UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

JAIR PAULINO DE SALES

DYNAMIC ENSEMBLE SELECTION FORECASTING SYSTEM BASED ON TREND CLASSIFICATION

Recife

2024

JAIR PAULINO DE SALES

DYNAMIC ENSEMBLE SELECTION FORECASTING SYSTEM BASED ON TREND CLASSIFICATION

Tese de Doutorado apresentada ao Programa de Pós-graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco, como requisito parcial para obtenção do título de Doutor em Ciência da Computação.

Área de Concentração: Inteligência Computacional

Orientador (a): Paulo S. G. de Mattos Neto

Coorientador (a): Paulo Renato Alves Firmino

Recife

2024

.Catalogação de Publicação na Fonte. UFPE - Biblioteca Central

de Sales, Jair Paulino.
Dynamic ensemble selection forecasting system based on trend classification / Jair Paulino de Sales. - Recife, 2024.
125f.: il.

Tese (Doutorado) - Universidade Federal de Pernambuco, Centro de Informática, Programa de Pós-Graduação em Ciência da Computação, 2024.
Orientação: Paulo Salgado Gomes de Mattos Neto.
Coorientação: Paulo Renato Alves Firmino.

1. Dynamic Ensemble Selection; 2. Trend Classification; 3. Model Selection; 4. Time Series; 5. Forecasting. I. Neto, Paulo Salgado Gomes de Mattos. II. Firmino, Paulo Renato Alves. III. Título.

UFPE-Biblioteca Central CDD 006.31

JAIR PAULINO DE SALES

DYNAMIC ENSEMBLE SELECTION FORECASTING SYSTEM BASED ON TREND CLASSIFICATION

Tese de Doutorado apresentada ao Programa de Pós-graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco, como requisito parcial para obtenção do título de Doutor em Ciência da Computação.

Área de Concentração: Inteligência Computacional.

BANCA EXAMINADORA

Prof. Dr. Ricardo Bastos Cavalcante Prudêncio
Centro de Informática - UFPE

Prof. Dr. Carlos Manuel Milheiro de Oliveira Pinto Soares
Faculdade de Engenharia - Universidade do Porto

Prof. Dr. Rafael Menelau Oliveira e Cruz
LOG-TI Departement - ÉTS-Montréal

Prof. Dr. Francisco Madeiro Bernardino Junior
Escola Politécnica - UPE

Prof. Dr. Tiago Alessandro Espínola Ferreira

Departamento de Estatística e Informática - UFRPE

Prof. Dr. Paulo Salgado Gomes de Mattos Netto (Orientador)

Centro de Informática - UFPE

Prof. Dr. Paulo Renato Alves Firmino (Coorientador)

Centro de Ciências e Tecnologias - UFCA

Recife

2024

Dedico este trabalho àqueles que, com fé e apoio, iluminaram minha jornada.

AGRADECIMENTOS

À força que rege nosso universo.

Às minhas duas mães, Joana D'arc e Maria do Socorro, ao meu pai, Severino Pedro, e aos meus dois irmãos, Jadiel Paulino e Jonas Paulino, com quem divido memórias inesquecíveis e cumplicidade inabalável. À Crístenes Sanches, por toda paciência, amor e companheirismo. Sou um grande sortudo por ter uma família tão maravilhosa.

Ao Professor Orientador, Paulo Salgado, agradeço por todos os momentos de aprendizado, pelas palavras de incentivo e pela confiança que depositou em mim. Seus ensinamentos me inspiraram a superar desafios e a buscar sempre evoluir. Sou grato por sua orientação exemplar e pelo papel fundamental que desempenhou na conclusão deste trabalho e em todo o meu processo de doutoramento.

Ao Professor Coorientador, Paulo Firmino, exemplo de profissional, amigo e mentor. Seus ensinamentos valiosos, a amizade sincera e o apoio constante foram essenciais para minha formação e para a concretização deste projeto. Agradeço pelos dez anos de aprendizado e cumplicidade, e espero que nossa jornada continue por muitos outros anos.

Agradeço imensamente aos professores membros da banca avaliadora, Dr. Ricardo Prudêncio, Dr. Carlos Soares, Dr. Rafael Cruz, Dr. Francisco Madeiro e Dr. Tiago Ferreira. Agradeço a cada um por sua dedicação, expertise e tempo dedicado à avaliação desta Tese.

Aos colegas dos grupos de pesquisa, ForecastAI e MESOR, em especial, Adriano Marabuco, Ruam Pastor e José Eduardo. Foram inúmeras horas em reuniões, conversas e desabafos. Estar ao lado de vocês torna tudo mais fácil.

Ao Banco Central do Brasil e Porto Digital, instituições que abriram portas e proporcionaram oportunidades incríveis de aprendizado e crescimento profissional através da participação em projetos de pesquisa e desenvolvimento. As experiências enriquecedoras que vivenciei nesses ambientes moldaram o profissional que sou hoje.

Agradeço, por fim, aos meus amigos, porto seguro e fonte de alegria. Vocês são a família que escolhi e sou eternamente grato por tê-los em minha vida. Obrigado por tudo!

Da Terra do fogo

Eu sou da Terra do fogo
Que sente no couro o agouro sagaz
Que foge da fome correndo sem nome
Subindo o monte com a reca atrás
Eu sou da terra da Luta que o povo disputa
E a morte astuta tenta levar
Correndo da sede caindo da rede
Com a fome feroz que vem pra matar

Jair Paulino de Sales

O Senhor é o meu pastor, nada me faltará (Salmo 23:1).

ABSTRACT

Dynamic Ensemble Selection systems (DES) have been proposed as an useful alternative for modeling and forecasting time series. The basic idea is to assess the performance of single models and select the best ones for predicting a new test instance. One of the most common selection strategies involves constructing regions of competence (RoC). In this case, based on a new test instance to be predicted, one evaluates which instances from the training and/or validation set are most similar using a similarity metric. However, the absence of similar patterns between the test and training/validation sets compromises the quality of the RoC and adversely affects the predictive capabilities of these systems. Besides, the choice of which similarity measure to adopt is a complex and ongoing research problem. Consequently, the following question arose: "How to conduct the selection phase considering structural changes in terms of trend in the time series, without relying on similarity measures?". This thesis proposes a new DES approach, Dynamic Ensemble Selection based on Trend Classification (DESTC), which uses trend analysis to select the models to be combined. Trend is the prevailing direction or pattern in data observed over time. DESTC consists of two main phases: the training phase (a), in which a pool of models is evaluated to determine the best ones for each trend class, and the testing phase (b), in which each new instance has its trend assessed, and the top-performing models are selected for prediction. To evaluate the predictive performance of DESTC, two experiments were conducted. In Experiment A, the proposed approach was applied to COVID-19 incidence time series data from eight countries and compared with single and ensemble-based algorithms from the literature. The proposed approach achieved superior forecasting performance and lower computational cost. In Experiment B, DESTC was further evaluated on time series exhibiting distinct characteristics from various phenomena. The results demonstrated that DESTC is a competitive alternative to other Multiple Predictor Systems (MPS). The main limitation of the proposed method is that DESTC tends to have lower predictive performance when the time series lacks a clear trend cycle pattern, making model selection based on trend classification impractical. Moreover, the results presented and discussed in both experiments demonstrate that the proposed method, DESTC, is a competitive alternative to other MPSs found in the literature.

Keywords: Dynamic Ensemble Selection, Trend Classification, Model Selection, Time Series, Forecasting.

RESUMO

Sistemas de Seleção Dinâmica têm sido propostos como uma alternativa útil para modelagem e previsão de séries temporais. Seu funcionamento avalia modelos em um conjunto (*pool*) para selecionar os mais competentes e os utilizar na previsão de novas instâncias de teste. Uma estratégia comum de seleção é a construção de regiões de competência (RoC), a partir da qual se avalia, com base na nova instância de teste, quais instâncias do conjunto de treinamento e/ou validação são mais semelhantes usando uma métrica de similaridade. No entanto, a ausência de padrões similares entre os conjuntos de teste e de treinamento/validação compromete a qualidade da RoC e afeta negativamente a capacidade preditiva desses sistemas. Além disso, a escolha de qual métrica de similaridade utilizar é um problema de pesquisa complexo e ainda em estudo. Neste sentido, surge a seguinte questão de pesquisa: "Como conduzir a fase de seleção considerando mudanças estruturais em termos de tendência na série temporal, sem depender de medidas de similaridade?". Esta tese propõe uma nova abordagem de seleção dinâmica, denominada *Dynamic Ensemble Selection based on Trend Classification* (DESTC), que utiliza análise de tendências para selecionar os modelos a serem combinados. O DESTC possui duas fases principais: a fase de treinamento (a), na qual um conjunto de modelos é avaliado para determinar os melhores para cada classe de tendência; e a fase de teste (b), na qual cada nova instância tem sua tendência avaliada, e os modelos com melhor desempenho são selecionados para a previsão. Para avaliar o desempenho preditivo do DESTC, foram conduzidos dois experimentos. No Experimento A, a abordagem proposta foi aplicada aos dados de séries temporais de incidência de COVID-19 de oito países e comparada com modelos únicos e ensembles já bem conhecidos na literatura. A abordagem proposta alcançou desempenho de previsão superior e menor custo computacional. No Experimento B, o DESTC foi avaliado em séries temporais que apresentam características diversas. Os resultados demonstraram que o DESTC é uma alternativa competitiva em relação a outros algoritmos. A principal limitação do método proposto é que o DESTC tem desempenho preditivo inferior quando a série temporal não possui um padrão bem definido de ciclos de tendência. Por fim, os resultados apresentados demonstram que o método proposto é uma alternativa competitiva em relação a outros sistemas de selação dinâmica encontrados na literatura.

Palavras-chaves: Sistemas de Seleção Dinâmica, Classificação de Tendência, Seleção de Modelos, Séries Temporais, Previsão.

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

ACF	Autocorrelation Function
ADE	Arbitrated Dynamic Ensemble
ANN	Artificial Neural Network
AR	Autoregressive models
ARIMA	Autoregressive Integrated Moving Average
ARM	Aggregating Ranking Measure
ARMA	Autoregressive Moving Average
CP	Consensus of Predictors
DES	Dynamic Ensemble Selection
DESTC	Dynamic Ensemble Selection based on Trend Classification
DS	Dynamic selection
DSNAW	Dynamic Selection based on the Nearest Windows
DTW	Dynamic Time Warping
DVS-OpOp	Dynamic Validation Set determination algorithm based on the similarity between the Output Profile of the test sample and the Output Profile of each training sample
ELM	Extreme Learning Machine
ES	Exponential smoothing
ETS	Exponential Smoothing State Space Approach
INPE	Instituto Nacional de Pesquisas Espaciais
LSTM	Long Short-Term Memory
MA	Moving Average models
MAE	Mean Absolute Error
MK	Mann-Kendall test
ML	Machine Learning

MLE	Maximum Likelihood Estimation
MLP	Multilayer Perceptron
MLR	Multiple Linear Regression
MPS	Multiple Predictor System
MSE	Mean Squared Error
PACF	Partial Autocorrelation Function
PD	Percentage Difference
PM	Performance Measure
PoE	Pool of Ensembles
RMSE	Root Mean Square Error
RNN	Recurrent Neural Network
RoC	Region of Competence
SARIMA	Seasonal ARIMA
SES	Simple Exponential Smoothing
SLR	Simple Linear Regression
SVM	Support Vector Machine
SVR	Support Vector Regressor
TSF	Transformer
TWF	Temporal-Window Framework
XGB	Extreme Gradient Boosting

LIST OF SYMBOLS

A_0	Initial value
a_i^*, a_i	Lagrange multipliers
b	Bias
b_t	Trend estimate in an ES model
B	Backshift notation
β	Trend parameter in an ES model
$\mathcal{C}$	Trend estimation method
C	Cyclic decomposition component
C_T	Long-term states
d	Number of differentiation
D	Number of seasonal differentiation
ϵ_t	Random error
ξ, ξ^*	Distance from actual values and and boundaries
f_{ens}	Convex combination of the single estimators
φ_n	Coefficient of a deterministic trend
ϕ	Coefficient of autoregressive component
g	Number of tied groups in MK test
H	Hidden layer
$\mathrm{H}^\dagger$	Moore-Penrose pseudoinverse
h	Time horizon
h_t	Short-term states
J	Set of best models

$\kappa(\cdot)$	Kernel function
l	Lag size
l_t	Level component
m	Number of models
n	Number of observations
η	Exponential decay parameter
Ψ	Set of trend classes
ψ	Class of trend
p	Autoregressive order
P	Pool of models
P'	Subset of P
P_{lit}	Performance of the model to be compared
P_{DESTC}	DESTC performance
q	Moving average order
Q	Seasonal moving average order
Q, K, V	Transformers matrixes
$Q_{j,k}$	Slope between a pair of data points (j, k)
$\rho(\cdot)$	Activation function
R	Random decomposition component
$R(C)$	Regression risk
R_{emp}	Empirical risk
r	Growth rate
S	Seasonality decomposition component

S_{MK}	Mann-Kendall Statistic
T	Trend decomposition component
θ	Coefficient of moving average component
T_t^D	Deterministic trend component
T_t^g	General trend component
T_t^S	Stochastic trend component
t_j	Number of data points in the j-th tied group
Ω_{J_ψ}	Combination of the J best models for the ψ trend class
U_a, V_a	Weight matrices
w_t^l	sliding window with l lags
w	Weight
x_i	n-dimensional input vector
y_i	Scalar to be predicted
$\hat{y}_i$	Predicted value of y_i
ζ_n	Linear coefficient
Z	Time series
z_t	Observation of a time series at time t
Z_{MK}	Mann-Kendall standardized statistic
$\hat{z}_{t+h}$	Forecast at time h
Z_w	New test instance
Z_{tr}, Z_v	Training and validation sets

CONTENTS

1 INTRODUCTION

Time series modeling and forecasting techniques are widely used across diverse research fields for their potential to enhance decision-making processes (BARRERA-ANIMAS et al., 2022; FENG; NIU, 2021; KIM, 2003; SEZER; GUDELEK; OZBAYOGLU, 2020; KAUSHIK et al., 2020; ZHAO et al., 2023; DARAGHMEH et al., 2021; KOUZIOKAS, 2019; DEB et al., 2017; RAHMAN et al., 2023). Recently, Multiple Predictor Systems (MPSs) have emerged as an alternative to improve the predictive performance of time series forecasting problems. Both empirical and theoretical studies provide evidence that these systems surpass the performance of single models (BROWN et al., 2005; CERQUEIRA et al., 2017a; HANANYA; KATZ, 2024).

A MPS generally is composed of three phases: (a) Generation, (b) Selection, and (c) Integration (Combination) (Figure 1). Initially, in (a), multiple forecasting models are trained to form a diverse and accurate pool. Here, diversity is crucial as different models capture various data patterns, compensating for individual errors. Subsequently, in (b), one or more models are selected for later combination (YAO; DAI; SONG, 2019; SILVA; DE MATTOS NETO; CAVALCANTI, 2021).

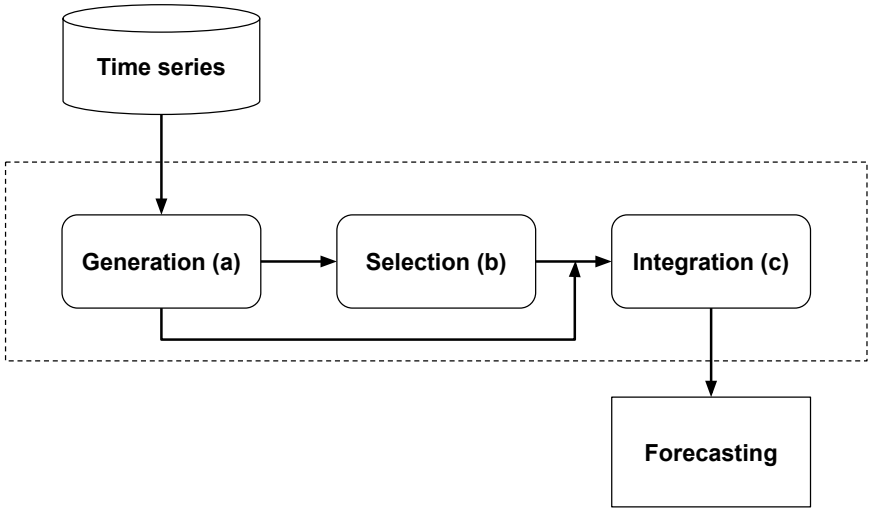


Figure 1 – Multiple Predictor System.

In the second phase, (b), the selection process can be static or dynamic. Static selection involves choosing a fixed set of models based on a training and/or validation set. These chosen models are then always combined for future predictions. In contrast, dynamic selection uses a different set of models for each new prediction (test pattern) (SILVA; DE MATTOS NETO; CAVALCANTI, 2021).

Lastly, in the final phase (c), selected models are combined. There are many combination strategies, with averaging and median being among the most common due to their low computational cost and good predictive performance (KOURENTZES; BARROW; CRONE, 2014). It is also common to use meta-models for combination; in these cases, a Machine Learning (ML) approach is trained to combine the estimates of the single models (GASTINGER et al., 2021; FERNÁNDEZ; SALINAS; TORRES, 2019).

1.1 PROBLEM STATEMENT

The fundamental principle behind dynamic selection systems is the recognition that different models exhibit expertise in distinct local regions of the feature space. This implies that no single model can proficiently estimate all test patterns. Therefore, the objective is to identify and select the most capable models for each new test pattern (YAO; DAI; SONG, 2019; SILVA; DE MATTOS NETO; CAVALCANTI, 2021).

In MPSs, establishing a Region of Competence (RoC) is a common strategy used for identifying the patterns most similar to a new test instance. In this context, a similarity metric can be employed to identify the patterns most similar to those in the training/validation set. However, choosing the appropriate similarity metric is a challenging task. Additionally, there are no guarantees that the new test instances to be predicted will be similar to those in the training and/or validation sets (SILVA et al., 2020; SILVA; DE MATTOS NETO; CAVALCANTI, 2021; SANTOS JUNIOR et al., 2022). The lack of similarity could be attributed to concept drift. This challenge became particularly evident during the COVID-19 pandemic.

As the pandemic progressed, the incidence time series of COVID-19 underwent significant structural changes globally, particularly in terms of trend (FIRMINO et al., 2020). Due to epidemiological factors such as transmission rates, mutations, and human behavior, these series began to display distinct cycles of alternating between positive and negative trends. This pattern indicates declines in incidence rates followed by increases due to rising case numbers, often referred to as "new waves" (AWADASSEID et al., 2020; CHUNG et al., 2021).

Regarding forecasting tasks, this scenario suggested that using similarity measures to construct RoC might not be the most effective approach, as the new instances to be predicted were very different from the instances in training and/or validation sets. Hence, the following question arose: "How to conduct the selection phase considering structural changes in terms of trend in the time series, without relying on similarity measures?". To address this question,

this thesis introduces a new selection strategy based on trend classification. Thus, the results presented here provide a foundation for developing new dynamic selection systems tailored to time series forecasting problems, with a focus on time dependency aspects.

1.2 OBJECTIVES

The main objectives of this thesis are:

- (i) to develop a new selection approach based on trend classification to enhance the forecasting of COVID-19 incidence time series;
- (ii) to assess the applicability of this new approach in time series from other domains.

The initial objective stemmed from the observation that pandemic time series exhibit distinct trend phases (increasing, decreasing, and no trend). This motivated the investigation of methods to identify the most suitable models for each phase. Subsequently, for each new prediction instance, the models most appropriate for the current trend class could be combined. As the trend of the series changes, the system dynamically selects the appropriate models to combine. The proposed approach introduces a new way of dynamically selecting models based on statistical information from the time series, specifically the trend. Additionally, by not depending on similarity metrics for model selection, DESTC avoids the problems directly linked to them, including the lack of similarity that may result from concept drift.

However, it is not clear how this method works for time series that do not behave like those from the COVID-19 pandemic. Some series do not have alternating cycles of trends, while others do not have any trend at all. Therefore, the second objective is to conduct an in-depth analysis of DESTC's functionality, thoroughly examining its mechanisms and identifying its limitations.

In addition to assessing the accuracy of the proposed approach, computational cost will also be evaluated. The expectation is that by eliminating the need to construct a RoC, there will be a reduction in computational costs. This aspect is crucial as it directly impacts the efficiency and scalability of the method, particularly in handling large datasets and real-time applications. The assessment of computational costs alongside accuracy provides a comprehensive evaluation of the proposed strategy's practical feasibility and potential benefits in various operational contexts.

1.3 CONTRIBUTIONS

The major contribution of this thesis is the introduction of a novel dynamic selection method for time series forecasting that leverages a time dependency aspect: trend. In this context, trend is defined as the general direction in which the data progresses over time. The selection strategy used in the proposed method, DESTC, involves identifying the best models for each trend class. Consequently, when a new test sample is introduced, its trend is evaluated, and the most appropriate models are selected based on this assessment. Initial experiments were conducted using COVID-19 incidence time series, with the results detailed in Chapter 4 and published in the following paper:

- DE SALES, J.P.; DE MATTOS NETO, P.S.G; FIRMINO, P.R.A. A dynamic ensemble approach based on trend analysis to COVID-19 incidence forecast. **Biomedical Signal Processing and Control**, 95, 106435, 2024. DOI: 10.1016/j.bspc.2024.106435.

Another contribution of this thesis is the assessment of the DESTC method when applied to time series with diverse characteristics, aiming to provide a deeper understanding of its advantages and limitations. In this context, time series from various domains, such as environmental, health, and financial, were evaluated. The computational cost was also evaluated in comparison to other dynamic selection methods. These results are presented in Chapter 5 and are also detailed in the following paper:

- DE SALES, J.P.; DE MATTOS NETO, P.S.G; FIRMINO, P.R.A. Enhancing forecasting accuracy with a dynamic ensemble selection approach based on trend classification. Under submission.

1.4 ORGANIZATION OF THE THESIS

Chapter 2 provides a theoretical background of time series analysis, trend classification, modeling and forecasting, multiple predictor systems, and performance measures. Chapter 3 presents the proposed approach in detail, discussing its motivation, mathematical formulation, and pseudo-code. In Chapter 4, Experiment A is outlined. This experiment was designed to evaluate the proposed approach using COVID-19 pandemic time series data in order to achieve the first main objective. Chapter 5 introduces Experiment B, which evaluates the proposed

approach using time series from different phenomena to achieve the second main objective. Chapter 6 concludes the thesis with a summary of the main findings. This chapter discusses the implications of the proposed approach and outlines potential directions for future research.

2 LITERATURE REVIEW

This chapter provides a literature review related to the topics covered in this thesis. Initially, it discusses time series, addressing their definition and presenting the concepts of trend, seasonality, cyclicality, and stationarity (Section 2.1). In Section 2.1.1,the concepts of deterministic, stochastic, and overall trends will be presented. Following that, two methods for trend analysis will be introduced: the Mann-Kendall statistical test (MANN, 1945) and Sen's Slope Estimator (SEN, 1968). Additionally, a set of time series forecasting models will be presented (Section 2.2), including both single (Section 2.2.1) and MPS alternatives (Section 2.2.2). Lastly, performance measures for evaluating forecasting models will be discussed (Section 2.3).

2.1 TIME SERIES

A time series, Z, is a set of ordered observations indexed over time (Equation 2.1) (MORETTIN; TOLOI, 2006),

$$Z = \{z_t \in \mathbb{R} \mid t = 1, 2, 3, ..., n\}, n \in \mathbb{N}, \tag{2.1}$$

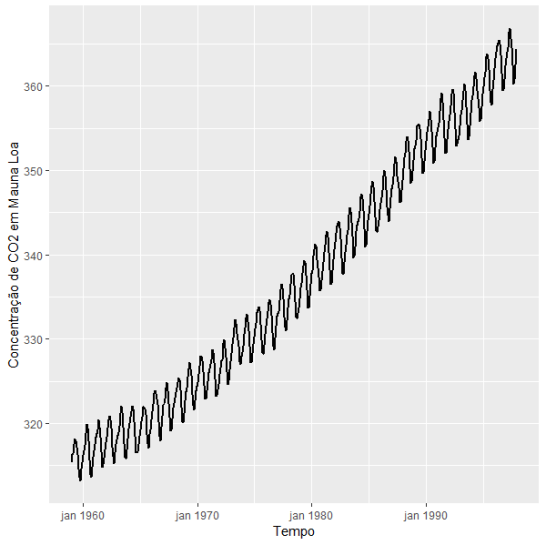
in which z_t denotes the value of the time series at time t and n is the number of observations available in the sample.

A time series is univariate (unidimensional) if the observations are of only one variable; otherwise, it is considered multivariate (multidimensional) (AMARAL, 2020). Hourly solar radiation, daily concentration of atmospheric particulate matter, and monthly cases of influenza are examples of time series data (BERRY et al., 2022). For simplicity, the term "time series" will be used in this text to denote univariate time series, as these will be the focus of study in this thesis.

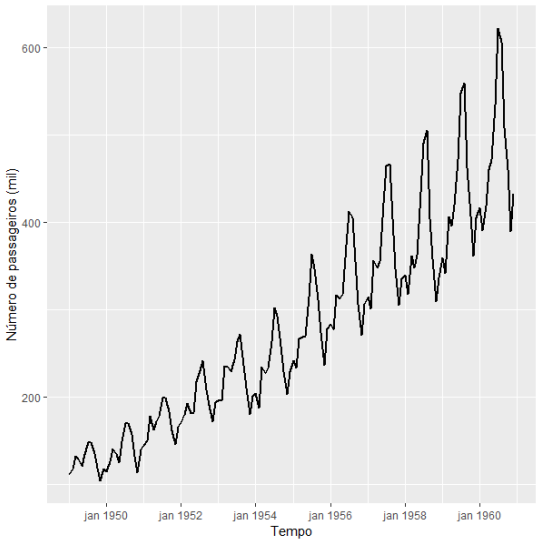
From a graphical perspective, a time series can be visualized in a two-dimensional representation, where the y-axis determines the observed values at time t (x-axis) (SILVA, 2021). This enables the identification of patterns such as trend, seasonality, and cycles, which facilitate understanding and inference about the studied phenomena (HYNDMAN; ATHANASOPOULOS, 2018).

A time series exhibits trend when it demonstrates non-periodic growth or decline in its

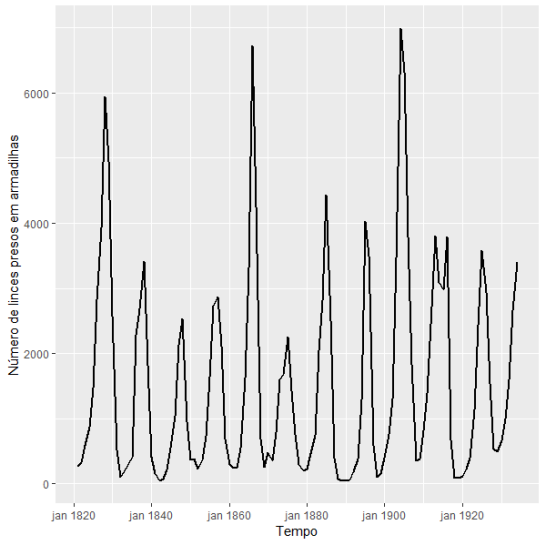
observations (HYNDMAN; ATHANASOPOULOS, 2018; COWPERTWAIT; METCALFE, 2009). The monthly concentration of CO_2 in Mauna Loa between 1959 and 1997 (Figure 2(a)) shows a long-term positive trend. In this example, a gradual growth can be observed throughout the recorded period.



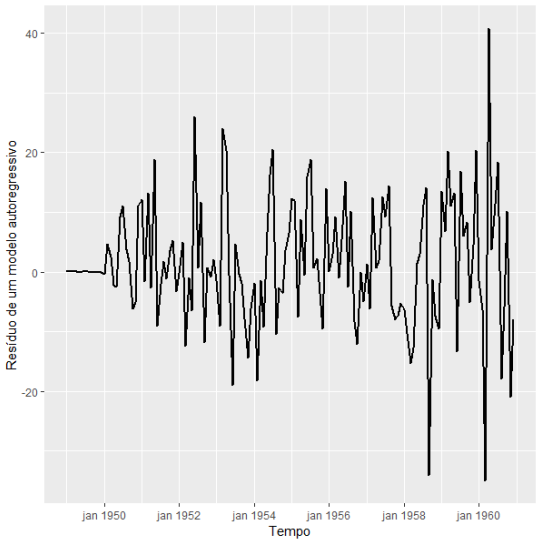
(a) Monthly concentration of CO_2 at Mauna Loa between 1959 and 1997.



(b) Monthly totals of international airline passengers between 1949 and 1960.



(c) Annual number of lynxes trapped in Canada between 1821 and 1934.



(d) Residuals of an autoregressive model for the time series AirPassengers.

Figure 2 – Four time series of distinct phenomena with different patterns.

Following, a seasonality pattern indicates that the studied phenomenon is influenced by seasonal factors, such as the end-of-year period effect on retail or the consequence of climatic seasons on the vegetation cover of a region. It is important to note that the seasonality of a time series has a known seasonal period (HYNDMAN; ATHANASOPOULOS, 2018). Figure 2(b)

presents the Air Passengers time series (BOX; JENKINS, 1976), which documents the monthly total of international airline passengers from 1949 to 1960. This time series exhibits an annual seasonal trend, with recurring patterns observed each year.

Cyclical behavior, similar to seasonal patterns, involves successive phases of growth and decline, yet the duration of its occurrence is not known. Moreover, while the duration of the causal effect in seasonal behavior is known, in cyclical behavior, it varies alongside its magnitude (HYNDMAN; ATHANASOPOULOS, 2018). Figure 2(c) presents the time series of the annual number of lynxes trapped in Canada between 1821 and 1934 (BROCKWELL; DAVIS, 1991). In this series, successive fluctuations in observed values can be analyzed; however, there is no clear temporal pattern, nor is there a well-established magnitude.

However, it is not always possible to identify the aforementioned patterns in time series. In some cases, observations exhibit random behavior, often difficult to predict using linear approaches (HYNDMAN; ATHANASOPOULOS, 2018). Figure 2(d) displays the residuals of an autoregressive model for the time series presented in Figure 2(a). In this series, discerning any clear behavior solely by visualizing its plot proves challenging.

Therefore, a practical way to analyze time series is to decompose them into the four Components, as shown in Equation 2.2 (FONSECA; MARTINS; TOLEDO, 2000).

$$Z = T + C + S + R, \tag{2.2}$$

in which T, S, C, and R represent the trend, seasonality, cyclic, and random components, respectively. In this case, the additive decomposition is presented. However, the components can also be modeled in a multiplicative way (Section 2.3) or by varying their relationships (Equations 2.4-2.6).

$$Z = T \cdot C \cdot S \cdot R \tag{2.3}$$

$$Z = T \cdot C \cdot S + R; \tag{2.4}$$

$$Z = T \cdot C + S \cdot R; \tag{2.5}$$

$$Z = T \cdot S + R. \tag{2.6}$$

Another important behavior in studies involving time series is stationarity. Essentially, it refers to a scenario where the statistical properties of the analyzed process remain constant over time, indicating a stable statistical equilibrium (CRAYER; CHAN, 2008). Hence, non-stationary

behaviors in time series can include trends, cycles, and random walks (HYNDMAN; ATHANASOPOULOS, 2018).

2.1.1 Trend analysis

According to Lamounier (2007), the trend of a time series can be classified as deterministic or stochastic. It is said to be deterministic when the variation in observations occurs as a function of time and in a certainly predictable manner, as Equation 2.7 shows (MATTOS, 2018).

$$z_t = T_t^D + \epsilon_t, \tag{2.7}$$

in which T_t^D is the deterministic trend component and ϵ_t is the random error (white noise). If T_t^D is linear, Equation 2.7 can be transformed into Equation 2.8:

$$z_t = \varphi_0 + \varphi_1 t + \epsilon_t, \tag{2.8}$$

where φ_0 and φ_1 are the linear coefficients, which can be estimated using the Method of Least Squares (MLS). In this function, the sign of the parameter φ_1 indicates whether the modeled series trend will be increasing ($\varphi_1 > 0$) or decreasing ($\varphi_1 < 0$) (FONSECA; MARTINS; TOLEDO, 2000).

However, depending on the phenomenon under study, T_t^D may exhibit other behaviors besides those described by a linear function. In these cases, more complex functions can be used, such as quadratic (Equation 2.9).

$$z_t = \varphi_0 + \varphi_1 t + \varphi_2 t^2 + \epsilon_t, \tag{2.9}$$

However, for many time series, it is not easy to find a function, linear or otherwise, capable of correctly estimating their trend (MATTOS, 2018). In these cases, growth and/or decay occur randomly, differing from the deterministic case in which there is a constant, φ_1, indicating the variation in the series level. When noise terms are independent, the stochastic trend component, T_t^S, can be written using Equations 2.10 and 2.11.

$$\Delta T_t^S = \epsilon_t, \tag{2.10}$$

$$T_t^S = T_{t-1}^S + \epsilon_t. \tag{2.11}$$

Moreover, many time series exhibit a mixed trend pattern, meaning there is a deterministic component responsible for a fixed alteration in the series level, as well as a stochastic pattern. This behavior can be classified as a general trend, T_t^G (Equation 2.12).

$$T_t^G = T_t^D + T_t^S. \tag{2.12}$$

According to Feo et al. (2022), studying the trend of time series is a very important topic in the field of data analysis, often being one of the first steps to be taken. In environmental sciences, for instance, assessing the trend can help identify changes in important variables such as ocean surface temperature (WATANABE et al., 2021; SHALTOUT, 2019), deforestation rates (SILVA et al., 2022; JAYATHILAKE et al., 2021), and solar irradiance (PUAH et al., 2021; SCAFETTA; WILLSON, 2019). The identification and modeling of trends also assist in performance analysis in financial systems (YIN; LI, 2021; DAS et al., 2022), as well as in understanding the epidemiological characteristics of diseases (TRINDADE et al., 2019; QIU; CAO; XU, 2021), thus supporting decision-making processes by both private companies and governmental entities.

The Mann-Kendall test (MK) (MANN, 1945; HAAN, 1977) and the Sen's Slope method (SEN, 1968) are classical and widely used approaches for trend detection (SAPLIOĞLU; GÜÇLÜ, 2022). These methods will be described below.

2.1.1.1 Mann-Kendall statistical test

The MK test is a widely-used non-parametric statistical method employed to assess trends within a dataset (MANN, 1945; WANG et al., 2020; AGBO; NKAJOE; EDET, 2023). It is particularly useful in analyzing time series data, where it aids in identifying monotonic trends, i.e., whether the data consistently increases or decreases over time. The MK test not only identifies the presence of a monotonic trend but also enables the evaluation of whether the trend is positive or negative. Additionally, MK offers a method to evaluate the intensity of trends (HAAN, 1977; LIBISELLER; GRIMVALL, 2002). Based on the studies of Mann (1945) and Kendall (1975), the MK test statistic, S_{MK}, is calculated based on the Equation 2.13.

$$S_{MK} = \sum_{j<i} \mathsf{sgn}(z_i - z_j), \tag{2.13}$$

in which sgn$(\cdot)$ is the sign function (Equation 2.14). The statistic S_{MK} has zero mean, and $Var(S_{MK})$ is calculated using Equation 2.15 (Equation with a correction term for ties):

$$\text{sgn}(x) = \begin{cases} 1, & \text{if } x > 0; \\ 0 & \text{if } x = 0; \\ -1 & \text{if } x < 0. \end{cases} \tag{2.14}$$

$$Var(S_{MK}) = \left\{ n(n-1)(2n+5) - \sum_{j=1}^{g} t_j(t_j-1)(2t_j+5) \right\} /18, \tag{2.15}$$

in which g is the number of tied groups in the data and t_j is the number of data points in the j-th tied group.

