is used to determine the importance of each feature and the rules that ensure proper separation of the data into different classes. Figure 4.12 clearly displays this concept in a simple feature space. The two classes are correctly separated by a linear function locally in a neighbour of the highlighted red cross. The linear model is not able to classify samples in different portion of the plot, however, it fits perfectly locally for the specific data instance. From this simple linear model, LIME generates

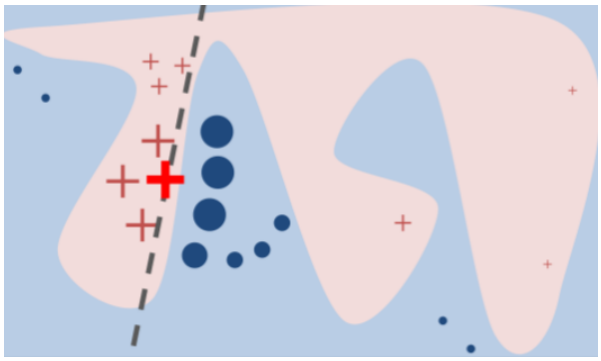


Figure 4.12: Linear model to approximate local behaviour in LIME [64].

rules that approximate the decision boundary of the complex model around the instance of interest.

The rules generated with tabular data are usually simple inequations that describe the threshold values of variables involved in the classification process. An example of a rule produced by LIME applied to the LSTM+XGBoost classifier is $Wind\_Gust > 18.7\ m/s$. Each of these rules are associated with an importance weight that specifies the contribution to the prediction of the instance under examination. Finally, it is relevant to notice that the number of rules produced with LIME is proportional to the number of features in the input data. Then, having a large dataset with many features produces a high volume of low-importance rules.

In the experiments done with this technique, LIME is applied to the model that performed best on the prediction metrics, namely the one described in Section 4.1.3. In addition, evaluations were made on the 24 hour forecast horizon, considering the most appropriate approach providing detailed descriptions for events that are predicted with high precision in the short term rather than for far future events predicted with low accuracy.

The significant contribution of all the features to the final prediction makes it difficult to evaluate a small number of explanatory rules. Considering black-box models implemented so far taking thousands features as inputs, the final prediction results from a combination of strongly non-linear functions applied to most of these variables. As a consequence LIME generates thousands of rules for each instance analyzed, making it impossible to define a subset of them as the ones mainly affecting forecasting process. For this reason, this technique has been used to evaluate, mainly quantitatively, the importance that each driver have on the final prediction, based on the number of rules involving a specific feature and the respective importance. Through this analysis, it is easy to understand the importance that the implemented model gives to the different datasets, meteorological variables, sub-areas from which they are derived, and steps in the time series.

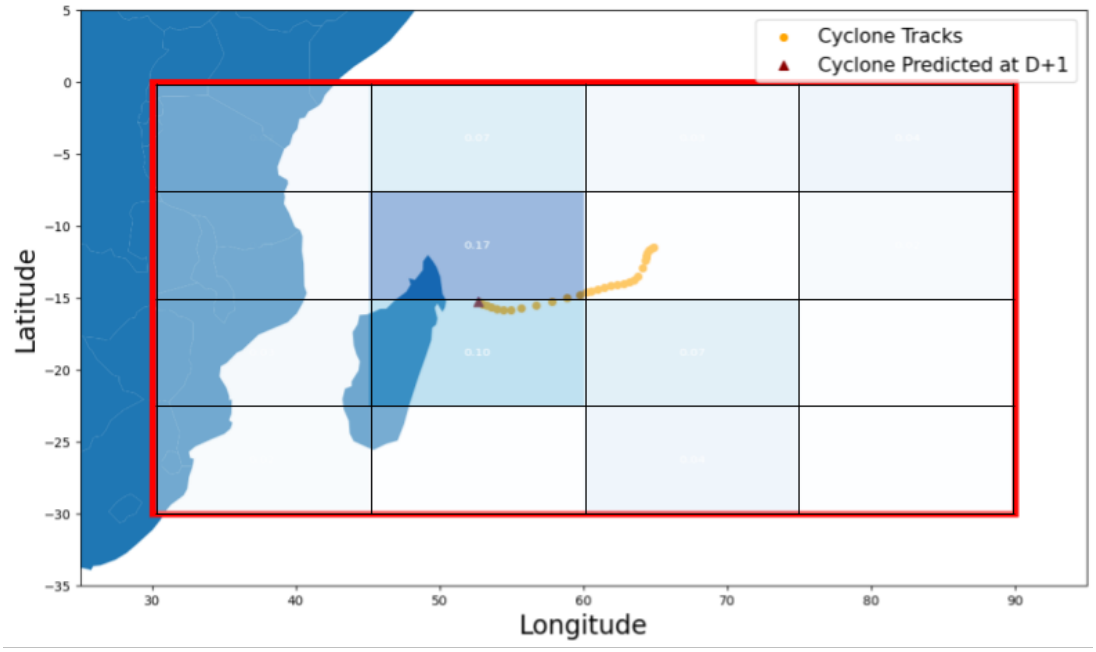


Figure 4.13: Contribution of the 16 zones to AVA cyclone prediction according to LIME (prediction on date 03-01-2018).

To provide an understanding of LIME's impact on each individual model prediction, some evaluations were obtained for a specific event. The sample considered for the following insights is part of the test set described in Section 4.1 and refers to the data instance for Cyclone AVA, which formed inside the target area on January 1, 2018, and dissipated on January 9 of the same year. All the samples for this specific TC are correctly predicted, being part of the true positive samples obtained from the test set.

Figures 4.13 shows the influence that certain subsets of variables have on the model's prediction. Specifically, in Figure 4.13, a heat-map is plotted showing the importance of features in each sub-zone according to the segmentation described in Section 3.2.2, based on the rules generated by LIME. This heat-map is aligned to the geographical representation of the target zone. Additionally, it is displayed the cyclone trajectory in the days prior to the forecast interval, obtained from the IBTrACS dataset. In this particular case, the variables related to the sub-zones where this TC moved, are the ones that contribute the most to the correct final forecast, confirming a model behavior that would be expected in such a situation.

Figure 4.14 displays the contributions of variables related to different datasets. It highlights the low influence of the time series related to global drivers for this specific forecast. Additionally, the local variables on the current day and the time series describing their trends over the previous ten days have a strong impact on the final outcome. Figure 4.15

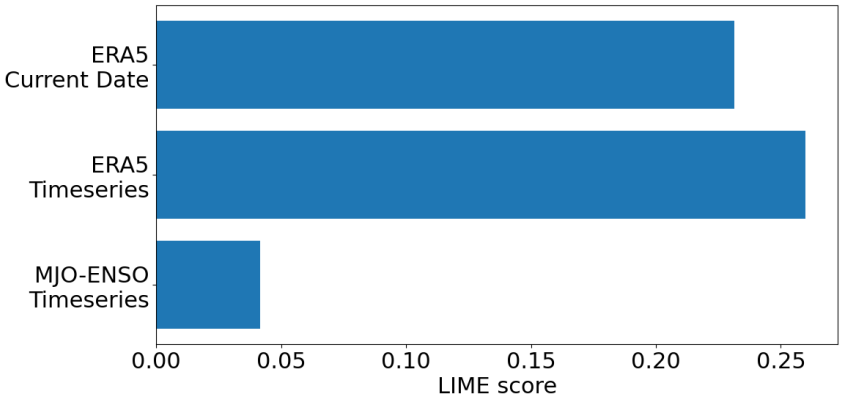


Figure 4.14: Contribution of different datasets to AVA Cyclone prediction according to LIME (prediction on date 03-01-2018).

