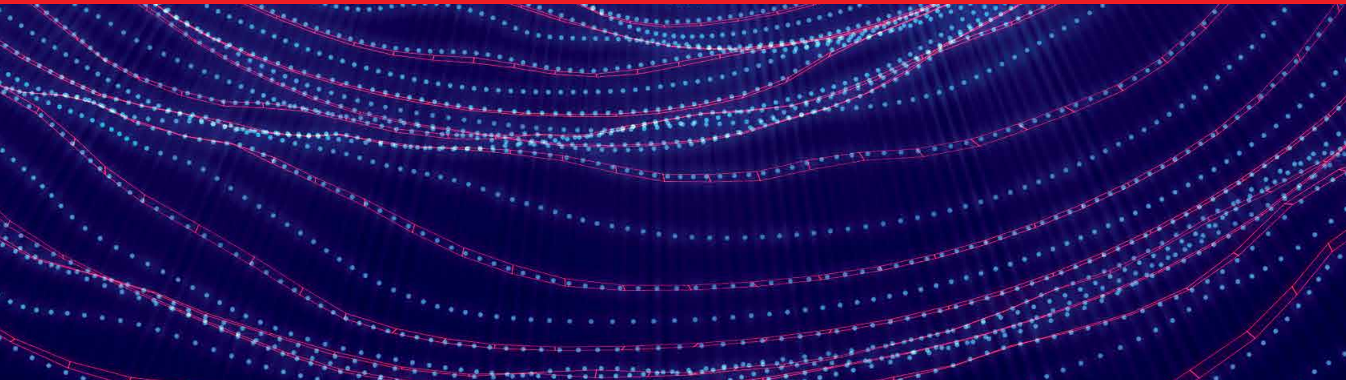
IntechOpen

Time Series Analysis

Recent Advances, New Perspectives and Applications

Edited by Jorge Rocha,
Sandra Oliveira and Cláudia M. Viana



Time Series Analysis - Recent Advances, New Perspectives and Applications

Edited by Jorge Rocha,
Sandra Oliveira and Cláudia M. Viana

Published in London, United Kingdom

Time Series Analysis - Recent Advances, New Perspectives and Applications
http://dx.doi.org/10.5772/intechopen.111223
Edited by Jorge Rocha, Sandra Oliveira and Cláudia M. Viana

Contributors
Alain Bourdier, Alicia Palacios-Orueta, Aulia Keiko Hubbansyah, Beatriz Arid Rudnick, Chris Aldrich, Cira Perna, Cláudia M. Viana, César Sáenz, Damaris Kirsch Pinheiro, Eduardo Cabal-Yepez, Efrain Ramirez-Velasco, Francisco Javier Villalobos-Pina, Hassan Bencherif, Ivan Sudibyo, Javier Litago, Jay Darji, Johann Faouzi, Jorge Rocha, Josue Augusto Reyes-Malanche, João Victor de Menezes Reichert, Klaus Wiese, Laura Recuero, Lawrence D. Jones, Michele La Rocca, Moacir Fernandes de Godoy, Nupur Biswas, Olivier Delage, René Tato Loua, Rob G. Sacco, Sandra Oliveira, Shashaanka Ashili, Thierry Portafaix, Toru Yazawa, Víctor Cicuéndez, Zaafri Ananto Husodo, Zhaoxing Gao

© The Editor(s) and the Author(s) 2024

The rights of the editor(s) and the author(s) have been asserted in accordance with the Copyright, Designs and Patents Act 1988. All rights to the book as a whole are reserved by INTECHOPEN LIMITED. The book as a whole (compilation) cannot be reproduced, distributed or used for commercial or non-commercial purposes without INTECHOPEN LIMITED's written permission. Enquiries concerning the use of the book should be directed to INTECHOPEN LIMITED rights and permissions department (permissions@intechopen.com).

Violations are liable to prosecution under the governing Copyright Law.

Individual chapters of this publication are distributed under the terms of the Creative Commons Attribution 3.0 Unported License which permits commercial use, distribution and reproduction of the individual chapters, provided the original author(s) and source publication are appropriately acknowledged. If so indicated, certain images may not be included under the Creative Commons license. In such cases users will need to obtain permission from the license holder to reproduce the material. More details and guidelines concerning content reuse and adaptation can be found at http://www.intechopen.com/copyright-policy.html.