The S_{MK} statistic indicates the direction of the trend (positive or negative) and its magnitude (based on the number of concordant and discordant pairs). However, S_{MK} alone does not provide a measure of the statistical significance of the trend. In this order, it is necessary to calculate Z_{MK} (Equation 2.16) (HELSEL; FRANS, 2006; HIRSCH; SLACK; SMITH, 1982):

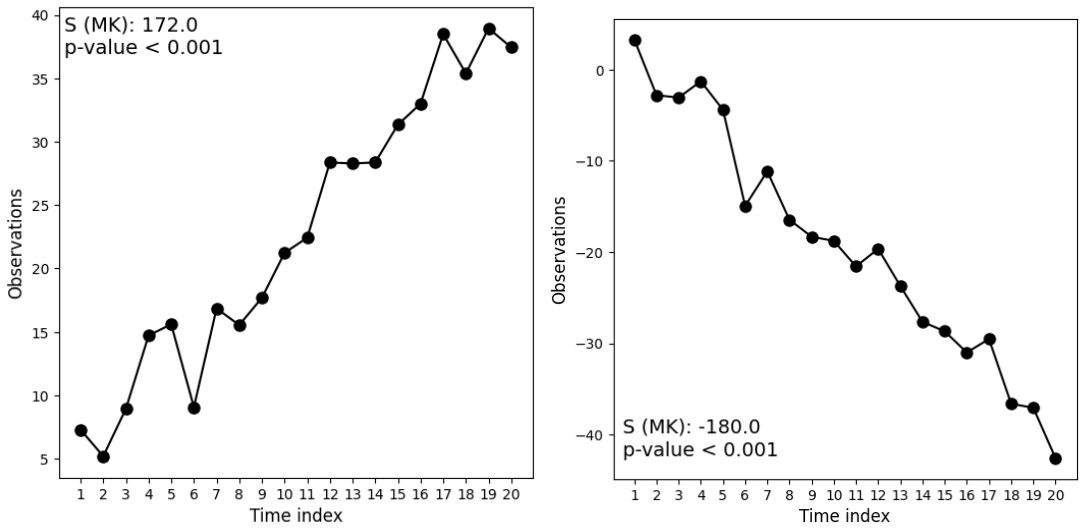
$$Z_{MK} = \begin{cases} \frac{S_{MK}-1}{\sqrt{\text{Var}(S_{MK})}}, & \text{if } S_{MK} > 0 \\ 0, & \text{if } S_{MK} = 0 \\ \frac{S_{MK}+1}{\sqrt{\text{Var}(S_{MK})}}, & \text{if } S_{MK} < 0 \end{cases} \tag{2.16}$$

The standardized Z_{MK} statistic is derived from S_{MK} and is used to determine the statistical significance of the trend. The null hypothesis of the MK test suggests that there is no monotonic trend in the data. Otherwise, the alternative hypothesis implies that there is a monotonic trend in the data. (POHLERT, 2020).

Null Hypothesis (H0):	There is no monotonic trend in the data.
Alternative Hypothesis (H1):	There is a monotonic trend in the data.

Positive (negative) values of S_{MK} indicate that most differences between previous, z_j, and successive, z_i, observations are positive (negative), demonstrating an increasing (decreasing) trend over time (BARI et al., 2016; HIRSCH; SLACK; SMITH, 1982). Therefore, the possible classifications for the instances are: positive trend (p-value $< \alpha$ and $S_{MK} > 0$), negative trend (p-value $< \alpha$ and $S_{MK} < 0$), and no trend (p-value $> \alpha$).

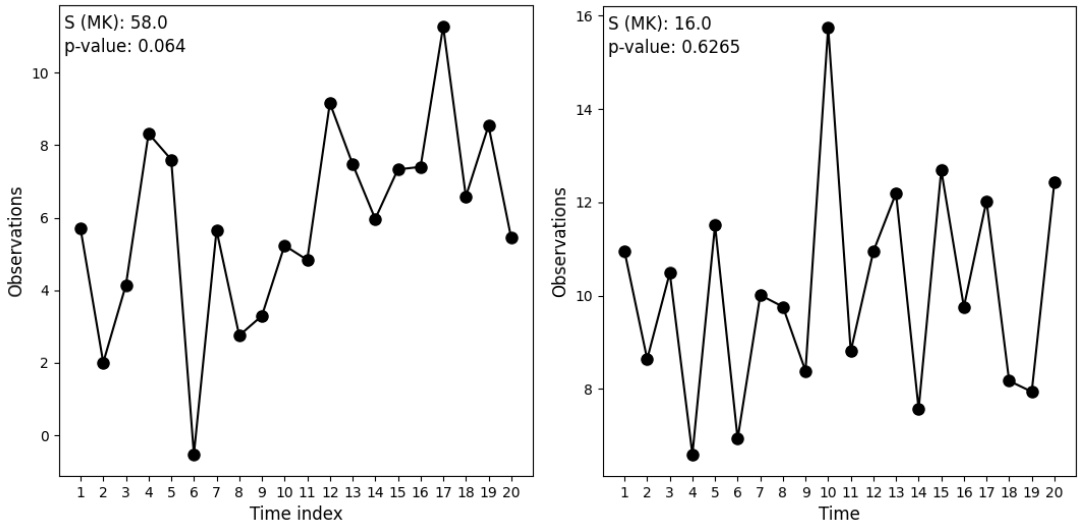
Figure 3 shows two examples of applying the MK test for trend detection. Initially, Figure 3(a) displays a time series with a positive trend, for which the p-value of the MK test less than 0.001 and $S_{MK} = 172.0$. Next, Figure 3(b) presents a time series with a decreasing trend, whose p-value was less than 0.001 and $S_{MK} = -180$.



(a) Time series with increasing trend. (b) Time series with decreasing trend.

Figure 3 – Time series with increasing and decreasing trends, respectively

Figure 4(a) presents a case where H0 is not rejected even with an apparent trend. In this time series, the data points exhibit an apparent positive trend over time. However, when applying the Mann-Kendall test, the result indicates that the trend is not statistically significant ($p > 0.05$). This means that, although there is an apparent trend in the data, it cannot be considered statistically significant based on the available dataset and the properties of the Mann-Kendall test. This discrepancy between visual trend and statistical detection underscores the importance of interpreting the results of statistical tests with caution and considering the specific context of each dataset. However, statistical tests have limitations and may not fully capture the complexity of data patterns. Figure 4(b) presents a case where there is no trend.



(a) H0 is not rejected even with an apparent trend.

(b) Time series with decreasing trend.

Figure 4 – Time series shows no trend according to the Mann-Kendall test.

2.1.1.2 Sen's Slope estimator

Sen's slope estimator, or Theil–Sen estimator, was proposed as an alternative to the parametric least-squares regression slope (SEN, 1968; THEIL, 1950). To mitigate the restrictive assumptions of regression analysis, Sen (1968) proposed a non-parametric approach that calculates the slope as the median of all possible slopes between each pair of successive time series values (ALMAZROUI; ŞEN, 2020; SILVA et al., 2015). It is robust against outliers and can be used even when residuals are not normally distributed. Sen's slope can be formulated as Equation 2.17.

$$Q_{j,k} = \frac{z_j - z_k}{j - k}, \tag{2.17}$$

in which z_j and z_k are successive values of a time series at time j and k ($j > k$), respectively. Sen's slope is the median of all $Q_{j,k}$ values.

Figure 5 presents an innovative application of the Sen's slope estimator for time series analysis. In this method, Sen's slope estimator is calculated using a sliding window across the time series. This approach allows for the evaluation of how the estimator evolves over time, providing a detailed view of changes in trend dynamics within the series. It enables the identification of trend variations and a deeper understanding of fluctuations throughout the analyzed period. Gray columns (Figure 5) represent the values of Sen's slope for the sliding

windows of the time series (black line). Sen's slope quantifies the strength of the trend; thus, a higher value indicates a stronger trend at that particular point in the series. This analysis will serve as a basis for discussing the results of Experiments A (Chapter 4) and B (Chapter 5).

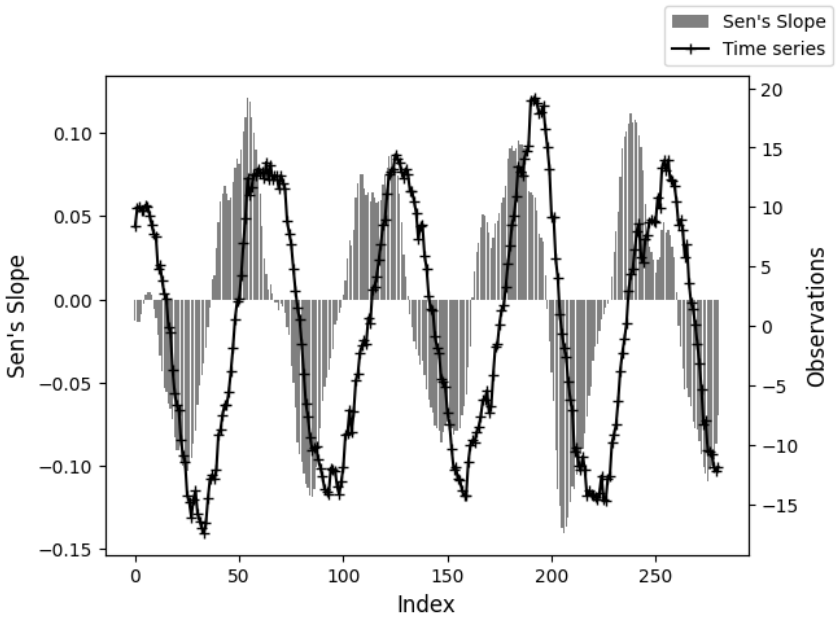


Figure 5 – Sen's Slope for a synthetic time series.

2.2 TIME SERIES MODELING AND FORECASTING

The development of time series forecasting models aids decision-making processes in various fields, including but not limited to industry (ALRASSAS et al., 2022), tourism (PRILISTYA; PERMANASARI; FAUZIATI, 2020), environmental sciences (ZAINI et al., 2022), finance (KUMAR; SINGH; JAIN, 2022), and health (RAHIMI; CHEN; GANDOMI, 2021). In summary, based on a time series Z, the aim is to predict z_{t+h}, in which h is the time horizon (Equation 2.18),

$$\hat{z}_{t+h} = f(w_t^l), \tag{2.18}$$

in which $\hat{z}_{t+h}$ is the h forecast, $f(\cdot)$ is a forecasting model and w_t^l is a sliding window with l lags. The forecasting horizon (h) can be established as one step ahead ($h = 1$) or multiple steps ahead ($h > 1$). One-step ahead predictions are commonly used to evaluate and compare new predictive methods. In turn, a sliding window is a set of past observations preceding the value to be predicted, which can be represented by $w_t^l = (z_t, z_{t-1}, \cdots, z_{t-l-1})$.

The number of lags l to to be used for constructing the temporal windows can be selected through the Autocorrelation Function (ACF), Partial Autocorrelation Function (PACF) (HYNDMAN; ATHANASOPOULOS, 2018), optimization methods (RIBEIRO et al., 2011; SURAKHI et al., 2020), sensitivity analysis (ZAIDAN et al., 2020), and experimentation (DOMINGOS; OLIVEIRA; DE MATTOS NETO, 2019). The selected lags do not have to be consecutive. One can imagine a scenario where, for a given series, the temporal windows are built from the observations z_{t-1}, z_{t-5}, e z_{t-7}, for example. In this case, the lag values were determined through the application of a selection algorithm.

In addition to defining w_t^l, the choice of forecasting models is of paramount importance for obtaining accurate results. In this regard, a wide range of alternatives can be found in the literature.

2.2.1 Single models

Single forecasting models are those methods that receive only past information from a time series as input and generate a forecast based on that information. In summary, they differ from MPS since the latter require a pool of models for generating forecasts (as will be seen in Section 2.2.2). Classical time series forecasting models are statistical methods that utilize historical patterns to make future predictions (HYNDMAN; ATHANASOPOULOS, 2018). They encompass techniques such as exponential smoothing and autoregressive models (Section 2.2.1.2 - 2.2.1.7). Moreover, it is common to utilize ML models for regression tasks in time series forecasting (LIM; ZOHREN, 2021; MASINI; MEDEIROS; MENDES, 2023; BARRERA-ANIMAS et al., 2022). These approaches aim to establish a relationship between past and future data, identifying the patterns and underlying structures within time series (Section 2.2.1.9 - 2.2.1.13).

2.2.1.1 Naïve model

The Naïve model, also known as Random Walk model (Equation 2.19), is commonly used as a benchmark. In this model, the prediction $\hat{z}_{t+1}$ is assumed to be equal to the value of observation z_t plus a normal fluctuation with zero mean and constant variance, making it mathematically simple. This model assumes that there are no patterns or underlying structure in the data, and that changes between periods are purely random. Despite its simplicity, the

Random Walk is useful as a benchmark for comparing more complex time series forecasting models. If a more sophisticated model does not significantly outperform the Random Walk forecast, it suggests that the model may not be effectively capturing patterns in the data.

$$\hat{z}_{t+1} = z_t + N(0, \sigma_z). \tag{2.19}$$

2.2.1.2 Exponential smoothing models

Exponential Smoothing model (ES) were proposed in the 1950s, making it one of the earliest explicitly created approaches for time series modeling and forecasting (BROWN, 1959; WINTERS, 1960). The basic idea is that the more recent an observation is, the higher its weight for the forecast. Additionally, the weights decay exponentially as observations get older (HYNDMAN; ATHANASOPOULOS, 2018).

The simplest ES model, known as Simple Exponential Smoothing (SES), is useful for modeling time series that do not exhibit trend and/or seasonality. Equation 2.20 presents the SES.

$$\hat{z}_{t+1} = \eta z_t + \eta(1-\eta) z_{t-1} + \eta(1-\eta)^2 z_{t-2} + \cdots + \eta(1-\eta)^n z_{t-n} + \epsilon_t, \tag{2.20}$$

where $0 < \eta < 1$ is the parameter of exponential decay and n is the size of the time series. The larger η is, the faster the decay will occur. As η approaches 1, the SES model approaches the Random Walk model.

Alternatively, ES models can be represented through components (HYNDMAN; ATHANASOPOULOS, 2018). In the case of SES, the only component present will be the level component l_t (Equation 2.21).

$$\text{Forecasting equation} : \hat{z}_{t+1} = l_t, \tag{2.21a}$$

$$\text{Level equation} : l_t = \eta z_t + (1-\eta) l_{t-1}. \tag{2.21b}$$

This alternative way of describing ES models is not very useful for the case of SES. However, it helps in understanding more complex methods, such as the Holt linear model (HOLT, 2004) (Equation 2.22), which incorporates the modeling of the trend of the time series.

$$\text{Forecasting equation : } \hat{z}_{t+1} = l_t + hb_t, \tag{2.22a}$$

$$\text{Level equation : } l_t = \eta z_t + (1-\eta)(l_{t-1} + b_{t-1}), \tag{2.22b}$$

$$\text{Trend equation : } b_t = \beta(l_t - l_{t-1}) + (1-\beta)b_{t-1}, \tag{2.22c}$$

in which b_t is the trend estimate of the series at time t, and β is the trend parameter ($0 < \beta < 1$).

Currently, based on the works of Hyndman and Athanasopoulos (2018), ES models can be grouped considering a state space approach notation, called Exponential Smoothing State Space Approach (ETS), which considers Error, Trend, and Seasonality components. In this notation, trend components can be none, additive, dampened additive, multiplicative, or dampened multiplicative. Seasonality components can be either additive or multiplicative. By combining the different options for trend and seasonality components, it is possible to construct 15 distinct types of models (Table 1). Furthermore, errors can be modeled as either additive or multiplicative, effectively doubling the number of possible configurations. For example, the notation ETS(A, N, M) represents a model with additive error (A), no trend (N), and multiplicative seasonality (M).

Table 1 – ES models according to ETS notation proposed by Hyndman and Athanasopoulos (2018).

Trend	Seasonality		
	N (None)	A (Additive)	M (Multiplicative)
N (None)	NN	NA	NM
A (Additive)	AN	AA	AM
Ad (Damped Additive)	AdN	AdA	AdM
M (Multiplicative)	MN	MA	MM
Md (Damped Multiplicative)	MdN	MdA	MdM

2.2.1.3 Autorregressive models

Autoregressive models (AR) models make forecasts based on a linear combination of past values of the time series of interest. The term "autoregressive" indicates that these models aim to describe the autocorrelation present in time series (HYNDMAN; ATHANASOPOULOS, 2018). Equation 2.23 presents AR models of order p, denoted by AR(p).

$$z_t = c + \phi_1 z_{t-1} + \phi_2 z_{t-2} + \cdots + \phi_p z_{t-p} + \epsilon_t, \tag{2.23}$$

in which c is an intercept (also known as a drift term), p is the order of the model, ϕ_p is the weight of the p-th previous observation, and ϵ_t is the error at time t.

2.2.1.4 Moving Average models

Moving Average Moving Average models (MA) models work similarly to AR; however, they do not explicitly use past values of the series but rather the residuals from previous forecasts (HYNDMAN; ATHANASOPOULOS, 2018). Equation 2.23 can be adapted to:

$$z_t = c + \epsilon_t + \theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2} + \cdots + \theta_q \epsilon_{t-q}, \tag{2.24}$$

where θ_q is the weight of the q-th previous error. In this case, the model is said to be an MA of order q and can be denoted by MA(q).

2.2.1.5 Autorregressive Moving Average models

Autoregressive Moving Average (ARMA) models combine the AR(p) and MA(q) components, which can be expressed by Equation 2.25 and denoted by ARMA(p,q).

$$z_t = c + \phi_1 z_{t-1} + \phi_2 z_{t-2} + \cdots + \phi_p z_{t-p} + \epsilon_t + \theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2} + \cdots + \theta_q \epsilon_{t-q}. \tag{2.25}$$

However, similarly to their predecessors (AR and MA), ARMA models have their limitations, ARMA models have two important limitations. The first one is that they only model linear processes, and the second one is that they do not handle non-stationary time series well (KARTHIKEYAN; KUMAR, 2013).

2.2.1.6 Autorregressive Integrated Moving Average models

Autoregressive Integrated Moving Average (ARIMA) models were developed to address the issue of non-stationarity. In these models, the term I refers to the adoption of the differencing method to stationarize time series. Equation 2.26 presents the ARIMA(p,d,q) model, where d

represents how many differencing steps were used to ensure the stationarity of the series, and $z_t^{(d)}$ is the differenced series itself.

$$z_t^d = c + \phi_1 z_{t-1}^d + \phi_2 z_{t-2}^d + \cdots + \phi_p z_{t-p}^d + \epsilon_t + \theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2} + \cdots + \theta_q \epsilon_{t-q}, \quad (2.26)$$

An alternative way to mathematically present ARIMA models involves the use of lag operators (backshift notation),

$$B_{z_t} = z_{t-} \quad (2.27)$$

In this case, ARIMA models can be described by Equation 2.28:

$$(1 - \phi_1 B - \cdots - \phi_p B^p)(1 - B)^d z_t = (1 + \theta_1 B + \cdots + \theta_q B^q)\epsilon_t, \quad (2.28)$$

where $(1 - \phi_1 B - \cdots - \phi_p B^p)$ represents the AR(p) component, $(1 - B)^d z_t$ represents differencing, and $(1 + \theta_1 B + \cdots + \theta_q B^q)$ represents the MA(q) component.

2.2.1.7 Seasonal Autorregressive Integrated Moving Average models

Seasonal ARIMA (SARIMA) model, in turn, was developed by adding seasonal components to the classical method. Thus, for each component, there exists a seasonal equivalent. Therefore, the SARIMA model can be represented by SARIMA(p,d,q)(P,D,Q)[m], where P, D, and Q represent, respectively, the seasonal autoregressive, seasonal differencing, and seasonal moving average components. Finally, m represents the seasonal period (HYNDMAN; ATHANASOPOULOS, 2018). For example, the SARIMA(1,1,1)(1,1,1)[4] model can be represented by Equation 2.29.

$$(1 - \phi_1 B)(1 - \Phi_1 B^4)(1 - B)(1 - B^4) z_t = (1 - \theta_1 B)(1 - \Theta_1 B^4)\epsilon_t \quad (2.29)$$

In addition to defining the parametric orders (e.g., defining p, q, P, and Q), it is necessary to estimate the values of these parameters (e.g., c, ϕ_i, θ_j). Maximum Likelihood Estimation (MLE) is a commonly used alternative at this stage. In summary, MLE estimates the parameters that maximize the likelihood of obtaining the observed data (HYNDMAN; ATHANASOPOULOS, 2018).

2.2.1.8 Linear Regression

A time series forecasting problem can be formulated as a regression task(URAS et al., 2020; HOPE, 2020; MASINI; MEDEIROS; MENDES, 2023; KUMARI; SINGH, 2023). In this approach, historical observations of the time series are used as predictor variables for estimating future values. Consider a training set $\{(x_i, y_i)\}_{i=1}^{n}$, in which x_i represent an n-dimensional input vector and y_i a scalar (target to be predicted). A Simple Linear Regression (SLR) model provides a linear relationship between x and y, such as:

$$\hat{y}_i = \zeta_0 + \zeta_1 x_i + \epsilon_i, \tag{2.30}$$

in which $\hat{y}_i$ is the predicted value, ζ_0 is the y-axis intercept, and ζ_1 the slope. This is the simplest mathematical way to determine the relationship between two variables. In this case, y being the future value and x the observation one step back. However, in real-world forecasting problems, it is common to use multiple observations to make predictions. Therefore, Multiple Linear Regression (MLR) becomes an essential tool. In this case, Equation 2.30 evolves to:

$$\hat{y}_i = \zeta_0 + \zeta_1 x_i + \zeta_2 x_2 + \cdots + \zeta_n x_n + \epsilon_i, \tag{2.31}$$

in which ζ_n is the linear coefficient of the x_n variable.

2.2.1.9 Support Vector Regressor

Once time series forecasting problems can be considered as regression tasks, ML techniques can be effectively applied. Support Vector Machine (SVM) is a widely used supervised ML algorithm developed based on the Theory of Empirical Risk Minimization, also known as the Vapnik-Chervonenkis Theory (VAPNIK, 1999; LI et al., 2009; HU et al., 2016). This supervised method can be utilized for both classification and regression problems. In the context of regression problems, it is referred to as Support Vector Regressor (SVR).

The main idea of SVR is to find a function that fits the data accurately, while allowing for a limited margin of error. Additionally, rather than the empirical risk minimization concept, the formulation corresponds to the structural risk minimization principle, i.e., rather than minimizing the prediction error on the training set, SVR aims to minimize an upper bound of the generalization error (LI et al., 2009; KANG; LI, 2016).

Consider a training set $\{(x_i, y_i)\}_{i=1}^{n}$. Equation 2.32 shows a nonlinear relationship between x_i and y_i described by a regression function,

$$f(x) = \omega^T \phi(x) + b, \tag{2.32}$$

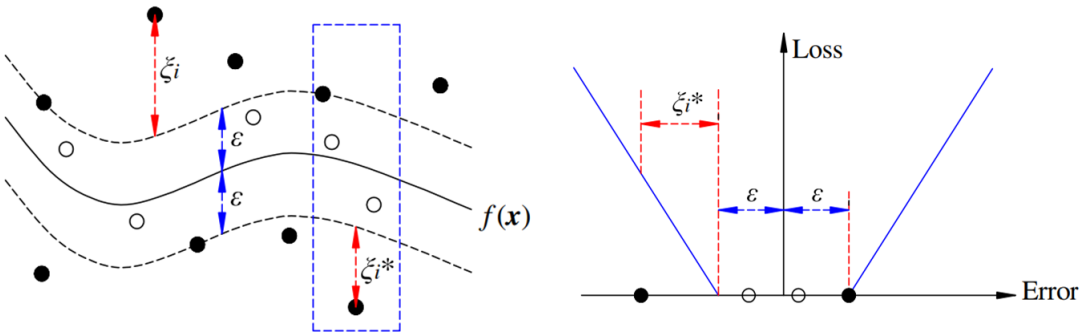
in which $f(x)$ is the forecasting value, $\phi(x)$ a nonlinear function, and ω and b are coefficients to be adjusted (KANG; LI, 2016; SILVA, 2020; SAPANKEVYCH; SANKAR, 2009).

Further, ω and b can be approximated by reducing the regularizing risk function:

$$R(C) = R_{emp} + \frac{1}{2}||\omega||^2 = C\frac{1}{l}\sum_{i=1}^{l} L_\epsilon[y_i, f(x_i)] + \frac{1}{2}||\omega||^2 \tag{2.33}$$

$$L_\epsilon(y_i, f(x_i)) = \begin{cases} |y_i - f(x_i)| - \epsilon, & |y_i - f(x_i)| \geq \epsilon \\ 0, & \text{otherwise.} \end{cases} \tag{2.34}$$

in which $R(C)$ and R_{emp} are the regression and empirical risks, respectively. First item of Equation 2.33 is the empirical error, which is estimated by the ϵ-insensitive loss function in Equation 2.34. In turn, the second item is the regularization. The C parameter represents the trade-off between the first and second terms of the equation. For better understanding, the ϵ parameter can be visualized as a tube size that corresponds to the approximation accuracy in the training set (Figure 6) (KANG; LI, 2016).



(a) ϵ-tube, support vectors, and slack variables (ξ_i)

(b) ϵ-insensitive loss function.

Figure 6 – SVR non-linear. Adapted from (KANG; LI, 2016).

Equation 2.33 presents the distance from actual values and corresponding boundary values of ϵ-tube, ξ and ξ^*, thus:

$$\begin{aligned} \text{min} \quad & \frac{1}{2}\|\mathrm{w}\|^2 + C\sum_{i=1}^{n}[L(\xi_i) + L(\xi_i^*)], \\ \text{s.t.} \quad & \begin{cases} y_i - \omega\phi(x_1) \leq \epsilon_i + \xi_i, \\ \omega\phi(x_1) - y_i \leq \epsilon_i + \xi_i^*, \\ \xi_i, \xi_i^* \geq 0, \end{cases} \end{aligned} \tag{2.35}$$

Moreover, the optimization problem of the Equation 2.35 can be transformed into a dual problem, with its solution provided in Equation 2.36, in which a_i^* and a_i are the Lagrange multipliers that can be obtained by resolving the dual problem, and $\kappa(x_i, x)$ is the kernel function. Most common kernel functions are linear, sigmoid, polynomial, and radial basis (LI et al., 2009; KANG; LI, 2016).

$$\begin{aligned} & f(x) = \sum_{i=1}^{n}(a_i^* - a_i)\kappa(x_i, x) + b \\ \text{s.t.} \quad & 0 \leq a_i^*, a_i \leq C \end{aligned} \tag{2.36}$$

2.2.1.10 Multilayer Perceptron

Artificial Neural Networks (ANNs) represent a class of ML algorithms commonly applied to classification and regression problems. They were developed to mimic the biological neural system, with the aim of replicating the learning and adaptation capabilities of the human brain (HAYKIN, 2009). Formally, an ANN can be defined as a massively parallel distributed processor composed of processing units (neurons), which have the inherent ability to store experiential knowledge and make it available for use (HAYKIN, 2009).

The neurons are organized in layers and connected by synaptic weights. Each neuron receives weighted inputs from other neurons, which are then combined through an activation function. This function introduces non-linearity and allows the ANN to learn complex patterns. The learning process iteratively adjusts these synaptic weights to minimize errors and enhance performance (HAYKIN, 2009; OLIVEIRA et al., 2021).

In terms of Multilayer Perceptron (MLP) architecture, two common approaches are the Single-Layer Feedforward Networks and the Multilayer Feedforward Networks. Single-Layer Feedforward Networks are the simplest, with connections (weights) directly linking inputs to outputs. On the other hand, Multilayer Feedforward Networks, also known as MLP, have one

or more hidden layers, each with its hidden units, between the inputs and outputs. Single-layer networks can only handle linear problems, while multilayer networks can capture more complex relationships, they are capable of solving non-linear problems (HAYKIN, 2009).

Figure 7 presents a basic model of an artificial neuron, which is widely used in more complex ANN (presented later). The synaptic weights (w_{k1}, w_{k2}, $\cdots$, w_{km}) indicate the relevance of the inputs. Next, a summation function aggregates inputs linearly based on their weights. Finally, an activation function will adjust the amplitude of the output signal (HAYKIN, 2009).

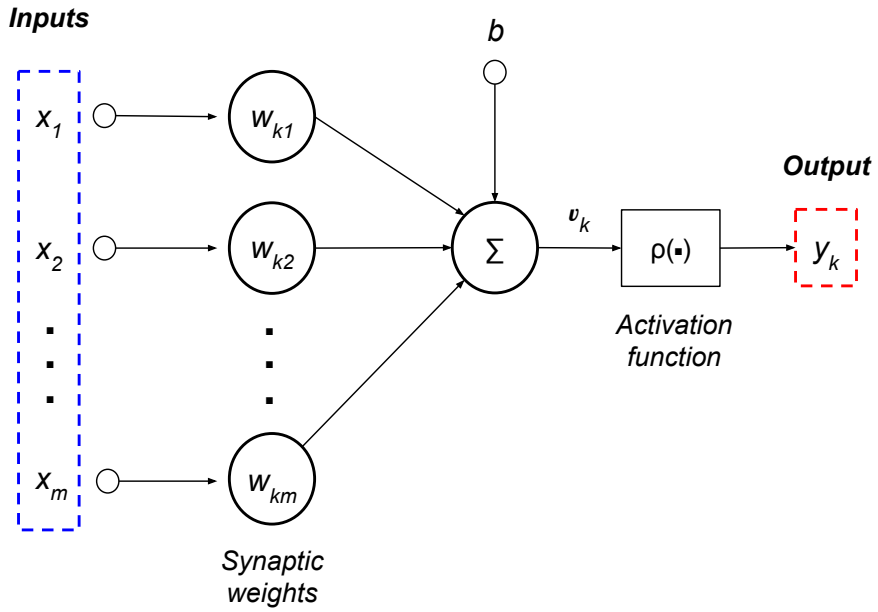


Figure 7 – Nonlinear model of a neuron. Adapted from Haykin (2009).

Mathematically, the artificial neuron can be described by Equations 2.37 and 2.38:

$$u_k = \sum_{j=1}^{m} w_{kj} x_j, \tag{2.37}$$

$$y_k = \rho(v_k + b), \tag{2.38}$$

where x_1, x_2, $\cdots$, and x_m are the input signals, w_{k1}, w_{k2}, $\cdots$, w_{km} are the respective synaptic weights of neuron k inputs, u_k is the linear combiner output due to the input signals, b is the bias, $\rho(\cdot)$ is the activation function, and y_k is the neuron output. The inclusion of the bias term b results in an affine transformation being applied to the output u_k of the linear combiner in the model of Figure 7 (Equation 2.39).

$$v_k = u_k + b \tag{2.39}$$

The training process of an MLP involves the iterative adjustment of weights to obtain a model with good generalization capability. For forecasting problems, the classical approach is to use a supervised algorithm, where synaptic weights are adjusted based on the difference between actual and predicted values (OLIVEIRA et al., 2021). In general, this training process is done using backpropagation algorithm (HAYKIN, 2009).

In the backpropagation algorithm, the first phase involves randomly initializing the weightS (Initialization). Thus, for each example in the training set, the input data is sequentially processed through the network's layers, with neurons applying synaptic weights and activation functions to generate output. This process continues until the final layer, where the network produces its prediction, which are compared to actual values to compute errors (Forward computation). The errors are then backpropagated through the network, enabling the calculation of weight gradients used to iteratively adjust the weights and minimize errors (Backward Computation). The forward and backward phases continue to be executed until a stopping criterion is met. This iterative adjustment process allows the network to learn and improve its predictions over time (HAYKIN, 2009).

Moreover, the activation functions currently represent an intense research field, with numerous works dedicated to the development of alternatives aimed at increasing performance or even reducing the complexity of prediction models. Among the alternatives, the activation functions commonly used are: Linear, Hyperbolic Tangent, Sigmoid, and ReLU (LESHNO et al., 1993; APICELLA et al., 2021).

2.2.1.11 Long Short-Term Memory

Recurrent Neural Network (RNN) are another well-established class of architecture in the time series forecasting literature (HOCHREITER; SCHMIDHUBER, 1997; OLIVEIRA et al., 2021; ABDEL-NASSER; MAHMOUD, 2019). They differ from Feedforward Networks (Section 2.2.1.10) by incorporating at least one feedback loop, a mechanism that allows the output of a neuron or layer to be fed back as input to the same network. This feature enables RNN to handle data sequences and capture temporal dependencies (HAYKIN, 2009).

LSTM networks, introduced by Hochreiter and Schmidhuber (1997), are a special kind of Recurrent RNN. Due to their powerful real-world predictive capabilities, LSTM networks have been widely applied in problems across different domains (YU et al., 2019; KRATZERT et al., 2018; PARK; YANG, 2022; GUO et al., 2021; ZHA et al., 2022). In summary, the key idea is that,

based on the input data, the LSTM is able of learning: (i) what to store in the long-term state; (ii) what to forget; and (iii) what to read from it (GÉRON, 2019). This process will be detailed below (GÉRON, 2019).

Figure 8 presents a LSTM cell. Vectors h_t and C_t represent, respectively, short-term and long-term states. One can see that the long-term state, C_t, flowing from left to right, initially passed through the *Forget gate*, which aims to control which information from the long-term state should be deleted. The *Input gate*, in turn, controls the information that should be added to the long-term state. Lastly, the *Output gate* is responsible for controlling which information from the long-term state will be read and used during this time interval (GÉRON, 2019).

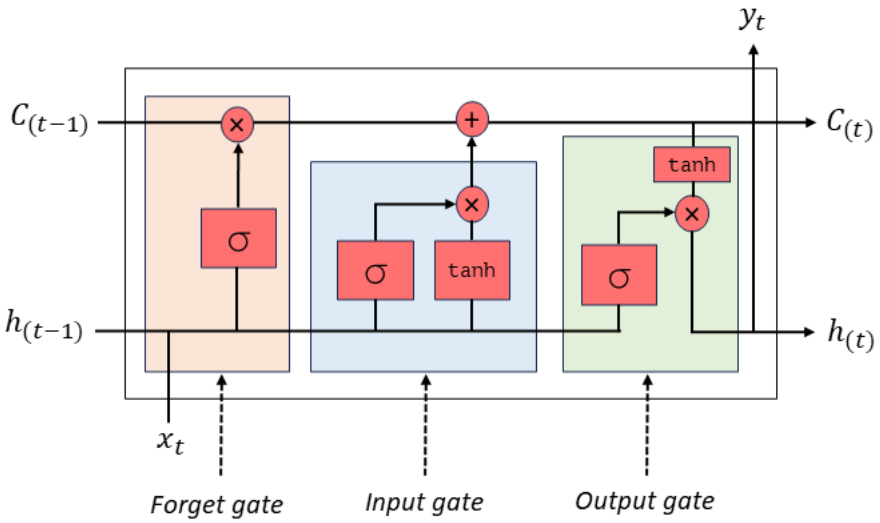


Figure 8 – LSTM cell. Adapted from Géron (2019), Mittal (2019), Matsumoto (2019).

In the *Forget gate* (orange rectangle in Figure 8), the vectors h and x passes through a sigmoid activation function (here denoted by σ), which returns values between 0 and 1 (f_t). The result is subsequently multiplied by the long-term state, obtaining C_t' (Equations 2.40 and 2.41, Figure 9(a)).

$$f_t = \sigma(U_a x_t + V_a h_{t-1} + b), \tag{2.40}$$

$$C_t' = f_t \cdot C_{t-1}, \tag{2.41}$$

in which U_a and V_a are weight matrices that connect the layer to the input vector and b is the bias. Next, it is determined which information from the short-term state should be inserted into the long-term state through the *Input gate*. For this, a state of activation containing the

new information, denoted by C_t^+, is defined, and a vector i, which will determine the relevance of this information (Equations 2.42, 2.43, and 2.44, Figure 9(b)).

$$i_t = \sigma(U_b x_t + V_b h_{t-1} + b), \tag{2.42}$$

$$C_t^+ = \tanh(U_c x_t + V_c h_{t-1} + b), \tag{2.43}$$

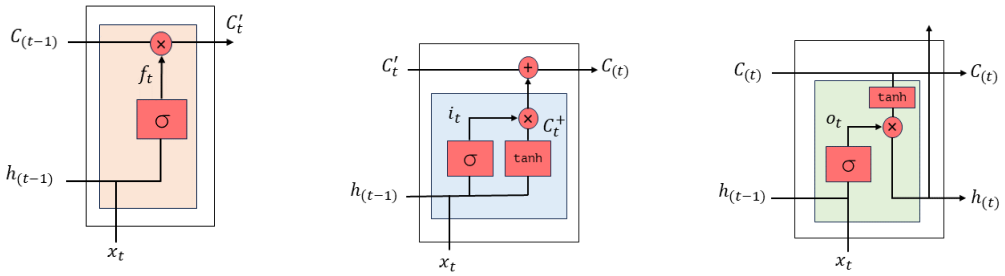
$$C_t = C_t' + i_t \cdot C_t^+. \tag{2.44}$$

Lastly, the output of the network, h_t, is calculated. In this case, the short-term state and the input vector passed through a sigmoid function. The result, o_t, is multiplied by the hyperbolic tangent of the long-term state (Equations 2.45 and 2.46, Figure 9(c)).

$$o_t = \sigma(U_d x_t + V_d h_{t-1} + b), \tag{2.45}$$

$$h_t = o_t \cdot \tanh(C_t). \tag{2.46}$$

In summary, LSTM networks are widely employed for modeling and forecasting time series because they can learn to recognize important inputs (*Input gate*), preserve them for the necessary duration (*Forget gate*), and extract them when appropriate (*Output gate*).