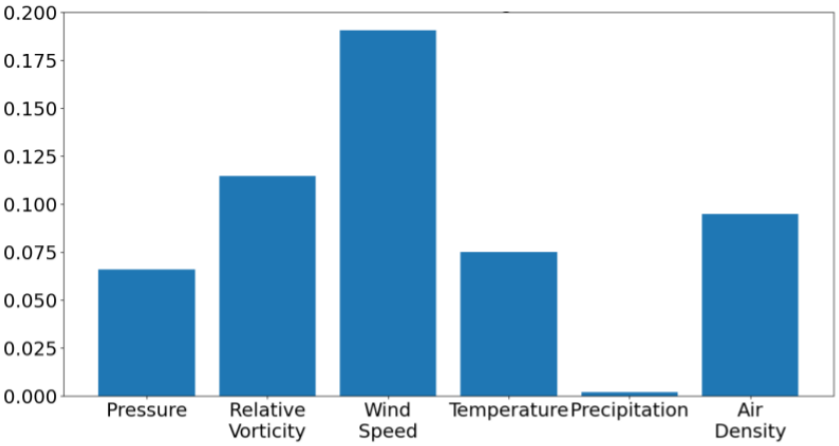


Figure 4.15: Contribution of meteorological variables to AVA Cyclone prediction according to LIME (prediction on date 03-01-2018).

illustrates the significance of specific meteorological variables in ERA5. Among these variables, winds and relative vorticities are the most descriptive of this cyclone, while precipitation is almost irrelevant.

4.3. Results and Discussion

This section analyzes the performance metrics of the different implemented models, comparing results obtained on the most relevant indicators. It also highlights the difficulties and disadvantages encountered when adopting LIME to describe predictions on different instances of the test set.

4.3.1. Prediction Metrics Comparison

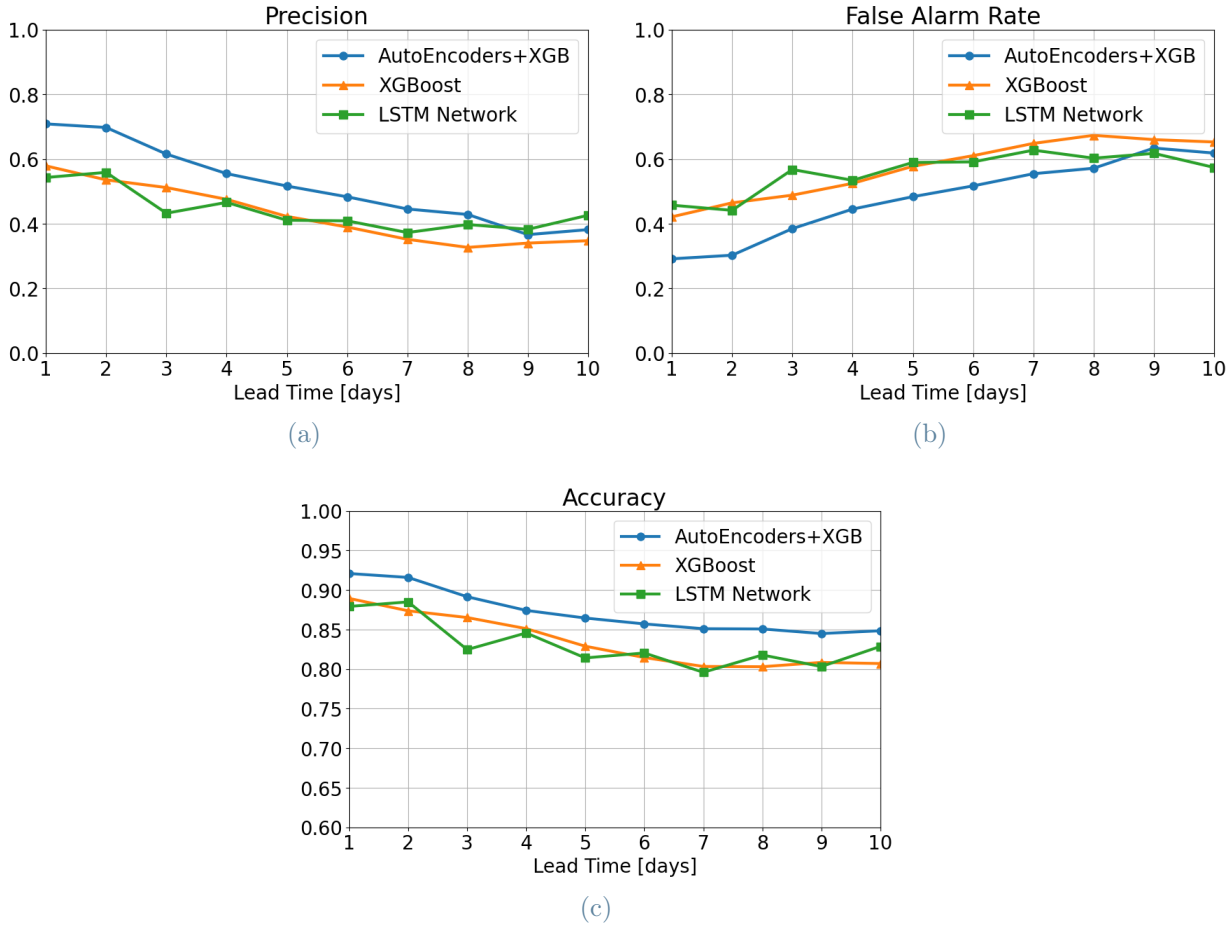
The specific predictive performance for each of the black-box models is discussed in detail in Section 4.1. However, for a direct comparison of each implementation on different metrics, Figure 4.16 provides an overall analysis of their capabilities by exploring predictions on different time horizons. This plot allows to derive observations in an intuitive way. The combination of XGBoost and LSTMs to build a hybrid model proved to be a useful solution in order to improve the robustness of the predictive model. The autoencoder-based model resulted in a deviation of approximately 10 % for the Precision and False Alarm Rate metrics compared to the other two solutions. To ensure the model behaves adequately for meteorological anomaly detection, it is important to aim for highly accurate forecasts with a low False Alarm Rate. This will result in a more trustworthy model that is less likely to produce forecast errors. Figure 4.16 (c) shows that the hybrid model has the highest accuracy of the three. However, as discussed in Section 3.4, the accuracy score is biased to the strong imbalance between the positive and negative classes, making it an unreliable indicator for overall model evaluation.

4.3.2. Limitations and Discussion on LIME

Although LIME can be an excellent method for providing interpretability for some test set instances, it is subject to limitations. Specifically, problems with the algorithm itself, such as the correct definition of a neighbourhood of a data instance [57], can arise, particularly if tabular data are involved, as in our case. One important aspect to consider regarding this technique is its strong instability. In fact, it is possible for very similar data points to produce significantly different descriptive rules, as discussed in [7].

Other limitations may be related to anomalies in some data instances. For example, there may be situations where more than one cyclone is present in the target zone on the same day or, as described in Section 3.1.3, some cyclones enter the target zone without being generated there. Although the model can classify some of these borderline cases accurately, anomalies such as these ones can lead LIME into providing inconsistent explanations.

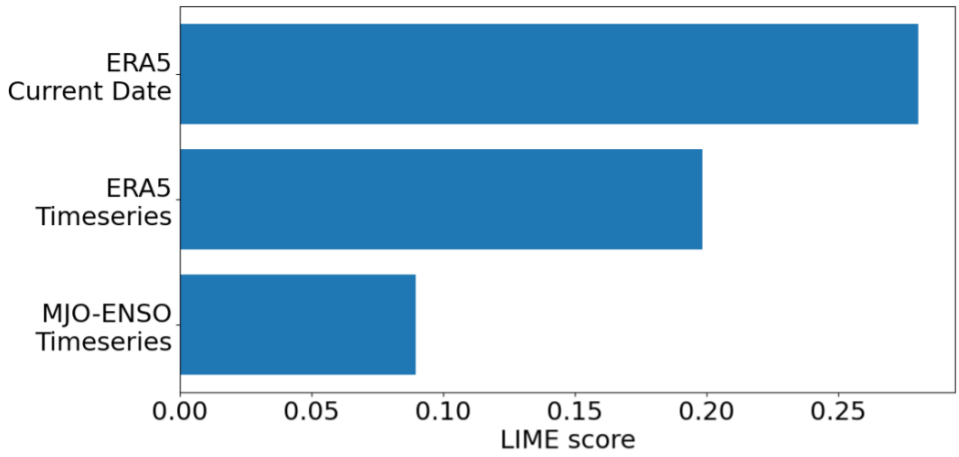
Despite its drawbacks, LIME enables us to gain a better understanding of how the implemented model works and how it is capable of capturing specific properties of cyclones. It is important to note that each predicted instance in the two classes may have unique characteristics, such as its location within the area of interest or the intensity of its activity. For instance, Figure 4.17 displays results from forecasting the Intense Tropical Cyclone Bejisa that occurred from January 27, 2014 to February 4, 2014 according to the