Notice

Statements and opinions expressed in the chapters are these of the individual contributors and not necessarily those of the editors or publisher. No responsibility is accepted for the accuracy of information contained in the published chapters. The publisher assumes no responsibility for any damage or injury to persons or property arising out of the use of any materials, instructions, methods or ideas contained in the book.

First published in London, United Kingdom, 2024 by IntechOpen
IntechOpen is the global imprint of INTECHOPEN LIMITED, registered in England and Wales, registration number: 11086078, 167-169 Great Portland Street, London, W1W 5PF, United Kingdom

British Library Cataloguing-in-Publication Data
A catalogue record for this book is available from the British Library

Additional hard and PDF copies can be obtained from orders@intechopen.com

Time Series Analysis - Recent Advances, New Perspectives and Applications
Edited by Jorge Rocha, Sandra Oliveira and Cláudia M. Viana
p. cm.
Print ISBN 978-0-85466-053-7
Online ISBN 978-0-85466-052-0
eBook (PDF) ISBN 978-0-85466-054-4

Meet the editors

Jorge Rocha is a geographer with an MSc in Geographic Information Systems, an MSc in Spatial Planning, and a Ph.D. in Geographic Information Science. Currently, he is an associate professor at the Institute of Geography and Spatial Planning and a member of the Modelling, Urban and Regional Planning, and Environmental Hazard and Risk Assessment and Management research groups of the Centre for Geographical Studies, all from the University of Lisbon (ULisboa), Portugal. His field of expertise is geosimulation and geocomputation involving artificial neural networks, graphs theory, cellular automata, and multi-agent systems. Dr. Rocha's works are quite diverse and include urban morphology, remote sensing, epidemiology, health geography, geomarketing, tourism, smart cities, and big data.

Sandra Oliveira is a researcher at the Centre for Geographical Studies, Institute of Geography and Spatial Planning, at the University of Lisbon. She graduated with a degree in geography from the University of Coimbra, Portugal. She has an MSc in Environmental Sciences from the University of Wollongong, Australia and a Ph.D. in Forest and Natural Resources Engineering from the University of Lisbon (ULisboa), Portugal. She was a consultant for Bioversity International in Ghana, working in biodiversity in cocoa agroforests, and a scientific officer at the Joint Research Centre of the European Commission, working in wildfires and forest protection in Europe. She works in environmental risk assessment and geographic information systems and her main interests are in climatic risks, wildfires, forest protection, and biodiversity.

Cláudia M. Viana is a geospatial data scientist and geographer who obtained a Ph.D. in Geography (2022) from the University of Lisbon (ULisboa), Portugal in 2022 with a grant from the Portuguese Foundation for Science and Technology (FCT). She earned a degree in geography and a master's in geographic information systems from the Institute of Geography and Territorial Planning, ULisboa, in 2012 and 2014, respectively. Currently, she is a post-doc junior researcher at the Centre for Geographical Studies and a member of various research groups, including Modelling and Spatial Planning, Associated Laboratory TERRA, both from ULisboa and CEIS20 from the University of Coimbra, Portugal. Dr. Viana serves as the principal investigator of the AgroecoDecipher project (2022.09372.PTDC) supported by FCT.

Contents

Preface **XIII**

Section 1
Foundations and Advancements in Time Series Analysis **1**

Chapter 1 **3**
Introductory Chapter: Time Series Analysis
by Cláudia M. Viana, Sandra Oliveira and Jorge Rocha

Chapter 2 **15**
Time Series Classification: A Review of Algorithms and Implementations
by Johann Faouzi

Section 2
Biomedical Time Series Analysis **49**

Chapter 3 **51**
A Heartbeat-Interval Time Series Analysis, Modified Detrended Fluctuation Analysis, mDFA, Distinguishes between Stressed- and Happy-Heartbeats: From Invertebrate Animals to Humans
by Toru Yazawa