(a) *Forget gate.* (b) *Input gate.* (c) *Output gate.*

Figure 9 – Detailed LSTM cell. Adapted from Géron (2019), Mittal (2019), Matsumoto (2019).

2.2.1.12 Extreme Learning Machine

Extreme Learning Machine (ELM), developed by Huang, Zhu and Siew (2006), is an architecture of ANN known for its fast training process. This algorithm is widely applied to both classification and regression problems (HUANG et al., 2011; DING; XU; NIE, 2014; WANG et al., 2022a). This modeling approach is advantageous for building a pool of models, as single

models train faster compared to other ANNs (LIU; WANG, 2010; LI; GOEL; WANG, 2016; YAO; DAI; SONG, 2019; LARREA et al., 2021).

ELM is structurally similar to a Single-Layer MLP. The main difference between them lies in the training method. In the case of ELM, the weights between the input and hidden layer are not iteratively adjusted. Meanwhile, the output weights are analytically calculated using matrix inversion techniques. For this reason, their training is very fast (the reason for its name) (HUANG; ZHU; SIEW, 2006; YAO; DAI; SONG, 2019):

Regarding ELM formulation, Equations 2.37 and 2.38 (Section 2.2.1.10) can be rewritten as 2.47 (HUANG; ZHU; SIEW, 2006).

$$\mathrm{H}\mu = \mathrm{T}, \tag{2.47}$$

in which,

$$\mathrm{H}(\mathrm{w}_1, \cdots, \mathrm{w}_H, b_1, \cdots, b_H, \mathrm{x}_1, \cdots, \mathrm{x}_n) = \begin{bmatrix} g(\mathrm{w}_1 \cdot \mathrm{x}_1 + b_1) & \cdots & g(\mathrm{w}_H \cdot \mathrm{x}_H + b_1) \\ \vdots & \cdots & \vdots \\ g(\mathrm{w}_1 \cdot \mathrm{x}_n + b_1) & \cdots & g(\mathrm{w}_H \cdot \mathrm{x}_n + b_1) \end{bmatrix}, \tag{2.48}$$

$$\mu = \begin{bmatrix} \mu_1^T \\ \vdots \\ \beta_H^T \end{bmatrix}, \quad \mathrm{e} \quad \mathrm{T} = \begin{bmatrix} \mathrm{y}_1^T \\ \vdots \\ \mathrm{y}^T \end{bmatrix}, \tag{2.49}$$

H is the output matrix of the hidden layer; the i-th column of H represents the i-th output node with respect to x_1, x_2, $\cdots$, x_n. In the case of ELM networks, the algorithm used follows four main steps (HUANG; ZHU; SIEW, 2006):

1. Random generation of weights w_1 and b_i;
2. Calculation of the output of the hidden layer H;
3. Calculation of the weight matrix $\hat{\mu} = \mathrm{H}^{\dagger}\mathrm{T}$;
4. Use of $\hat{\mu}$ to predict $\mathrm{T} = \mathrm{H}\hat{\mu}$,

in which $\mathrm{H}^{\dagger}$ is the Moore-Penrose pseudoinverse of the matrix H (BANERJEE, 1973).

2.2.1.13 Transformers

Transformer models represent one of the most efficient methodologies for processing sequential data, particularly in domains such as natural language processing, computer vision, and speech recognition (VASWANI et al., 2017; LIN et al., 2022; DONG; XU; XU, 2018; ZENG et al., 2023). More recently, Transformer (TSF)-based approaches have been designed to address time series forecasting problems, with a primary emphasis on long-term prediction (ZENG et al., 2023; AHMED et al., 2023).

Despite the numerous variations that have been proposed, the underlying principles of TSF remain unchanged. In summary, TSF is a sequence-to-sequence approach structured in an encoder-decoder configuration, which receives (input) a sequence of words from a source language and then generates the translation in other target language (output) (VASWANI et al., 2017; AHMED et al., 2023).

Commonly, neural sequence transduction models are structured with an encoder-decoder architecture, in which the encoder maps a sequence of symbols (input) $(s_1, \cdots, s_n)$ to a sequence of continuous representation $r = r_1, \cdots, r_n$. Further, given r, encoder generates an output sequence $(l_1, \cdots, l_n)$ of symbols one by one. During each step, the model operates in an autoregressive manner, incorporating the previously generated symbols as additional input when generating the subsequent symbol. In the case of TSF, the same architecture is used, employing stacked self-attention and point-wise, fully connected layers for both the encoder and decoder (GRAVES, 2013; SUTSKEVER; VINYALS; LE, 2014; VASWANI et al., 2017).

Figure 10 present the TSF architecture (VASWANI et al., 2017). The left side of the Figure illustrates the encoder, which consists of a stack of identical layers, each comprising two sub-layers: a multi-head self-attention mechanism, and a fully connected feed-forward network. Right side of the Figure 10 shows the decoder, also composed of stacked layers. However, in addition to the two layers present in the encoder, there is a third layer, which performs multi-head attention over the output of the encoder stack. In both encoder and decoder, residual connections are employed around sub-layers, followed by normalization.

According to the authors (VASWANI et al., 2017), the attention function can be described as mapping a query and a set of key-value pairs to an output, in which output, query, and keys are vectors. Besides, the output is calculated as a weighted sum of the values, in which the weight assigned to each value is computed by a function of the query with the corresponding key.

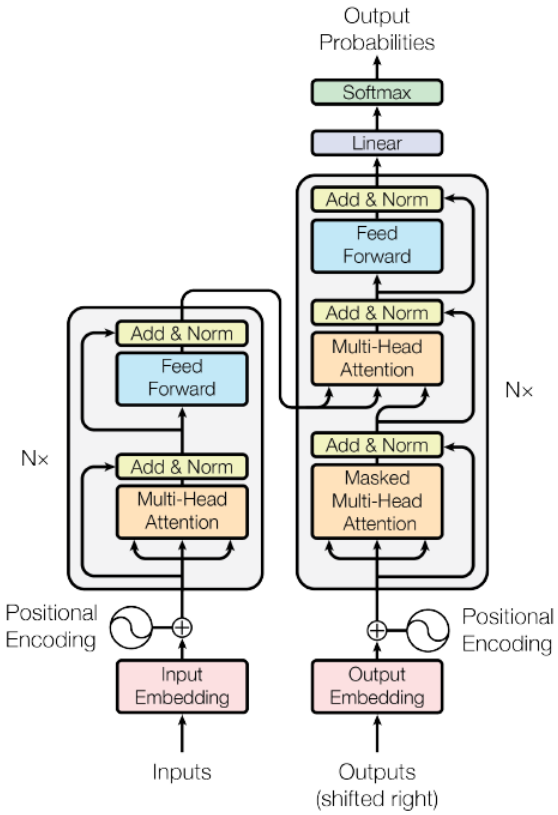


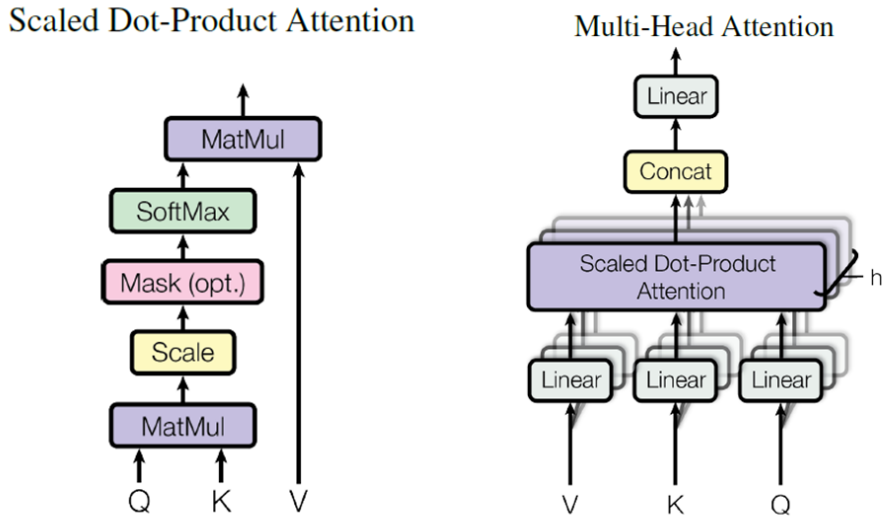
Figure 10 – Transformer architecture (VASWANI et al., 2017).

Vaswani et al. (2017) used an attention function, called "Scaled Dot-Product Attention", in which the attention on set of queries are computed simultaneously, organized together into a matrix Q. Likewise, keys and values are also organized together into matrices K and V. In this way, the matrix of outputs is computed based on Equation 2.50,

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V, \tag{2.50}$$

in which d_k is the dimension of the key vectors.

However, instead of performing a single attention function (with key, values, and queries), the authors proposed to project (linearly) the queries, keys and values h times with different, learned linear projections. On each of these projected versions of queries, keys and values, the attention function is performed in parallel (VASWANI et al., 2017).



(a) Scaled Dot-Product Attention.

(b) Multi-Head Attention consists of several attention layers running in parallel.

Figure 11 – Scaled Dot-Product Attention and Multi-Head Attention (VASWANI et al., 2017).

Following, each layer has a fully connected feed-forward network. This network is applied to each position separately and identically (Equation 2.51).

$$\text{FFN}(X) = \max(0, xW_1 + b_1)W_2 + b_2 \tag{2.51}$$

2.2.2 Multiple Predictor System

The development of Multiple Predictor Systems (MPS) has been gaining prominence in the scientific literature as a robust approach capable of reducing bias and variance in predictions (GHEYAS; SMITH, 2011; KOURENTZES; BARROW; CRONE, 2014; GÉRON, 2022; LV et al., 2022). These systems are normally constructed through three phases: (i) Generation, (ii) Selection, and (iii) Integration (CRUZ; SABOURIN; CAVALCANTI, 2018; YAO; DAI; SONG, 2019; SILVA; DE MATTOS NETO; CAVALCANTI, 2021). Initially, in (i), a pool (P) of M forecasting models is created (Section 2.2.2.1). Next, in (ii), a subset ($P' \subset P$) of m models is selected (Section 2.2.2.2). Finally, in (iii), the selected models are combined.

2.2.2.1 Generation

The objective of this phase is to generate a pool, P, of M models ($M > 1$), $P = \{m_1, m_2, \cdots, m_M\}$. It is important to emphasize that P should be generated with accurate and diverse models (VALENTINI; DIETTERICH, 2004; SILVA; DE MATTOS NETO; CAVALCANTI, 2021). The main idea of MPS is that each model in P is capable of capturing specific patterns in a time series; therefore, their combination will assist in the final prediction outcome, which justifies the required diversity.

Clemen (1989) conducted a literature review on the methodologies for combining forecasts. According to the author, the basic idea is that different forecasting models were capable of capturing distinct aspects of the available information. Further, Perrone (1993) and Hashem (1997) explored the combination of regression models. These authors discuss the degree of redundancy in the information obtained from single models: if all models carry the same information, no benefits may be expected from combining them. However, despite the authors acknowledging the importance of redundancy in the information arising from single models, they did not deeply discuss the relationship between pool generation, diversity, and performance.

Krogh and Vedelsby (1994), in a seminal study, argued that combining the outputs of several networks (or other predictors) is only beneficial when they exhibit disagreement on certain inputs. The authors proved that, at any given instance, the quadratic error of an ensemble is guaranteed to be less than or equal to the average quadratic error of the component estimator (Equation 2.52) (KROGH; VEDELSBY, 1994; BROWN et al., 2005):

$$(f_{ens} - d)^2 = \sum_i w_i (f_i - d)^2 - \sum_i w_i (f_i - f_{ens})^2, \tag{2.52}$$

in which f_{ens} is a convex combination ($\sum_i w_i = 1$) of the single estimators ($f_{ens} = \sum_i w_i f_i$), f is a single model, and d is the target. The first term of the Equation 2.52, $(f_{ens} - d)^2$, indicates the MPS error, while the second indicates the weighted average error of the single models, $\sum_i w_i (f_i - d)^2$. Finally, $\sum_i w_i (f_i - f_{ens})^2$ represents the Ambiguity term and measures the variability among the single models. Since this value is always positive, subtracting it from the first term ensures that the MPS will have a error lower than the average single error (BROWN et al., 2005). Equation 2.52 shows the Ambiguity decomposition.

According to Equation 2.52, the larger the Ambiguity term, the greater the reduction in

MPS error. Nevertheless, as the variability among the individuals increases, the value of the first term also rises. This demonstrates that diversity alone is insufficient; achieving the lowest overall MPS error requires striking the right balance between diversity (the Ambiguity term) and single accuracy (the average error term) (KROGH; VEDELSBY, 1994; BROWN et al., 2005).

One can also discuss MPS diversity through the bias-variance decomposition (BROWN et al., 2005). The generalization error of an estimator can be divided into two components: bias and variance. These components operate in opposition: efforts to reduce the bias often lead to an increase in variance, and conversely (GEMAN; BIENENSTOCK; DOURSAT, 1992; UEDA; NAKANO, 1996). According to Brown et al. (2005), the mean squared error of an MPS depends critically on the amount of error correlation between individual models. Ideally, one would aim to reduce the covariance without inducing any increments in the bias or variance terms.

The generation of a pool can be classified as homogeneous or heterogeneous (YAO; DAI; SONG, 2019; SILVA; DE MATTOS NETO; CAVALCANTI, 2021). If the M models are trained based on the same algorithm, P is said to be homogeneous (YAO; DAI; SONG, 2019). Otherwise, if they are constructed using different algorithms, P is said to be heterogeneous (SANTOS JUNIOR et al., 2022). In the homogeneous case, two methods are commonly used. The first method involves training multiple single models with the same algorithm but varying its parameters - for example: generating M MLP models with different architectures (YAO; DAI; SONG, 2019). The second approach is based on the adoption of resampling techniques (HASHEM, 1997; SILVA et al., 2020; SILVA; DE MATTOS NETO; CAVALCANTI, 2021). In this cases, algorithm such as bagging can be used to generate multiple distinct models from a single training set. The main idea is to train different models using various samples of the data, meaning variables and instances are selected in different ways, and each of these combinations leads to a distinct model.

2.2.2.2 Selection

The objective of this phase is to select a subset ($P' \subset P$) of the best-performing models based on a predefined strategy. A MPS can select only one model (predictor selection), or multiple models (ensemble selection) (SILVA; DE MATTOS NETO; CAVALCANTI, 2021).

Moreover, a MPS can also be classified as static or dynamic. They are considered static when the selected models are always the same for all new instances that arrive at the system. A common strategy adopted in static MPS is the selection of the best models based on

some performance measure for a validation set (FONSECA; GOMEZ, 2016; ADHIKARI; VERMA; KHANDELWAL, 2015) (*pruning by ranking*). The application of Meta-Learning (LEMKE; GABRYS, 2010) and optimization techniques has also been documented in the literature (ABBASIMEHR; SHABANI; YOUSEFI, 2020; SIN; WANG, 2017).

In the case of dynamic MPSs, models are selected whenever a new instance arrives at the system, meaning there is no fixed set of models (YAO; DAI; SONG, 2019; SILVA et al., 2020; SILVA; DE MATTOS NETO; CAVALCANTI, 2021). Since selection occurs for each new instance, dynamic MPSs are typically more complex and have higher computational costs. Dynamic MPSs are detailed in Section 2.2.3.

2.2.2.3 Integration

The third and final phase of an MPS aims to generate predictions by combining models selected from the Generation and Selection stages. Combination techniques can be classified as trainable and non-trainable (SILVA; DE MATTOS NETO; CAVALCANTI, 2021). For non-trainable techniques, average, weighted average, and median are the most commonly used techniques, although estimating the mode is also feasible using kernel density estimation (KOURENTZES; BARROW; CRONE, 2014; WANG et al., 2022b; FATICHI; IVANOV; CAPORALI, 2013; SANTOS JUNIOR et al., 2022). Kourentzes, Barrow and Crone (2014) present a comprehensive discussion of these techniques. The authors argued that the mode operator represents a viable alternative to the average and median operators in forecasting applications.

On the other hand, trainable techniques use ML algorithms to map the relationship between the single models predictions and the observed values of the time series (GÉRON, 2022). Saadallah, Tavakol and Morik (2021) propose a meta-learning approach for aggregating linearly weighted ensembles. In this systems, the combination strategy of the ensemble is depicted as a sequential decision-making process capable of capturing temporal dynamics in time-series data. In turn, Adhikari and Agrawal (2012) present a weighted nonlinear ensemble technique that considers both the forecasts of single models and the correlations among them during the combination process.

2.2.3 Dynamic Ensemble Selection

Dynamic Ensemble Selection (DES) are ensemble approaches in which single models (previously trained) are selected on the fly, based on each new instance to be predicted (CERQUEIRA et al., 2017a; CRUZ; SABOURIN; CAVALCANTI, 2018). DS term is frequently used when only one model is selected, whereas DES indicates that more than one model are selected (SILVA; DE MATTOS NETO; CAVALCANTI, 2021). These approaches have been widely discussed in classification tasks, as their results often outperform standard combination approaches (CRUZ; SABOURIN; CAVALCANTI, 2018; BRITTO JUNIOR; SABOURIN; OLIVEIRA, 2014; ZYBLEWSKI; SABOURIN; WOŹNIAK, 2021; ELMI et al., 2023; CABRAL; OLIVEIRA, 2021; MOURA; CAVALCANTI; OLIVEIRA, 2021). However, its application to time series forecasting remains less explored.

One of the earliest publications addressing DS for forecasting tasks dates back to 2006, when Yankov, DeCoste and Keogh (2006) proposed a method to classify time series into two classes, which are better predicted using one of two possible neural network predictors. The authors demonstrated that the proposed approach reduced prediction errors on the test set in the developed experiment. Further, Widodo and Budi (2011), Widodo and Budi (2012) and Budi, Aji and Widodo (2014) proposed a selection approach in which single models are selected through a similarity analysis between the new instance being predicted and those from the training and/or validation sets. The authors use two similarity measures, Euclidean and Dynamic Time Warping (DTW).

Moreover, some authors proposed selection strategies guided by the performance of models within a RoC. The RoC is composed by the patterns in the in-sample (validation and/or training set), which are more similar to a new test pattern to be predicted. Initially proposed by Woods, Kegelmeyer and Bowyer (1997) for classification tasks and adapted by Rooney et al. (2004) for regression problems, the approach of dynamically selecting models based on RoC has proven to be a robust alternative (SILVA; DE MATTOS NETO; CAVALCANTI, 2021; CRUZ; SABOURIN; CAVALCANTI, 2018).

A dynamic MPS based on defining RoC, entitled Dynamic Selection based on the Nearest Windows (DSNAW), was proposed by Silva, de Mattos Neto and Cavalcante (SILVA; DE MATTOS NETO; CAVALCANTI, 2021). The authors presented a DES approach that selects one or more competent models according to their performance in the RoC composed of the nearest antecedent windows to the new target time window. The experiment used ten well-known time series. In this case, bagging algorithm (MORETTI et al., 2015) was used to generate the

pool. In their study, the authors included two variations for the DES constructed from RoC, varying the combination method: average (DESLA$_a$) and median (DESLA$_m$). LA stands for local accuracy.

Santos Junior et el., (2022) also proposed a DES based on RoC to solar irradiance forecasting, however, using a heterogeneous pool composed of seven well-discussed algorithms in the literature (SANTOS JUNIOR et al., 2022). The proposed approach achieved an overall accuracy that surpasses that of individual models. In this case, it is worth noting that constructing a heterogeneous pool can assist in increasing diversity, an important factor for a MPS.

Regarding the use of RoC to construct DESs, some limitations can be identified. First, it is known that the quality of RoC is a key factor in the accuracy of the MPS; however, choosing the appropriate similarity measure and RoC size is a challenging task. Further, the dependence on the chosen similarity metric, once different metrics capture different aspects of time series, thus, it is assumed that one may be more (or less) appropriate for a given series (YU et al., 2006; SILVA, 2020; YULIANTI et al., 2023). It is known that some metrics are sensitive to outliers, which can hinder their use in series that present anomalous values Kumar, Chhabra and Kumar (2014). The construction of RoCs is a computationally expensive activity, since each new instance to be predicted must be compared with all others in the validation and/or test set. Lastly, there are no guarantees that the new data to be predicted will be similar to the time series on which the MPS is based.

Besides, Yao, Dai and Song (2019) presented DES methods based on Consensus of Predictors (CP). Methods based on CP start from generating multiple Pool of Ensembles (PoE) instead of pools of individual models. Next, the PoE with the highest consensus is selected. Consensus evaluation can be performed by estimating variance or applying clustering methods. The authors assume that the higher the extent of CP, the better the predictive performance of the PoE. According to the authors, a key advantage of their proposed approaches is their independence from information specific to the RoC, overcoming limitations imposed by its definition. However, as limitation, the effectiveness of CP-based methods can significantly depend on specific parameters, such as the number of ensembles in the PoE and the criteria used to evaluate consensus. Furthermore, due to the limited number of studies on the method, it is non clear how variability in the input data can affect the stability of consensus and the selection of the most suitable PoE.

Yao, Dai and Song (2019) also proposed the Dynamic Validation Set determination algorithm based on the similarity between the Output Profile of the test sample and the Output

Profile of each training sample (DVS-OpOp). According to authors, the method follows the logic of select instances (in validation set) similar to the unseen instance (new observation to be predicted). Nevertheless, the similarity is measured by combining the output profile of the unseen instance with the output profiles of the training set. In this case, the output profile is a vector that consists of the predicted values obtained by the single models for that instance. This algorithm's main advantage is that it is not constrained by how the local region is defined in feature space. As limitation, DVS-OpOp ignores the local information of the new test instance and only takes into account its global knowledge.

Another approach was proposed by Neto et al. (2020). The authors propose a Temporal-Window Framework (TWF), in which the time series is split into subsets (each subset may intercept its neighbours). Each new instance will be evaluated, and the model will be selected based on the similarity between the partitions and the new instance to be predicted. The results of the experiment demonstrated that the proposed method outperformed the other models used for comparison. The limitations include the significant dependence of performance on parameter values, which vary based on the characteristics of the time series. Adjusting these parameters can improve results, but it is a costly and problem-specific task.

Another DS approach is the Arbitrated Dynamic Ensemble (ADE) (CERQUEIRA et al., 2017b; CERQUEIRA et al., 2019). The authors propose a strategy that adaptively combines forecasters using metalearning techniques. In this case, the method effectively modeled the expertise of base learners across different parts of the time series, allowing the ensemble model to adapt consistently. Regarding limitations, one of the main issues is the inability to directly model interdependencies between forecasters, which might be important for accounting for diversity among the models. Moreover, metalearning methodologies introduce an additional layer of complexity that can hinder interpretability.

Table 2 presents the main advantages and limitations of the approaches employed for DS in time series forecasting problems.

In the context of applying DES methods to time series forecasting problems, one of the most common limitations is the computational cost, especially due to the inherent nature of the problem: the search for the optimal models to be combined in order to forecast a new test instance. Generally, computational costs escalate with the increasing number of single models and the size of the time series. This occurs because the space for selection becomes larger. Additionally, the challenge of parameter selection can be a significant limitation, as improper choices can undermine forecasting performance. Increasing the number of parameters

Table 2 – DS methods, advantages, limitations, and references.

Method	Advantages	Limitations	References
RoC	Mathematically simple. Easy to implement. Most commonly used criterion for DS.	Quality of RoC influenced by similarity metric and size. High computational cost. Loss of temporal information.	SILVA (2020), Silva, DE MATTOS NETO and Cavalcanti (2021), SANTOS JUNIOR et al. (2022)
CP	Does not need to extract information from local regions.	Influenced by criteria used to evaluate consensus and number of ensemble. High computational cost. Low interpretability. Limited number of studies.	Yao, Dai and Song (2019)
DVS-OpOp	It is not restricted by the definition of the local region in feature space.	Ignores the local information of the new test instance and only takes into account its global knowledge. High computational cost. Low interpretability. Limited number of studies.	Yao, Dai and Song (2019)
TWF	To train predictors specialized in different parts of the time series.	Significant dependence of performance on parameter values. High computational cost. Low interpretability. Limited number of studies.	Neto et al. (2020)
ADE	To model their expertise in the different parts of the time series and adapt the ensemble weights.	Inability to directly model interdependencies between forecasters. High computational cost. Low interpretability.	Cerqueira et al. (2017b), Cerqueira et al. (2019)

complicates the task further, and complex parameters that are difficult to understand present an additional challenge. Another significant concern is the interpretability of the methods, as understanding how the selection process occurs can be challenging in some cases. Besides, RoC-based methods are among the most well-described in the literature (SILVA, 2020; SILVA; DE MATTOS NETO; CAVALCANTI, 2021; SANTOS JUNIOR et al., 2022). However, the choice of

similarity measures for constructing the RoC and the variability in input data can affect the effectiveness and stability of the selected models. The limited number of publications is also considered a limitation as it hinders a thorough understanding of the proposed methods.

The MPS presented in this thesis, DESTC, addresses some of the limitations of other algorithms. The first difference is that, by not requiring the creation of a RoC, it avoids issues related to the choice of similarity metrics. Additionally, it is mathematically simple and has a low computational cost.

2.3 PERFORMANCE MEASURES

Some classical performance measures are Mean Squared Error (MSE), Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and Theil's U. MSE (Equation 2.53) is calculated from the average of the squares of the differences between the observed values and the values predicted by the model. Therefore, the lower the value, the better the performance of the evaluated model. MSE is a performance measure sensitive to outliers.

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(z_t - \hat{z}_t)^2, \tag{2.53}$$

in which z_t is the observed value of the series at time t, $\hat{z}_t$ is the MPS forecast, and n is the time series size.

RMSE (Equation 2.54) is the square root of MSE. It is a performance measure that presents the magnitude of the error in the same unit of measurement as the time series, aiding in their interpretations. Like MSE, lower values indicate better performance.

$$\text{RMSE} = \sqrt{\text{MSE}(z_t, \hat{z}_t)} \tag{2.54}$$

MAE (Equation 2.55) measures the average of the absolute differences between the observed values and those predicted by the model. Just like RMSE, the result of this performance measure is given in the same dimension as the modeled time series. The lower its value, the better the model will be.

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|z_t - \hat{z}_t|, \tag{2.55}$$

Theil's U (Equation 2.56), meanwhile, compares models with a Random Walk model. if Theil's U < 1, the model is better than the naive predictor; otherwise if Theil's U > 1, the

compared model is worse. The Random Walk model is commonly used as a benchmark. In this model, the prediction $\hat{z}_{t+1}$ is assumed to be equal to the value of observation z_t, making it mathematically simple. Therefore, Theil's U is a relative metric that facilitates comparison with a naive predictor, unlike MSE, which is an absolute metric that provides a direct measure of mean squared error.

$$\text{Theil's U} = \frac{\sum_{t=1}^{N}(z_t - \hat{z}_t)^2}{\sum_{t=1}^{n}(z_t - z_{t-1})^2}, \tag{2.56}$$

Aggregating Ranking Measure (ARM) (Equation 2.57) quantifies the average rank of models for a given set of time series. The closer the ARM is to 1, the better the performance. An ARM $= 1$ indicates that the model outperforms all others for all performance measures used.

$$\text{ARM} = \frac{\sum_{i=1}^{\text{PM}} \text{rank}(m_i)}{n}, \tag{2.57}$$

in which PM is the number of performance measures used and $rank(m_i)$ is the raking of the m-th model regarding i-th performance measures.

Moreover, when developing new forecasting methodologies, it is valuable to compare them with alternatives from the literature. In this scenario, one can utilize the Percentage Difference (PD) (Equation 2.58) between the new method and those already proposed. A comparison in percentage terms can be made based on an individual performance measure, such as RMSE.

$$\text{PD} = \frac{P_{\text{lit}} - P_{DESTC}}{P_{\text{lit}}} \cdot 100, \tag{2.58}$$

in which P_{lit} and P_{DESTC} are, respectively, a model in the literature and the proposed one. These are the RMSEs obtained by the models from the literature and a new model, respectively. Positive values indicate an improvement of the proposed approach compared to a certain model from the literature; otherwise, the model from the literature is superior.

3 PROPOSED APPROACH

DS systems have emerged as a powerful tool for time series forecasting, offering superior predictive performance compared to traditional individual modeling and static MPS. Their ability to adapt to changing data patterns and select the most suitable forecasting models from a pool of candidates makes them well-suited for a wide range of applications. One of the most common strategies for model selection is the use of similarity metrics to identify, within the training or validation set, which instances are most similar to a new test instance. However, this approach faces two main challenges. The first challenge is choosing the optimal similarity metric. With numerous options available in the literature, each with its own strengths and weaknesses, selecting the right one becomes a complex task. Secondly, there is no guarantee that similar instances are stored in the pool compared to those that need to be predicted. This absence of similar data can significantly complicate the selection process, potentially leading to suboptimal model choices and unreliable forecasts.

These difficulties became evident during the COVID-19 pandemic (AWADASSEID et al., 2020; CHUNG et al., 2021). As COVID-19 drastically affected healthcare systems globally, many countries implemented strategies to reduce the spread of outbreaks, such as developing decision support systems for monitoring, predicting, and controlling the situation (MUHAREB; GIACAMAN, 2020; RAHIMI; CHEN; GANDOMI, 2021). However, developing accurate models for predicting COVID-19 time series poses a significant challenge due to the complex behavior of the data (FIRMINO et al., 2020). While numerous researchers have implemented models to forecast incidence and mortality, the results often suffer from low accuracy; primarily attributable to the unknown behavior of the disease and incorrect modeling assumptions (PETROPOULOS; MAKRIDAKIS, 2020; IOANNIDIS; CRIPPS; TANNER, 2020; LUO, 2021). Additionally, as a consequence of the evolution in the number of cases during the pandemic and the changes in data patterns, the new values to be predicted became significantly different from the past values of the time series. Consequently, attempting to use DS for COVID-19 incidence series encounters difficulties.

More specifically, it was observed that the COVID-19 incidence time series was characterized by cycles of three consecutive phases: exponential increase, plateau, and then decrease (FIRMINO et al., 2020). Initially, exponential growth represents the early stage of an outbreak, which has a high spread due to a vulnerable population. The second phase initiates

when the number of new cases begins to stabilize due to adopting mitigation strategies. Finally, in the third phase, the incidence time series exhibit a decreasing pattern. However, before the pandemic is fully controlled, factors such as virus mutation and inefficient reopening strategies can lead to a further increase in the number of cases, which can indicate the beginning of a new cycle (a new wave) (SAHA et al., 2021).

Based on this pattern, the Dynamic Ensemble Selection based on Trend Classification (DESTC) is proposed. The general idea is to evaluate the best models for each class of the trend using a validation set. Then, for each new instance to be predicted, the best models for the current trend class are combined. As the trend of the series changes, the system dynamically selects the appropriate models to combine. The approach proposed introduces a new way of dynamically selecting models based on statistical information from the time series, specifically the trend. Additionally, by not depending on similarity metrics for model selection, DESTC avoids the problems directly linked to them. Furthermore, the proposal offers a computational cost advantage, as estimating trends is to be more computationally economical compared to the construction of RoC or the utilization of meta-learning alternatives.

Figure 12 illustrates an example of DESTC focusing on the positive trend (dashed red circles). DESTC selects the most appropriate forecasting models for this trend (validation set) and combines them in order to forecast new patterns (test set). DESTC addresses three different trend classes (positive, negative, or none). The evaluation ensures that the most suitable models are selected for each trend type, leading to highly accurate and reliable forecasts across all trend patterns.

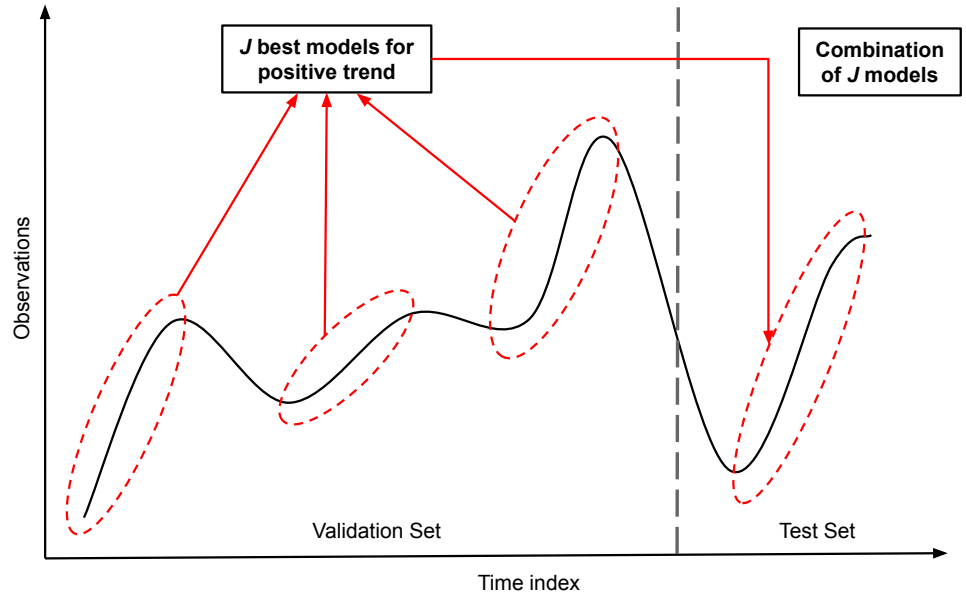


Figure 12 – Illustrated example of DESTC. Dashed red circles represent positive trend patterns in the time series. DESTC selects the most appropriate forecasting models for this trend pattern (in validation set) combines them (test set).

For trend classification, various methods may be employed. However, in the experiments presented in this study, the Mann-Kendall statistical test (Section 2.1.1.1) will be utilized, given its well-established use in the literature. The Sen Slope estimator (Section 2.1.1.2) can also be employed for this purpose; however, here it will be used solely to discuss the results. Another similarly simple alternative would be trend classification based on the coefficients of a linear regression.

3.1 FORMAL DEFINITION

Let Ψ be the set of trend classes identified in the time series (e.i., positive, negative, neutral). For each trend $\psi \in \Psi$, there exists a set of models, J_ψ, that have been evaluated and classified as optimal for that trend. Given a new test instance Z_w, its trend firstly is classified based on an estimation method $\mathcal{C}(\cdot)$ (Equation 3.1), and the prediction $\hat{Z}_{w+t}$ is computed as the combination of the models selected for the current trend (Equation 3.2). DESTC is presented below,

$$\psi = \mathcal{C}(Z_w), \tag{3.1}$$

$$\hat{Z}_{w+t} = \Omega_{J_\psi}(Z_w) \tag{3.2}$$

in which Ω_{J_ψ} is the combination method of J best models for the ψ trend class, J_ψ. The combination method can be mean, median, mode, or any other alternative that combines estimates from single models.

3.2 PSEUDO-CODE

A pseudo-code for DESTC is presented in Algorithm 1. The inputs of algorithm are training (Z_{tr}) and validation (Z_v) sets, pool of models P, and window size (w) for trend classification. The output is the prediction $\hat{Z}_{w+t}$ for a new test instance (Z_t). The Algorithm is divided into two phases: (a) training and (b) test.