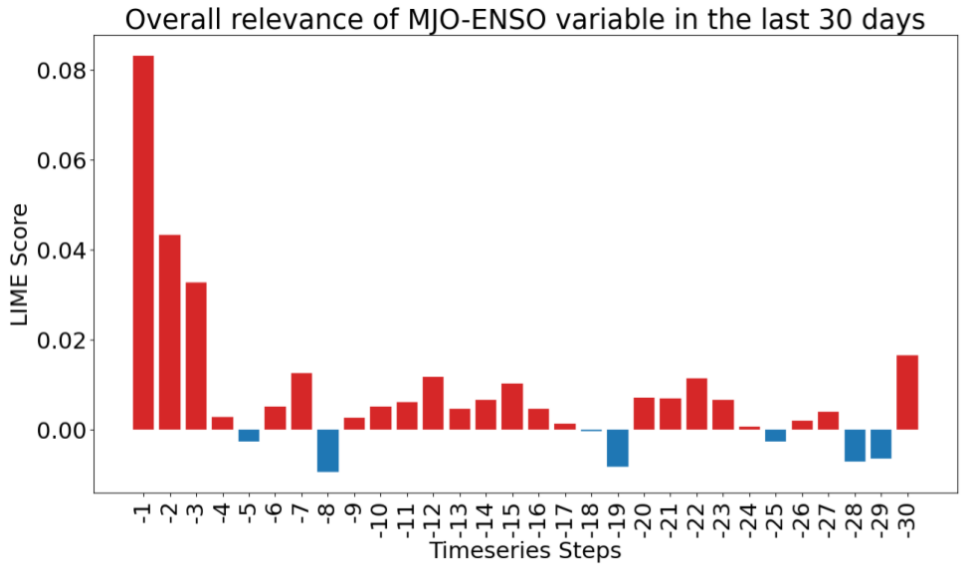
(a) (b) (c)

Figure 4.16: Comparison of Precision, False Alarm Rate and Accuracy on test set for the three black-box models implemented.

IBTrACS. This cyclone is a unique case as it coincides with the El Nino period, one of the major global-scale anomalies described in Section 2.1.2. The figure shows that the contribution of the variables relative to the global drivers is more relevant for this instance than in the previous analysis of the AVA cyclone in section 4.2.1. This confirms that the model we are using has captured the overall contribution of ENSO phenomenon to this specific TC activity.



(a) Contribution of local and global drivers to the predriction.



(b) MJO-ENSO variables relevance for each step of the timeseries.

Figure 4.17: Contribution of global drivers to Bejisa Cyclone prediction according to LIME (prediction on date 02-01-2014). Positive scores represent contribution to the *TC presence* class prediction while negative ones to the *TC absence* class

5 | White-Box Models

In Machine Learning, "white-boxes" refers to models that can provide transparency in the prediction processes. These methods supply a clear explanation of how they work, allowing experts in the application domain to gain additional knowledge about the resulting outcome. White-boxes often rely on simpler algorithms involved in the training and inference operations than those involved in the black-box domain.

Models based on traditional regressive methods, such as linear or logistic regression, are are an example of simple white-box methods. Monotonicity and linearity are the main properties that guarantee a high degree of explanatory power in such models. These two characteristics define the function that associates the input values to the prediction produced. In this case, the predictive function has an increasing or decreasing trend and a linear behavior. In such cases, it is straightforward to determine the threshold that these algorithms use to classify a new data point based on its features. However, they have limitations in capturing the nonlinear dependencies in complex datasets, such as the one used in this work.

In addition, these methods don't take advantage of the interactions between features, such as how the relationships between two or more input values of different features may affect the final predictions [57]. Therefore, it is necessary to rely on techniques that may be more effective in the forecasting scenario of this thesis, such as decision trees or rule-based methods.

5.1. Dimensionality Reduction for White-Boxes

White-box models are interpretable by construction, but they may struggle to capture the necessary patterns for accurate classification in high-dimensional data [58]. As stated in Chapter 3, the reference dataset used in this work includes records with high dimensionality, which makes it challenging to define suitable white-boxes. For example, when a white-box model identifies patterns among thousands of features, its complexity becomes correlated with this dimension [60]. As data complexity increases and more features are

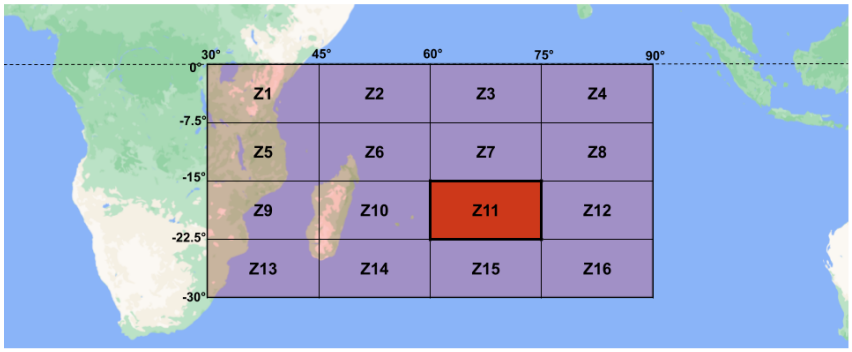


Figure 5.1: New target area for the white boxes models.

involved, the structure of white-box models grows to provide more accurate predictions. However, rule-based or path-based models may hinder the interpretation of their global behavior and the understanding of long and complex decision paths for classifying positive examples, such as the presence of TCs. This limitation prevents the utilization of all available data from the original datasets, as successfully demonstrated with the black-box models in Chapter 4. However, observations on dimensionality reduction and feature selection techniques can be helpful in effectively reducing the size of the source dataset. In particular, the Kernel Density Estimation and Features Selection, described in Sections 3.2.2 and 3.3.1, are the approaches supporting further decisions about selecting and reducing the features involved.

The first reduction is applied to the geographical distribution of samples, re-defining the target area used for the TCs classification task. In particular, the Z11 area, described in Section 3.3.1, is considered a new sub-basin target zone. Z11 covers an ocean surface of more than one million km^2 with coordinates of 60°W, 75°W, 15°S and 22.5°S. The choice of this particular area is the result of the features selection and the TCs distribution analysis. According to the description in Section 3.3.1, Z11 is one of the most expressing zones for TCs detection in the entire South-West Indian Ocean basin. Additionally, the high number of TCs collected in the IBTrACS from 1980 to 2022 makes it a perfect candidate. Figure 5.1 displays the new target zone, with a reference for coordinates and position inside the previously considered target area.

In the second step, a further reduction of the local and global drivers is carried out. Large-scale drivers are not considered anymore, as their contribution to final predictions is too specific for particular seasonal conditions. On the other side, only a subset of local drivers is selected, according to the tree-based feature selection results shown in Figure 3.8. Table 5.1 displays the variables with the corresponding atmospheric levels. These features compose the source dataset to train and evaluate the white-box models. For each

meteorological driver, we consider the mean and standard deviation of all data points in Z11 as valid contributors. Therefore, the dataset consists of 16 meteorological contributors and one temporal indicator (Day of the Year), which specifies the number of days in the year for each record. The new dataset has 17 values per record, significantly fewer than the source. This reduction is sufficient for obtaining highly expressive white-box models.

Additionally, the white-box forecasting methods only evaluate variables for the present day, without considering the contribution of previous time steps. The training and evaluation phases employ the same train-test dataset partitioning, as shown in Table 4.2 for black-box implementations.

The following sections discuss the implementation of different models and the related results. They include two inherently explainable techniques that provide global rules or decision paths to detect TCs in a detailed scenario. These are the traditional Decision Trees (DT) and the Bayesian Rule Lists (BRL). Section 5.4 provides a detailed comparison between the two methods, evaluating how different approaches yield the same qualitative predictive results.

Variables	Atm Levels
Surface Pressure	-
Sea Surface Temperature	-
Relative Vorticity	850hPa
Air Density	-
Wind Gust Speed	-
Wind Speed	1000hPa, 850hPa, 300hPa
Day of the Year	-

Table 5.1: Selected variables for white-box models implementation.