Chapter 4 **75**
Use of the Gini Coefficient for the Analysis of Heart Rate Variability in Sick and Healthy Individuals
by Moacir Fernandes de Godoy, Beatriz Arid Rudnick and João Victor de Menezes Reichert

Chapter 5 **85**
Handling Missing Data in the Time-Series Data from Wearables
by Jay Darji, Nupur Biswas, Lawrence D. Jones and Shashaanka Ashili

Section 3
Methodological Approaches in Time Series Analysis **101**

Chapter 6 **103**
Variability Analysis of Observational Time Series: An Overview of the Decomposition Methods for Non-Stationary and Noisy Signals
by Olivier Delage, Hassan Bencherif, Thierry Portafaix, Alain Bourdier, René Tato Loua and Damaris Kirsch Pinheiro

Chapter 7 **129**
Segmentation of High-Dimensional Matrix-Variate Time Series
by Zhaoxing Gao

Chapter 8 **145**
Neural Networks in Nonlinear Time Series: A Subsampling Model Selection Procedure
by Michele La Rocca and Cira Perna

Section 4
Environmental Time Series Analysis **165**

Chapter 9 **167**
Analysis and Modeling of Rainfed Crops Dynamics Based on NDVI Time Series in Central Spain
by César Sáenz, Víctor Cicuéndez, Laura Recuero, Klaus Wiese, Alicia Palacios-Orueta and Javier Litago

Section 5
Financial and Economic Time Series Analysis **185**

Chapter 10 **187**
Momentum Periods of Feedback Trading toward Exchange Rate Volatility in ASEAN Countries
by Ivan Sudibyo, Zaafri Ananto Husodo and Aulia Keiko Hubbansyah

Section 6
Engineering and Technological Applications of Time Series Analysis **219**

Chapter 11 **221**
Electric Fault Diagnosis in Induction Machines Using Motor Current Signature Analysis (MCSA)
by Francisco Javier Villalobos-Pina, Josue Augusto Reyes-Malanche, Eduardo Cabal-Yepez and Efrain Ramirez-Velasco

Chapter 12 **241**
A Comparative Analysis of Image Encoding of Time Series for Anomaly Detection
by Chris Aldrich

Section 7
Time Series Analysis in Cultural Contexts **269**

Chapter 13 271
Fractal Dynamics and Fibonacci Sequences: A Time Series Analysis of Cultural Attractor Landscapes
by Rob G. Sacco

Preface

Welcome to *Time Series Analysis – Recent Advances, New Perspectives and Applications*. The study of time series data has evolved into a dynamic and indispensable field with profound implications across various disciplines. In this book, we embark on a comprehensive journey through recent advancements, innovative perspectives, and practical applications in time series analysis.

In today's big data era, we live in a data-driven world and the ability to extract meaningful insights from temporal data is more crucial than ever. Time series analysis serves as a powerful toolkit for uncovering patterns, revealing processes, predicting trends, and making informed decisions in diverse domains. From biomedical research to environmental monitoring, from financial markets to cultural studies, time series analysis offers invaluable tools and methodologies for understanding complex temporal phenomena.

This volume brings together a varied group of contributions from leading experts in the field, each offering unique insights and expertise in their respective areas. We begin by exploring the latest advancements in analyzing biomedical data, with chapters dedicated to topics such as heart rate variability analysis, stress detection, and health monitoring using innovative time series techniques.

Moving forward, we delve into methodological approaches that push the boundaries of traditional time series analysis. From advanced decomposition methods to sophisticated segmentation techniques, these chapters highlight the latest developments in extracting meaningful information from complex time-dependent data.

The book then takes a multidisciplinary approach, exploring the applications of time series analysis in environmental sciences, economics, finance, and engineering. Whether it is modeling ecological dynamics, forecasting market trends, or diagnosing faults in industrial machinery, time series analysis provides indispensable tools for systems understanding.

Finally, we devote a section to the cultural dimensions of time series analysis, exploring how it can offer unique insights into cultural patterns, artistic movements, and societal dynamics. From analyzing cultural attractor landscapes to uncovering hidden temporal structures in artistic expressions, the chapter highlights the intersection of time series analysis with the humanities and social sciences.