The first step of the training phase is training the P models. DESTC is a model-agnostic approach. However, P training must be performed to generate accurate models capable of modeling different local patterns. Here, Z_{tr} and Z_v are used for this task. Instances of the validation set (Z_v) are classified according to their trend, based on the concepts (positive, negative, neutral). There are several trend classifier methods, and any of them can be used for this task in the DESTC.

Following, each model of P_t is evaluated in the validation set (Z_v) using a performance measure, such as Root Mean Square Error (Eq. 2.54), to create three lists indicating the order of the best models for each trend class (ranking lists). The ranking for positive trend, for example, will contain the best j models considering only the instances that were classified as positive trend.

Lastly, regarding (b), when the system receives a new instance (Z_w), the trend is first evaluated and, based on the ranking list with the same trend classification, the J best models are selected (selection step). After selection, the J single forecasts are obtained and combined (integration step). For a better understanding of the DESTC, Figure 13 illustrates the proposed approach.

Algorithm 1 Dynamic Ensemble Selection based on Trend Classification - DESTC

Require:

Training (Z_{tr}) and validation (Z_v) sets
Pool of algorithms (P)
Window size (w) for trend classification

Ensure:

Prediction $\hat{Z}_{w+t}$ for new test instance Z_t

// Training phase (a)

$P_t \leftarrow \text{TrainPoolModels}(Z_{tr}, Z_V, P)$
$T_v \leftarrow \text{TrendClassification}(Z_v, w)$
for each trend $\psi \in \Psi$ **do**
    $\text{RankList}_\psi \leftarrow \text{CreateRankLists}(P_t, \psi)$
end for

// Test phase (b)

$\psi_{\text{current}} \leftarrow \text{IdentifyCurrentTrend}(Z_t, w)$
$J \leftarrow \text{GetModelsForTrend}(P_t, \text{RankList}_\psi, \psi_{\text{current}})$
$\hat{Z}_{w+t} \leftarrow \text{Combine(J)}$
return $\hat{Z}_{w+t}$

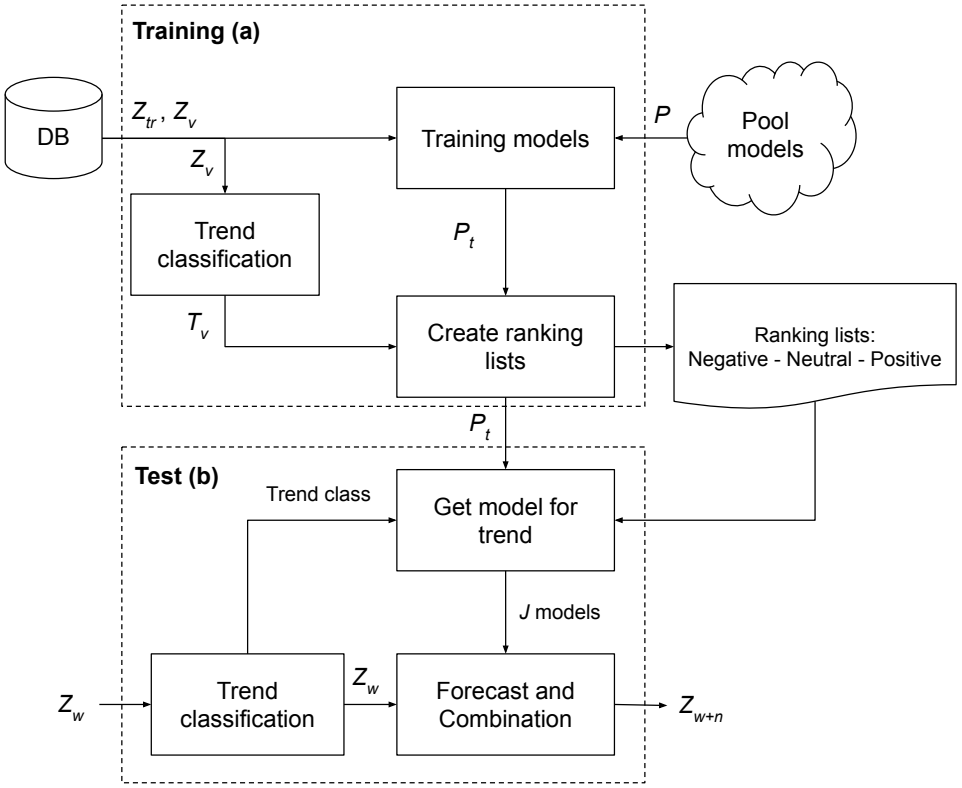


Figure 13 – Flowchart of DESTC. The training phase (a) involves creating the pool (P) based on the training set (Z_{tr}), and selecting the best models, by trend class, using the validation set (Z_v). In the test phase (b), a new instance (Z_W) is classified according to its trend, and the best models (based on the ranking list) are combined.

4 EXPERIMENT A - COVID-19 TIME SERIES FORECASTING

The primary objective of this thesis is (i) to develop a new selection approach based on trend classification to enhance the forecasting of COVID-19 incidence time series. Consequently, this chapter aims to evaluate the proposed approach, DESTC. To illustrate the overall performance of DESTC, this chapter presents an experiment focused on forecasting COVID-19 incidence time series.

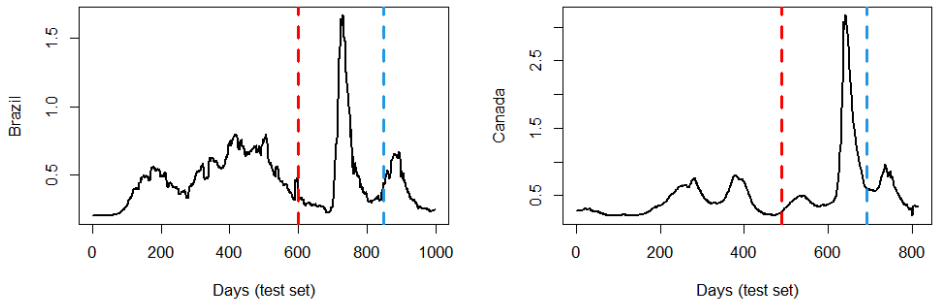
4.1 EXPERIMENTAL PROTOCOL

The experimental evaluation used eight time series from different countries. These time series (Table 3) exhibit distinct statistical patterns, including minimum, maximum, average, median, standard deviation, and coefficient of variation. Additionally, the temporal ranges (start and end dates) vary across the datasets, allowing the proposed approach to be evaluated under diverse scenarios. A 7-days rolling mean was applied in the original time series in order to reduce the weekly seasonal effect (ZIVOT et al., 2003). This preprocessing step aims to highlight local trends that happen in the COVID-19 incidence time series, minimizing the "weekend effect" (SOUKHOVOLSKY et al., 2021). This effect occurs because the patients with initial symptoms tend to go to hospitals only after the weekend. After that, the time series were normalized to the interval [0.2, 0.8] and split following the temporal order into training (60%), validation (25%), and test (15%) sets. Figures 14 and 15 present the time series training, validation, and test sets.

Table 3 – Start and end dates, sample sizes, number of peaks (NP), minimum, maximum, average, median, standard deviation (SD), and coefficient of variation (CV) from each time series used in the experiments.

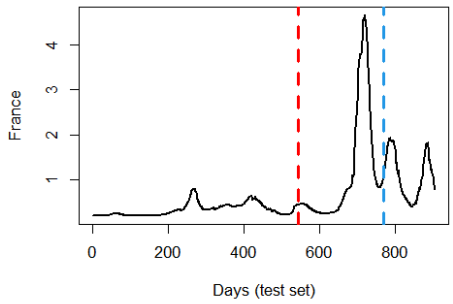
Country	Start	End	Sample size	NP	Min.	Max.	Average	Median	SD	CV(%)
Brazil	02/01/2020	10/31/2022	1003	4	0	189526	34876.8	30134.9	30468.1	87.4
Canada	04/01/2020	06/30/2022	820	5	273	42191	4800.1	3009.9	6330.7	131.9
France	02/01/2020	07/31/2022	911	4	0	354886	36095.3	15544.6	61566.5	170.6
Germany	02/01/2020	08/31/2022	942	4	1	233161	34234.1	11380.9	52059.7	152.1
Italy	02/01/2020	08/31/2022	942	4	0	181822	23249.1	11705.6	34255.5	147.3
Spain	05/01/2020	07/31/2022	821	5	289	141967	15895.3	9603.7	24097.7	151.6
UK	02/01/2020	07/31/2022	911	4	0	214687	25731.2	15511.5	32740.5	127.2
US	02/01/2020	10/31/2022	1003	4	0	809735	96289.1	64247.8	118586.2	123.2

In the modeling stage, a heterogeneous pool of fifty forecasting models was created to

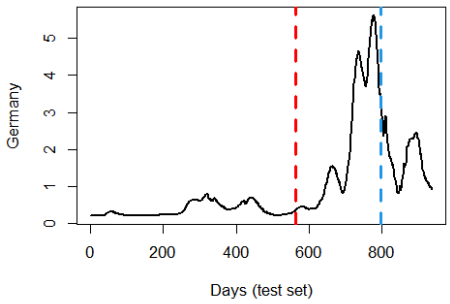


(a) Brazil.

(b) Canada



(c) France



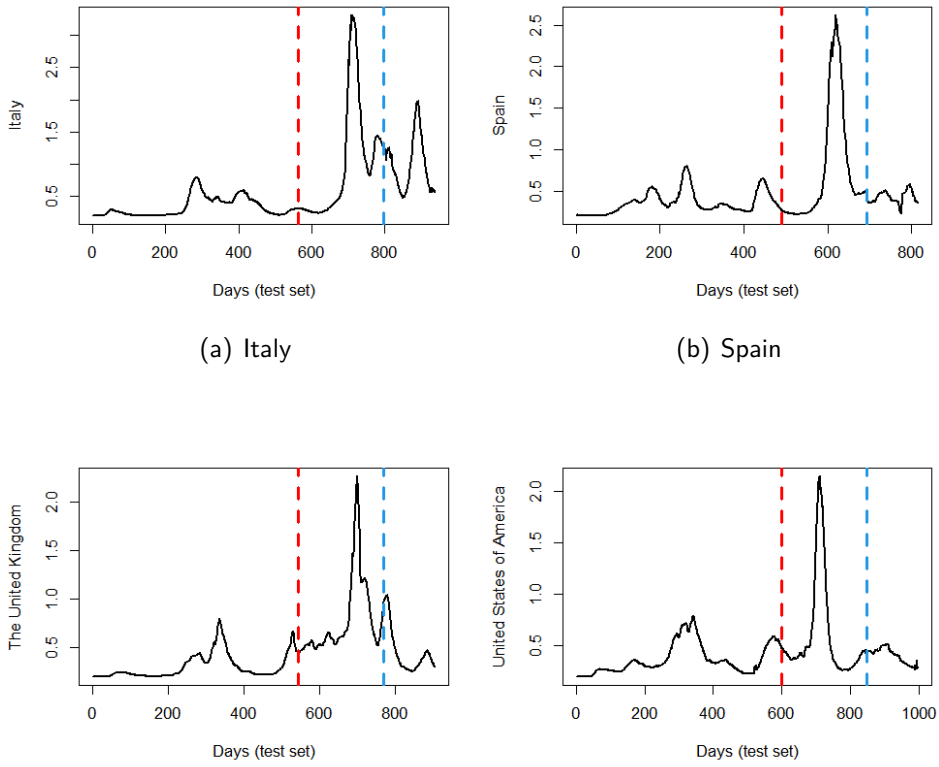
(d) Germany

Figure 14 – Training (before red line), validation (between red and blue lines), and test (after blue line) sets for COVID-19 incidence time series in Brazil, Canada, France, and Germany.

generate the ensembles. This pool size was determined based on preliminary studies using validation sets, which indicated that, for these cases, increasing the number of models did not lead to a significant change in the diversity of the pool and in the oracle's RMSE.

The heterogeneous pool consists of seven single individual models: two classical time series models (ARIMA and ETS) and five other machine learning models, ELM, LSTM, MLP, SVR, and Extreme Gradient Boosting (XGB). XGB is an algorithm based on gradient boosting trees (CHEN; GUESTRIN, 2016; QIU et al., 2022; CHEN, 2015). Further, for increasing the diversity of the P pool, ELM models were created using different parameter combinations. This modeling approach was chosen because ELM models have a lower computational training cost compared to other training algorithms such as MLP and SVR (YAO; DAI; SONG, 2019).

Regarding DESTC, the trend classification task was performed using the Mann-Kendall statistical test (MANN, 1945) (Section 2.1.1.1). This is a classic alternative, commonly applied in literature. Other parameters such as the numbers of models for each trend class (ie., $n_{neutral}$, $n_{positive}$, and $n_{negative}$) and the combination method are presented in Table 4. These parameters



(a) Italy (b) Spain

(c) UK (d) US

Figure 15 – Training (before red line), validation (between red and blue lines), and test (after blue line) sets for COVID-19 incidence time series in Italy, Spain, UK, and US.

were selected through a grid search strategy.

Classical models (ARIMA and ETS (HYNDMAN; ATHANASOPOULOS, 2018)) are trained without a validation step. Training and validation sets were joined for parametric estimation for these models. The parameters of ML models (SVR, ELM, MLP, LSTM, and XGB) were chosen by grid search (Table 4). The parameters used to train SVR, MLP, and LSTM models are based on (OLIVEIRA; SILVA; DE MATTOS NETO, 2021; ADHIKARI; VERMA; KHANDELWAL, 2015). ELM and XGB models are trained based on (SONG; DAI, 2017).

Further, in order to compare DESTC with more recent techniques from literature, Transformers (TSF) (HEATON, 2022) models were also implemented. These models have received attention in recent years in the field of time series modeling and forecasting (HEATON, 2022). In these experiments, TSF models are trained according to Heaton (2022).

Eight ensemble-based approaches were developed using the same P pool of the DESTC. Inspired by the literature (SILVA; DE MATTOS NETO; CAVALCANTI, 2021; MAALIW et al., 2021), these ensembles were created to perform a more comprehensive comparison. Two non-trainable

ensembles with average (eSA) and median (eSM), three trainable ensembles using ELM (eELM), MLP (eMLP) (FERNANDES; EBECKEN; ESQUERDO, 2017), and SVR (eSVR) (DE MATTOS NETO et al., 2021), three dynamic selection approaches to choose one model (DSLA), and more than one predictor using average (DESLA$_a$) and median (DESLA$_m$) combinations (SILVA; DE MATTOS NETO; CAVALCANTI, 2021). All parametric selections were based on grid search (Table 4). It is important to mention that for all trainable ensembles (eELM, eMLP, and eSVR), the combination model receives the forecasts of all models of the P pool as input).

Three well-know Performance Measure (PM) were used to evaluate each model, Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and Theil's U (Equations 2.54, 2.55, and 2.56. Diebold–Mariano (DM) statistical test (DIEBOLD; MARIANO, 1991) was applied to the residuals of the models to verify if there are (or not) statistical differences between the proposed and alternative approaches. This test is a classic alternative for assessing the quality of forecasting models (DIEBOLD, 2015; MOHAMMED; MOUSA, 2020). A 0.05 significance level was considered. The null hypothesis of the DM test is that both models exhibit similar prediction performances. Otherwise, the alternative hypothesis is that there is a difference between the models forecasting performance. Additionally, the Bayesian signed-rank test (BENAVOLI et al., 2017) was employed to comparatively evaluate the performance of the MPS in this experiment, utilizing the Python package Autorank (HERBOLD, 2020).

Table 4 – Grid search parameters used by model.

Models	Parameters	Values
ARIMA, SARIMA	p, d, q, P, D, Q, Z	(HYNDMAN; ATHANASOPOULOS, 2018)
ETS	E, T, S	(HYNDMAN; ATHANASOPOULOS, 2018)
SVR, eSVR	Lag	5, 7, 10
	Kernel	linear, radial
	Gamma	0.01, 0.1, 1, 10, 100, 1000
	Cost	0.01, 0.1, 1, 10, 100, 1000
	Epsilon	0.0001, 0.001, 0.01
	Tolerance	0.001
ELM, eELM	Lag	5, 7, 10
	Activation function	relu, tansig, rbf
	Units in hidden layer	25, 50, 100, 150, 200
MLP, eMLP	Lag	5, 7, 10
	Activation function	sigmoid
	Units in hidden layer	10, 15, 20
	Algorithm	adam
LSTM	Lag	5, 7, 10
	Activation function	relu
	Units in hidden layer	10, 100, 500
	Algorithm	adam
TSF	Lag	5, 7, 10
	Head size	5, 10, 20
	Number of attention heads	1, 2, 4
	Units in hidden layers	10, 25, 50, 100
DSLA	k	10, 20
	n	1
	Distance	euclidian
$DESLA_a$	k	5, 10, 20, 25
	n	5, 10, 20, 25
	Combination method	average
	Distance	euclidian
$DESLA_m$	k	5, 10, 20, 25
	n	5, 10, 20, 25
	Combination method	median
	Distance	euclidian
DESTC	$n_{neutral}$	3, 5, 10, 25
	$n_{positive}$	3, 5, 10, 25
	$n_{negative}$	3, 5, 10, 25
	Combination method	median, average

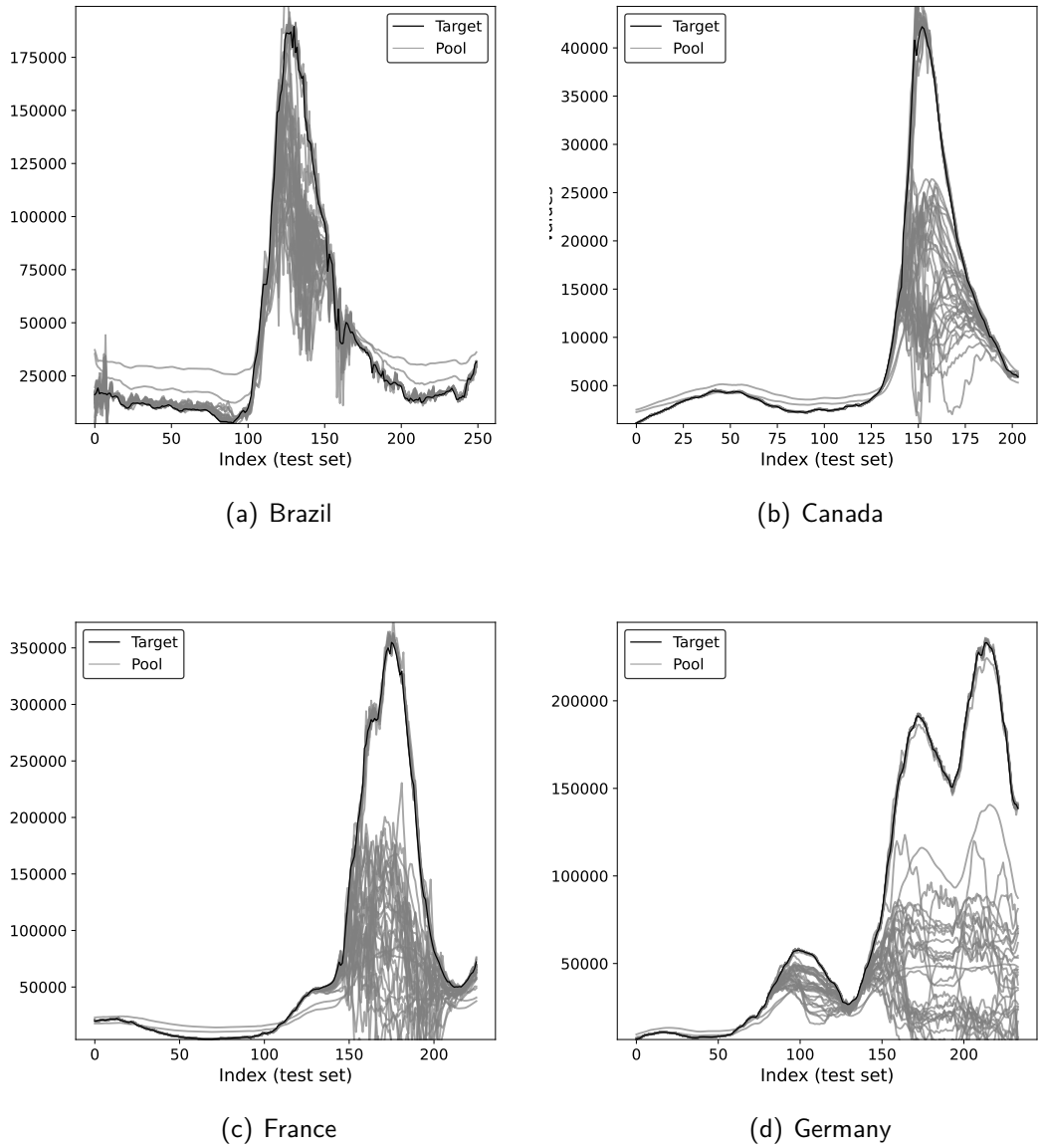
4.2 RESULTS

The results will be presented in two subsections. Initially, the pool generated for the construction of the DESTC and other ensembles will be presented (Section 4.2.1). Following that, the results regarding the performance of the models will be discussed (Section 4.2.2).

4.2.1 Pool generation

The pools generated for the eight COVID-19 time series incidence used in this experiment are presented in Figures 16 and 17 (validation sets). Black lines represent the observed values of a time series, while the gray are the forecasts generated by the models contained within the

respective pool.



(a) Brazil

(b) Canada

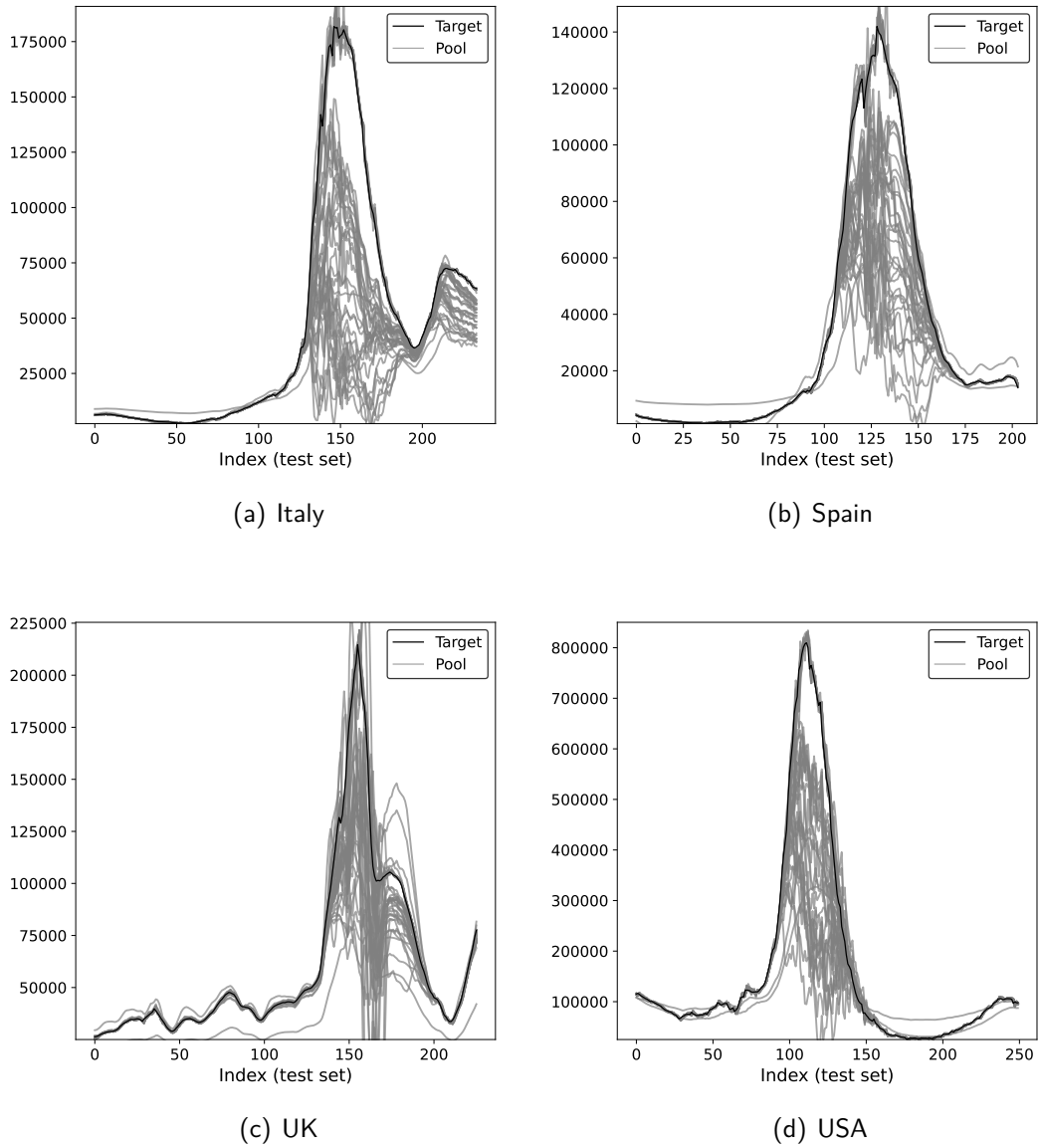
(c) France

(d) Germany

Figure 16 – Generated pools (validation sets) for Brazil, Canada, France, and Germany time series.

In general, it can be observed that the pools contain a diversity of models, although some of them may lack predictive capability, as they underestimate the series (predicted values lower than the actual ones). The observed behavior is likely attributed to the absence of the temporal pattern from the validation set in the training set (Figures 14 and 15). This phenomenon, prevalent during the COVID-19 pandemic, was a primary factor contributing to the poor performance of forecasting models. However, in all cases, it is possible to observe the presence of models that adequately followed the growth and decline of the curves in the time series. Consequently, if the dynamic selectors are able to appropriately select these models, the

accuracy of the forecasting results is expected to improve.



(a) Italy

(b) Spain

(c) UK

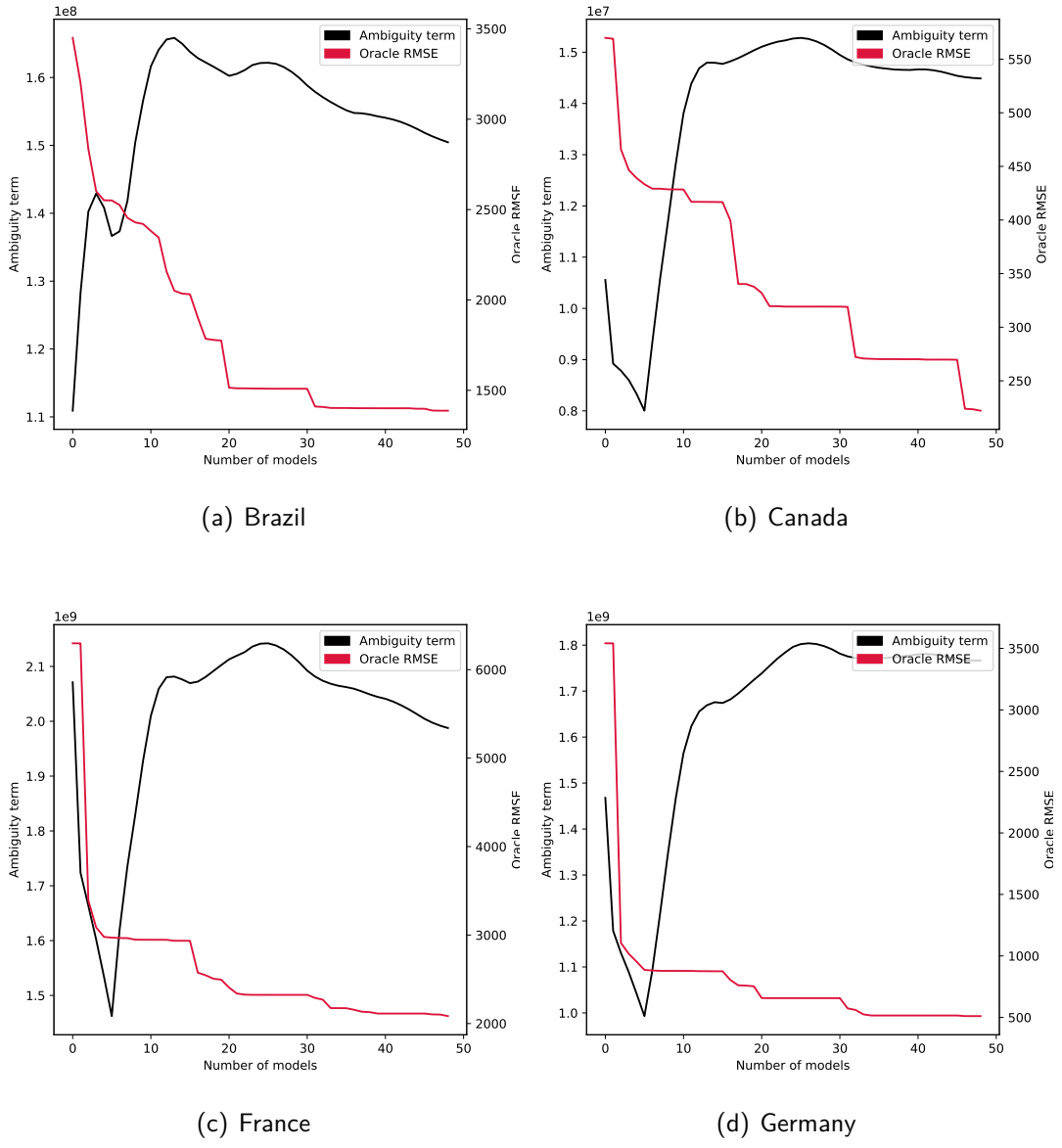
(d) USA

Figure 17 – Generated pools (validation sets) for Italy, Spain, UK, and USA time series.

Moreover, Figures 18 and 19 show the ambiguity term and the RMSE across all cases. All cases exhibit a similar behavior. Initially, the ambiguity term increases as the number of models in the pool rises. This ascent levels off around the 30 models. This suggests that the diversity of the pool increases up to approximately thirty models; beyond this point, the ambiguity term either decreases (Brazil, France, Spain, UK, and USA) or stabilizes (Canada, Germany, and Italy).

The Oracle's RMSE exhibits an inverse relationship with the number of models. The Oracle represents the highest accuracy achievable by the pool, as it chooses the forecast that is closest

to each point in the test sample (OLIVEIRA; SILVA; DE MATTOS NETO, 2021). Initially, the RMSE values exhibit a consistent decline until reaching approximately 30 models. Beyond this point, the RMSE stabilizes, indicating that adding more models does not significantly enhance the Oracle's performance. Consequently, systems employing dynamic selection strategies are likely to exhibit a stabilized performance.



(a) Brazil (b) Canada

(c) France (d) Germany

Figure 18 – Ambiguity term and RMSE of the Oracle for pools with 50 models in Brazil, Canada, France, and Germany time series.

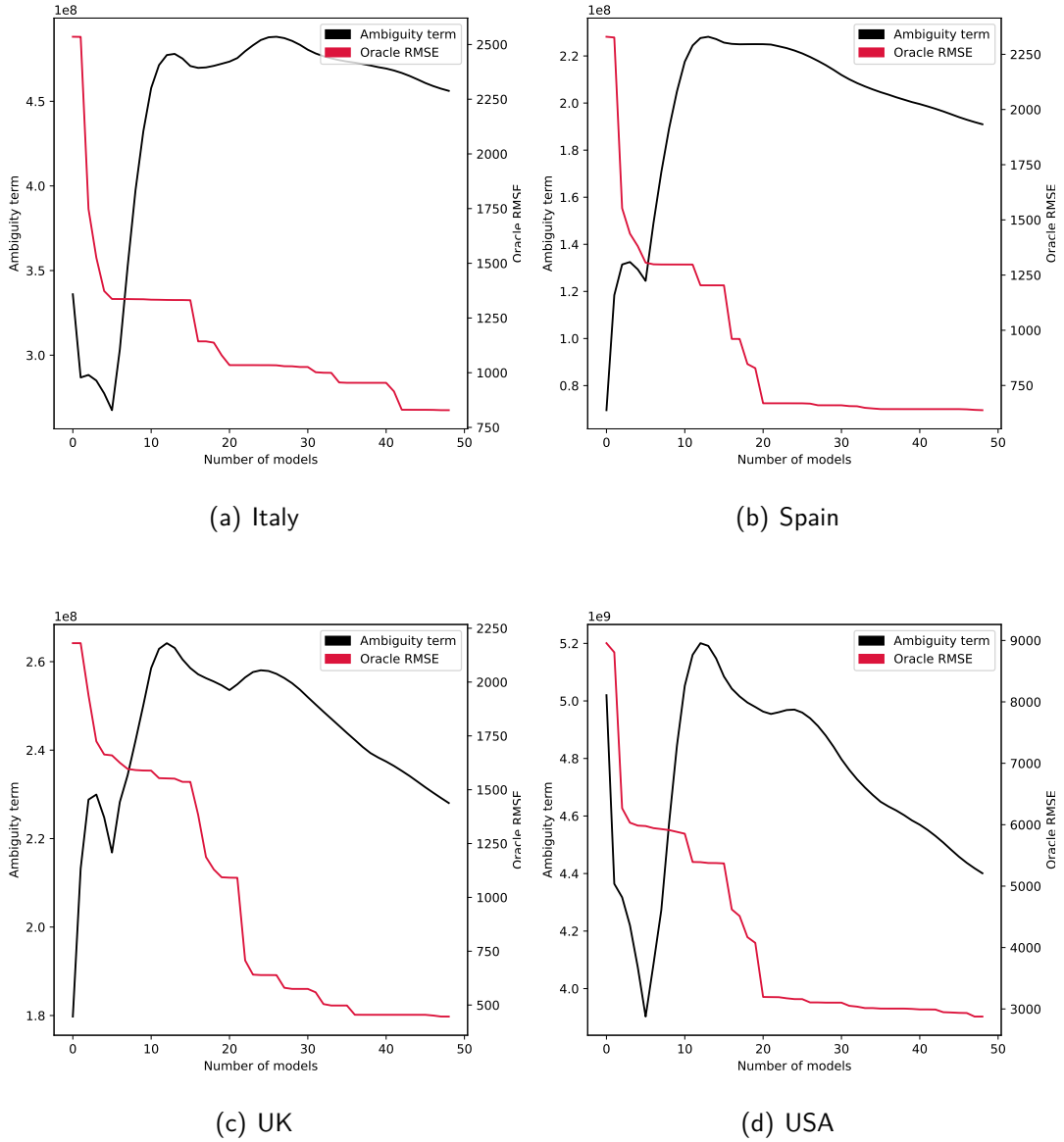


Figure 19 – Ambiguity term and RMSE of the Oracle for pools with 50 models in Italy, Spain, UK, and USA time series.

4.2.2 Results

Tables 5 and 6 present the comparison of DESTC with single models and ensemble-based approaches. Table 5 shows RMSE, MAE, and Theil's U values achieved by DESTC and single models (ARIMA, ETS, SVR, ELM, MLP, LSTM, and TSF) and XGB for eight time series of COVID-19 incidence. DESTC demonstrated superior performance compared to single models, especially in the time series of Brazil, France, Germany, and US, in which the proposed approach achieved better values in most performance measures.

DESTC demonstrated good accuracy but did not excel across all performance metrics for the time series data from Italy and the UK, ranking among the top-performing models. These results show that the proposed approach was able to model better the different behaviors of COVID-19 incidence than the single models. Indeed, the results show that no literature model attained a consistent performance in all considered data sets.

Table 5 – Performance measures (RMSE, MAE, and Theil's U) for proposed approach (DESTC) and literature models (ARIMA, ETS, SVR, ELM, XGB, LSTM, MLP, and TSF) on test set. Bold values represent the top performance across all models by time series, while underlined values denote the second best performance.