5.2. Decision Trees

Decision trees are a commonly used ML algorithm for solving classification and regression tasks. As shown in Figure 5.2, they are sequential methods that partition the dataset through a series of decision nodes. Each internal node of the tree evaluates a single attribute of the input sample against a defined threshold. Different branches are produced for each of the possible outcomes of this evaluation. The leaf nodes contain the outcomes of the model [41], which is one of the two binary classes adopted for TCs detection.

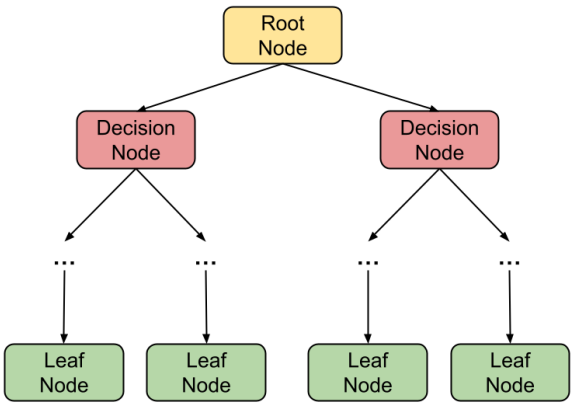


Figure 5.2: Structure for a Decision Tree model.

Various construction algorithms have been developed over the years to learn a decision tree structure from a training set. In this thesis, the CART algorithm is considered for tree composition [11]. CART splits the source dataset into different subsets, selecting the node with the lowest Gini impurity at each iteration. Gini impurity is a measure used to evaluate the quality of a split. It measures the probability of incorrectly classifying a randomly chosen element in the dataset if it were randomly labeled according to the class distribution in the subset. The algorithm ends when reaching a stopping criterion, such as a maximum depth for the generated tree or a minimum number of samples required to be present in a leaf node.

The trees produced by this algorithm offer valuable insights into the explanations for TC classification. Specifically, we can identify the importance of the features closest to the root nodes, which are the ones that most effectively separate the dataset. Furthermore, we can visualize the numerical thresholds that identify the decision paths to classify a sample with the presence of TC.

Defining appropriate values for the hyperparameters that determine the tree structure is crucial. In particular, it is essential to limit the maximum depth of the tree to produce a globally interpretable model. Models with excessive depth, such as those with hundreds of nodes, may have opaque classification behavior from the perspective of the application domain experts who use them. Additionally, it is relevant to find the appropriate balance between model complexity and generalization to avoid critical phenomena such as overfitting. This work identifies that limiting the maximum depth of the tree is an acceptable choice to achieve explainability and generalization over the testing set.

Figure 5.3 shows the tree structure produced by limiting the maximum depth to six. Each node indicates the class and the number of samples involved. The highlighted leaves

represent the positive (TC presence) classification of samples. Table 5.2 displays the prediction performance on the same metrics adopted to evaluate the black-box models. The reported results indicate that this model hardly provide precise predictions for cyclone detection beyond 24 hours. This behavior may be due to the simplifications made to obtain a more comprehensible tree, such as reducing the dimensionality of the dataset or imposing limitations on the structural definition of the model, such as maximum depth. For the following comparisons with other transparent methods, only the models trained to 24 hours horizon are taken into consideration, since they appear to be the most effective ones.

Metrics	Day+1 (24 h)	Day+2 (48 h)	Day+3 (72 h)
Accuracy	0.98	0.96	0.95
Precision	0.80	0.59	0.17
Recall	0.66	0.30	0.02
F1-Score	0.72	0.40	0.04
False Alarm Rate	0.20	0.41	0.83

Table 5.2: Performance metrics for the first three forecasting horizons.

5.3. Bayesian Rule Lists

Methods based on rule lists identify another possible option for white-box models. Rule-based techniques are highly interpretable due to the ability of rules to represent global behaviors through expressions close to the natural language. The rules generated by these models consist of IF-THEN statements, where the IF introduces one or more conditions, representing a partition of the source dataset defined. The second part of the rule specifies the outcome of the predictive process. The cascading contributions of each of these rules compose the global classification function.

In this thesis, the Bayesian Rule Lists (BRLs) are adopted as the reference for rule-based methods. They combine Bayesian inference techniques and rule-based systems to produce a highly interpretable model [46]. Compared to the DT adopted in the previous section, BRL offer a different representation of the classification paths. The rules generated in this model do not represent conjunctions of valid conditions to reach a leaf, as in the case of trees. Instead, they are conjunctions of negated predicates before the asserted one, which enables identification in one of the classes with a related probability. Table 5.3 provides an example of the differences between the rules generated with the BRL classifier and the

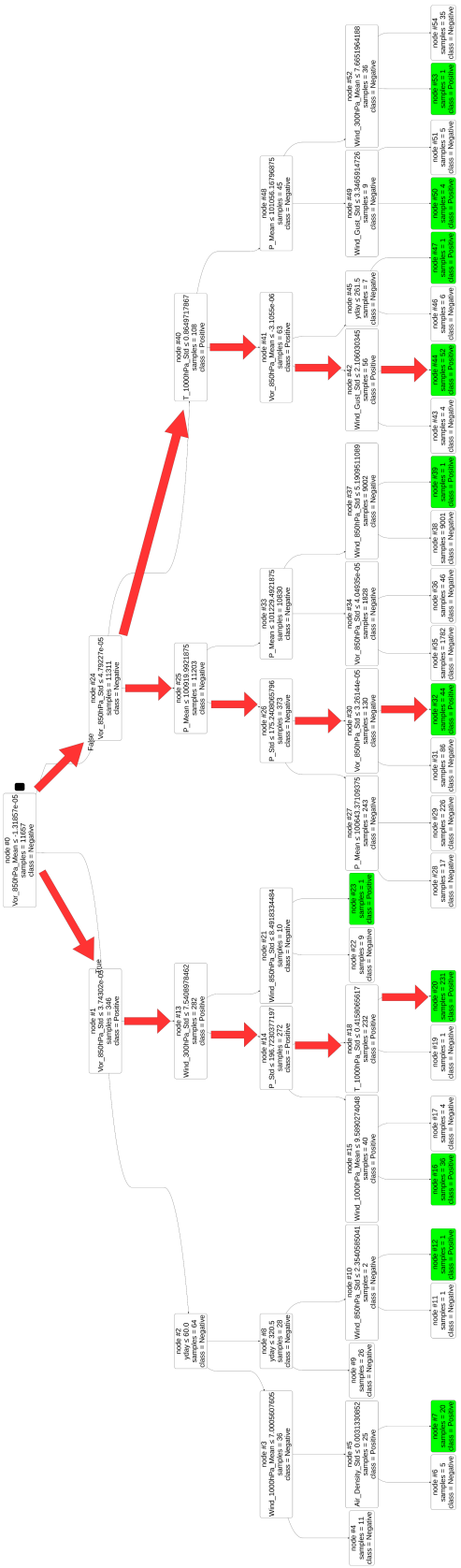


Figure 5.3: Decision Tree Classifier for TCs in Zone 11. The three decision paths that cover the majority of samples are highlighted in red.

rules extracted from a classification path in the DT, to classify positive examples (TC presence).

Models	Classification Rules
Decision Tree	$\text{Vor\_850hPa\_Mean} \leq -1.32 \cdot 10^{-5} \wedge$ $\text{Vor\_850hPa\_Std} > 3.74 \cdot 10^{-5} \wedge$ $\text{Wind\_300hPa\_Std} > 7.54 \wedge \text{P\_Std} > 196.72 \wedge$ $\text{T\_1000hPa\_Std} \leq 0.42$
Bayesian Rule List	$\neg\ (\text{P\_Mean} > 101230 \wedge \text{Wind\_Gust\_Std} \leq 2.11) \wedge$ $\neg\ (\text{P\_Mean} > 101230 \wedge \text{Vor\_850hPa\_Std} < 3.26 \cdot 10^{-5}) \wedge$ $\neg\ (\text{P\_Std} \leq 175 \wedge \text{Vor\_850hPa\_Mean} \geq -3.11 \cdot 10^{-6}) \wedge$ $\neg\ (\text{Vor\_850hPa\_Std} \leq 3.26 \cdot 10^{-5}) \wedge$ $\neg\ (\text{P\_Mean} > 101230 \wedge \text{T\_1000hPa\_Std} > 0.92) \wedge$ $\neg\ (\text{Vor\_850hPa\_Mean} \geq -3.11 \cdot 10^{-6}) \wedge$ $\neg\ (\text{P\_Std} \leq 175) \wedge$ $(-1.32 \cdot 10^{-5} < \text{Vor\_850hPa\_Mean} \leq -3.11 \cdot 10^{-6})$

Table 5.3: An example of the two classification rules produced by DT and BRL models to classify positive examples (TC presence).