We hope that this book serves as a valuable resource for researchers, specialists, and students alike, providing a comprehensive overview of recent advances, new perspectives, and exciting applications in time series analysis. Whether you are an expert looking to stay in touch with the latest developments or an enthusiast willing to explore the fascinating world of time-varying data, we invite you to join us on this intellectual journey.

Jorge Rocha, Cláudia M. Viana and Sandra Oliveira
Institute of Geography and Spatial Planning,
University of Lisbon,
Lisbon, Portugal

Associate Laboratory TERRA,
Lisbon, Portugal

Section 1

Foundations and Advancements in Time Series Analysis

Chapter 1

Introductory Chapter: Time Series Analysis

Cláudia M. Viana, Sandra Oliveira and Jorge Rocha

1. Introduction

Time series, defined as sequentially observed data points over time [1], find applications across diverse domains such as economics and engineering. The statistical analysis of time series is crucial, and Chatfield's taxonomy identifies six main categories: Economic and Financial Time Series, Physical Time Series, Marketing Time Series, Process Control Data, Binary Processes, and Point Processes.

To effectively categorize time series, consideration of features like seasonality, trend, and outliers is essential [1]. Seasonality reflects recurring patterns over time intervals, while trend represents a systematic linear or nonlinear component. Outliers are observations distant from others, often indicating anomalies. The categorization and analysis of time series are pivotal for drawing meaningful inferences from the diverse structures encountered in engineering, science, sociology, and economics [2].

The objectives of time series analysis encompass description, explanation, prediction, and control. Description involves plotting observations over time to reveal patterns, while explanation explores relationships between variables. Prediction focuses on forecasting future values, and control utilizes time series to enhance control over physical or economic systems.

Possible applications span from land use-cover [3, 4] and agriculture changes [5, 6], tourism [7, 8], socioeconomic vulnerability [9], epidemiology [10], and health [11]. This chapter delves into advanced approaches for time series analysis.

2. Time series analysis

Time Series Analysis (TSA) involves constructing predictive models that generate a target variable or label based on sequential observations across a defined period, that is, data that is time-dependent. The analysis of time series involves studying the relationships between variables that change over time [12].

There are two types of time series: deterministic and stochastic. Deterministic time series can be predicted with certainty based on previous experience, while stochastic time series have random fluctuations [13]. Time series analysis is widely used in various fields such as economics, finance, and health research [12, 14]. It helps in identifying patterns, forecasting future values, and understanding the underlying dynamics of the data [14–16]. As such, it has applications in various domains and is important for making inferences about the future based on past observations [12, 13, 15, 17, 18].

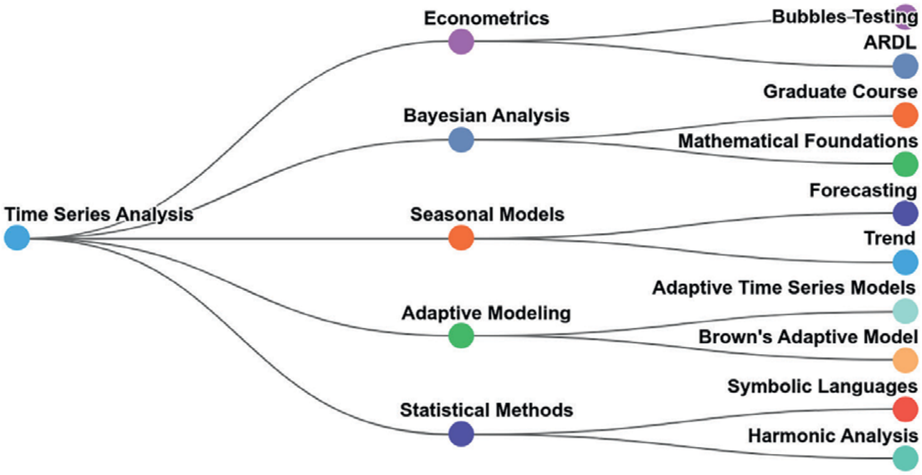


Figure 1.
Different approaches in time series analysis.