Countries	Metrics	ARIMA	ETS	SVR	ELM	LSTM	MLP	TSF	DESTC
Brazil	RMSE	2187.5	2104.4	1877.4	2070.5	2169.8	2056.3	1956.2	**1806.4**
	MAE	1124.6	1122.8	1061.3	1056.5	1140.9	1080.0	**1021.3**	1024.0
	Theil's U	1.081	1.000	0.750	0.969	1.062	0.955	0.875	**0.736**
Canada	RMSE	398.90	372.31	428.21	606.91	432.41	485.50	**356.20**	377.28
	MAE	220.83	**201.98**	247.18	356.82	258.45	355.02	208.17	204.35
	Theil's U	1.150	1.003	1.326	2.666	1.347	1.704	**0.9517**	1.020
France	RMSE	3898.1	3879.9	4122.8	4070.4	4987.9	14250.8	3702.4	**3655.6**
	MAE	2395.3	2461.5	2518.4	2446.6	3242.5	10175.3	2308.1	**2222.9**
	Theil's U	0.743	0.736	0.8313	0.810	1.215	9.746	0.658	**0.653**
Germany	RMSE	2814.4	2736.7	2982.9	2851.9	6802.8	17328.5	3223.5	**2426.6**
	MAE	1975.2	**1515.0**	1693.2	1721.0	5204.8	13405.2	1913.6	1554.7
	Theil's U	0.864	0.797	0.929	0.891	4.990	32.598	1.088	**0.627**
Italy	RMSE	1725.3	1687.0	**1581.1**	2202.9	3948.9	5546.9	1764.5	1621.5
	MAE	1060.1	**935.3**	1018.8	1563.6	3213.5	3678.1	1205.8	1071.3
	Theil's U	0.528	0.511	**0.446**	0.863	2.739	5.444	0.477	0.459
Spain	RMSE	1984.0	1498.7	1371.6	1889.4	**1264.1**	1608.1	1287.4	1449.5
	MAE	1154.4	775.4	718.4	1040.6	696.7	989.7	**664.1**	811.9
	Theil's U	2.426	1.384	1.1593	2.199	**0.985**	1.593	1.065	1.295
UK	RMSE	451.7	419.9	362.0	**320.5**	2629.5	3069.0	479.1	369.8
	MAE	265.1	243.7	**214.0**	228.0	2504.6	2340.8	306.9	227.0
	Theil's U	0.107	0.093	0.070	**0.055**	3.557	4.871	0.123	0.073
US	RMSE	6456.6	5934.1	6254.2	6043.8	5547.8	5908.5	5448.9	**5012.9**
	MAE	2781.0	2554.1	2724.9	2671.8	2422.9	4283.7	2444.9	**2414.3**
	Theil's U	1.451	1.225	1.361	1.271	1.071	1.214	1.069	**0.873**

Table 6 shows the performance measures of DESTC, static ensembles (eSA, eSM, eSVR, eELM, eMLP), and dynamic ensembles (DSLA, DESLA$_m$, and DESLA$_a$). Compared with ensemble-based approaches, DESTC achieved the best overall performance. For the Canada series, eSA attained the best RMSE, and Theil'U values, while eSM obtained the best MAE value. In general terms, the DESLA$_m$ and DESLA$_a$ were competitive alternatives. These findings corroborate the hypothesis that the forecasting approaches based on dynamic selection

Table 6 – Performance measures (RMSE, MAE, and Theil's U) for proposed approach (DESTC), static ensemble models (eSA, eSM, eSVR, eELM, eSVR, eMLP), and dynamic selection ensembles (DSLA, DESLA$_m$, DESLA$_a$) on test set. Bold values represent the top performance across all models by time series, while underlined values denote the second best performance.

Countries	Metrics	eSA	eSM	eSVR	eELM	eMLP	DSLA	DESLAm	DESLAa	DESTC
Brazil	RMSE	2138.0	2166.4	14047.7	5592.4	67805.2	2096.1	2065.3	1882.8	**1806.4**
	MAE	1399.5	1206.9	12420.8	4167.7	40961.9	1077.9	1058.1	1047.2	**1024.0**
	Theil	1.033	1.060	44.584	7.048	1038.255	0.993	0.964	0.801	**0.736**
Canada	RMSE	**369.76**	371.62	846.76	1263.96	13526.40	397.40	371.47	374.17	377.28
	MAE	205.51	**202.86**	611.93	939.32	7944.37	227.40	210.41	211.30	204.35
	Theil	**0.989**	0.998	5.188	11.504	1288.694	1.141	0.998	1.012	1.020
France	RMSE	13103.3	9947.6	27052.9	47515.3	90595.1	3887.1	3661.4	10710.2	**3655.6**
	MAE	9582.8	7124.3	18039.6	31050.5	67933.8	2529.7	2364.9	7721.5	**2222.9**
	Theil	8.263	4.770	34.801	110.424	380.543	0.739	0.656	5.545	**0.653**
Germany	RMSE	20860.5	20978.7	16403.0	20430.6	52859.9	54661.3	6620.3	16975.8	**2426.6**
	MAE	16754.9	16095.1	13966.0	15708.0	45064.2	44597.7	4551.0	13604.1	**1554.7**
	Theil	47.379	48.087	30.460	46.509	312.918	328.679	4.713	31.447	**0.627**
Italy	RMSE	8162.9	6358.0	29795.3	6899.9	77833.9	15182.6	10658.1	9479.3	**1621.5**
	MAE	5063.2	3911.4	24307.2	5426.4	61505.2	9183.3	6779.7	5979.2	**1071.3**
	Theil	11.842	7.141	155.900	8.540	1086.775	41.028	20.211	15.968	**0.459**
Spain	RMSE	1504.3	1480.5	3978.6	5663.6	64790.5	1629.2	1583.9	1581.3	**1449.5**
	MAE	842.50	817.89	3018.53	3956.17	48282.18	879.67	870.79	888.97	**811.87**
	Theil	1.395	1.351	9.753	19.768	2585.126	1.636	1.546	1.541	**1.295**
UK	RMSE	573.46	631.69	49133.57	9206.22	69567.07	2629.47	667.70	469.63	369.80
	MAE	298.81	288.97	37852.36	6633.53	53323.33	2504.55	329.33	325.68	**226.97**
	Theil	0.129	0.143	1290.801	42.867	2588.047	3.557	0.183	0.114	**0.073**
US	RMSE	5544.7	5894.2	21868.5	6814.7	295974.5	5547.8	5752.0	5672.5	**5012.9**
	MAE	2548.7	2631.8	14275.5	4713.8	199947.4	2422.9	2578.8	2572.1	**2414.3**
	Theil	1.070	1.209	16.555	1.608	3047.480	1.071	1.151	1.120	**0.873**

can be a fruitful alternative to deal with time series that present local regimes with distinct behavior.

Table 7 presents the PD between proposed approach and single and ensemble models according to Equation 2.58. Overall, DESTC stands out as one of the most consistent forecasting methods considering the experiments conducted. Most of the gains achieved by the DESTC had significant percentage difference values. This behavior can be seen in the average and median of the percentage difference. The percentage difference in all analyzed data sets was positive, with average values higher than 10% in 14 out of 15 comparisons.

Figure 20 apresent the boxplots of the ARM of the DESTC against single and ensemble models. For this calculation, the models were ranked according to Equation 2.57 (ARM) considering RMSE, MAE, and Theil's U values. The lower the ARM, the better the model performance is. For example, in the France series, DESTC attained rank 1 for four performance measures, so its ARM is 1. The proposed approach presented better performance with lower

Table 7 – Percentage difference (PD) according to Equation 2.58 between DESTC approach and literature models. Bold values indicate models that were more competitive compared to DESTC. Negative values indicate that the compared model outperformed DESTC.

Model	Brazil	Canada	France	Germany	Italy	Spain	UK	US	Average	Median	SD
ARIMA	17.42	5.42	6.22	13.78	6.02	26.94	18.12	22.36	14.54	15.60	8.12
ETS	14.16	-1.33	5.78	**11.33**	3.88	3.28	11.94	15.52	8.07	8.56	6.00
SVR	**3.78**	11.89	11.33	18.65	**-2.55**	-5.68	-2.15	19.85	6.89	**7.56**	9.92
ELM	12.76	37.84	10.19	14.92	26.39	23.28	**-15.37**	17.06	15.88	15.99	15.44
LSTM	16.75	12.75	26.71	64.33	58.94	**-14.67**	85.94	9.64	32.55	21.73	33.78
MLP	12.15	22.29	74.35	86.00	70.77	9.86	87.95	15.16	47.32	46.53	35.31
TSF	7.66	**-5.92**	1.26	24.72	8.11	-12.60	22.81	**8.00**	**6.76**	7.83	12.82
eSA	15.51	-2.03	72.10	88.37	80.14	3.64	35.51	9.59	37.85	25.51	36.99
eSM	16.62	-1.52	63.25	88.43	74.50	2.09	41.46	14.95	37.47	29.04	34.56
eSVR	87.14	55.44	86.49	85.21	94.56	63.57	99.25	77.08	81.09	85.85	15.00
eELM	67.70	70.15	92.31	88.12	76.50	74.41	95.98	26.44	73.95	75.45	21.85
eMLP	97.34	97.21	95.96	95.41	97.92	97.76	99.47	98.31	97.42	97.55	1.28
DSLA	13.82	5.06	5.96	95.56	89.32	11.03	85.94	9.64	39.54	12.43	42.18
DESLAm	12.54	-1.57	**0.16**	63.35	84.79	8.48	44.62	12.85	28.15	12.69	32.18
DESLAa	4.06	-0.83	65.87	85.71	82.89	8.33	21.26	11.63	34.86	16.44	36.85

median and standard deviation. Among the single models from the literature, TSF reached the best performance. The figure shows a consistent better performance of the proposed approach compared to ensemble-based models. Also, it is possible to note that the dynamic selection methods achieved good results among ensemble-based models. This result shows that the DESTC achieved a more stable performance, being able to adapt to different regimes and patterns present in the COVID-19 incidence time series.

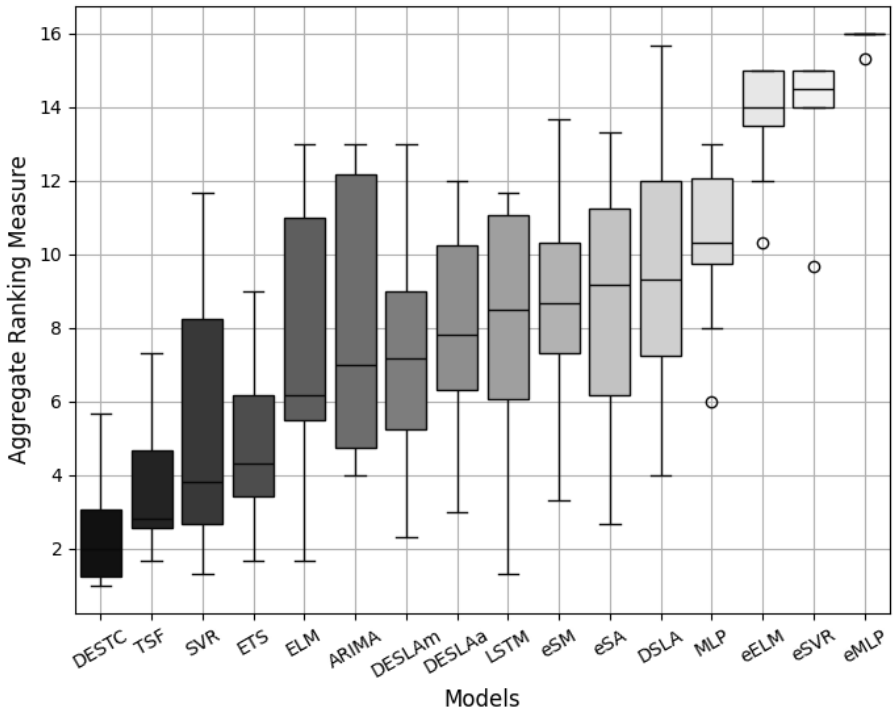


Figure 20 – Boxplot of ARM for single models and ensembles.

Table 8 shows the results of the Diebold-Mariano tests from the comparison of the residuals of the models to verify the statistical difference between the proposed approach and other MPSs. In this case, there is statistical evidence that DESTC achieved superior results compared to other algorithms. Additionally, Table 9 presents paired comparisons using the Bayesian signed rank test, where DESTC also demonstrated better predictive performance than other forecasting alternatives. These findings underscore the superior forecasting performance of DESTC for COVID-19 time series incidence.

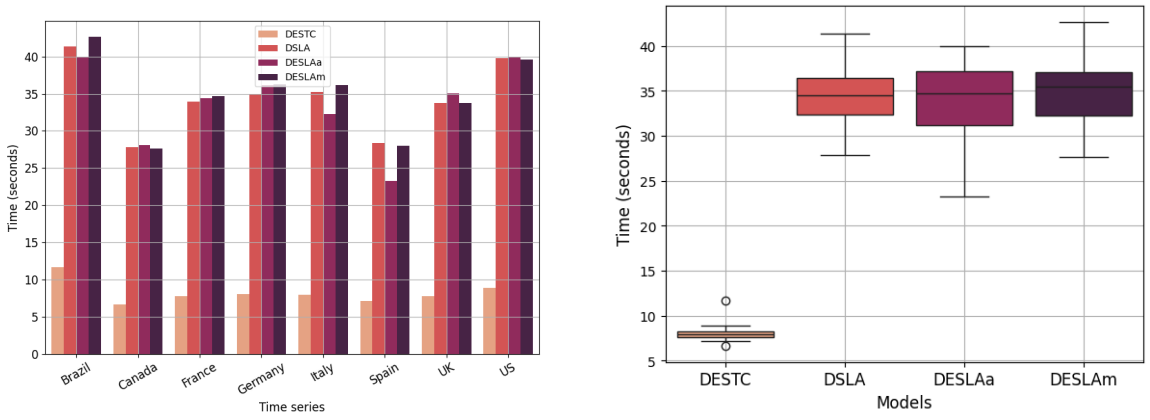
Table 8 – Paired comparison between residuals of the MPSs using Diebold-Mariano test ($\alpha = 5\%$)(rows versus columns). "$+$" in red ("$-$" in blue) indicates that the model listed in the row is better (worse) than the one listed in the column. "$=$" indicates that there is no statistical difference.

Models	DESTC	DESLAa	DESLAm	eSA	eSM	DSLA	eELM	eMLP	eSVR
DESTC	$=$	$+$	$+$	$+$	$+$	$+$	$+$	$+$	$+$
DESLAm	-	$+$	$=$	$+$	$+$	$+$	$+$	$+$	$+$
DESLAa	-	$=$	-	$+$	$+$	$+$	$+$	$+$	$+$
eSA	-	-	-	$=$	-	$+$	$+$	$+$	$+$
eSM	-	-	-	$+$	$=$	$+$	$+$	$+$	$+$
DSLA	-	-	-	-	-	$=$	$=$	$+$	$+$
eELM	-	-	-	-	-	$=$	$=$	$+$	$+$
eSVR	-	-	-	-	-	-	-	$+$	$=$
eMLP	-	-	-	-	-	-	-	$=$	-

Table 9 – Paired comparison between MPSs using the Bayesian signed rank test (rows versus columns). "+" in red ("−" in blue) indicates that the model listed in the row is better (worse) than the one listed in the column. "=" indicates that there is no difference.

ARM	DESTC	DESLAm	DESLAa	eSM	eSA	DSLA	eELM	eSVR	eMLP
DESTC	=	+	+	+	+	+	+	+	+
DESLAm	-	=	=	=	=	+	+	+	+
DESLAa	-	=	=	=	=	=	+	+	+
eSM	-	=	=	=	=	=	+	+	+
eSA	-	=	=	=	=	=	+	+	+
DSLA	-	-	=	=	=	=	+	+	+
eELM	-	-	-	-	-	-	=	=	+
eSVR	-	-	-	-	-	-	=	=	+
eMLP	-	-	-	-	-	-	-	-	=

Finally, Figure 21 shows the computational (processing) time during the test phase, in seconds, for the four implemented dynamic models (DSLA, DESLA_a, DESLA_m, and DESTC). In this context, processing time refers to the duration required for a previously trained model to generate predictions on the test set. Figure 21(a) presents the processing time by time series and Figure 21(b) shows the results by model. The proposed approach demonstrated the shortest computational processing time across all evaluated cases. The difference is significantly larger because the DSLA, DESLA_a, and DESLA_m methods select models based on a RoC constructed using similarity calculations between test instances and all those in the validation set. Meanwhile, DESTC simply calculates the trend of the instance and selects the models based on this information.



(a) Processing time by time series.

(b) Boxplot of processing time by DS models.

Figure 21 – Processing time (in seconds) of test phase by time series.

4.3 DISCUSSION

Forecasting COVID-19 incidence presents a significant challenge. One notable aspect of these series is the presence of trend cycles, which highlight the distinct phases of the pandemic. In this context, an innovative method, DESTC, was proposed to address this dynamic.

From the conducted experiments, it can be observed that DESTC emerges as a competitive forecasting alternative when dealing with COVID-19 incidence time series. Performance measures demonstrate that for the modeled series, DESTC ranked among the top approaches in nearly all cases. Unlike single models, the proposal can adapt to the trend changes observed in the COVID-19 incidence time series, choosing specific models for each trend class, which justifies the superior performance for the presented cases.

Figures 22 and 23 present a detailed analysis of the trend in the time series of incidence in France and Germany, considering the validation and test sets. In the figures, colored vertical bars represent the Sen's Slope estimator (right y-axis), calculated based on the last 10 observations of each instance. Additionally, the trend for each instance was estimated using the Mann-Kendall test. The observations of the time series are also showed (left y-axis). Blue, red, and green bars indicate increasing, decreasing, and no trend, respectively. In both cases, for which the DESTC achieved good predictive performance (see Tables 5 and 6), it can be observed that the trend detection method was able to track the changes in the time series.

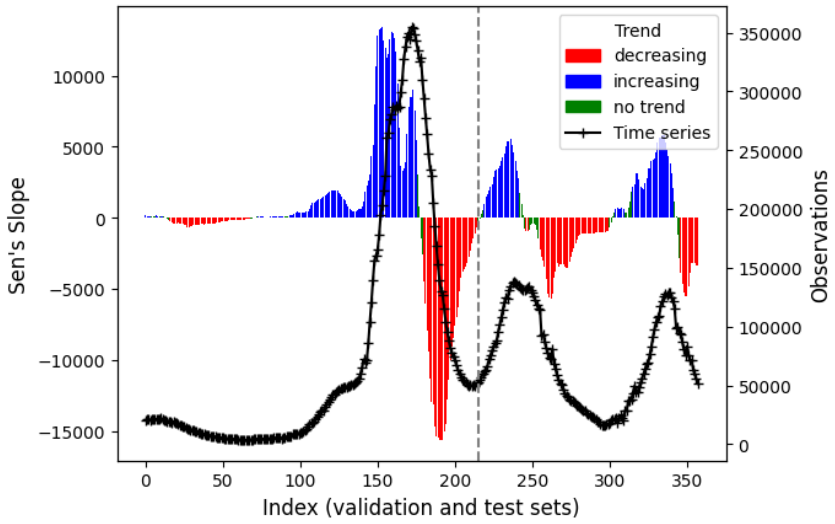


Figure 22 – Trend analysis based on Sen's Slope estimator and Mann-Kendall statistical test to France time series.

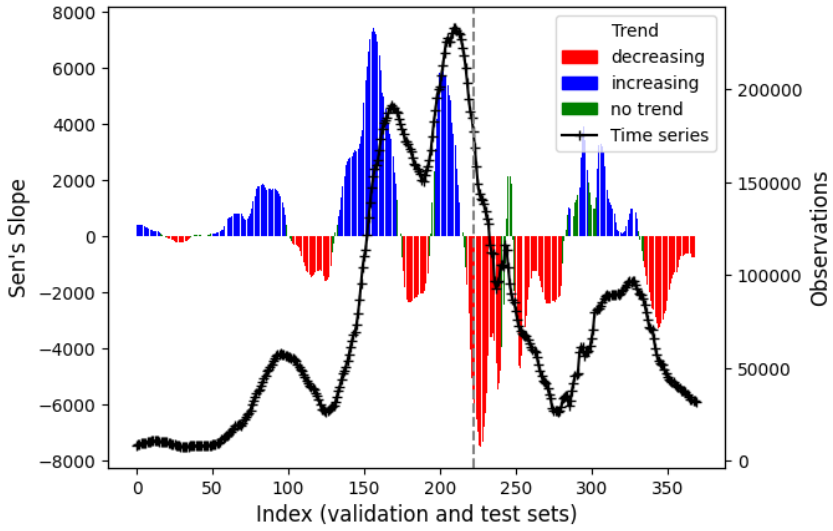


Figure 23 – Trend analysis based on Sen's Slope estimator and Mann-Kendall statistical test to Germany time series.

On the other hand, Figures 24 and 25 show the time series related to the cases in Spain and the Canada, two of the most challenging cases for the DESTC (see Tables 5 and 6). In the case of Spain (Figure 24), it is apparent that, despite the alternation between trend classes, a pronounced discrepancy exists in the pattern observed between the validation set (prior to the gray vertical line) and the test set (after gray vertical line). The case of Canada is also important, as despite an apparent negative trend in the test set (from index observations 250 to 300), the Mann-Kendall test was not able to classify correctly (the bars should be red, but they are green). This case is similar to the one presented in the Section 2.1.1.1, Figure 4(a), where the Mann-Kendall test was unable to correctly classify the trend of a time series with noise, once H0 is not rejected even with an apparent trend. In these cases, it appears that the DESTC was not able to appropriately select the best models.

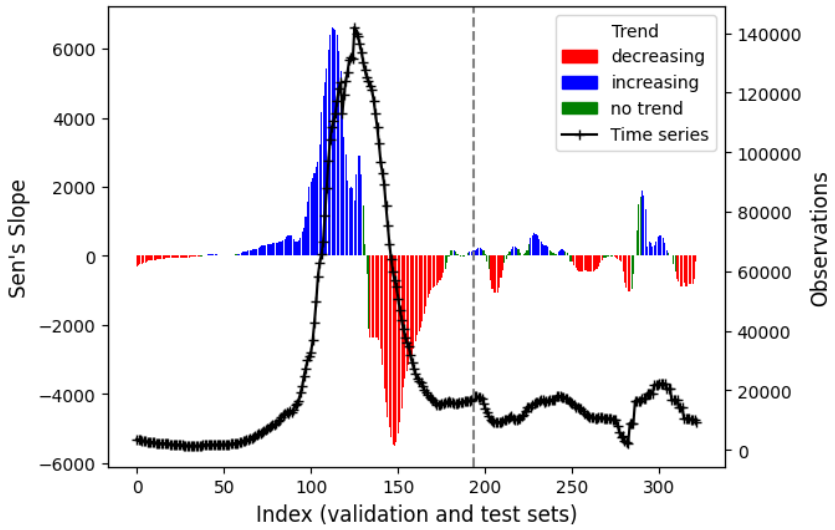


Figure 24 – Trend analysis based on Sen's Slope estimator and Mann-Kendall statistical test to Spain time series.

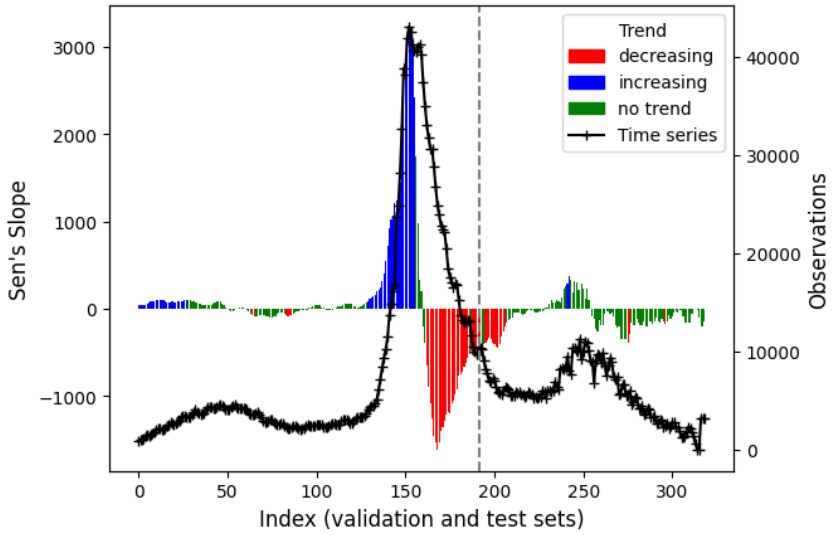


Figure 25 – Trend analysis based on Sen's Slope estimator and Mann-Kendall statistical test to Canada time series.

Besides, in order to enhance comprehension and facilitate a detailed analysis of the DESTC functioning, Table 10 shows the number of the validation (Z_v) and test (Z_t) samples classified by DESTC into decreasing, increasing, and no trends for two specific cases (Italy and Canada). These cases were chosen because they presented different behaviors in the data sets. For each country, the average rank of the models selected by DESTC for each concept was calculated. This rank corresponds to the RMSE value of all pool models in the Z_t set. So, the rank varies in the range [1, 50], where the lower the rank, the better the performance of the model regarding the pool. In Italy, the proposed approach was notably very effective once it

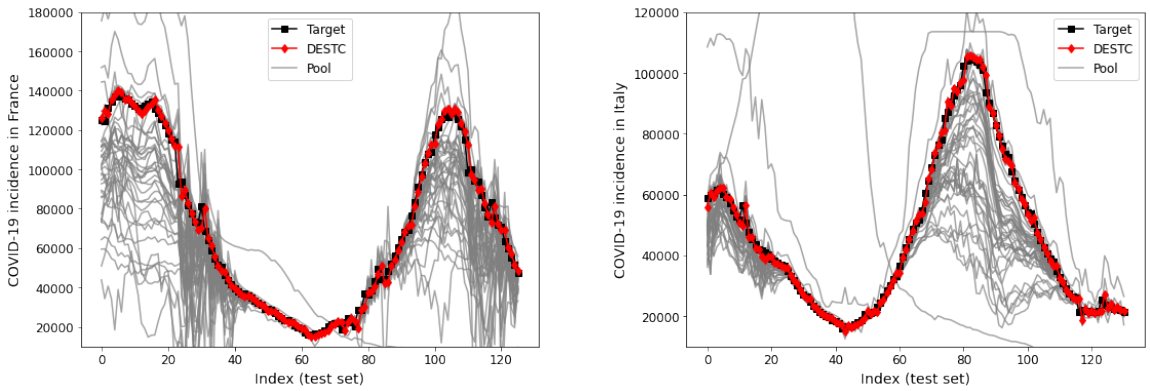
successfully identified, during the training phase, the most promising models for combination in the testing phase. The average rank, jointly with performance values (Tables 5 and 6), shows that DESTC selected accurate models for the three concepts. On the other hand, DESTC did not select the best models for the Canada series. Only for decreasing concept, DESTC obtained a low average rank value. For increasing and no trend concepts, DESTC obtaining a high average rank of 33.7 and 36.8, respectively.

Table 10 – Average rank of the selected models in test set (Z_t) by trend class to Italy and Canada time series. Length column shows the number of instances classified by trend class in validation set (Z_v) and Z_t.

Country	Trend	Length		Average rank
		Z_v	Z_t	
Italy	Decreasing	99	72	13.7
	Increasing	108	39	8.0
	None	17	20	9.1
Canada	Decreasing	82	64	9.6
	Increasing	89	22	33.7
	None	23	26	36.8

Figures 26(a) and 26(b) show the DESCT (red lines) in the test sets of France and Italy, respectively, in comparison to the observed values (black lines) and the generated pools (gray lines). In both scenarios, the generated pool exhibited diversity and DESTC effectively selecting the models for combination. Regarding Italian case (Figure 26(b)), the generated pool seems to underestimate the observed values (most gray lines are below the black one). This pattern suggests that the majority of individual models did not adequately capture the peak of the pandemic. Furthermore, it is also associated with the low accuracy of the combined models utilized in this study (Table 6), once they tend to underestimate the forecasts. Hence, despite DESTC being able to make accurate predictions, enhancing the diversity of pools could potentially amplify its accuracy.

Additionally, it is important to address the underperformance of the dynamic models employed in the experiment (DSLA, DESLAa, and DESLAm). These models depend on a similarity measure to construct RoCs (SILVA; DE MATTOS NETO; CAVALCANTI, 2021). However, the significant differences between the validation and test sets may explain why these alternatives performed poorly. The chosen similarity measure might not have been able to accurately identify the most similar instances, thereby compromising the overall system performance. Thus, the proposed method represents a valuable alternative to the problem under consideration



(a) DESCT in France.

(b) DESCT in Italy.

Figure 26 – Comparison of DESTC model (red lines) performance in the test sets of France (a) and Italy (b), alongside the observed values (black lines) and the generated pools (gray lines).

since it selects the best models not relying on a similarity measure but rather based on a trend classification.

In this study, ETS was one of the most competitive models. The performance of this approach in predicting time series related to COVID-19 has been discussed in literature (PETROPOULOS; MAKRIDAKIS; STYLIANOU, 2022; LARABI-MARIE-SAINTE et al., 2022). With respect to ensemble methods, eSA and eSM showed the best results, while those based on ML (eSVR, eELM, and eMLP) had the worst.

In terms of processing time, DESTC exhibited the lowest result for all modeled time series when compared to the other dynamic models. This occurs because constructing regions of competence based on similarity measures (as done in DSLA, DESLA_a, and DESLA_m) is computationally more costly than selecting models based on their trend classes (as done in the case of DESTC).

Thus, the proposed forecasting approach, DESTC, was able to improve the forecast of COVID-19 incidence when compared to alternatives from the literature. It should be noted that this is one of the first studies that address the adoption of DS and DES techniques to COVID-19 time series forecast.

Regarding limitations, two points warrant discussion. Firstly, as illustrated by the case of Canada, DESTC did not consistently identify the most appropriate models for combination. In this instance, the time series structure changed significantly enough that the optimal models for each trend class also shifted. Thus, relying solely on trend classes for selection proved ineffective in ensuring model accuracy. Secondly, the parameters used in DESTC (Table 4) are selected through a validation step; however, the validation patterns can be different from those

of testing. In this case, exploring alternatives to improve model selection phase is necessary.

5 EXPERIMENT B - DIVERSE TIMES SERIES FORECASTING

This chapter aims to address the second main objective of this thesis: (ii) to assess the applicability of this new approach in time series from other domains. DESTC dynamically selects the best models for different trend classes and optimally combines them, ensuring more accurate forecasts. In the context, DESTC will be evaluated using 10 time series from diverse phenomena (social, economic, and environmental).

5.1 EXPERIMENTAL PROTOCOL

Ten time series representing diverse phenomena and with distinct statistical patterns were utilized in this experiment (Table 11). The first three refer to monthly wildfire spots in the Brazilian states of Alagoas (WDF-AL, Figure 27), Bahia (WDF-BA, Figure 28), and Ceará (WDF-CE, Figure 29). These time series were obtained from the Wildfire Monitoring Program of the Instituto Nacional de Pesquisas Espaciais (INPE) (INPE, 2024). Additionally, two other environmental time series were used: the Canadian lynx series (LYNX, Figure 30), a well-known historical dataset recording the annual number of lynx collected by the Hudson's Bay Company in Canada (CAMPBELL; WALKER, 1977), and the sunspot time series (SUNS, Figure 31), which tracks the number of sunspots observed on the surface of the sun over time (SILSO, 2024). Following, two economic time series were used, referring to the weekly opening prices of ITSA4 (Figure 32) and NASDAQ (Figure 33) stocks. Both economic series were obtained from Yahoo Finance (Yahoo Finance, 2024). Lastly, time series of dengue incidence in Ceará (DEN-CE, 34) and Iquitos (DEN-IQ, 35) (BENEDUM et al., 2020), and mumps in New York were used (MUC, 36) (DETHLEFSEN; LUNDBYE-CHRISTENSEN, 2006).

The WDF-AL, WDF-BA, and WDF-CE time series exhibit seasonal patterns, though the first series is notably noisier and with lower maximum and minimum values. In turn, LYNX and SUNS time series are characterized as cyclic time series due to the unclear presence of seasonality. Both the ITSA4 and NASDAQ time series show no distinct trend or seasonality behavior, resembling more of a random walk. DEN-CE and DEN-IQ present substantial variability in their values, lacking any discernible trend or seasonality. Lastly, the MUC time series displays considerable variability, showing alternating cycles of trend.

Table 11 – Description of the time series used for modeling and forecasting. Stationarity was evaluated using the Dickey-Fuller hypothesis test assuming a significance level of 5%.

Time series	Length	Min	Max	Mean	Median	STD	Time unit	Stationarity
WDF-AL	288	1	153.0	21.5	12.5	24.9	Monthly	Stationary
WDF-BA	288	4	10076.0	1036.6	303.5	1776.8	Monthly	Stationary
WDF-CE	288	1	4816.0	437.6	56.5	806.9	Monthly	Non-stationary
LYNX	114	39	6991.0	1538.0	771.0	1585.8	Yearly	Stationary
SUNS	315	0	190.2	49.7	40.0	40.2	Yearly	Stationary
ITSA4	229	0	11.6	8.6	8.5	1.1	Weekly	Non-stationary
NASDAQ	264	6847.2	16601.0	12124.8	12266.4	2514.2	Weekly	Non-stationary
DEN-CE	216	66	29665.0	4517.3	2525.0	5413.9	Monthly	Non-stationary
DEN-IQ	598	0	116.0	8.5	5.0	11.9	Weekly	Stationary
MUC	534	20	1956.0	487.7	355.0	384.3	Monthly	Stationary

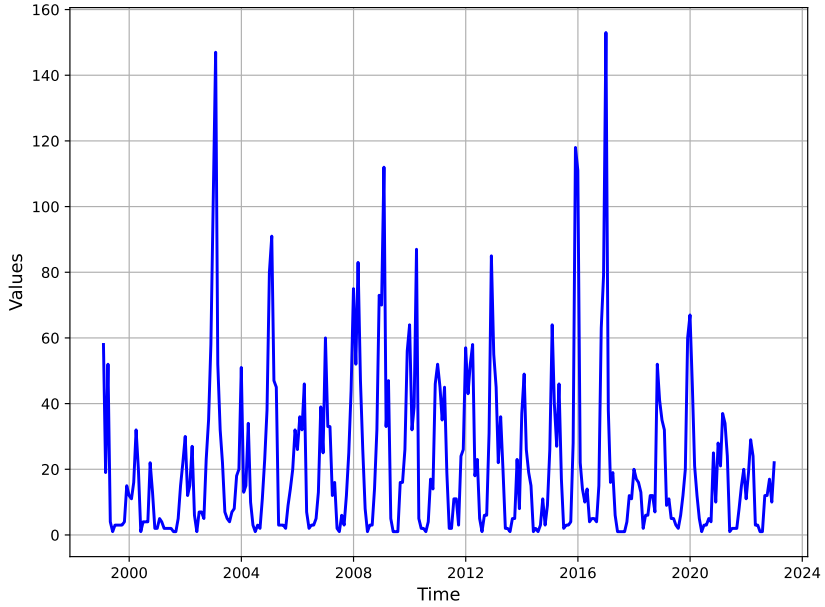


Figure 27 – WDF-AL time series.

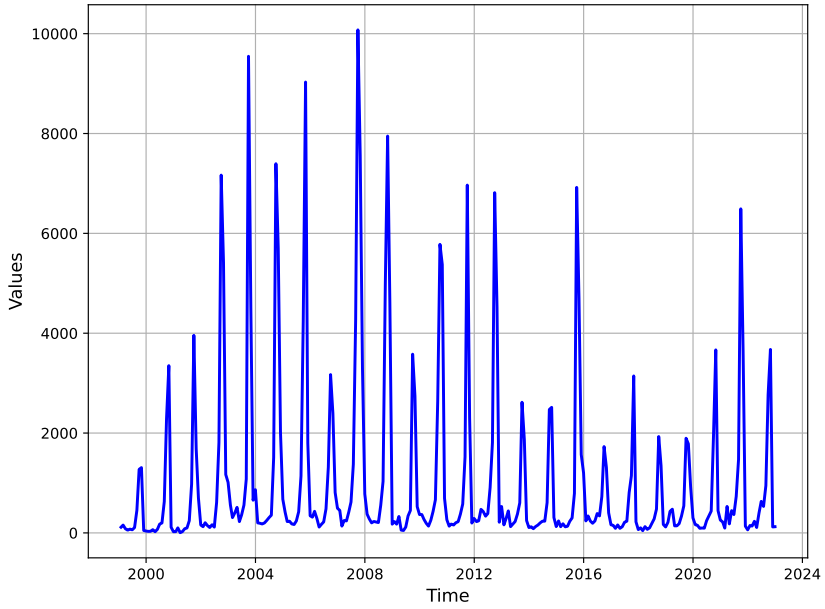


Figure 28 – WDF-BA time series.