BRLs rely on probabilistic analysis of logical rules derived from a dataset. When features have continuous values, such as local drivers variables, calculate the probability for each possible value can be complicated. Therefore, it is essential to discretize the dataset and simplify the process. For this thesis we conducted several experiments to partition the continuous values of features into discrete bins using different evaluation methods. Each variable was discretized using different strategies, including uniform discretization, quantile discretization, and discretization by k-means clustering[1]. No significant differences are observed among the three discretization strategies, so the uniform one is chosen for its simplicity. Based on the results obtained from various implementations, it appears that six bins provide an effective division.

The training process for BRL involves multiple steps. First, frequent patterns in the data are analyzed to pre-mine a set of most relevant rules. This pre-processing step typically employs algorithms such as Apriori or FP-Growth [4, 30]. Bayesian inference is then applied to calculate the posterior distribution of rules. Its computation involves the prior

[1] *imodels* is the python library used for BRLs and discretization techniques (`https://csinva.io/imodels/discretization/discretizer.html`).

Trained RuleListClassifier

IF	P_Mean > 101230	and Wind_Gust_Std <= 2.11	THEN probability of class 1: 0.3% (0.2%-0.5%)
ELSE IF	P_Mean > 101230	and Vor_850hPa_Std <= 3.26e-05	THEN probability of class 1: 0.6% (0.2%-1.1%)
ELSE IF	P_Std <= 175	and Vor_850hPa_Mean >= -3.11e-06	THEN probability of class 1: 2.2% (1.5%-3.0%)
ELSE IF	Vor_850hPa_Std <= 3.26e-05		THEN probability of class 1: 10.7% (8.6%-13.1%)
ELSE IF	P_Mean > 101230	and T_1000hPa_Std > 0.92	THEN probability of class 1: 6.0% (2.5%-11.0%)
ELSE IF	Vor_850hPa_Mean >= -3.11e-06		THEN probability of class 1: 26.4% (15.6%-38.9%)
ELSE IF	P_Std <= 175		THEN probability of class 1: 29.4% (22.2%-37.1%)
ELSE IF	-1.32e-05 < Vor_850hPa_Mean <= -3.11e-06		THEN probability of class 1: 55.8% (48.5%-62.9%)
ELSE IF	Wind_850hPa_Mean <= 11.5		THEN probability of class 1: 82.8% (75.6%-88.9%)
ELSE IF	P_Std > 196		THEN probability of class 1: 97.8% (94.8%-99.5%)
ELSE			probability of class 1: 33.3% (1.3%-84.2%)

Figure 5.4: Bayesian Rule List classifier for TCs in Zone 11.

distribution of each rule multiplied by the likelihood of observing the data, given the rule. In conclusion, rules that maximize the posterior distribution are selected, prioritizing the minimum number of rules involved and their minimum length.

Figure 5.4 shows the model resulting from this training process. The source dataset employed in the training and evaluation phases is the same as described in Section 5.1. The list of rules comprises ten logical rules, each one containing two predicates at most. The posterior distribution's value for each of the rules in the list represents the probability of classifying the related input to the positive class. In addition, each rule includes a range of this probability. This range specifies the lower and upper probability bounds for samples to be classified into the positive class by the specific rule. The list classifies the presence of cyclones with higher probability the deeper it goes into the rules. The final ELSE statement identifies examples where a specific pattern in the data was hard to find and no specific logical rule is related to them. It is the negation of all the previous rules in the list. The probabilistic range for this case covers both negative and some positive samples.

Although not being able to find relevant rules for all the TCs, the performance metrics show partially satisfactory results. Table 5.4 provides detailed information on the performance evaluation of this model, which seems consistent with the metrics scores obtained using decision trees. As previously mentioned, predictions beyond 24 hours are too imprecise and lack predictive capabilities. Therefore, they are not included in this evaluation.

Metrics	Day+1 (24 h)
Accuracy	0.98
Precision	0.78
Recall	0.70
F1-Score	0.74
False Alarm Rate	0.22

Table 5.4: Performance metrics for the BRL model in the 24 h prediction.

It is important to note that BRL training is computationally intensive due to the highly procedural learning algorithm, which does not allow for efficient parallelization of the process. Section 5.4 addresses further comparative analyses of the different training period behaviors.

5.4. Results and Discussion

The two white-box techniques proposed for detecting TCs in the Z11 area achieved comparable results on the evaluation metrics for predictions with a forecasting horizon of 24 hours. The recall values indicate that approximately 70% of the cyclones in the test set are correctly classified, while the FAR level shows that 20% of the positive examples predicted are false alarms. Overall, the performance results are lower than those obtained with a Gradient Boosting black-box model on the Z11 area, as it is reported on Table 5.5. This is due to the higher complexity of black-boxes that cover a wider range of test cases. In addition, it is relevant to note that white-box techniques are implemented after appropriate problem reduction and are not suitable to cover large-scale generalizations. The target region used for DTs and BRLs is a limited portion of the entire South West basin of the Indian Ocean and could be strongly subject to bias. Then, it is difficult to state that the same rules and paths derived from white-boxes in this thesis can detect TCs in the entire basin. However, they can provide relevant insights into the drivers and threshold values that characterize this phenomenon in this specific condition.

Metrics	Decision Tree	Bayesian Rule Lists	Gradient Boosting
Accuracy	0.98	0.98	0.99
Precision	0.80	0.78	0.88
Recall	0.66	0.70	0.82
F1-Score	0.72	0.74	0.85
False Alarm Rate	0.20	0.22	0.12

Table 5.5: Performance metrics compared between white-box (DT and BRL) and black-box (GB) models in the 24 h horizon and Z11 area.

The structures of DTs and BRLs exhibit similar patterns. BRLs can be transformed into DTs where rules identify internal nodes and leaves represent the probabilities of positive classifications associated with each of them. Therefore, the overall tree structure of BRLs is an unbalanced cascading tree. Considering BRLs in the form of unbalanced trees is helpful to exctract the decision paths generated by this model. As a result, the DTs and BRLs can be directly compared with the similarities and differences in the extracted paths.

By analyzing the identified patterns of the two classifiers in detail, it is possible to make observations on the variables that contribute the most. Exploring DT structure in Figure 5.3, it can be seen that eleven paths attribute classification in the positive class. Many of them do not generalize across multiple samples, identifying the classification of only one or a few examples. However, the three most relevant paths are capable of addressing the majority of the cyclones identified, making them the most effective rules for describing TCs in Z11 through DTs. The Bayesian rules, on the other hand, rely on the three rules highlighted in Figure 5.4 to trigger positive classifications. Each of these rules, along with the negation of the previous ones, represents a path to address TCs detection.

In Table 5.6 the classification patterns are derived from the two models and converted into conjunctions of first-order logic rules. For the DT only the three paths that identify the majority of TCs are displayed. It is evident that many of them exhibit similarities, and some are nearly identical. When comparing the two rules that classify positive examples with a higher probability for the two different models, some variables are related to the same threshold value, such as mean relative vorticity or standard deviation of surface pressure. Additionally, there are still consistent rules in other cases, such as mean surface temperature and standard deviation of vorticity. Furthermore, the rules generated by the Bayesian model are typically longer. Conversely, those produced by the tree

algorithm explore the feature space more extensively and are restricted in length by the hyperparameter that specifies the maximum depth of the tree.

When comparing DTs and BRLs, it is relevant to consider the computational effort required for the training on the data. The two methods use different approaches to search the hypothesis space for the model that optimizes classification capabilities. Constructing decision trees is generally an efficient and easily scalable process, even when handling large datasets. In contrast, building Bayesian lists can be expensive and complicated to scale to datasets with large samples and features. These differences are observed by comparing the training time of the two methods, evaluated with equal computational power. Decision trees take just over 140 ms to be generated, while the Bayesian lists have a training period of more than 11 minutes. Considering these aspects is crucial when evaluating models with similar predictive outcomes. It enables experts to make informed decisions on models that may be suitable for scaling on larger datasets.