To perform time series analysis, it is important to convert nonstationary data into stationary data using techniques like differencing [13]. Time series data, common in various fields, present unique challenges due to random noise and interdependencies between measurements at different time points. Autocorrelations and partial autocorrelations of the series can indicate the degree of correlation between each point and earlier values in the series [18]. Autocorrelation measures the correlation between a data point and its lagged values, while partial autocorrelation measures the correlation between a data point and its lagged values after removing the effects of intermediate lags. Recent developments have led to prominent approaches (**Figure 1**).

2.1 Forecasting

Forecasting entails predicting unseen values within an observed time series, crucial in domains like economics and production planning. Despite a plethora of forecasting methods, challenges persist in achieving satisfactory generalization capabilities [19]. There is still the need to choose the most suitable methodology assuming a set of preconditions. Being a form of extrapolation, forecasting stands all the risks of it. The forecasting horizon introduces risks of error escalation, requiring careful model adaptation based on incoming information.

2.2 Anomaly detection

Anomaly Detection, synonymous with outlier or novelty detection, identifies abnormal data within a dataset [20]. Anomalies signify rare events, prompting critical actions in diverse domains such as network security and healthcare [21]. Anomalies can be categorized as point anomalies (deviations from normal patterns), contextual anomalies (anomalies in specific environments), and collective anomalies (erratic behavior in a group of similar data instances) [22].

2.3 Case-based reasoning

Case-based Reasoning (CBR) replicates human problem-solving by drawing comparisons between previously solved cases and applying similar solutions to new cases. It relies on specific knowledge and maintains a case base for reference, contributing to problem-solving and learning [23].

2.4 Bayesian optimization

Bayesian Optimization addresses global optimization problems by iteratively developing a statistical model of the unknown objective function [24]. It balances exploitation and exploration using an acquisition function, with Expected Improvement (EI) commonly employed [25]. This approach proves beneficial when dealing with black-box functions and limited samples [26].

2.5 Competitor analysis

Competitor Analysis in time series analysis considers frameworks aiding in forecasting and anomaly detection, ranging from to meteorological readings to Internet of Things (IoT) data [27]. Understanding competitors in these areas involves examining tools, methodologies, and their applicability to specific domains. Naturally, to have a superior understand of trending, anomaly occurrences, and forecasting, one has to perform analysis on these data structures. This process can be smoothed with the help of frameworks that provide the researcher with tools to perform forecasting and anomaly detection. In addition, comparative summaries provide insights into the strengths and weaknesses of each competitor.

3. Advancements in time series analysis

The latest advancements in TSA have focused on addressing the challenges posed by the increasing amount of data and the need for more efficient algorithms for accurate forecasting. These advancements offer more efficient ways of analyzing and predicting data compared to traditional approaches [28]. These advancements have been driven by the demand for accurate forecasting and decision-making in various domains such as finance, healthcare, and environmental protection [29, 30].

One can emphasize that new technologies and smarter algorithms are being developed to analyze large time series collections [28]. Likewise, the development of Python packages like tsfresh has accelerated the process of feature extraction from time series data [31]. There are also some emerging applications in various scenarios, such as subject theme evolution, academic influence evaluation, network sentiment analysis, and technology trend analysis [32].

New methods in time series analysis offer more efficient and accurate ways of analyzing and predicting data compared to traditional approaches [27]. The use of advanced techniques, such as nonlinear time series analysis, has improved the ability to model and predict complex systems [33].

However, implementing new methods can be challenging, and the impact on forecasting accuracy and model performance depends on the specific techniques used. For example, the systematic framework proposed in TsP-SA enables qualitative comparison and assessment of different time series prediction techniques, leading to improved forecasting accuracy [34].

4. Persisting issues in TSA

There are some common challenges and limitations in time series analysis. These include dealing with nonstationarity, missing data, outliers, and the curse of dimensionality [13, 14]. Nonstationarity refers to the situation where the statistical properties of a time series change over time, making it difficult to model and forecast accurately [13]. The selection and evaluation of the most suitable method for a specific task can be challenging due to the lack of comprehensive categorization and comparison of techniques [34].