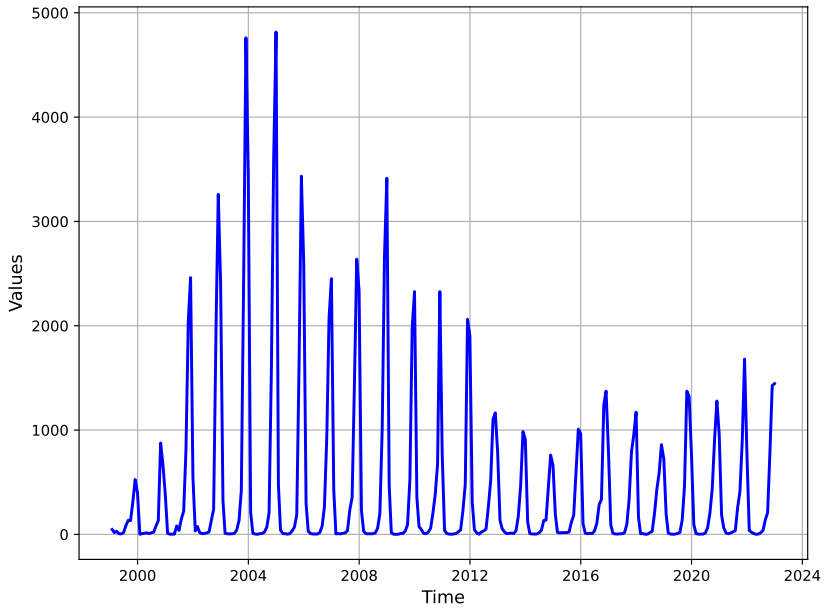


Figure 29 – WDF-CE time series.

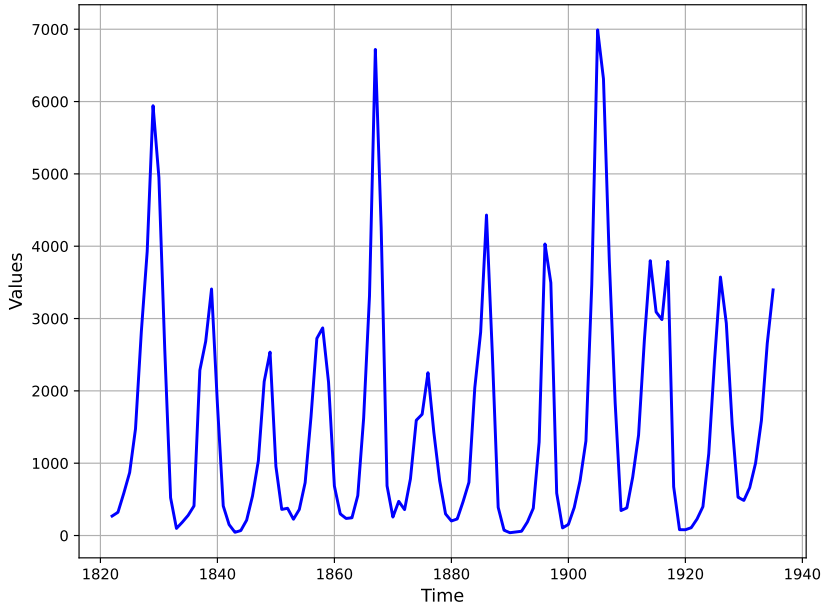


Figure 30 – LYNX time series.

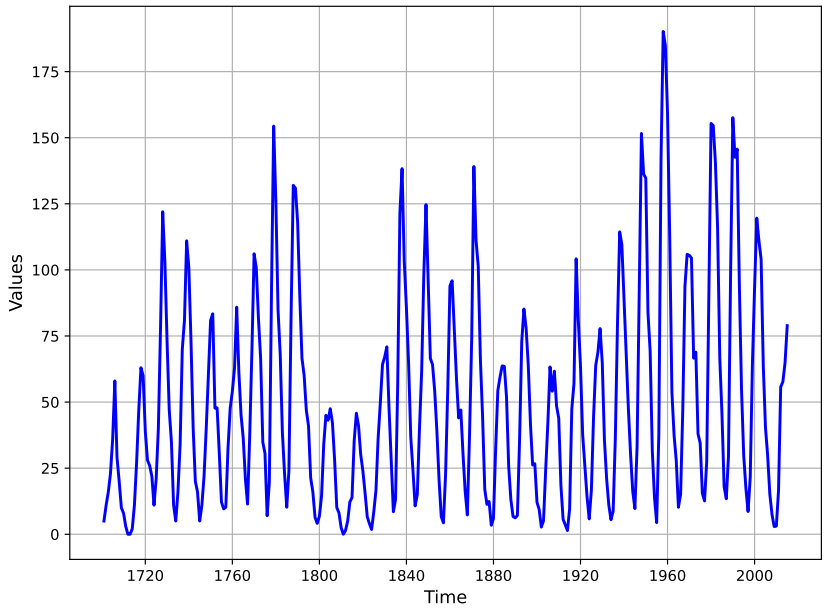


Figure 31 – SUNS time series.

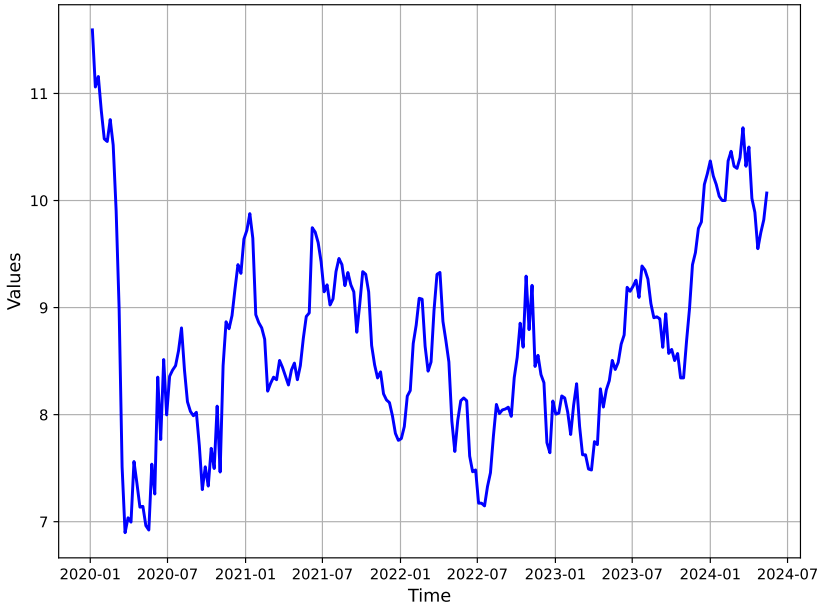


Figure 32 – ITSA4 time series.

Figure 33 – NASDAQ time series.

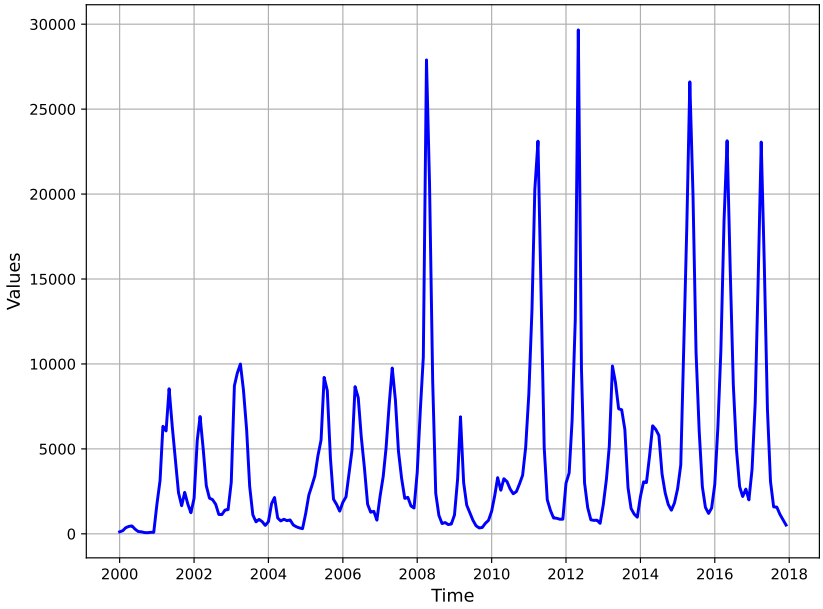


Figure 34 – DEN-CE time series.

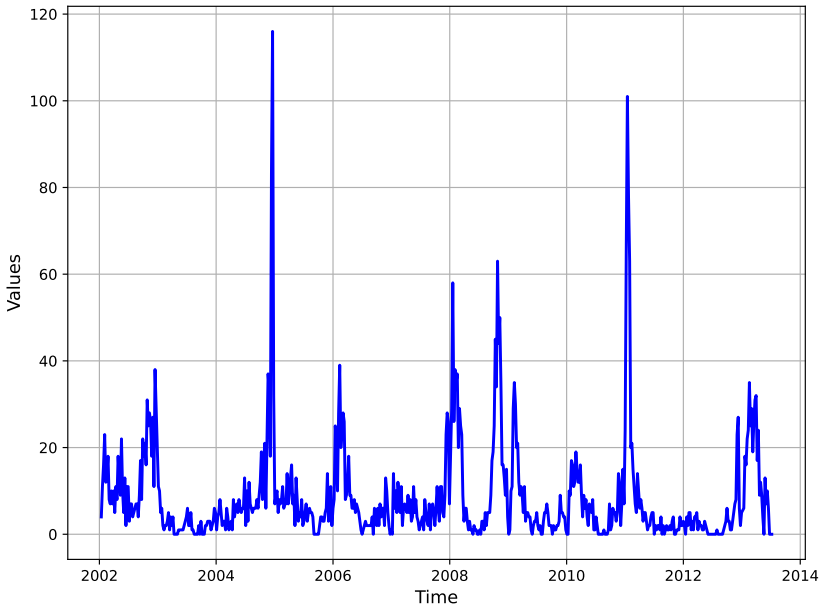


Figure 35 – DEN-IQ time series.

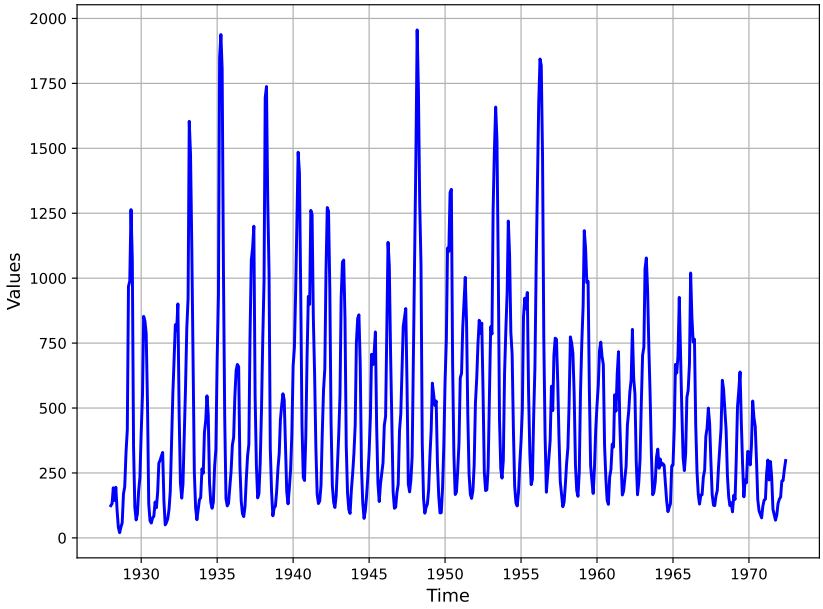


Figure 36 – MUC time series.

In the preprocessing stage, the time series were initially split in temporal order into training (60%), validation (25%), and test (15%) sets. Subsequently, they were normalized to the interval [0.2, 0.8] using the maximum and minimum values from the training and validation sets. In the modeling stage, a mixed pool of 70 forecasting models was created to generate the ensembles. The number of models selected was based on preliminary tests using the validation sets. The heterogeneous pool consists of six distinct single models: two classical time series models (ARIMA and ETS) and four other machine learning models, ELM, LSTM, MLP, and SVR. Further, for increasing the diversity of the P pool, ELM models were created using different hyperparameter combinations (Table 12). ELM was chosen due to its computational cost advantages.

Regarding DESTC, the trend classification task was performed using the Mann-Kendall statistical test (MANN, 1945) (Section 2.1.1.1). Other parameters such as the number of models for each trend class (ie., $n_{neutral}$, $n_{positive}$, and $n_{negative}$), combination method and lag size are presented in Table 12. These parameters were selected through a grid search strategy. Classical models (ARIMA and ETS (HYNDMAN; ATHANASOPOULOS, 2018)) are trained without a validation step. In these cases, training and validation sets were joined for parametric estimation. The parameters of ML models (SVR, ELM, MLP, and LSTM) were chosen by grid search (Table 12). The parameters used to train SVR, MLP, and LSTM models are based on (OLIVEIRA; SILVA; DE MATTOS NETO, 2021; ADHIKARI; VERMA; KHANDELWAL, 2015). Further,

in order to compare DESTC with more recent techniques from literature, TSF models were also trained (HEATON, 2022).

Seven ensemble-based approaches were developed using the same P pool of the DESTC. Inspired by the literature (SILVA; DE MATTOS NETO; CAVALCANTI, 2021; MAALIW et al., 2021; SANTOS JUNIOR et al., 2022), these ensembles were created to perform a more comprehensive comparison. Two non-trainable ensembles with average (eSA) and median (eSM), three trainable ensembles using ELM (eELM), MLP (eMLP) (FERNANDES; EBECKEN; ESQUERDO, 2017), and SVR (eSVR) (DE MATTOS NETO et al., 2021). Regarding DS approaches, two algorithms were used: the first selects only one model for forecasting (DSLA), while the second, DESLA, selects multiple models and combines them through average, median or mode. This modeling approach differs from Experiment A (Section 4), in which two methods, DESLA$_a$ and DESLA$_m$, combined results based on average and median, respectively. Here, the combination method is a parameter to be optimized in the grid search, with the possibilities being: average, median, and mode. The mode combination method was based by Kourentzes, Barrow and Crone (2014) and this strategy was presented in Sales et al. (2023). Additionally, a baseline model was constructed by combining the top J models (regarding RMSE) without employing trend class-based selection. This will help understand whether trend classification contributes to the final performance or not. All parametric selections were based on grid search (Table 12).

RMSE (Equations 2.54), MAE (Equation 2.55), and Theil's U (Equation 2.56) were used to assess the quality of the model forecasts. Just like in Experiment A (Chapter 4), the Bayesian signed rank test was chosen to comparatively evaluate the performance of the MPSs in this experiment. The Python package Autorank was used (HERBOLD, 2020).

Table 12 – Grid search parameters used by model.

Models	Parameters	Values
ARIMA, SARIMA	p, d, q, P, D, Q, Z	(HYNDMAN; ATHANASOPOULOS, 2018)
ETS	E, T, S	(HYNDMAN; ATHANASOPOULOS, 2018)
SVR, eSVR	Lag	5, 7, 10
	Kernel	linear, radial
	Gamma	0.01, 0.1, 1, 10, 100, 1000
	Cost	0.01, 0.1, 1, 10, 100, 1000
	Epsilon	0.0001, 0.001, 0.01
	Tolerance	0.001
ELM, eELM	Lag	5, 7, 10
	Activation function	relu, tansig, rbf
	Units in hidden layer	10, 15, 20, 25, 50, 100, 150, 200
MLP, eMLP	Lag	5, 7, 10
	Activation function	sigmoid
	Units in hidden layer	10, 15, 20
	Algorithm	adam
LSTM	Lag	5, 7, 10
	Activation function	relu
	Units in hidden layer	10, 100, 500
	Algorithm	adam
TSF	Lag	5, 7, 10
	Head size	5, 10, 20
	Number of attention heads	1, 2, 4
	Units in hidden layers	10, 25, 50, 100
DSLA	k	10
	n	1
	Distance	euclidian
DESLA	k	10, 20
	n	3, 5, 10
	Combination method	median, average, mode
	Distance	euclidian
DESTC	$n_{neutral}$	3, 5, 10
	$n_{positive}$	3, 5, 10
	$n_{negative}$	3, 5, 10
	Combination method	median, average, mode

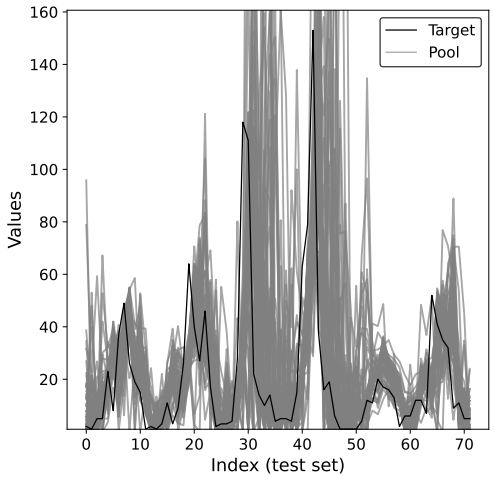
5.2 RESULTS

Below, the results regarding the generation of the pool for each of the analyzed time series and the predictive performance of the proposed models will be presented.

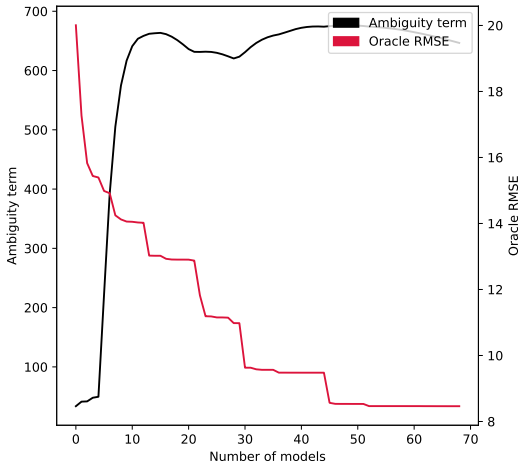
5.2.1 Pool generation

Figures 37 to 45 present the pool generated for each time series as well as the ambiguity term and the RMSE of the oracle. In contrast to the previous experiment (Experiment A - Chaper 4), which focuses solely on COVID-19 incidence series, it is observed that the ambiguity term and the oracle RMSE tend to reach a plateau after the inclusion of 30 models, particularly noticeable for the final two time series analyzed: DEN-CE and DEN-IQ. This observation suggests that, for these specific time series, a greater number of models than in Experiment A

(Chapter 4) may be necessary for the dynamic selection system to achieve improved predictive performance.

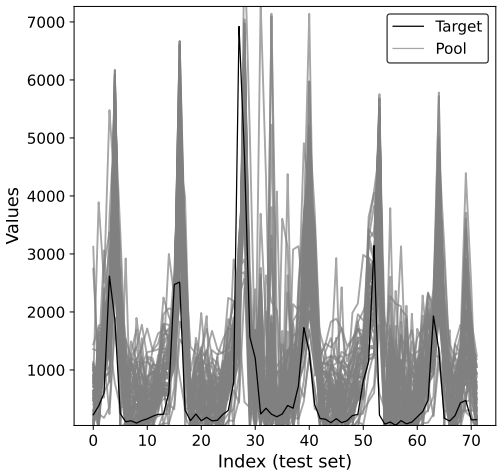


(a) WDF-AL pool.

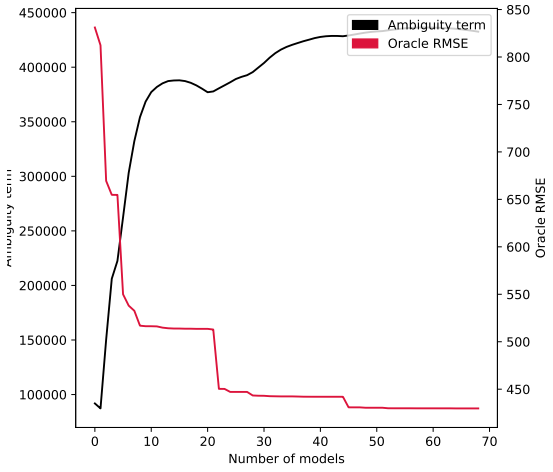


(b) Ambiguity term and RMSE of the oracle.

Figure 37 – Pool generated for WDF-AL time series as well as the ambiguity term and the RMSE of the oracle (validation set).

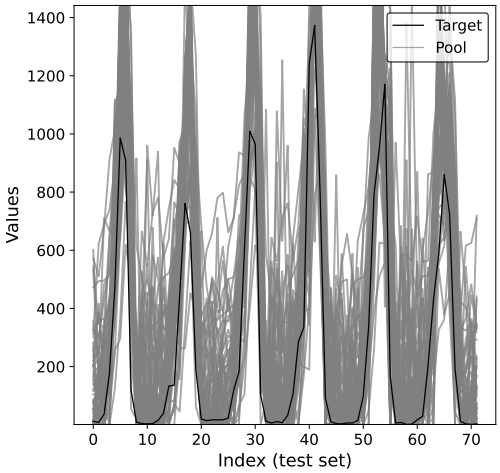


(a) WDF-BA pool.

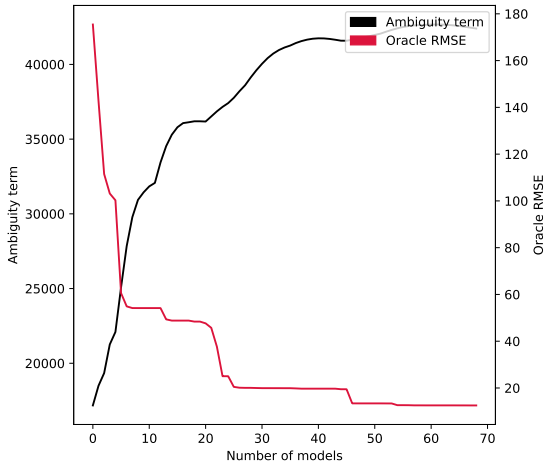


(b) Ambiguity term and RMSE of the oracle.

Figure 38 – Pool generated for WDF-BA time series as well as the ambiguity term and the RMSE of the oracle (validation set).

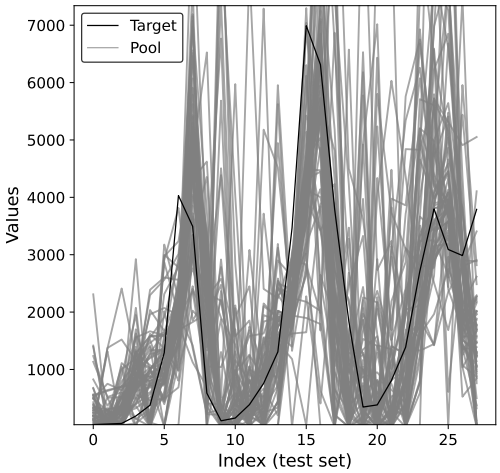


(a) WDF-CE pool.

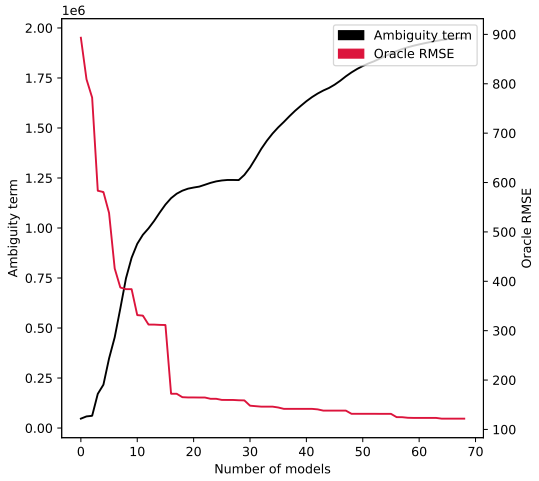


(b) Ambiguity term and RMSE of the oracle.

Figure 39 – Pool generated for WDF-CE time series as well as the ambiguity term and the RMSE of the oracle (validation set).

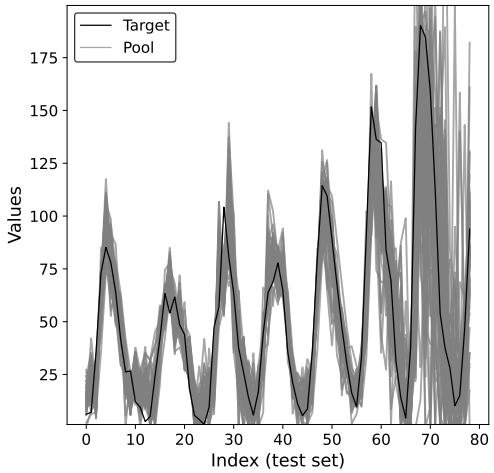


(a) LYNX pool.

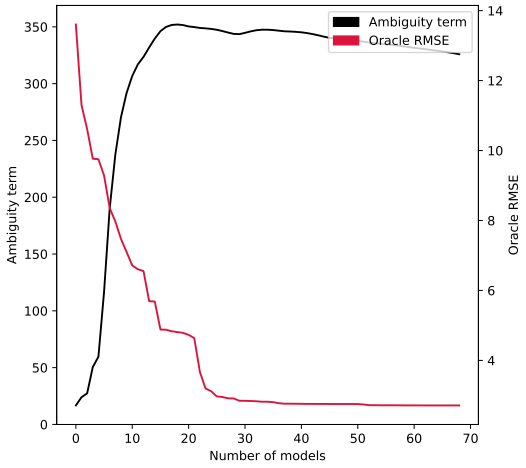


(b) Ambiguity term and RMSE of the oracle.

Figure 40 – Pool generated for LYNX time series as well as the ambiguity term and the RMSE of the oracle (validation set).

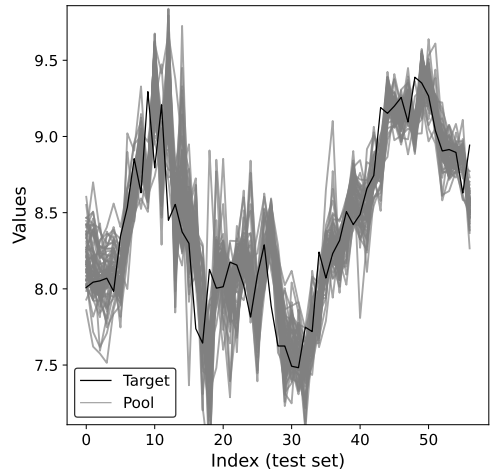


(a) SUNS pool.

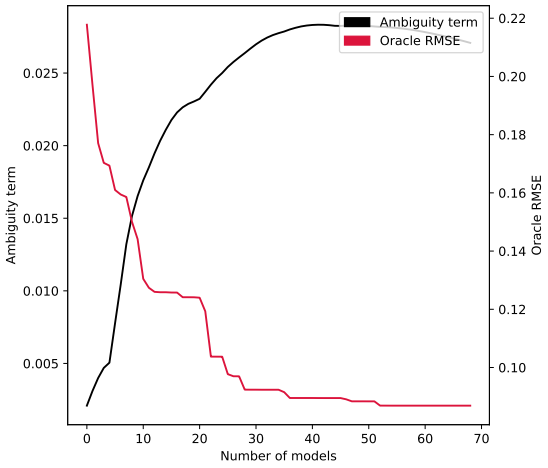


(b) Ambiguity term and RMSE of the oracle.

Figure 41 – Pool generated for SUNS time series as well as the ambiguity term and the RMSE of the oracle (validation set).

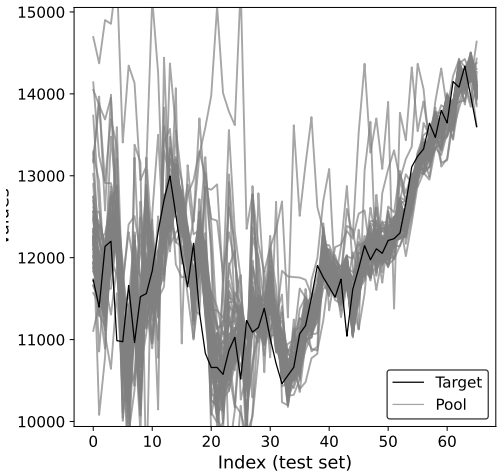


(a) ITSA4 pool.

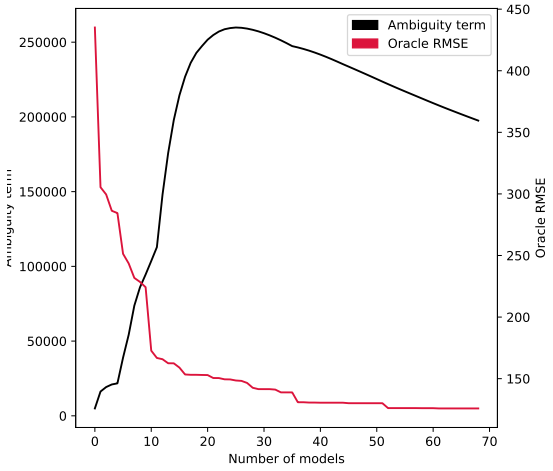


(b) Ambiguity term and RMSE of the oracle.

Figure 42 – Pool generated for ITSA4 time series as well as the ambiguity term and the RMSE of the oracle (validation set).

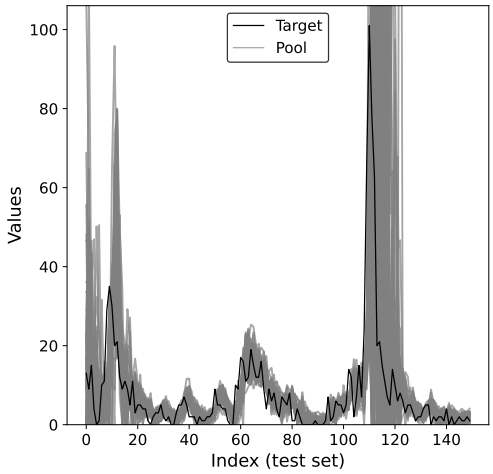


(a) NASDAQ pool.

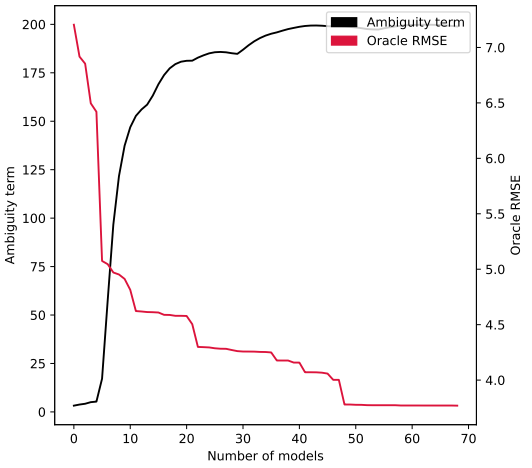


(b) Ambiguity term and RMSE of the oracle.

Figure 43 – Pool generated for NASDAQ time series as well as the ambiguity term and the RMSE of the oracle (validation set).

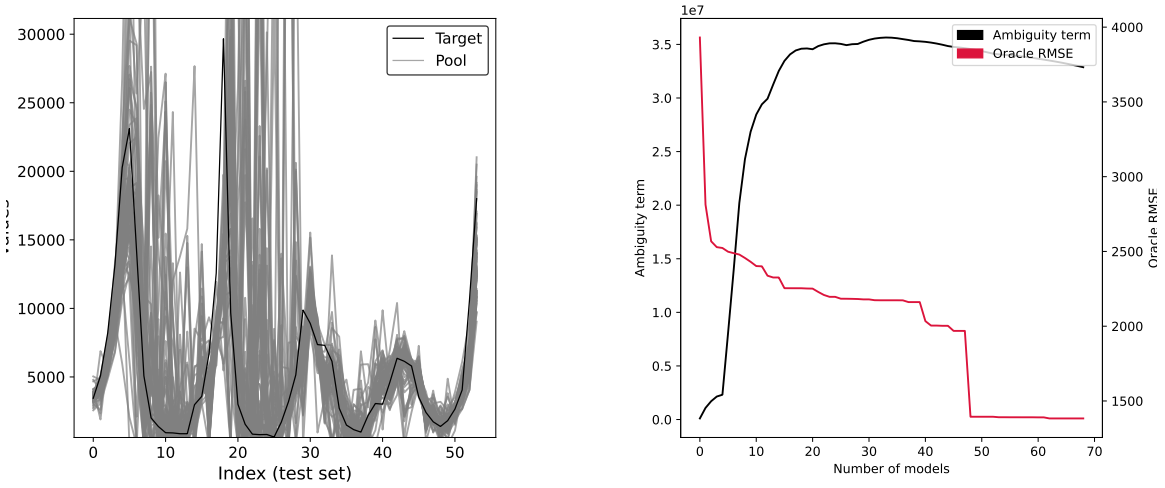


(a) DEN-IQ pool.



(b) Ambiguity term and RMSE of the oracle.

Figure 44 – Pool generated for DEN-IQ time series as well as the ambiguity term and the RMSE of the oracle (validation set).



(a) DEN-CE pool.

(b) Ambiguity term and RMSE of the oracle.

Figure 45 – Pool generated for DEN-CE time series as well as the ambiguity term and the RMSE of the oracle (validation set).

5.2.2 Results

Tables 13 and 14 present the comparison of DESTC with single models and ensemble-based approaches in terms of RMSE, MAE, and Theil's U. Additionally, both Tables also present the rank of DESTC for each of the evaluated cases. Regarding single models (Table 13), DESTC achieved a TOP 3 rank in 6 out of 10 cases (WDF-AL, WDF-BA, WDF-CE, LYNX, SUNS and MUC) but was the best (rank 1 to all PM) in only two cases: WDF-BA and WDF-CE.

Table 14 shows the performance measures of DESTC, static ensembles (eSA, eSM, eSVR, eELM, eMLP, BL), and dynamic ensembles (DSLA and DESLA). In this case, DESTC was in the TOP 3 best combination alternatives in 6 out of 10 cases. Further, whether compared with single or ensemble models, DESTC demonstrated good average performance, showing strong predictive capabilities. The series with the poorest metrics were the financial ones (ITSA4 and NASDAQ), for which no predictive model achieved a Theil's U lower than 0.85. In these particular cases, most ensemble models yielded poor results.

Table 13 – Performance measures (RMSE, MAE, and Theil's U) for proposed approach (DESTC) and literature models (ARIMA, ETS, SVR, ELM, LSTM, MLP) on test set. Bold values represent the top performance across all models by time series, while underlined values denote the second best performance. R_{DESTC} represents the rankings of the proposed approaches compared to other models.