In conclusion, white-box models demonstrate effectiveness in gaining detailed knowledge of patterns that identify TCs in local environmental data. The obtained results demonstrate the potential of these models for forecasting extreme events. DTs and BRLs produce explanations related to the intrinsic nature of their model structure. Therefore, they are more effective in providing detailed rules than the LIME technique applied to black-box models. As described in Section 4.3.2, LIME can be highly limited to provide clear and reliable explanations. These considerations emphasize the importance of finding the appropriate balance between model complexity to have accurate predictors and highly interpretable methods that can provide detailed descriptions of how the model works.

Decision Tree Paths	Probability	Number of Samples
Vor_850hPa_Mean $\leq -1.32 \cdot 10^{-5} \wedge$ Vor_850hPa_Std $> 3.74 \cdot 10^{-5} \wedge$ Wind_300hPa_Std $> 7.54 \wedge$ P_Std $> 196.72 \wedge$ T_1000hPa_Std ≤ 0.42	96.1%	231
$-1.32 \cdot 10^{-5} <$ Vor_850hPa_Mean $\leq -3.11 \cdot 10^{-6} \wedge$ Vor_850hPa_Std $> 4.79 \cdot 10^{-5} \wedge$ T_1000hPa_Std $\leq 0.87 \wedge$ Wind_Gust_Std > 2.11	88.46%	52
Vor_850hPa_Mean $> -1.32 \cdot 10^{-5} \wedge$ $3.26 \cdot 10^{-5} <$ Vor_850hPa_Std $\leq 4.79 \cdot 10^{-5} \wedge$ P_Mean $\leq 100920 \wedge$ P_Std > 175.24	61.36%	44
Bayesian Rule Lists	Probability	Probability Range
P_Mean $\leq 101230 \wedge$ Wind_Gust_Std $> 2.11 \wedge$ Vor_850hPa_Std $> 3.26 \cdot 10^{-5} \wedge$ T_1000hPa_Std $\leq 0.92 \wedge$ Vor_850hPa_Mean $\leq -1.32 \cdot 10^{-5} \wedge$ Wind_850hPa_Mean $> 11.5 \wedge$ P_Std > 196	97.8%	$(94.8\% - 99.5\%)$
P_Mean $\leq 101230 \wedge$ Wind_Gust_Std $> 2.11 \wedge$ Vor_850hPa_Std $> 3.26 \cdot 10^{-5} \wedge$ P_Std $> 175 \wedge$ T_1000hPa_Std $\leq 0.92 \wedge$ Vor_850hPa_Mean $\leq -1.32e-05 \wedge$ Wind_850hPa_Mean ≤ 11.5	82.8%	$(75.6\% - 88.9\%)$
P_Mean $\leq 101230 \wedge$ Wind_Gust_Std $> 2.11 \wedge$ Vor_850hPa_Std $> 3.26 \cdot 10^{-5} \wedge$ P_Std $> 175 \wedge$ T_1000hPa_Std $\leq 0.92 \wedge$ $-1.32 \cdot 10^{-5} <$ Vor_850hPa_Mean $\leq -3.11 \cdot 10^{-6}$	82.8%	$(75.6\% - 88.9\%)$

Table 5.6: Classification rules for BRLs and DTs.

6 | Conclusion and Future Work

This final chapter summarizes the goal of this thesis and discusses the questions that remain unanswered. It also highlights the main limitations encountered during the development of the solutions. In conclusion, some ideas are proposed for future work to improve the knowledge of interpretability of Machine Learning models applied to the analysis of Tropical Cyclones.

6.1. Goals and Open Questions

The main goal of this thesis is to assess the potential of Explainable Artificial Intelligence in enhancing interpretability for forecasting extreme weather events. Specifically, we focus on predicting the presence of Tropical Cyclones in the South-West Indian Ocean basin.

Scholars often prioritize predictive performance over system transparency, which can be problematic in fields like environmental sciences and medical applications where technology supports human decision-making processes. For this reason, ensuring that systems are transparent and understandable to the involved domain experts, should be a priority for future works. In recent years, data-driven approaches have been widely adopted to predict anomalous environmental events, making them effective contributors to improve forecasting accuracy. However, many implemented systems represent black-boxes, making it impossible to observe their behavior. Few works have contributed to the transparency and the deep description of how these systems act in the TCs detection. In this context, this thesis aims to contribute to the assessment of traditional interpretable methods for forecasting and describing TCs activities.

The analysis of the results in this thesis identifies unsolved problems in the context of TCs forecasting. Currently, effective long-term prediction of this phenomenon is not possible. As noted in the results described in Section 4.3 and 5.4, broadening predictive horizons renders the process unfeasible. Another unresolved aspect is the ability to create models that can be robust for different regions of the planet. Cyclones develop in various basins, and all the considerations made in this thesis take only the South-West Indian Ocean

basin into account. The unique geographic and environmental conditions of each cyclone basin bring to the generation of anomalies that are profoundly different. It is then difficult to adequately generalize the problem over multiple geographic regions.

6.2. Limitations

Predicting cyclones is a complex problem subject to several limitations. Some of them concern data quality, complex cyclone dynamics, and non-linearities associated with meteorological variables.

One of the limitations that mainly affects the results in this thesis concerns the data dimensionality. The ERA5 reanalysis system described in Section 3.1.2, provides data on local drivers with high-resolution and detailed descriptions of the target area. Previous experiments have demonstrated that the ERA5 records effectively describe TCs activity [10]. However, the high dimensionality of the data poses a challenge when simple and interpretable models are required. Therefore, this work aims to reduce dimensionality and make the learning process manageable with limited computational power.

Additionally, the modeling process involves a series of approximations when white-box models are considered. It is essential to reduce the number of features to produce rule-based models that are easily interpretable. As a consequence of these evaluations, the white-box methods are limited to cover a specific sub-region of the South-West Indian Ocean basin. Therefore, the paths and rules mined can't be generalized to the entire target area. Other relevant aspects may comprise some specific techniques that are applied to improve the interpretability of predictions made by black-box methods. For instance, the post-hoc technique LIME, described in Section 4.2.1, presents some limitations, due to its instability and the difficulties in describing models' behaviors when large datasets are involved.

It is also important to consider limitations related to the generation of cyclones inside or outside the target region. As described in Section 3.1.3, some anomalies in the dataset used for classification labels represent the presence of cyclones that form outside the target area and subsequently move within it. In more rare occasions, the reverse condition occurs. These anomalous behaviors introduce significant noise into the data, which can hinder the model's ability to generalize effectively on those samples that describe the genesis of a cyclone.

6.3. Future Work

In the context of predicting TCs, future work may explore additional aspects to improve knowledge about the interpretability of the data-driven methods applied. One approach could be to redefine the problem and evaluate it from a different perspective. For instance, a sensible choice could be analyzing how the activity of already formed cyclones changes, focusing on tracking or intensity variation. This approach can be advantageous because it allows the use of meteorological data related to mature extreme events that focus on a restricted area concerning cyclone activity. By using a cleaner, less noisy, and dimensionally-limited dataset, it may be possible to achieve good results in terms of predictability and explainability of the phenomenon. Additionally, forecasting the intensity variation or geographic location of TCs is one of the most relevant aspects of mitigating the damages caused by catastrophic events.

Further exploration could include the investigation of a correlation between global drivers and large-scale cyclone variations. This thesis is ineffective in establishing a concrete correlation between global meteorologic oscillations and the predictions produced by the models. This result is mainly due to the definition of the forecasting problem. The short-term forecast through binary classification demonstrates that the contribution of global drivers doesn't particularly impact results. As described in several papers [12], global anomalies such as MJOs or ENSOs can affect the seasonality of cyclones in specific areas. Considering this, possible future works could assess how these phenomena affect the frequency of cyclones during a particular period of the year, specifically during the cyclonic seasons. This analysis may provide insight into the extent of large-scale phenomenon impact on TCs development.