In terms of advancements and trends in time series analysis, there is ongoing research in developing more efficient algorithms and techniques to handle large-scale time series data [28]. Additionally, there is a focus on integrating time series analysis with other data mining and machine learning (ML) techniques to improve prediction accuracy and discover hidden patterns [35].

Time series analysis is a valuable tool for exploring, analyzing, and forecasting data indexed over time. It involves concepts such as autocorrelation, ARIMA (Autoregressive Integrated Moving Average) models, and stochastic volatility models. ARIMA models are commonly used in time series analysis. They combine autoregressive (AR), moving average (MA), and differencing components to model the trend, seasonality, and noise in a time series [14]. In the other hand, stochastic volatility models aim to model the change over time in the variability or volatility of a time series. These models are particularly useful in finance and economics to capture the volatility clustering phenomenon observed in financial markets [14].

TSA finds applications in finance and economics, helping in predicting future values and analyzing the effects of economic policies in market forecasting, risk management, and customer requirement analysis [36]. However, it also faces challenges such as nonstationarity and missing data. Ongoing advancements include the development of more efficient algorithms and the integration of time series analysis with other techniques for improved prediction and pattern discovery.

5. Conclusion

The state-of-the-art in time series analysis shows the prevalence of ML approaches, demonstrating excellent results in forecasting and anomaly detection. While classical models maintain relevance, Bayesian Optimization enhances the quality of results, as evidenced in specific scenarios. The discussion underscores the continued importance of classical models and the evolving role of ML in time series analysis.

The demand for time series forecasting spans various challenging domains and data analytics issues. This process involves utilizing models to interpret past sequences of values at evenly spaced intervals to predict subsequent values along the same time axis. Initially, methods like Holt-Winters [37] and later Box and Jenkins [38] laid the foundation. Subsequently, statistical models such as ARIMA, X11ARIMA, X12ARIMA [39, 40], Seasonal Autoregressive Integrated Moving Average (SARIMA) [38], Seasonal

Autoregressive Integrated Moving Average with Exogenous factors (SARIMAX), and Support Vector Machine (SVM) [41] were introduced to this domain.

However, a limitation arises with these models in discerning exogenous input features [42]. Additionally, Autoregressive Moving Average (ARMA) models' linear basis poses challenges in learning and predicting the nonlinear dynamics of time series [43]. Nonlinear machine learning approaches such as kernel methods [44], Gaussian processes [45], and ensemble methods [46] have shown significant improvements but still lack the ability to assimilate true nonlinear relationships.

Neural networks, on the other hand, offer promising solutions. Multi-Layer Perceptron (MLP) was among the first neural networks used for time series forecasting, followed by recurrent networks [47]. Long Short Term Memory (LSTM) networks have significantly improved accuracy by incorporating information from long-term dependencies [48]. Fuzzy approaches and fuzzy neural networks have also been explored [49].

In recent years, deep learning approaches, particularly in domains like natural language processing, have inspired algorithms applicable to time series forecasting [50]. Attention-based architectures [51], which excel in sequence prediction tasks [52, 53], hold promise in this domain. Most recent deep learning architectures have been Vanilla Long Short Term Memory (V-LSTM) [54], Gated Recurrent Unit (GRU) [55], Bidirectional LSTM (BD-LSTM) [56], Autoencoder LSTM (AELSTM) [57], Convolutional Neural Networks combined with LSTM (CNN-LSTM) [58], Attention Mechanism Network [59], and the Transformer network [60].

Acknowledgements

We acknowledge GEOMODLAB (Laboratory for Remote Sensing, Geographical Analysis, and Modeling) of the Center of Geographical Studies/IGOT for providing the required equipment and software. This research was supported by the Portuguese Foundation for Science and Technology (FCT) under grant number 2022.09372.PTDC and grant number 2022.05015.PTDC and Centre for Geographical Studies—University of Lisbon and FCT under grant numbers UIDB/00295/2020 and UIDP/00295/2020.