Time series	PM	ARIMA	ETS	SVR	ELM	LSTM	MLP	TSF	DESTC	R_{DESTC}
WDF-AL	RMSE	9.506	9.804	**8.478**	10.384	12.491	8.831	10.008	9.026	3
	MAE	7.988	7.212	**6.402**	7.885	10.14	6.941	8.015	6.797	2
	Theil's U	0.916	1.000	**0.759**	1.122	1.646	0.824	0.89	0.851	3
WDF-BA	RMSE	1255.5	1680.9	1319.8	1230.7	1237.2	1348.6	1170.5	**1081**	1
	MAE	828.2	925.3	678.2	759.7	754.1	1031.8	669.5	**665.4**	1
	Theil's U	0.65	1.172	0.722	0.622	0.633	0.742	0.619	**0.485**	1
WDF-CE	RMSE	302.4	380.6	206.6	202.6	418.6	634.5	228.3	**163.6**	1
	MAE	272.1	208.7	114.1	111.4	240.6	585	138.2	**107.3**	1
	Theil's U	0.709	1.141	0.336	0.323	1.381	3.09	0.411	**0.211**	1
LYNX	RMSE	404.5	571.9	438	**278.6**	812.3	554.5	632.9	317.4	2
	MAE	321.6	474.3	357.7	243.8	668.6	**456.7**	439.1	**213.2**	1
	Theil's U	0.381	0.324	0.473	**0.149**	1.624	0.592	0.433	0.250	2
SUNS	RMSE	20.5	30.6	19.1	17.00	19.00	22.7	16.7	**15.9**	1
	MAE	16.2	23.6	15.5	13.5	14.4	16.7	12.9	**12.0**	1
	Theil's U	0.431	1	0.384	**0.294**	0.354	0.525	0.315	0.296	2
ITSA4	RMSE	2.022	2.025	1.988	1.98	1.963	1.976	**0.408**	2.011	6
	MAE	0.582	0.572	0.789	0.817	0.582	0.629	**0.345**	0.658	6
	Theil's U	0.997	1.000	0.963	0.955	**0.94**	0.953	3.529	0.986	5
NASDAQ	RMSE	336.9	337.4	**325.6**	341.4	334.3	424.1	338.7	367.9	7
	MAE	270.9	274	263.6	271.5	**235.5**	363.6	271.6	309.2	7
	Theil's U	1.000	1.001	0.924	1.023	**0.862**	1.39	0.978	1.156	7
DEN-CE	RMSE	3716.2	4800.5	4827.6	2548.1	4075.1	4841.6	**2444.4**	3375	3
	MAE	2509.9	3747.3	3352.4	1972.2	3157.2	3445.5	**1722**	2637.1	4
	Theil's U	0.590	1.000	0.931	0.293	0.707	0.998	**0.270**	0.485	3
DEN-IQ	RMSE	5.287	5.193	**5.169**	5.588	6.254	5.937	5.245	6.398	8
	MAE	3.927	**3.269**	3.475	3.804	3.826	3.676	3.735	4.052	8
	Theil's U	0.973	0.939	**0.93**	1.087	1.361	1.226	1.011	1.425	8
MUC	RMSE	99.7	81.3	78.0	84.9	89.8	88.1	**61.4**	72.0	2
	MAE	87.1	64.5	58.0	70.2	74.8	74.8	**45.9**	56.3	2
	Theil's U	1.385	0.918	0.844	1.000	1.125	1.083	**0.498**	0.722	2

Table 14 – Performance measures (RMSE, MAE and Theil's U) for proposed approach (DESTC), static ensemble models (eSA, eSM, eELM, eSVR, and eMLP), and dynamic selection ensembles (DSLA and DESLA) on test set. Bold values represent the top performance across all models by time series, while underlined values denote the second best performance. R_{DESTC} represents the rankings of the proposed approaches compared to other models.

Time series	PM	eSA	eSM	eSVR	eELM	eMLP	DSLA	BL	DESLA	DESTC	R_{DESTC}
WDF-AL	RMSE	10.132	10.122	11.47	16.332	15.724	*9.347*	11.633	10.022	**9.026**	1
	MAE	7.762	7.549	9.195	12.93	13.779	**6.659**	8.415	6.947	*6.797*	2
	Theil	1.073	1.077	1.385	2.812	2.398	*0.905*	0.957	1.061	**0.851**	1
WDF-BA	RMSE	1189.2	1179.5	1105.0	6155.4	1380.2	1745.7	**1055.1**	1151.4	*1081.0*	2
	MAE	770.4	762.7	**566.0**	2544.6	870.0	1042.3	*636.3*	746.6	665.4	3
	Theil	0.585	0.577	*0.507*	15.721	0.786	1.264	0.576	0.548	**0.485**	1
WDF-CE	RMSE	188.7	*178.5*	326.1	1557.4	422.9	408.6	183.9	238.4	**163.6**	1
	MAE	143.5	117.6	224.2	824.5	217.6	229.5	**107**	171	*107.3*	2
	Theil	0.279	0.251	0.835	19.096	1.405	1.287	*0.246*	0.442	**0.211**	1
LYNX	RMSE	321.3	**222.5**	539.6	4598	1783.1	580.8	457.6	*267.1*	317.4	3
	MAE	274.9	**191.1**	472.6	3052.4	1482	459	364.2	230.8	*213.2*	2
	Theil	0.215	**0.085**	0.728	50.121	7.219	0.362	0.353	*0.111*	0.25	4
SUNS	RMSE	18.9	18.0	**15.8**	24.7	85.5	21.3	22.5	*17.4*	**15.9**	2
	MAE	14.4	13.9	12.1	18.1	74.4	16.9	14.6	12.7	**12.0**	1
	Theil	0.36	0.322	**0.265**	0.723	8.69	0.457	0.373	0.299	*0.296*	2
ITSA4	RMSE	2.004	2.013	2.211	2.801	9.929	1.987	**1.684**	*1.976*	2.011	5
	MAE	0.728	0.694	0.826	2.415	9.725	*0.626*	**0.464**	0.653	0.658	4
	Theil	0.979	0.988	1.192	1.912	23.128	*0.962*	0.963	**0.951**	0.986	5
NASDAQ	RMSE	402.8	*371.0*	571.4	5235.1	10747.8	401.1	387.4	436.1	**367.9**	1
	MAE	349.7	313.3	516.4	4697.4	8045.5	338.8	**287.1**	383.7	*309.2*	2
	Theil	1.383	1.181	2.885	243.596	1027.365	1.368	**0.995**	1.593	*1.156*	2
DEN-CE	RMSE	3640.7	**2624.6**	3105.4	32269.5	10664.1	4075.1	4521.4	*2929.4*	3375	4
	MAE	2349.4	**1793.1**	*2148.7*	21410.6	7838.5	3157.2	2220.9	2325.7	2637.1	6
	Theil	0.599	*0.301*	0.435	46.067	4.899	0.707	**0.188**	0.388	0.485	5
DEN-IQ	RMSE	5.795	*5.638*	6.72	12.819	22.381	6.32	**5.153**	5.716	6.398	6
	MAE	3.713	3.601	4.774	8.575	13.724	3.806	**3.385**	*3.572*	4.052	6
	Theil	1.169	*1.107*	1.568	5.709	17.439	1.391	**0.978**	1.137	1.425	6
MUC	RMSE	75.6	75.6	*71.3*	102.9	506.5	94.0	90.9	**68.3**	72.0	3
	MAE	60.00	59.00	58.9	81.2	368.9	77.0	71.6	**53.9**	*56.3*	2
	Theil	0.786	0.789	*0.701*	1.477	34.841	1.133	0.730	**0.646**	0.722	3

Figure 46 presents the boxplots of the ARM of the DESTC against single and ensemble models. For this calculation, the models were ranked according to Equation 2.57 (ARM) considering RMSE, MAE, and Theil's U values. The lower the ARM, the better the model performance is. The proposed approach presented better performance with lower median. Among the single models from the literature, TSF reached the best performance. Also, it is possible to note that the dynamic selection methods achieved good results among ensemble-based models.

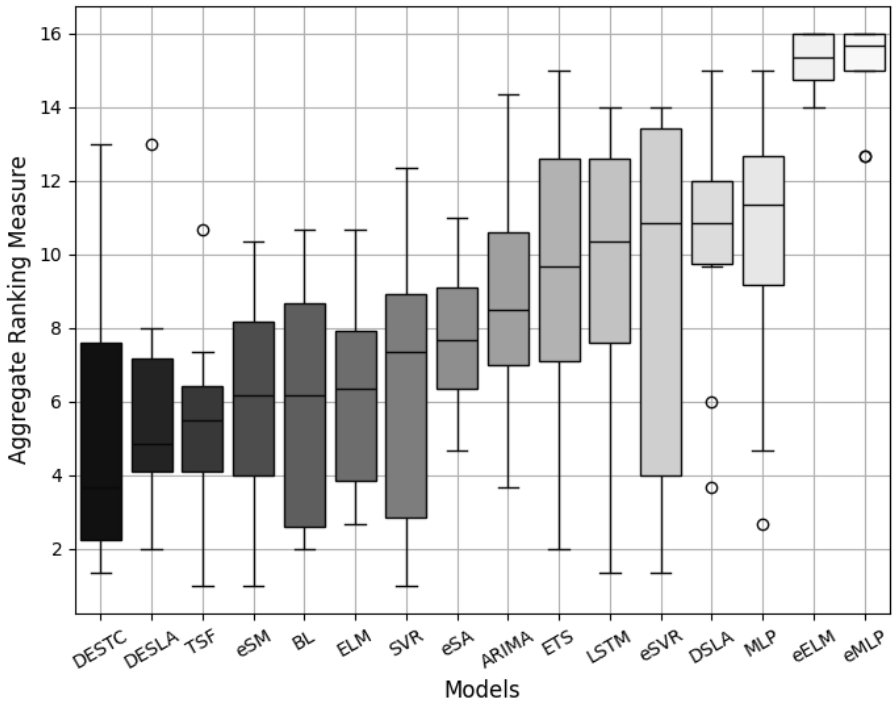


Figure 46 – Boxplot of ARM for single models and ensembles.

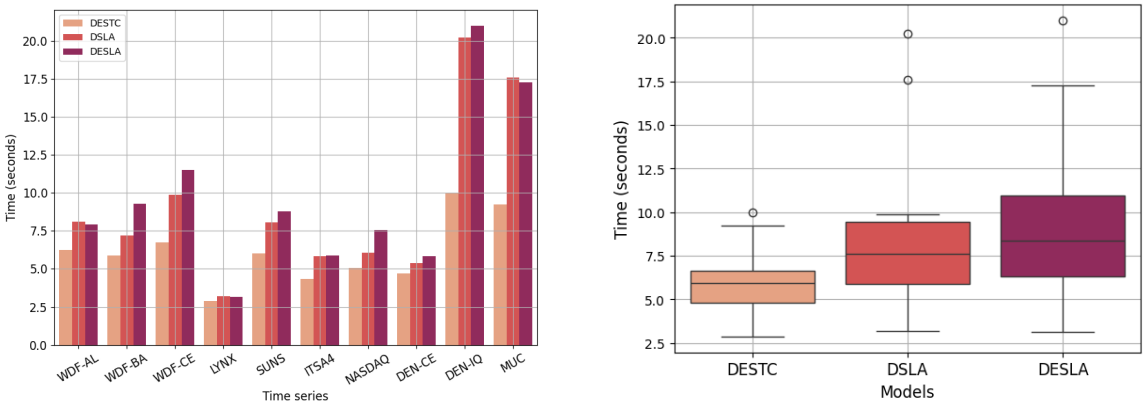
Table 15 presents the Bayesian signed rank test regarding the paired comparison of DESTC and other MPSs. In this experiment, DESTC did not achieve an overall better performance than DESLA, eSM, and BL, but it performed better than eSA, eSVR, DSLA, eELM, and eMLP. This result can be explained because, unlike Experiment A (Chapter 4), there are cases where DESTC exhibits low predictive performance (ITSA4, NASDAQ, and DEN-IQ), which will be discussed in more detail in the Session 5.3.

Table 15 – Paired comparison between MPSs using the Bayesian signed rank test. (rows versus columns). "+" in red ("−" in blue) indicates that the model listed in the row is better (worse) than the one listed in the column. "=" indicates that there is no difference.

ARM	DESTC	DESLA	eSM	BL	eSA	eSVR	DSLA	eELM	eMLP
DESTC	=	=	=	=	+	+	+	+	+
DESLA	=	=	=	=	+	+	+	+	+
eSM	=	=	=	=	+	=	+	+	+
BL	=	=	=	=	=	=	+	+	+
eSA	-	-	-	=	=	=	+	+	+
eSVR	-	-	=	=	=	=	=	+	+
DSLA	-	-	-	-	-	=	=	+	+
eELM	-	-	-	-	-	-	-	=	=
eMLP	-	-	-	-	-	-	-	-	=

Following, Figure 47 presents the processing time (in seconds) in the test phase across different time series regarding DS models (DESTC, DSLA, and DESLA). According to Figure

47(a), in all cases, DESTC achieved a shorter processing time than other DS analyzed. This difference is greater as the size of the series increases (Table 11). In both experiments, DESTC has demonstrated superior computational cost efficiency. This advantage arises from DESTC's approach of abstaining from constructing RoC or training meta-models during the selection phase, which are commonly practiced in the literature. This finding reinforces the potential of the approach presented here. Compared to Experiment A (Chapter 4), the difference between DESTC and other dynamic models arises because, in the first experiment, the series were larger (approximately one thousand observations).



(a) Processing time by time series.

(b) Boxplot of processing time by DS models.

Figure 47 – Processing time (in seconds) of test phase by time series.

Finally, the predictive performance of DESTC was evaluated in a scenario where noise was added to a time series in a preliminary study. The objective was to assess the performance of the DS methods in the presence of noise. Thus, pseudo-artificial time series were created from the original series by adding a randomly normally distributed value with a zero mean and a standard deviation of 0.1, 0.3, or 0.5 of the original series. Figure 48 presents the results of the experiment. In addition to DESTC, the models DSLA and DESLA were evaluated. The original series (without added noise) is represented by the first bar. The remaining bars indicate the series with noise, with darker bars representing higher amounts of added noise. The y-axis shows the RMSE values for each model across the four series.

As a result, it is observed that the addition of noise affects the models at different levels of intensity, following the logic that more noise leads to higher RMSE values. The models that were least and most affected were DESTC and DSLA, respectively. Additionally, it is observed that the models constructed based on the RoC (DESLA and DSLA) appear to be more negatively affected to the addition of noise. This may occur because the calculation

of similarities for constructing the regions of competence might be adversely affected by the noise. However, this is a preliminary analysis that requires further evaluation.

Walmsley et al. (2022) provide a thorough discussion on the impact of label noise in DS systems, asserting that these systems are severely affected. Although their focus is on classification problems, similar effects occur in regression problems. In this regard, it is pertinent to further analyze the impacts of label noise in DS methods within the context of time series forecasting.

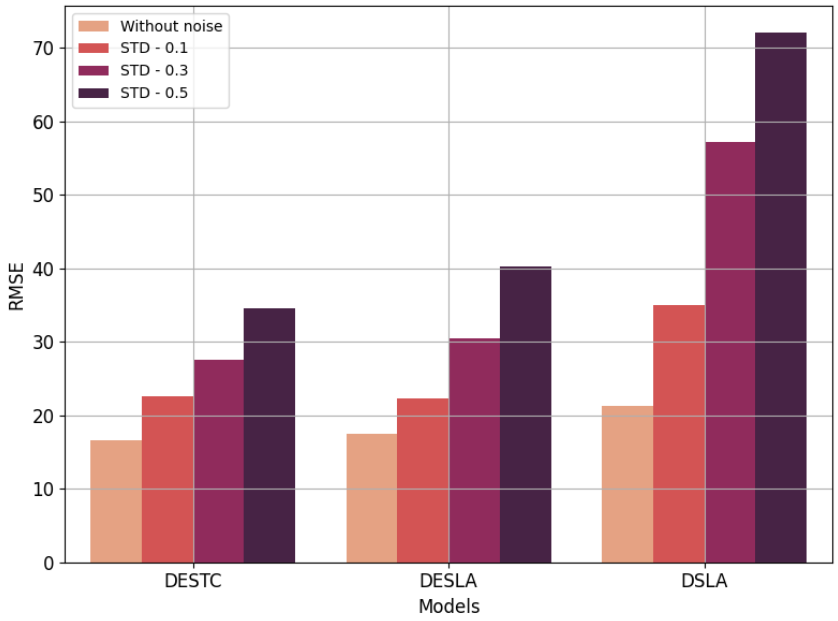


Figure 48 – Impact of noise addition on DS models (DESTC, DSLA, DESLA).

5.3 DISCUSSION

This experiment was conducted using ten time series from diverse domains, including environmental science (WDF-AL, WDF-BA, WDF-CE, LYNX, and SUNS), economics (ITSA4 and NASDAQ), and healthcare (DEN-CE, DEN-IQ, and MUC). In general, these time series exhibit characteristics that differ from those of COVID-19 incidence data, which showed alternating trend classes that typically persisted for weeks. Additionally, the pandemic time series diverged significantly in terms of trend intensity and observed value levels over time.

Regarding WDF time series, DESTC achieved good predictive performance (see Tables 13 and 14). Figures 49 and 50 show the trend analysis utilizing the Sen's Slope estimator and Mann-Kendall statistical test, along with their respective time series. Based on the Figures, one can see that the proposed approach was able to identify different local trends in the time

series. However, unlike the cases in Experiment A (Chapter 4), most of the instances here are classified as having no trend (especially for WDF-BA case, Figure 50). Nevertheless, even in this scenario, the DESTC was able to appropriately select models for combination, resulting in forecasts that were comparatively good relative to other adopted algorithms.

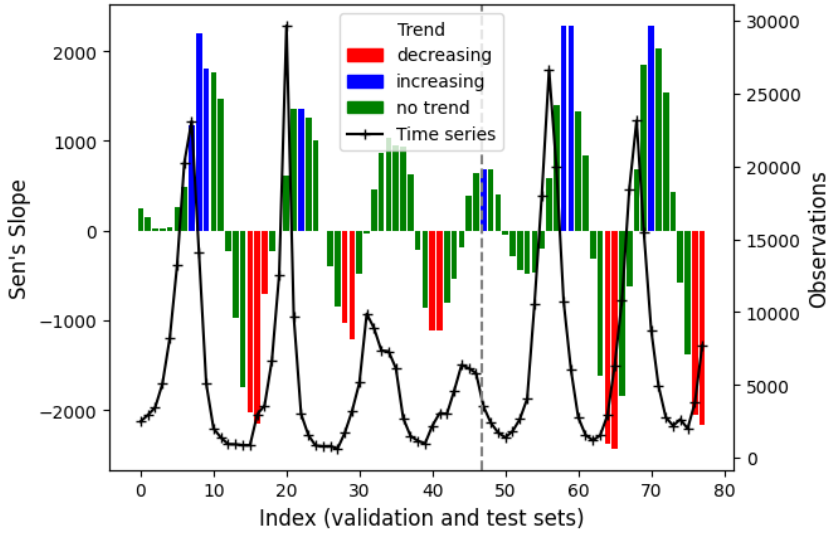


Figure 49 – Trend analysis based on Sen's Slope estimator and Mann-Kendall statistical test to WDF-AL time series.

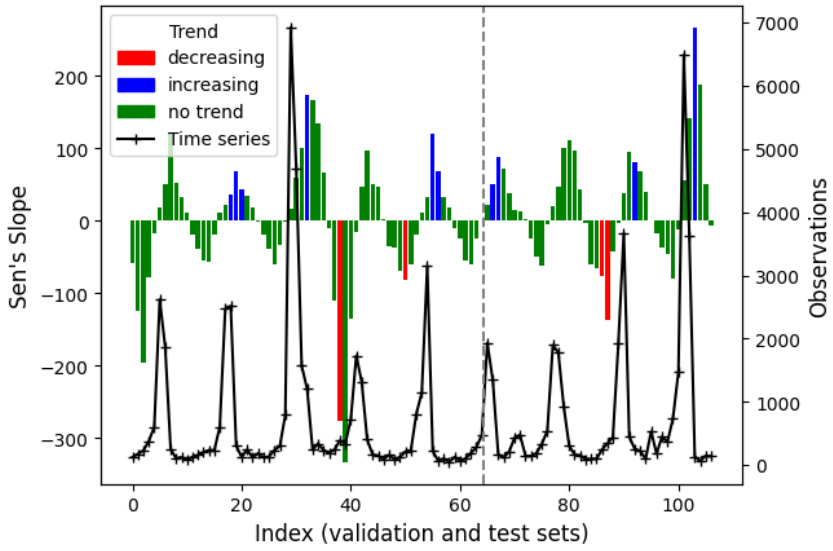


Figure 50 – Trend analysis based on Sen's Slope estimator and Mann-Kendall statistical test to WDF-BA time series.

Otherwise, for cases of dengue fever incidence (DEN-CE and DEN-IQ), the method achieved less competitive results, especially for the DEN-IQ (see Tables 13 and 14). Figure 51 shows the trend analysis for DEN-IQ series. Two key points emerge: Firstly, the series lacks a clear

trend cycle pattern, making model selection based on this logic impractical. Although there are instances classified with a positive trend in the test set, these are almost non-existent in the validation set. Secondly, there is a noticeable difference in peak patterns between validation and test sets, explaining the challenge in accurate predictions, not just for DESTC but for other algorithms as well.

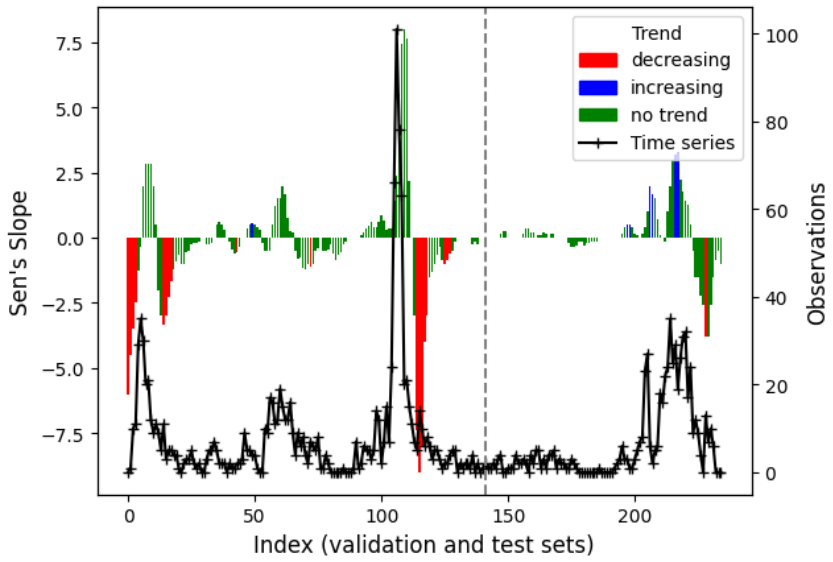


Figure 51 – Trend analysis based on Sen's Slope estimator and Mann-Kendall statistical test to DEN-IQ time series.

Another noteworthy case is that of the SUNS series. During the grid search, a lag size of 5 was selected, meaning the trend was estimated based on the previous 5 values of each observation. However, this led DESTC to classify all instances as no trend, as show in Figure 52 (all vertical bars are green). In this case, the performance of DESTC is close to BL, since it selects the best J models without considering trend classification.

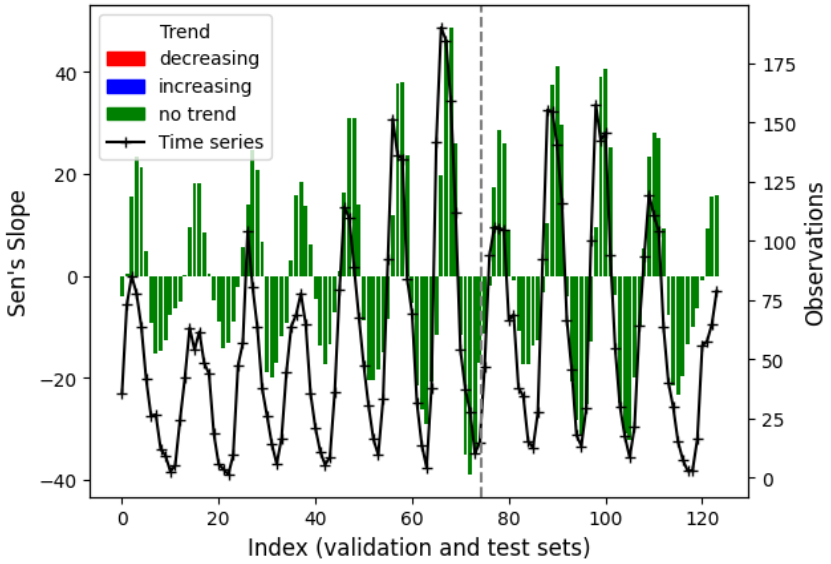


Figure 52 – Trend analysis based on Sen's Slope estimator and Mann-Kendall statistical test to SUNS time series.

The financial time series (ITSA4 and NASDAQ) also deserve attention. Although DESTC did not perform well compared to single models, it achieved good performance when compared to other MPSs, especially for the NASDAQ case. This may be explained by the possibility that, since these series are similar to a random walk, the strategy for generating the pool may have influenced the quality of the MPSs predictions. Figure 53 presents the trend analysis of the NASDAQ series. It is noticeable that the method was able to track overall trend changes but struggled to classify the apparent negative trend at the beginning of the test set.

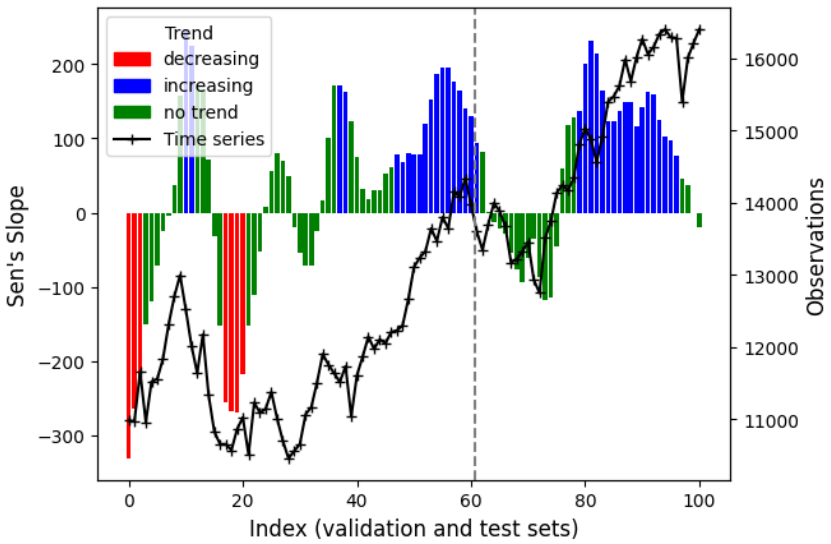


Figure 53 – Trend analysis based on Sen's Slope estimator and Mann-Kendall statistical test to NASDAQ time series.

Furthermore, regarding the preliminary analyses conducted with the addition of noise to a time series, it was observed that among the dynamic selection methods, DESTC exhibited the least negative impact as more noise was introduced. This reduced impact may stem from its trend detection being mathematically simpler compared to RoC alternatives (DSLA and DESLA). However, further studies are necessary to delve into, discuss, and comprehensively validate this analysis. Noise evaluation is particularly relevant in problems involving real-world time series data, as this type of data often exhibits such behavior, impacting the accuracy and reliability of predictive models. Thus, understanding how different models handle noise can lead to more robust and effective forecasting strategies.

6 CONCLUSION

In this thesis, a DES system, DESTC, based on trend classification was proposed for time series forecasting problems. The proposed approach consists of two main phases: the training phase (a), in which a pool of models is evaluated to determine the best ones for each trend class, and the testing phase (b), in which each new instance has its trend assessed, and the top-performing models are selected for prediction. DESTC was initially proposed to address the challenge of predicting COVID-19 incidence time series, as both single and MPS alternatives have shown low predictive performance. Thus, to evaluate the predictive performance of DESTC, two experiments were conducted.

The Experiment A (Chapter 4) was conducted in order to achieve the first main objective: **(i) to develop a new selection approach based on trend classification to enhance the forecasting of COVID-19 incidence time series**. In this case, the proposed approach was applied to time series of COVID-19 incidence in different countries and compared with single (ARIMA, ETS, SVR, ELM, MLP, LSTS, and TSF) and ensemble-based (eSA, eSM, eSVR, eMLP, eELM, DSLA, DESLAa and DESLAm) algorithms. Generally, the proposed approach obtained higher forecasting performance. However, some limitations were identified. As shown in the example of Canada, DESTC did not consistently select the most suitable models for combination, which could be due to sudden changes in the structure of the time series. In this instance, the time series structure changed significantly enough that the optimal models for each trend class also shifted. Thus, relying solely on trend classes for selection proved ineffective in ensuring model accuracy. Additionally, the parameters in DESTC are chosen through a validation process. However, the patterns observed during validation may not accurately reflect those encountered during testing. Therefore, exploring alternative methods to improve the model selection phase would be beneficial.

Furthermore, Experiment B (Chapter 5) was conducted in order to achieve the second main objective: **(ii) to assess the applicability of this new approach in time series from other domains**. In this case, DESTC was applied on ten time series with distinct characteristics from different phenomena. From the results, DESTC proved to be a competitive alternative when compared to other MPSs. Nevertheless, other limitations could be discussed. Firstly, DESTC tends to have lower predictive performance when the time series lacks a clear trend cycle pattern, making model selection based on this logic impractical, just as it happens in

the case of DEN-IQ. Chapter 5 also presented a preliminary study regarding the creation of semi-synthetic time series through the addition of noise to the time series. As result, DESTC showed the least negative impact compared to other dynamic selection methods (DSLA and DESLA). However, further studies are needed to thoroughly validate these findings.

Moreover, DESTC has shown superior performance in terms of computational cost efficiency in both experiments. This advantage stems from DESTC's approach of not creating a RoC or training meta-models during the selection stage, practices that are commonly found in the literature. As showed in Experiments A and B, as the time series length increases, the difference in processing time between DESTC and other dynamic selection methods becomes more pronounced. This finding reinforces the potential of the approach presented here.

Overall, DESTC is a novel, efficient, and simple dynamic ensemble selection system specifically designed for time series forecasting with trend cycles. The results presented makes DESTC particularly attractive for real-world applications where processing speed and resource limitations are crucial considerations.

6.1 FUTURE WORKS

Some research questions emerge as future work:

- While DESTC focuses solely on trend classification, a potentially fruitful avenue for further investigation lies in incorporating additional statistical characteristics for model selection. Seasonality, for instance, represents a crucial factor that could significantly influence the underlying patterns in the data. To address this, exploring methodologies capable of performing model selection based on multiple characteristics, including both trend and seasonality, is highly recommended.

- In this thesis, only the Mann-Kendall statistical test was used for trend detection. While this approach provides a useful starting point, expanding the analysis to include a variety of trend classification techniques would enhance the robustness of the results and provide a deeper insight into the underlying trends. This could mean considering established alternatives and possibly even creating new methodologies tailored to the specific data and research objectives.

- The current trend classification based on three discrete classes (increasing, decreasing,

and no trend) could present limitations in capturing the nuances of real-world data. Expanding upon this classification scheme by incorporating additional classes could offer a more granular and informative representation of the underlying trends. A particularly promising approach for achieving this involves the application of fuzzy logic.

- Regarding ensemble training, developing an algorithm to determine the optimal number of models in the pool for each time series could improve the efficiency of forecasting systems. This algorithm could utilize both the accuracy of individual models and diversity of the pool.

- Dynamic selection methods that construct RoC are well disseminated in the literature. Therefore, we aim to develop a hybrid methodology that combines this approach with DESTC. The idea is to leverage the advantages of both proposals.

- Assess the occurrence of concept drift and adapt the DESTC based on this information.

- The codes related to the current project are being organized into an R package. The package presents functions with a pool generation strategy, the DESCT method, and performance measures.

REFERENCES

ABBASIMEHR, H.; SHABANI, M.; YOUSEFI, M. An optimized model using lstm network for demand forecasting. *Computers & Industrial Engineering*, Elsevier, v. 143, p. 106435, 2020.

ABDEL-NASSER, M.; MAHMOUD, K. Accurate photovoltaic power forecasting models using deep lstm-rnn. *Neural Computing and Applications*, Springer, v. 31, p. 2727–2740, 2019.

ADHIKARI, R.; AGRAWAL, R. Combining multiple time series models through a robust weighted mechanism. In: IEEE. *2012 1st International Conference on Recent Advances in Information Technology (RAIT)*. [S.l.], 2012. p. 455–460.

ADHIKARI, R.; VERMA, G.; KHANDELWAL, I. A model ranking based selective ensemble approach for time series forecasting. *Procedia Computer Science*, Elsevier, v. 48, p. 14–21, 2015.

AGBO, E. P.; NKAJOE, U.; EDET, C. O. Comparison of mann–kendall and şen's innovative trend method for climatic parameters over nigeria's climatic zones. *Climate Dynamics*, Springer, v. 60, n. 11, p. 3385–3401, 2023.

AHMED, S.; NIELSEN, I. E.; TRIPATHI, A.; SIDDIQUI, S.; RAMACHANDRAN, R. P.; RASOOL, G. Transformers in time-series analysis: A tutorial. *Circuits, Systems, and Signal Processing*, Springer, v. 42, n. 12, p. 7433–7466, 2023.

ALMAZROUI, M.; ŞEN, Z. Trend analyses methodologies in hydro-meteorological records. *Earth Systems and Environment*, Springer, v. 4, p. 713–738, 2020.

ALRASSAS, A. M.; AL-QANESS, M. A.; EWEES, A. A.; REN, S.; SUN, R.; PAN, L.; ELAZIZ, M. A. Advance artificial time series forecasting model for oil production using neuro fuzzy-based slime mould algorithm. *Journal of Petroleum Exploration and Production Technology*, Springer, p. 1–13, 2022.

AMARAL, T. *Mineração de Regras de Exceção em Séries Temporais Multivariadas*. Master's Thesis (Master's Thesis) — Instituto de Ciências Matemáticas e de Computação - Universidade de São Paulo, 2020.

APICELLA, A.; DONNARUMMA, F.; ISGRÒ, F.; PREVETE, R. A survey on modern trainable activation functions. *Neural Networks*, Elsevier, v. 138, p. 14–32, 2021.

AWADASSEID, A.; WU, Y.; TANAKA, Y.; ZHANG, W. Initial success in the identification and management of the coronavirus disease 2019 (COVID-19) indicates human-to-human transmission in Wuhan, China. *International Journal of Biological Sciences*, Ivyspring International Publisher, v. 16, n. 11, p. 1846, 2020.

BANERJEE, K. *Generalized inverse of matrices and its applications*. [S.l.]: Taylor & Francis, 1973.

BARI, S. H.; RAHMAN, M. T. U.; HOQUE, M. A.; HUSSAIN, M. M. Analysis of seasonal and annual rainfall trends in the northern region of bangladesh. *Atmospheric Research*, Elsevier, v. 176, p. 148–158, 2016.

BARRERA-ANIMAS, A. Y.; OYEDELE, L. O.; BILAL, M.; AKINOSHO, T. D.; DELGADO, J. M. D.; AKANBI, L. A. Rainfall prediction: a comparative analysis of modern machine learning algorithms for time-series forecasting. *Machine Learning with Applications*, Elsevier, v. 7, p. 100204, 2022.

BENAVOLI, A.; CORANI, G.; DEMŠAR, J.; ZAFFALON, M. Time for a change: a tutorial for comparing multiple classifiers through bayesian analysis. *Journal of Machine Learning Research*, v. 18, n. 77, p. 1–36, 2017. Available at: <http://jmlr.org/papers/v18/16-305.html>.

BENEDUM, C. M.; SHEA, K. M.; JENKINS, H. E.; KIM, L. Y.; MARKUZON, N. Weekly dengue forecasts in iquitos, peru; san juan, puerto rico; and singapore. *Plos Neglected Tropical Diseases*, Public Library of Science San Francisco, CA USA, v. 14, n. 10, p. e0008710, 2020.

BERRY, I.; RAHMAN, M.; FLORA, M. S.; SHIRIN, T.; ALAMGIR, A.; KHAN, M. H.; ANWAR, R.; LISA, M.; CHOWDHURY, F.; ISLAM, M. A. et al. Seasonality of influenza and coseasonality with avian influenza in Bangladesh, 2010–19: a retrospective, time-series analysis. *The Lancet Global health*, Elsevier, v. 10, n. 8, p. e1150–e1158, 2022.

BOX, G.; JENKINS, G. Time series analysis: forecasting and control. 1976.

BRITTO JUNIOR, A. S.; SABOURIN, R.; OLIVEIRA, L. E. Dynamic selection of classifiers—a comprehensive review. *Pattern Recognition*, Elsevier, v. 47, n. 11, p. 3665–3680, 2014.

BROCKWELL, P. J.; DAVIS. *Time Series and Forecasting Methods*. [S.l.]: Springer, 1991.

BROWN, G.; WYATT, J.; HARRIS, R.; YAO, X. Diversity creation methods: a survey and categorisation. *Information Fusion*, Elsevier, v. 6, n. 1, p. 5–20, 2005.

BROWN, R. G. Statistical forecasting for inventory control. McGraw-Hill, 1959.

BUDI, I.; AJI, R. F.; WIDODO, A. Prediction of research topics using ensemble of best predictors from similar dataset. *International Journal of Computer and Information Engineering*, v. 8, n. 1, p. 86–92, 2014.

CABRAL, J. T. H. d. A.; OLIVEIRA, A. L. Ensemble effort estimation using dynamic selection. *Journal of Systems and Software*, Elsevier, v. 175, p. 110904, 2021.

CAMPBELL, M.; WALKER, A. A survey of statistical work on the mackenzie river series of annual canadian lynx trappings for the years 1821-1934 and a new analysis. *Journal of the Royal Statistical Society: Series A (General)*, Wiley Online Library, v. 140, n. 4, p. 411–431, 1977.