Finally, it is worth considering that this work did not utilize all the current state-of-the-art methods for various predictive problems. Future work may explore the implementation of deep models, such as Deep Convolutional Networks, to learn from a large and structured dataset. An approach to address the problem with these models could be the treatment of the target area as a 3-dimensional matrix, with latitude, longitude, and meteorological variables as its dimensions. Additionally, it may be beneficial to learn from time series data using advanced techniques such as the attention mechanism. This approach emphasizes the input values that the model prioritizes, enabling interpretive techniques to be derived from the resulting predictions.

Bibliography

[1] Explanation on the products for the madden julian oscillation (mjo). URL https://ds.data.jma.go.jp/tcc/tcc/products/clisys/mjo/explanation_mjo.html.

[2] The accumulated cyclone energy (ace) index, 2002. URL https://www.cpc.ncep.noaa.gov/products/outlooks/hurricane2003/August/background_information.html.

[3] Extreme events detection. Technical report, CLINT - Climate Intelligence, 2022. URL https://131.175.15.9/share.cgi/CLINT_D31_ECMWF_WP3_F_EE_Detect.pdf?ssid=e915a0c903ff4e1caff808e7aa33e245&fid=e915a0c903ff4e1caff808e7aa33e245&open=normal&ep=.

[4] R. Agrawal, R. Srikant, et al. Fast algorithms for mining association rules. In *Proceedings of the International Conference on Very Large Data Bases, VLDB*, volume 1215, pages 487–499. Santiago, Chile, 1994.

[5] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama. Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2623–2631, 2019.

[6] S. Alemany, J. Beltran, A. Perez, and S. Ganzfried. Predicting hurricane trajectories using a recurrent neural network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 468–475, 2019.

[7] D. Alvarez-Melis and T. S. Jaakkola. On the robustness of interpretability methods. *arXiv preprint arXiv:1806.08049*, 2018.

[8] A. B. Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. García, S. Gil-López, D. Molina, R. Benjamins, et al. Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. *Information Fusion*, 58:82–115, 2020.

[9] C. Bentéjac, A. Csörgő, and G. Martínez-Muñoz. A comparative analysis of gradient boosting algorithms. *Artificial Intelligence Review*, 54:1937–1967, 2021.

[10] G.-F. Bian, G.-Z. Nie, and X. Qiu. How well is outer tropical cyclone size represented in the era5 reanalysis dataset? *Atmospheric Research*, 249:105339, 2021.

[11] L. Breiman. *Classification and regression trees*. Routledge, 2017.

[12] S. J. Camargo and A. H. Sobel. Western north pacific tropical cyclone intensity and enso. *Journal of Climate*, 18(15):2996–3006, 2005.

[13] S. J. Camargo, M. K. Tippett, A. H. Sobel, G. A. Vecchi, and M. Zhao. Testing the performance of tropical cyclone genesis indices in future climates using the hiram model. *Journal of Climate*, 27(24):9171–9196, 2014.

[14] G. Chandrashekar and F. Sahin. A survey on feature selection methods. *Computers & Electrical Engineering*, 40(1):16–28, 2014.

[15] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. Smote: synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16: 321–357, 2002.

[16] R. Chen, X. Wang, W. Zhang, X. Zhu, A. Li, and C. Yang. A hybrid cnn-lstm model for typhoon formation forecasting. *GeoInformatica*, 23:375–396, 2019.

[17] R. Chen, W. Zhang, and X. Wang. Machine learning in tropical cyclone forecast modeling: A review. *Atmosphere*, 11(7):676, 2020.

[18] T. Chen and C. Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 785–794, 2016.

[19] Z. Chen, X. Yu, G. Chen, and J. Zhou. Cyclone intensity estimation using multispectral imagery from the fy-4 satellite. In *Proceedings of the International Conference on Audio, Language and Image Processing (ICALIP)*, pages 46–51. IEEE, 2018.

[20] P.-S. Chu. Enso and tropical cyclone activity. *Hurricanes and Typhoons: Past, Present, and Potential*, 297:332, 2004.

[21] F. M. De Luca. An empirical evaluation of ai-based methods for modeling the genesis of tropical cyclones. Master's thesis, Politecnico di Milano, 2023. URL `https://hdl.handle.net/10589/204577`.

[22] F. Doshi-Velez and B. Kim. Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*, 2017.

[23] F. K. Došilović, M. Brčić, and N. Hlupić. Explainable artificial intelligence: A survey. In *Proceedings of the International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, pages 0210–0215. IEEE, 2018.

[24] K. Emanuel. 100 years of progress in tropical cyclone research. *Meteorological Monographs*, 59:15–1, 2018.

[25] K. A. Emanuel. The maximum intensity of hurricanes. *Journal of Atmospheric Sciences*, 45(7):1143–1155, 1988.

[26] K. A. Emanuel and D. S. Nolan. Tropical cyclone activity and the global climate system. In *Proceedings of the Preprints Conference on Hurricanes and Tropical Meteorology, Miami, FL, Amer. Meteor. Soc. A*, volume 10, 2004.

[27] Y. Freund and R. E. Schapire. Experiments with a new boosting algorithm. In *Proceedings of the International Conference on International Conference on Machine Learning*, page 148–156, 1996.

[28] J. H. Friedman. Greedy function approximation: a gradient boosting machine. *Annals of Statistics*, pages 1189–1232, 2001.

[29] D. Galea, J. Kunkel, and B. N. Lawrence. Tcdetect: a new method of detecting the presence of tropical cyclones using deep learning. *Artificial Intelligence for the Earth Systems*, 2(3):e220045, 2023.

[30] J. Han, J. Pei, and Y. Yin. Mining frequent patterns without candidate generation. *ACM Sigmod Record*, 29(2):1–12, 2000.

[31] H. He and E. A. Garcia. Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering*, 21(9):1263–1284, 2009.

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

[33] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.

[34] J. R. Hope and C. J. Neumann. An operational technique for relating the movement of existing tropical cyclones to past tracks. *Monthly Weather Review*, 98(12):925–933, 1970.

[35] I. R. Jenkins, L. O. Gee, A. Knauss, H. Yin, and J. Schroeder. Accident scenario

generation with recurrent neural networks. In *Proceedings of the International Conference on Intelligent Transportation Systems (ITSC)*, pages 3340–3345. IEEE, 2018.

[36] G. N. Kiladis, M. C. Wheeler, P. T. Haertel, K. H. Straub, and P. E. Roundy. Convectively coupled equatorial waves. *Reviews of Geophysics*, 47(2), 2009.

[37] S. Kim, H. Kim, J. Lee, S. Yoon, S. E. Kahou, K. Kashinath, and M. Prabhat. Deep-hurricane-tracker: Tracking and forecasting extreme climate events. In *Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1761–1769. IEEE, 2019.

[38] P. J. Klotzbach. The madden–julian oscillation's impacts on worldwide tropical cyclone activity. *Journal of Climate*, 27(6):2317–2330, 2014.

[39] K. R. Knapp, M. C. Kruk, D. H. Levinson, H. J. Diamond, and C. J. Neumann. The international best track archive for climate stewardship (ibtracs) unifying tropical cyclone data. *Bulletin of the American Meteorological Society*, 91(3):363–376, 2010.

[40] T. R. Knutson, J. L. McBride, J. Chan, K. Emanuel, G. Holland, C. Landsea, I. Held, J. P. Kossin, A. Srivastava, and M. Sugi. Tropical cyclones and climate change. *Nature Geoscience*, 3(3):157–163, 2010.

[41] S. B. Kotsiantis. Decision trees: a recent overview. *Artificial Intelligence Review*, 39: 261–283, 2013.

[42] M. A. Kramer. Nonlinear principal component analysis using autoassociative neural networks. *AIChE journal*, 37(2):233–243, 1991.

[43] D. Lafleur, B. Barrett, and G. Henderson. Some climatological aspects of the madden-julian oscillation (mjo). *Journal of Climate*, 28:150501115958009, 05 2015.

[44] S. L. Lavender and J. L. McBride. Global climatology of rainfall rates and lifetime accumulated rainfall in tropical cyclones: Influence of cyclone basin, cyclone intensity and cyclone size. *International Journal of Climatology*, 41(S1):E1217–E1235, 2021.

[45] J. A. Leloup, Z. Lachkar, J.-P. Boulanger, and S. Thiria. Detecting decadal changes in enso using neural networks. *Climate Dynamics*, 28:147–162, 2007.