Author details

Cláudia M. Viana[1,2], Sandra Oliveira[1,2] and Jorge Rocha[1,2*]

1 Institute of Geography and Spatial Planning, University of Lisbon, Lisbon, Portugal

2 Associate Laboratory TERRA, Lisbon, Portugal

*Address all correspondence to: jorge.rocha@edu.ulisboa.pt

References

[1] Chatfield C, Xing H. The Analysis of Time Series: An Introduction with R. Boca Raton: CRC Press; 2019

[2] Brockwell PJ, Davis RA. Introduction to Time Series and Forecasting. New York: Springer; 2002

[3] Viana CM, Girão I, Rocha J. Long-term satellite image time-series for land use/land cover change detection using refined open source data in a rural region. Remote Sensing [Internet]. 2019;**11**(9):1104. Available from: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85065724909&doi=10.3390%2Frs11091104&partnerID=40&md5=5c5a2eddfb9361b48d49287661087add

[4] Viana CM, Pontius RG, Rocha J. Four fundamental questions to evaluate land change models with an illustration of a cellular automata–markov model. Annals of the American Association of Geographers. 2023;**113**(10):2497-2511. DOI: 10.1080/24694452.2023.2232435

[5] Viana CM, Freire D, Abrantes P, Rocha J. Evolution of agricultural production in Portugal during 1850-2018: A geographical and historical perspective. Land [Internet]. 2021;**10**(8):776. Available from: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85111691871&doi=10.3390%2Fland10080776&partnerID=40&md5=ec428276f8a03f807833dea6dbbc0293

[6] Ribeiro C, Viana CM, Girão I, Figueiredo E, Rocha J. The spatiotemporal links between urban and rural regions through the sale and consumption of agri-food products. Sustainability [Internet]. 2023;**15**(15):12038. Available from: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85167813491&doi=10.3390%2Fsu151512038&partnerID=40&md5=0af3506bc51251a07068fd08d456d190

[7] Encalada-Abarca L, Ferreira CC, Rocha J. Measuring tourism intensification in urban destinations: An approach based on fractal analysis. Journal of Travel Research [Internet]. 2022;**61**(2):394-413. Available from: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85100040880&doi=10.1177%2F0047287520987627&partnerID=40&md5=a604838e099f4c76e92281e88b824ddd

[8] Encalada-Abarca L, Ferreira CC, Rocha J. Revisiting city tourism in the longer run: An exploratory analysis based on LBSN data. Current Issues in Tourism [Internet]. 2024;**27**(4):584-599. Available from: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85150605889&doi=10.1080%2F13683500.2023.2182669&partnerID=40&md5=dbee951a47278cefa150b41d16fb1794

[9] Santos PP, Zêzere JL, Pereira S, Rocha J, Tavares AO. A novel approach to measuring spatiotemporal changes in social vulnerability at the local level in Portugal. International Journal of Disaster Risk Science [Internet]. 2022;**13**(6):842-861. Available from: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85142850511&doi=10.1007%2Fs13753-022-00455-w&partnerID=40&md5=bbf27704a0e918712cc0198d1cac523f

[10] Oliveira S, Capinha C, Rocha J. Predicting the time of arrival of the Tiger mosquito (Aedes albopictus) to new countries based on trade patterns of tyres and plants. Journal of Applied Ecology [Internet]. 2023;**60**(11):2362-2374. Available from: https://www.scopus.

com/inward/record.uri?eid=2-s2.0-85170397762&doi=10.1111%2F1365-2664.14503&partnerID=40&md5=7be08cde73fda1d61d51ab621993709b

[11] Silva M, Betco I, Capinha C, Roquette R, Viana CM, Rocha J. Spatiotemporal Dynamics of COVID-19 Infections in Mainland Portugal. Sustainability. 2022;**14**

[12] León-Álvarez AL, Betancur-Gómez JI, Jaimes-Barragán F, Grisales-Romero H. Clinical and epidemiological rounds. Time series. Iatreia [Internet]. 2016;**29**(3):373-381. Available from: https://www.scopus.com/inward/record.uri?eid=2-s2.0-84977585305&doi=10.17533%2Fudea.iatreia.v29n3a12&partnerID=40&md5=6432935ac8123d424bbfac4e3b24b84e

[13] Mathelinea D, Chandrashekar R, Mawengkang H. Stationarity test for medicine time series data. AIP Conference Proceedings [Internet]. 2023;**2714**(1):30049. Available from. DOI: 10.1063/5.0128444