CERQUEIRA, V.; TORGO, L.; OLIVEIRA, M.; PFAHRINGER, B. Dynamic and heterogeneous ensembles for time series forecasting. In: IEEE. *2017 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*. [S.l.], 2017. p. 242–251.

CERQUEIRA, V.; TORGO, L.; PINTO, F.; SOARES, C. Arbitrated ensemble for time series forecasting. In: SPRINGER. *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2017, Skopje, Macedonia, September 18–22, 2017, Proceedings, Part II 10*. [S.l.], 2017. p. 478–494.

CERQUEIRA, V.; TORGO, L.; PINTO, F.; SOARES, C. Arbitrage of forecasting experts. *Machine Learning*, Springer, v. 108, p. 913–944, 2019.

CHEN, T. Xgboost: extreme gradient boosting. *R package version 0.4-2*, v. 1, n. 4, 2015.

CHEN, T.; GUESTRIN, C. Xgboost: A scalable tree boosting system. In: *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*. [S.l.: s.n.], 2016. p. 785–794.

CHUNG, H. W.; APIO, C.; GOO, T.; HEO, G.; HAN, K.; KIM, T.; KIM, H.; KO, Y.; LEE, D.; LIM, J. et al. Effects of government policies on the spread of covid-19 worldwide. *Scientific Reports*, Nature Publishing Group UK London, v. 11, n. 1, p. 20495, 2021.

CLEMEN, R. T. Combining forecasts: A review and annotated bibliography. *International Journal of Forecasting*, Elsevier, v. 5, n. 4, p. 559–583, 1989.

COWPERTWAIT, P. S.; METCALFE, A. V. *Introductory time series with R*. [S.l.]: Springer, 2009.

CRAYER, J.; CHAN, K. *Time Series Analysis with application in R*. [S.l.]: Springer, 2008.

CRUZ, R. M.; SABOURIN, R.; CAVALCANTI, G. D. Dynamic classifier selection: Recent advances and perspectives. *Information Fusion*, Elsevier, v. 41, p. 195–216, 2018.

DARAGHMEH, M.; AGARWAL, A.; MANZANO, R.; ZAMAN, M. Time series forecasting using facebook prophet for cloud resource management. In: IEEE. *2021 IEEE International Conference on Communications Workshops (ICC Workshops)*. [S.l.], 2021. p. 1–6.

DAS, A. K.; MISHRA, D.; DAS, K.; MOHANTY, A. K.; MOHAMMED, M. A.; AL-WAISY, A. S.; KADRY, S.; KIM, J. A deep network-based trade and trend analysis system to observe entry and exit points in the forex market. *Mathematics*, MDPI, v. 10, n. 19, p. 3632, 2022.

DE MATTOS NETO, P. S.; FIRMINO, P. R. A.; SIQUEIRA, H.; TADANO, Y. D. S.; ALVES, T. A.; OLIVEIRA, J. F. L. D.; MARINHO, M. H. D. N.; MADEIRO, F. Neural-based ensembles for particulate matter forecasting. *IEEE Access*, IEEE, v. 9, p. 14470–14490, 2021.

DEB, C.; ZHANG, F.; YANG, J.; LEE, S. E.; SHAH, K. W. A review on time series forecasting techniques for building energy consumption. *Renewable and Sustainable Energy Reviews*, Elsevier, v. 74, p. 902–924, 2017.

DETHLEFSEN, C.; LUNDBYE-CHRISTENSEN, S. Formulating state space models in r with focus on longitudinal regression models. *Journal of Statistical Software*, v. 16, p. 1–15, 2006.

DIEBOLD, F. X. Comparing predictive accuracy, twenty years later: A personal perspective on the use and abuse of diebold–mariano tests. *Journal of Business & Economic Statistics*, Taylor & Francis, v. 33, n. 1, p. 1–1, 2015.

DIEBOLD, F. X.; MARIANO, R. S. *Comparing predictive accuracy I: An asymptotic test*. [S.l.], 1991.

DING, S.; XU, X.; NIE, R. Extreme learning machine and its applications. *Neural Computing and Applications*, Springer, v. 25, p. 549–556, 2014.

DOMINGOS, S. d. O.; OLIVEIRA, J. F. de; DE MATTOS NETO, P. S. An intelligent hybridization of arima with machine learning models for time series forecasting. *Knowledge-Based Systems*, Elsevier, v. 175, p. 72–86, 2019.

DONG, L.; XU, S.; XU, B. Speech-transformer: a no-recurrence sequence-to-sequence model for speech recognition. In: IEEE. *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. [S.l.], 2018. p. 5884–5888.

ELMI, J.; EFTEKHARI, M.; MEHRPOOYA, A.; RAVARI, M. R. A novel framework based on the multi-label classification for dynamic selection of classifiers. *International Journal of Machine Learning and Cybernetics*, Springer, v. 14, n. 6, p. 2137–2154, 2023.

FATICHI, S.; IVANOV, V.; CAPORALI, E. Assessment of a stochastic downscaling methodology in generating an ensemble of hourly future climate time series. *Climate Dynamics*, Springer, v. 40, p. 1841–1861, 2013.

FENG, Z.-k.; NIU, W.-j. Hybrid artificial neural network and cooperation search algorithm for nonlinear river flow time series forecasting in humid and semi-humid regions. *Knowledge-Based Systems*, Elsevier, v. 211, p. 106580, 2021.

FEO, G.; GIORDANO, F.; NIGLIO, M.; PARRELLA, M. L. Financial time series classification by nonparametric trend estimation. In: *Mathematical and Statistical Methods for Actuarial Sciences and Finance: MAF 2022*. [S.l.]: Springer, 2022. p. 241–246.

FERNANDES, J. L.; EBECKEN, N. F. F.; ESQUERDO, J. C. D. M. Sugarcane yield prediction in Brazil using NDVI time series and neural networks ensemble. *International Journal of Remote Sensing*, Taylor & Francis, v. 38, n. 16, p. 4631–4644, 2017.

FERNÁNDEZ, C.; SALINAS, L.; TORRES, C. E. A meta extreme learning machine method for forecasting financial time series. *Applied Intelligence*, Springer, v. 49, p. 532–554, 2019.

FIRMINO, P. R. A.; SALES, J. P. de; JÚNIOR, J. G.; SILVA, T. A. da. A non-central beta model to forecast and evaluate pandemics time series. *Chaos, Solitons & Fractals*, Elsevier, v. 140, p. 110211, 2020.

FONSECA, J. S. da; MARTINS, G. de A.; TOLEDO, G. L. *Estatística Aplicada* . [S.l.]: Editora Atlas SA, 2000.

FONSECA, R.; GOMEZ, P. Automatic model selection in ensembles for time series forecasting. *IEEE Latin America Transactions*, IEEE, v. 14, n. 8, p. 3811–3819, 2016.

GASTINGER, J.; NICOLAS, S.; STEPIĆ, D.; SCHMIDT, M.; SCHÜLKE, A. A study on ensemble learning for time series forecasting and the need for meta-learning. In: IEEE. *2021 International Joint Conference on Neural Networks (IJCNN)*. [S.l.], 2021. p. 1–8.

GEMAN, S.; BIENENSTOCK, E.; DOURSAT, R. Neural networks and the bias/variance dilemma. *Neural Computation*, v. 4, n. 1, p. 1–58, 1992.

GÉRON, A. *Mãos à Obra: Aprendizado de Máquina com Scikit-Learn & TensorFlow*. [S.l.]: Alta Books, 2019.

GÉRON, A. *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow*. [S.l.]: O'Reilly Media, Inc., 2022.

GHEYAS, I. A.; SMITH, L. S. A novel neural network ensemble architecture for time series forecasting. *Neurocomputing*, Elsevier, v. 74, n. 18, p. 3855–3864, 2011.

GRAVES, A. Generating sequences with recurrent neural networks. *arXiv preprint arXiv:1308.0850*, 2013.

GUO, A.; BEHESHTI, R.; KHAN, Y. M.; LANGABEER, J. R.; FORAKER, R. E. Predicting cardiovascular health trajectories in time-series electronic health records with lstm models. *BMC Medical Informatics and Decision Making*, Springer, v. 21, p. 1–10, 2021.

HAAN, C. T. Statistical methods in hydrology: Ames. *Iowa University Press*, 1977.

HANANYA, R.; KATZ, G. Dynamic selection of machine learning models for time-series data. *Information Sciences*, Elsevier, p. 120360, 2024.

HASHEM, S. Optimal linear combinations of neural networks. *Neural Networks*, Elsevier, v. 10, n. 4, p. 599–614, 1997.

HAYKIN, S. *Neural networks and learning machines, 3/E*. [S.l.]: Pearson Education India, 2009.

HEATON, J. *Applications of Deep Neural Networks - Keras Transformers for Time series*. [S.l.], 2022. Available at: <https://github.com/jeffheaton/t81_558_deep_learning/blob/master/t81_558_class_10_5_keras_transformers.ipynb>.

HELSEL, D. R.; FRANS, L. M. Regional kendall test for trend. *Environmental science & technology*, ACS Publications, v. 40, n. 13, p. 4066–4073, 2006.

HERBOLD, S. Autorank: A python package for automated ranking of classifiers. *Journal of Open Source Software*, v. 5, n. 48, p. 2173, 2020.

HIRSCH, R. M.; SLACK, J. R.; SMITH, R. A. Techniques of trend analysis for monthly water quality data. *Water Resources Research*, v. 18, n. 1, p. 107–121, 1982.

HOCHREITER, S.; SCHMIDHUBER, J. Long short-term memory. *Neural Computation*, MIT press, v. 9, n. 8, p. 1735–1780, 1997.

HOLT, C. C. Forecasting seasonals and trends by exponentially weighted moving averages. *International Journal of Forecasting*, Elsevier, v. 20, n. 1, p. 5–10, 2004.

HOPE, T. M. Linear regression. In: *Machine Learning*. [S.l.]: Elsevier, 2020. p. 67–81.

HU, W.; YAN, L.; LIU, K.; WANG, H. A short-term traffic flow forecasting method based on the hybrid pso-svr. *Neural Processing Letters*, Springer, v. 43, p. 155–172, 2016.

HUANG, G.-B.; ZHOU, H.; DING, X.; ZHANG, R. Extreme learning machine for regression and multiclass classification. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, IEEE, v. 42, n. 2, p. 513–529, 2011.

HUANG, G.-B.; ZHU, Q.-Y.; SIEW, C.-K. Extreme learning machine: theory and applications. *Neurocomputing*, Elsevier, v. 70, n. 1-3, p. 489–501, 2006.

HYNDMAN, R. J.; ATHANASOPOULOS, G. *Forecasting: principles and practice*. [S.l.]: OTexts, 2018.

INPE. *Instituto Nacional de Pesquisas Espaciais - Programa Queimadas do INPE*. 2024. Available at: <https://terrabrasilis.dpi.inpe.br/queimadas/portal/>.

IOANNIDIS, J. P.; CRIPPS, S.; TANNER, M. A. Forecasting for COVID-19 has failed. *International Journal of Forecasting*, Elsevier, 2020.

JAYATHILAKE, H. M.; PRESCOTT, G. W.; CARRASCO, L. R.; RAO, M.; SYMES, W. S. Drivers of deforestation and degradation for 28 tropical conservation landscapes. *Ambio*, Springer, v. 50, p. 215–228, 2021.

KANG, F.; LI, J. Artificial bee colony algorithm optimized support vector regression for system reliability analysis of slopes. *Journal of Computing in Civil Engineering*, American Society of Civil Engineers, v. 30, n. 3, p. 04015040, 2016.

KARTHIKEYAN, L.; KUMAR, D. N. Predictability of nonstationary time series using wavelet and EMD based ARMA models. *Journal of Hydrology*, Elsevier, v. 502, p. 103–119, 2013.

KAUSHIK, S.; CHOUDHURY, A.; SHERON, P. K.; DASGUPTA, N.; NATARAJAN, S.; PICKETT, L. A.; DUTT, V. Ai in healthcare: time-series forecasting using statistical, neural, and ensemble architectures. *Frontiers in Big Data*, Frontiers Media SA, v. 3, p. 4, 2020.

KENDALL, M. Rank correlation methods, charles griffin. *Google Sch*, 1975.

KIM, K. J. Financial time series forecasting using support vector machines. *Neurocomputing*, Elsevier, v. 55, n. 1-2, p. 307–319, 2003.

KOURENTZES, N.; BARROW, D. K.; CRONE, S. F. Neural network ensemble operators for time series forecasting. *Expert Systems with Applications*, Elsevier, v. 41, n. 9, p. 4235–4244, 2014.

KOUZIOKAS, G. N. Neural network-based road accident forecasting in transportation and public management. In: SPRINGER. *Data Analytics: Paving the Way to Sustainable Urban Mobility: Proceedings of 4th Conference on Sustainable Urban Mobility (CSUM2018), 24-25 May, Skiathos Island, Greece*. [S.l.], 2019. p. 98–103.

KRATZERT, F.; KLOTZ, D.; BRENNER, C.; SCHULZ, K.; HERRNEGGER, M. Rainfall–runoff modelling using long short-term memory (LSTM) networks. *Hydrology and Earth System Sciences*, Copernicus Publications Göttingen, Germany, v. 22, n. 11, p. 6005–6022, 2018.

KROGH, A.; VEDELSBY, J. Neural network ensembles, cross validation, and active learning. *Advances in Neural Information Processing Systems*, v. 7, 1994.

KUMAR, G.; SINGH, U. P.; JAIN, S. An adaptive particle swarm optimization-based hybrid long short-term memory model for stock price time series forecasting. *Soft Computing*, Springer, v. 26, n. 22, p. 12115–12135, 2022.

KUMAR, V.; CHHABRA, J. K.; KUMAR, D. Performance evaluation of distance metrics in the clustering algorithms. *INFOCOMP Journal of Computer Science*, v. 13, n. 1, p. 38–52, 2014.

KUMARI, S.; SINGH, S. K. Machine learning-based time series models for effective co2 emission prediction in india. *Environmental Science and Pollution Research*, Springer, v. 30, n. 55, p. 116601–116616, 2023.

LAMOUNIER, W. M. Tendência, ciclos e sazonalidade nos preços spot do café brasileiro na nybot. *Gestão & Produção*, SciELO Brasil, v. 14, p. 13–23, 2007.

LARABI-MARIE-SAINTE, S.; ALHALAWANI, S.; SHAHEEN, S.; ALMUSTAFA, K. M.; SABA, T.; KHAN, F. N.; REHMAN, A. Forecasting COVID-19 parameters using time-series: KSA, USA, Spain, and Brazil comparative case study. *Heliyon*, Elsevier, v. 8, n. 6, p. e09578, 2022.

LARREA, M.; PORTO, A.; IRIGOYEN, E.; BARRAGÁN, A. J.; ANDÚJAR, J. M. Extreme learning machine ensemble model for time series forecasting boosted by pso: application to an electric consumption problem. *Neurocomputing*, Elsevier, v. 452, p. 465–472, 2021.

LEMKE, C.; GABRYS, B. Meta-learning for time series forecasting and forecast combination. *Neurocomputing*, Elsevier, v. 73, n. 10-12, p. 2006–2016, 2010.

LESHNO, M.; LIN, V. Y.; PINKUS, A.; SCHOCKEN, S. Multilayer feedforward networks with a nonpolynomial activation function can approximate any function. *Neural Networks*, Elsevier, v. 6, n. 6, p. 861–867, 1993.

LI, D.-Y.; XU, W.; ZHAO, H.; CHEN, R.-Q. A svr based forecasting approach for real estate price prediction. In: IEEE. *2009 International Conference on Machine Learning and Cybernetics*. [S.l.], 2009. v. 2, p. 970–974.

LI, S.; GOEL, L.; WANG, P. An ensemble approach for short-term load forecasting by extreme learning machine. *Applied Energy*, Elsevier, v. 170, p. 22–29, 2016.

LIBISELLER, C.; GRIMVALL, A. Performance of partial Mann–Kendall tests for trend detection in the presence of covariates. *Environmetrics: The Official Journal of the International Environmetrics Society*, Wiley Online Library, v. 13, n. 1, p. 71–84, 2002.

LIM, B.; ZOHREN, S. Time-series forecasting with deep learning: a survey. *Philosophical Transactions of the Royal Society A*, The Royal Society Publishing, v. 379, n. 2194, p. 20200209, 2021.

LIN, T.; WANG, Y.; LIU, X.; QIU, X. A survey of transformers. *AI Open*, Elsevier, v. 3, p. 111–132, 2022.

LIU, N.; WANG, H. Ensemble based extreme learning machine. *IEEE Signal Processing Letters*, IEEE, v. 17, n. 8, p. 754–757, 2010.

LUO, J. Forecasting COVID-19 pandemic: Unknown unknowns and predictive monitoring. *Technological Forecasting and Social Change*, Elsevier, v. 166, p. 120602, 2021.

LV, S.-X.; PENG, L.; HU, H.; WANG, L. Effective machine learning model combination based on selective ensemble strategy for time series forecasting. *Information Sciences*, Elsevier, v. 612, p. 994–1023, 2022.

MAALIW, R. R.; BALLERA, M. A.; MABUNGA, Z. P.; MAHUSAY, A. T.; DEJELO, D. A.; SEÑO, M. P. An ensemble machine learning approach for time series forecasting of COVID-19 cases. In: IEEE. *2021 IEEE 12th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*. [S.l.], 2021. p. 0633–0640.

MANN, H. B. Nonparametric tests against trend. *Econometrica: Journal of the Econometric Society*, JSTOR, p. 245–259, 1945.

MASINI, R. P.; MEDEIROS, M. C.; MENDES, E. F. Machine learning advances for time series forecasting. *Journal of Economic Surveys*, Wiley Online Library, v. 37, n. 1, p. 76–111, 2023.

MATSUMOTO, F. *Redes Neurais | LSTM*. 2019. <https://medium.com/turing-talks/turing-talks-27-modelos-de-predi%C3%A7%C3%A3o-lstm-df85d87ad210>. Accessed: 2023-Aug-10.

MATTOS, R. S. de. Tendências e raízes unitárias. *Texto didático. Juiz de Fora: UFJF*, p. 61, 2018.

MITTAL, A. *Understanding RNN and LSTM*. 2019. <https://aditi-mittal.medium.com/understanding-rnn-and-lstm-f7cdf6dfc14e>. Accessed: 2023-Aug-10.

MOHAMMED, F. A.; MOUSA, M. A. Applying diebold–mariano test for performance evaluation between individual and hybrid time-series models for modeling bivariate time-series data and forecasting the unemployment rate in the USA. In: SPRINGER. *Theory and Applications of Time Series Analysis: Selected Contributions from ITISE 2019 6*. [S.l.], 2020. p. 443–458.

MORETTI, F.; PIZZUTI, S.; PANZIERI, S.; ANNUNZIATO, M. Urban traffic flow forecasting through statistical and neural network bagging ensemble hybrid modeling. *Neurocomputing*, Elsevier, v. 167, p. 3–7, 2015.

MORETTIN, P. A.; TOLOI, C. Análise de séries temporais. In: *Análise de séries temporais*. [S.l.: s.n.], 2006.

MOURA, T. J.; CAVALCANTI, G. D.; OLIVEIRA, L. S. Mine: A framework for dynamic regressor selection. *Information Sciences*, Elsevier, v. 543, p. 157–179, 2021.

MUHAREB, R.; GIACAMAN, R. Tracking COVID-19 responsibly. *The Lancet*, Elsevier, 2020.

NETO, P. S. de M.; CAVALCANTI, G. D.; FIRMINO, P. R.; SILVA, E. G.; FILHO, S. R. V. N. A temporal-window framework for modelling and forecasting time series. *Knowledge-Based Systems*, Elsevier, v. 193, p. 105476, 2020.

OLIVEIRA, D. D.; RAMPINELLI, M.; TOZATTO, G. Z.; ANDREÃO, R. V.; MÜLLER, S. M. Forecasting vehicular traffic flow using MLP and LSTM. *Neural Computing and Applications*, Springer, v. 33, p. 17245–17256, 2021.

OLIVEIRA, J. F. de; SILVA, E. G.; DE MATTOS NETO, P. S. A hybrid system based on dynamic selection for time series forecasting. *IEEE Transactions on Neural Networks and Learning Systems*, IEEE, 2021.

PARK, S.; YANG, J.-S. Interpretable deep learning LSTM model for intelligent economic decision-making. *Knowledge-Based Systems*, Elsevier, v. 248, p. 108907, 2022.

PERRONE, M. Improving regression estimates: Averaging methods for variance reduction with extensions to general convex measure optimization. *PhD Thesis, Brown University*, 1993.

PETROPOULOS, F.; MAKRIDAKIS, S. Forecasting the novel coronavirus COVID-19. *PloS One*, Public Library of Science San Francisco, CA USA, v. 15, n. 3, p. e0231236, 2020.

PETROPOULOS, F.; MAKRIDAKIS, S.; STYLIANOU, N. Covid-19: Forecasting confirmed cases and deaths with a simple time series model. *International Journal of Forecasting*, Elsevier, v. 38, n. 2, p. 439–452, 2022.

POHLERT, T. *trend: Non-Parametric Trend Tests and Change-Point Detection*. [S.l.], 2020. R package version 1.1.4. Available at: <https://cran.r-project.org/web/packages/trend/index.html>.

PRILISTYA, S. K.; PERMANASARI, A. E.; FAUZIATI, S. Tourism demand time series forecasting: A systematic literature review. In: IEEE. *2020 12th International Conference on Information Technology and Electrical Engineering (ICITEE)*. [S.l.], 2020. p. 156–161.

PUAH, B. K.; CHONG, L. W.; WONG, Y. W.; BEGAM, K.; KHAN, N.; JUMAN, M. A.; RAJKUMAR, R. K. A regression unsupervised incremental learning algorithm for solar irradiance prediction. *Renewable Energy*, Elsevier, v. 164, p. 908–925, 2021.

QIU, H.; CAO, S.; XU, R. Cancer incidence, mortality, and burden in china: a time-trend analysis and comparison with the united states and united kingdom based on the global epidemiological data released in 2020. *Cancer Communications*, Wiley Online Library, v. 41, n. 10, p. 1037–1048, 2021.

QIU, Y.; ZHOU, J.; KHANDELWAL, M.; YANG, H.; YANG, P.; LI, C. Performance evaluation of hybrid WOA-XGBoost, GWO-XGBoost and BO-XGBoost models to predict blast-induced ground vibration. *Engineering with Computers*, Springer, v. 38, n. Suppl 5, p. 4145–4162, 2022.

RAHIMI, I.; CHEN, F.; GANDOMI, A. H. A review on COVID-19 forecasting models. *Neural Computing and Applications*, Springer, p. 1–11, 2021.

RAHMAN, M. M.; SHAKERI, M.; KHATUN, F.; TIONG, S. K.; ALKAHTANI, A. A.; SAMSUDIN, N. A.; AMIN, N.; PASUPULETI, J.; HASAN, M. K. A comprehensive study and performance analysis of deep neural network-based approaches in wind time-series forecasting. *Journal of Reliable Intelligent Environments*, Springer, v. 9, n. 2, p. 183–200, 2023.

RIBEIRO, G. H.; DE MATTOS NETO, P. S.; CAVALCANTI, G. D.; TSANG, R. Lag selection for time series forecasting using particle swarm optimization. In: IEEE. *The 2011 International Joint Conference on Neural Networks*. [S.l.], 2011. p. 2437–2444.

ROONEY, N.; PATTERSON, D.; ANAND, S.; TSYMBAL, A. Dynamic integration of regression models. In: SPRINGER. *Multiple Classifier Systems: 5th International Workshop, MCS 2004, Cagliari, Italy, June 9-11, 2004. Proceedings 5*. [S.l.], 2004. p. 164–173.

SAADALLAH, A.; TAVAKOL, M.; MORIK, K. An actor-critic ensemble aggregation model for time-series forecasting. In: IEEE. *2021 IEEE 37th International Conference on Data Engineering (ICDE)*. [S.l.], 2021. p. 2255–2260.

SAHA, S.; TANMOY, A. M.; TANNI, A. A.; GOSWAMI, S.; SIUM, S. M. A.; SAHA, S.; ISLAM, S.; HOODA, Y.; MALAKER, A. R.; ANIK, A. M. et al. New waves, new variants, old inequity: a continuing COVID-19 crisis. *BMJ Global Health*, BMJ Specialist Journals, v. 6, n. 8, p. e007031, 2021.

SALES, J. P. de; LIMA, J. E. de C.; FIRMINO, P. R. A.; DE MATTOS NETO, P. S. de. Oceanic niño index forecasting based on dynamic ensemble selection. Galoá, 2023.

SANTOS JUNIOR, D. S. d. O.; DE MATTOS NETO, P. S.; OLIVEIRA, J. F. de; SIQUEIRA, H. V.; BARCHI, T. M.; LIMA, A. R.; MADEIRO, F.; DANTAS, D. A.; CONVERTI, A.; PEREIRA, A. C. et al. Solar irradiance forecasting using dynamic ensemble selection. *Applied Sciences*, MDPI, v. 12, n. 7, p. 3510, 2022.

SAPANKEVYCH, N. I.; SANKAR, R. Time series prediction using support vector machines: a survey. *IEEE Computational Intelligence Magazine*, IEEE, v. 4, n. 2, p. 24–38, 2009.

SAPLIOĞLU, K.; GÜÇLÜ, Y. S. Combination of Wilcoxon test and scatter diagram for trend analysis of hydrological data. *Journal of Hydrology*, Elsevier, v. 612, p. 128132, 2022.

SCAFETTA, N.; WILLSON, R. C. Comparison of decadal trends among total solar irradiance composites of satellite observations. *Advances in Astronomy*, Hindawi Limited, v. 2019, p. 1–14, 2019.

SEN, P. K. Estimates of the regression coefficient based on Kendall's tau. *Journal of the American Statistical Association*, Taylor & Francis, v. 63, n. 324, p. 1379–1389, 1968.

SEZER, O. B.; GUDELEK, M. U.; OZBAYOGLU, A. M. Financial time series forecasting with deep learning: A systematic literature review: 2005–2019. *Applied Soft Computing*, Elsevier, v. 90, p. 106181, 2020.

SHALTOUT, M. Recent sea surface temperature trends and future scenarios for the red sea. *Oceanologia*, Elsevier, v. 61, n. 4, p. 484–504, 2019.

SILSO. *Sunspot Index and Long-term Solar Observations - Royal Observatory of Belgium*. 2024. <https://www.sidc.be/SILSO/datafiles>.

SILVA, C. A.; GUERRISI, G.; FRATE, F. D.; SANO, E. E. Near-real time deforestation detection in the brazilian amazon with sentinel-1 and neural networks. *European Journal of Remote Sensing*, Taylor & Francis, v. 55, n. 1, p. 129–149, 2022.

SILVA, E. G. *Uma Abordagem de Seleção Dinâmica de Preditores Baseada nas Janelas Temporais Mais Recentes*. Phd Thesis (PhD Thesis) — Centro de Informática - Universidade Federal de Recife, 2021.

SILVA, E. G.; CAVALCANTI, G. D.; OLIVEIRA, J. F. L. de; DE MATTOS NETO, P. S. On the evaluation of dynamic selection parameters for time series forecasting. In: IEEE. *2020 International Joint Conference on Neural Networks (IJCNN)*. [S.l.], 2020. p. 1–7.

SILVA, E. G.; DE MATTOS NETO, P. S.; CAVALCANTI, G. D. A dynamic predictor selection method based on recent temporal windows for time series forecasting. *IEEE Access*, IEEE, v. 9, p. 108466–108479, 2021.

SILVA, R. M. D.; SANTOS, C. A.; MOREIRA, M.; CORTE-REAL, J.; SILVA, V. C.; MEDEIROS, I. C. Rainfall and river flow trends using mann–kendall and sen's slope estimator statistical tests in the cobres river basin. *Natural Hazards*, Springer, v. 77, p. 1205–1221, 2015.

SILVA, T. A. *Estudo do desempenho da combinação de preditores baseados em cópulas e máquinas de vetor de suporte para séries temporais úteis ao desenvolvimento sustentável*. Master's Thesis (Master's Thesis) — Universidade Federal Rural de Pernambuc, 2020.

SIN, E.; WANG, L. Bitcoin price prediction using ensembles of neural networks. In: IEEE. *2017 13th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)*. [S.l.], 2017. p. 666–671.

SONG, G.; DAI, Q. A novel double deep elms ensemble system for time series forecasting. *Knowledge-Based Systems*, Elsevier, v. 134, p. 31–49, 2017.

SOUKHOVOLSKY, V.; KOVALEV, A.; PITT, A.; SHULMAN, K.; TARASOVA, O.; KESSEL, B. The cyclicity of coronavirus cases:"waves" and the "weekend effect". *Chaos, Solitons & Fractals*, Elsevier, v. 144, p. 110718, 2021.

SURAKHI, O. M.; ZAIDAN, M. A.; SERHAN, S.; SALAH, I.; HUSSEIN, T. An optimal stacked ensemble deep learning model for predicting time-series data using a genetic algorithm—an application for aerosol particle number concentrations. *Computers*, MDPI, v. 9, n. 4, p. 89, 2020.

SUTSKEVER, I.; VINYALS, O.; LE, Q. V. Sequence to sequence learning with neural networks. *Advances in Neural Information Processing Systems*, v. 27, 2014.

THEIL, H. A rank-invariant method of linear and polynomial regression analysis. *Indagationes Mathematicae*, v. 12, n. 85, p. 173, 1950.

TRINDADE, F. F.; FERNANDES, G. T.; NASCIMENTO, R. H. F.; JABBUR, I. F. G.; CARDOSO, A. de S. Epidemiological profile and trend analysis of HIV/AIDS/. *Journal Health NPEPS*, v. 4, n. 1, p. 153–165, 2019.

UEDA, N.; NAKANO, R. Generalization error of ensemble estimators. In: IEEE. *Proceedings of International Conference on Neural Networks (ICNN'96)*. [S.l.], 1996. v. 1, p. 90–95.

URAS, N.; MARCHESI, L.; MARCHESI, M.; TONELLI, R. Forecasting bitcoin closing price series using linear regression and neural networks models. *PeerJ Computer Science*, PeerJ Inc., v. 6, p. e279, 2020.

VALENTINI, G.; DIETTERICH, T. G. Bias-variance analysis of support vector machines for the development of SVM-based ensemble methods. *Journal of Machine Learning Research*, v. 5, n. Jul, p. 725–775, 2004.

VAPNIK, V. N. An overview of statistical learning theory. *IEEE Transactions on Neural Networks*, IEEE, v. 10, n. 5, p. 988–999, 1999.

VASWANI, A.; SHAZEER, N.; PARMAR, N.; USZKOREIT, J.; JONES, L.; GOMEZ, A. N.; KAISER, Ł.; POLOSUKHIN, I. Attention is all you need. *Advances in Neural Information Processing Systems*, v. 30, 2017.

WALMSLEY, F. N.; CAVALCANTI, G. D.; SABOURIN, R.; CRUZ, R. M. An investigation into the effects of label noise on Dynamic Selection algorithms. *Information Fusion*, Elsevier, v. 80, p. 104–120, 2022.

WANG, F.; SHAO, W.; YU, H.; KAN, G.; HE, X.; ZHANG, D.; REN, M.; WANG, G. Re-evaluation of the power of the mann-kendall test for detecting monotonic trends in hydrometeorological time series. *Frontiers in Earth Science*, Frontiers Media SA, v. 8, p. 14, 2020.

WANG, J.; LU, S.; WANG, S.-H.; ZHANG, Y.-D. A review on extreme learning machine. *Multimedia Tools and Applications*, Springer, v. 81, n. 29, p. 41611–41660, 2022.

WANG, X.; HYNDMAN, R. J.; LI, F.; KANG, Y. Forecast combinations: an over 50-year review. *International Journal of Forecasting*, Elsevier, 2022.

WATANABE, M.; DUFRESNE, J.-L.; KOSAKA, Y.; MAURITSEN, T.; TATEBE, H. Enhanced warming constrained by past trends in equatorial pacific sea surface temperature gradient. *Nature Climate Change*, Nature Publishing Group UK London, v. 11, n. 1, p. 33–37, 2021.

WIDODO, A.; BUDI, I. Model selection for time series forecasting using similarity measure. In: IEEE. *2011 International Conference on Advanced Computer Science and Information Systems*. [S.l.], 2011. p. 221–226.

WIDODO, A.; BUDI, I. Model selection of ensemble forecasting using weighted similarity of time series. *Jurnal Ilmu Komputer dan Informasi*, v. 5, n. 1, p. 40–49, 2012.

WINTERS, P. R. Forecasting sales by exponentially weighted moving averages. *Management Science*, v. 6, n. 3, p. 324–342, 1960.

WOODS, K.; KEGELMEYER, W. P.; BOWYER, K. Combination of multiple classifiers using local accuracy estimates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, IEEE, v. 19, n. 4, p. 405–410, 1997.

Yahoo Finance. *Yahoo Finance - Stock Market Live, Quotes, Business*. 2024. <https://finance.yahoo.com/>.

YANKOV, D.; DECOSTE, D.; KEOGH, E. Ensembles of nearest neighbor forecasts. In: SPRINGER. *Machine Learning: ECML 2006: 17th European Conference on Machine Learning Berlin, Germany, September 18-22, 2006 Proceedings 17*. [S.l.], 2006. p. 545–556.

YAO, C.; DAI, Q.; SONG, G. Several novel dynamic ensemble selection algorithms for time series prediction. *Neural Processing Letters*, Springer, v. 50, p. 1789–1829, 2019.

YIN, X.; LI, J. Financial market trend analysis based on autoregressive conditional heteroscedasticity model and bp neural network prediction. *The International Journal of Electrical Engineering & Education*, SAGE Publications Sage UK: London, England, p. 0020720920988497, 2021.

YU, J.; AMORES, J.; SEBE, N.; TIAN, Q. A new study on distance metrics as similarity measurement. In: IEEE. *2006 IEEE International Conference on Multimedia and Expo*. [S.l.], 2006. p. 533–536.

YU, Y.; SI, X.; HU, C.; ZHANG, J. A review of recurrent neural networks: LSTM cells and network architectures. *Neural Computation*, v. 31, n. 7, p. 1235–1270, 2019.

YULIANTI, L. P.; TRISETYARSO, A.; SANTOSO, J.; SURENDRO, K. Comparison of distance metrics for generating cluster-based ensemble learning. In: *Proceedings of the 2023 12th International Conference on Software and Computer Applications*. [S.l.: s.n.], 2023. p. 26–33.

ZAIDAN, M. A.; SURAKHI, O.; FUNG, P. L.; HUSSEIN, T. Sensitivity analysis for predicting sub-micron aerosol concentrations based on meteorological parameters. *Sensors*, MDPI, v. 20, n. 10, p. 2876, 2020.

ZAINI, N.; EAN, L. W.; AHMED, A. N.; MALEK, M. A. A systematic literature review of deep learning neural network for time series air quality forecasting. *Environmental Science and Pollution Research*, Springer, p. 1–33, 2022.

ZENG, A.; CHEN, M.; ZHANG, L.; XU, Q. Are transformers effective for time series forecasting? In: *Proceedings of the AAAI Conference on Artificial Intelligence*. [S.l.: s.n.], 2023. v. 37, n. 9, p. 11121–11128.

ZHA, W.; LIU, Y.; WAN, Y.; LUO, R.; LI, D.; YANG, S.; XU, Y. Forecasting monthly gas field production based on the CNN-LSTM model. *Energy*, Elsevier, p. 124889, 2022.

ZHAO, X.; LI, K.; ANG, C. K. E.; CHEONG, K. H. A deep learning based hybrid architecture for weekly dengue incidences forecasting. *Chaos, Solitons & Fractals*, Elsevier, v. 168, p. 113170, 2023.

ZIVOT, E.; WANG, J.; ZIVOT, E.; WANG, J. Rolling analysis of time series. *Modeling financial time series with S-Plus®*, Springer, p. 299–346, 2003.

ZYBLEWSKI, P.; SABOURIN, R.; WOŹNIAK, M. Preprocessed dynamic classifier ensemble selection for highly imbalanced drifted data streams. *Information Fusion*, Elsevier, v. 66, p. 138–154, 2021.