[46] B. Letham, C. Rudin, T. H. McCormick, and D. Madigan. Interpretable classifiers using rules and Bayesian analysis: Building a better stroke prediction model. *The Annals of Applied Statistics*, 9(3):1350 – 1371, 2015.

[47] J. Li, K. Cheng, S. Wang, F. Morstatter, R. P. Trevino, J. Tang, and H. Liu. Feature selection: A data perspective. *ACM Computing Surveys (CSUR)*, 50(6):1–45, 2017.

[48] G.-F. Lin and L.-H. Chen. Application of an artificial neural network to typhoon rainfall forecasting. *Hydrological Processes: An International Journal*, 19(9):1825–1837, 2005.

[49] I.-I. Lin, S. J. Camargo, C. M. Patricola, J. Boucharel, S. Chand, P. Klotzbach, J. C. L. Chan, B. Wang, P. Chang, T. Li, and F.-F. Jin. *ENSO and Tropical Cyclones*, chapter 17, pages 377–408. American Geophysical Union (AGU), 2020.

[50] O. Loyola-Gonzalez. Black-box vs. white-box: Understanding their advantages and weaknesses from a practical point of view. *IEEE Access*, 7:154096–154113, 2019.

[51] S. W. Lubis and C. Jacobi. The modulating influence of convectively coupled equatorial waves (ccews) on the variability of tropical precipitation. *International Journal of Climatology*, 35(7):1465–1483, 2015.

[52] A. Mamalakis, I. Ebert-Uphoff, and E. Barnes. *Explainable Artificial Intelligence in Meteorology and Climate Science: Model Fine-Tuning, Calibrating Trust and Learning New Science*, pages 315–339. Springer International Publishing, Cham, 2022.

[53] D. Matsuoka, M. Nakano, D. Sugiyama, and S. Uchida. Deep learning approach for detecting tropical cyclones and their precursors in the simulation by a cloud-resolving global nonhydrostatic atmospheric model. *Progress in Earth and Planetary Science*, 5(1):1–16, 2018.

[54] A. F. Mavume, L. Rydberg, M. Rouault, and J. R. Lutjeharms. Climatology and landfall of tropical cyclones in the south-west indian ocean. *Western Indian Ocean Journal of Marine Science*, 8(1), 2009.

[55] M. J. McPhaden, S. E. Zebiak, and M. H. Glantz. Enso as an integrating concept in earth science. *Science*, 314(5806):1740–1745, 2006.

[56] C. E. Menkes, M. Lengaigne, P. Marchesiello, N. C. Jourdain, E. M. Vincent, J. Lefèvre, F. Chauvin, and J.-F. Royer. Comparison of tropical cyclogenesis indices on seasonal to interannual timescales. *Climate Dynamics*, 38:301–321, 2012.

[57] C. Molnar. *Interpretable Machine Learning*. Leanpub, 2 edition, 2022. URL `https://christophm.github.io/interpretable-ml-book`.

[58] C. Molnar, G. Casalicchio, and B. Bischl. Interpretable machine learning–a brief history, state-of-the-art and challenges. In *Proceedings of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 417–431. Springer, 2020.

[59] C. J. Neumann. An alternate to the hurran (hurricane analog) tropical cyclone forecast system. Technical report, United States, National Weather Service, Southern Region, Scientific Services Division, 1972. URL `https://repository.library.noaa.gov/view/noaa/3605`.

[60] T. Oates and D. Jensen. The effects of training set size on decision tree complexity. In *Proceedings of the International Workshop on Artificial Intelligence and Statistics*, pages 379–390. PMLR, 1997.

[61] E. Palmen. On the formation and structure of tropical hurricanes. *Geophysica*, 3(1): 26–38, 1948.

[62] B. Pan, X. Xu, and Z. Shi. Tropical cyclone intensity prediction based on recurrent neural networks. *Electronics Letters*, 55(7):413–415, 2019.

[63] R. Pradhan, R. S. Aygun, M. Maskey, R. Ramachandran, and D. J. Cecil. Tropical cyclone intensity estimation using a deep convolutional neural network. *IEEE Transactions on Image Processing*, 27(2):692–702, 2017.

[64] M. T. Ribeiro, S. Singh, and C. Guestrin. "why should i trust you?": Explaining the predictions of any classifier. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, page 1135–1144. Association for Computing Machinery, 2016.

[65] C. Roy and R. Kovordányi. Tropical cyclone track forecasting techniques—a review. *Atmospheric Research*, 104:40–69, 2012.

[66] D. Saraswat, P. Bhattacharya, A. Verma, V. K. Prasad, S. Tanwar, G. Sharma, P. N. Bokoro, and R. Sharma. Explainable ai for healthcare 5.0: opportunities and challenges. *IEEE Access*, 2022.

[67] R. Snaiki and T. Wu. Knowledge-enhanced deep learning for simulation of tropical cyclone boundary-layer winds. *Journal of Wind Engineering and Industrial Aerodynamics*, 194:103983, 2019.

[68] J. Tan, S. Chen, and J. Wang. Western north pacific tropical cyclone track forecasts by a machine learning model. *Stochastic Environmental Research and Risk Assessment*, 35:1113–1126, 2021.

[69] M. K. Tippett, S. J. Camargo, and A. H. Sobel. A poisson regression index for tropical cyclone genesis and the role of large-scale vorticity in genesis. *Journal of Climate*, 24(9):2335–2357, 2011.

[70] I. Tomek. Two modifications of cnn. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-6(11):769–772, 1976.

[71] D. Vale, A. El-Sharif, and M. Ali. Explainable artificial intelligence (xai) post-hoc explainability methods: Risks and limitations in non-discrimination law. *AI and Ethics*, 2(4):815–826, 2022.

[72] C. Velden, B. Harper, F. Wells, J. L. Beven, R. Zehr, T. Olander, M. Mayfield, C. uard, M. Lander, R. Edson, et al. The dvorak tropical cyclone intensity estimation technique: A satellite-based method that has endured for over 30 years. *Bulletin of the American Meteorological Society*, 87(9):1195–1210, 2006.

[73] Y. Wang, W. Zhang, and W. Fu. Back propogation (bp)-neural network for tropical cyclone track forecast. In *Proceedings of the International Conference on Geoinformatics*, pages 1–4. IEEE, 2011.

[74] Z. Wang, J. Zhao, H. Huang, and X. Wang. A review on the application of machine learning methods in tropical cyclone forecasting. *Frontiers in Earth Science*, 10: 902596, 2022.

[75] S. Węglarczyk. Kernel density estimation and its application. In *Proceedings of the ITM Web of Conferences*, volume 23, page 00037. EDP Sciences, 2018.

[76] C.-C. Wei. Forecasting surface wind speeds over offshore islands near taiwan during tropical cyclones: Comparisons of data-driven algorithms and parametric wind representations. *Journal of Geophysical Research: Atmospheres*, 120(5):1826–1847, 2015.

[77] M. C. Wheeler and H. H. Hendon. An all-season real-time multivariate mjo index: Development of an index for monitoring and prediction. *Monthly Weather Review*, 132(8):1917–1932, 2004.

[78] D. L. Wilson. Asymptotic properties of nearest neighbor rules using edited data. *IEEE Transactions on Systems, Man, and Cybernetics*, 2:408–421, 1972.

[79] C.-C. Young and W.-C. Liu. Prediction and modelling of rainfall–runoff during typhoon events using a physically-based and artificial neural network hybrid model. *Hydrological Sciences Journal*, 60(12):2102–2116, 2015.

[80] J. A. Zehnder. "tropical cyclone". encyclopedia britannica, 2024. URL `https://www.britannica.com/science/tropical-cyclone`.

[81] C. Zhang. Madden-julian oscillation. *Reviews of Geophysics*, 43(2), 2005.

[82] T. Zhang, W. Lin, Y. Lin, M. Zhang, H. Yu, K. Cao, and W. Xue. Prediction of tropical cyclone genesis from mesoscale convective systems using machine learning. *Weather and Forecasting*, 34(4):1035–1049, 2019.

[83] W. Zhang, B. Fu, M. S. Peng, and T. Li. Discriminating developing versus non-developing tropical disturbances in the western north pacific through decision tree analysis. *Weather and Forecasting*, 30(2):446–454, 2015.

List of Figures

List of Tables