[14] De Jong P. Time series analysis. In: Frees EW, Derrig RA, Meyers G, editors. Predictive Modeling Applications in Actuarial Science. Cambridge: Cambridge University Press; 2014. pp. 427-448. Available from: https://www.scopus.com/inward/record.uri?eid=2-s2.0-84953865079&doi=10.1017%2FCBO9781139342674.017&partnerID=40&md5=3262f51f0506452273100f26b3e4fbd8

[15] Žáček M. Introduction to time series. In: Volna E, Kotyrba M, Janosek M, editors. Pattern Recognition and Classification in Time Series Data. Hershey, PA: IGI Global; 2017. pp. 32-52. DOI: 10.4018/978-1-5225-0565-5.ch002

[16] Kulp CW, Niskala BJ. Characterization of time series data. In: Skiadas CH, Skiadas C, editors. Handbook of Applications of Chaos Theory [Internet]. New York: Chapman and Hall/CRC; 2017. pp. 211-230. Available from: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85052770091&doi=10.1201%2Fb20232&partnerID=40&md5=47dae08ebe64bb0bb6195ea52938bb39

[17] Ivanović M, Kurbalija V. Time series analysis and possible applications. In: Biljanovic P, Butkovic Z, Skala K, Grbac TC, Cicin-Sain M, Sruk V, et al., editors. 39th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO). Opatija, Croatia: IEEE; May 30 - June 3, 2016. pp. 473-479. DOI: 10.1109/MIPRO.2016.7522190

[18] Chattopadhyay AK, Chattopadhyay T. Time series analysis. In: Statistical Methods for Astronomical Data Analysis. Springer Series in Astrostatistics. Vol. 3. New York, NY: Springer; 2014. DOI: 10.1007/978-1-4939-1507-1_9

[19] Makridakis S, Spiliotis E, Assimakopoulos V. The M4 competition: Results, findings, conclusion and way forward. International Journal of Forecasting. 2018;**34**(4):802-808

[20] Ahmed M, Naser Mahmood A, Hu J. A survey of network anomaly detection techniques. Journal of Network and Computer Applications [Internet]. 2016;**60**:19-31. Available from: https://www.sciencedirect.com/science/article/pii/S1084804515002891

[21] Mason AC. Artificial Intelligence Cybersecurity Threats: Determining Strategy and Decision-Making Effects. Ann Arbor, Michigan: ProQuest, Northcentral University; 2020

[22] Ahmed M, Mahmood AN, Islam MR. A survey of anomaly detection

techniques in financial domain. Future Generation Computer Systems. 2016;**55**:278-288

[23] Kolodner JL. An introduction to case-based reasoning. Artificial Intelligence Review [Internet]. 1992;**6**(1):3-34. DOI: 10.1007/BF00155578

[24] Mockus J. The Bayesian approach to local optimization BT. In: Mockus J, editor. Bayesian Approach to Global Optimization. Mathematics and Its Applications. Vol. 37. Springer, Dordrecht: Springer Netherlands; 1989. pp. 125-156. DOI: 10.1007/978-94-009-0909-0_7

[25] Mockus J, Tiesis V, Zilinskas A. The application of {B}ayesian methods for seeking the extremum. Towards Global Optimization. 1978;**2**(117-129):2

[26] Brochu E, Cora VM, de Freitas N. A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. ArXiv [Internet]. 2010;abs/1012.2599. Available from: https://api.semanticscholar.org/CorpusID:1640103

[27] Song YX. Time series analysis process of dynamic data in internet of things system. Journal of Physics: Conference Series [Internet]. 2021;**1856**(1):12010. DOI: 10.1088/1742-6596/1856/1/012010

[28] Palpanas T, Beckmann V. Report on the first and second interdisciplinary time series analysis workshop (ITISA). SIGMOD Record [Internet]. 2019;**48**(3):36-40. Available from: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85077393353&doi=10.1145%2F3377391.3377400&partnerID=40&md5=cfbd117ea2ff1a82bd02387432495816

[29] Kapila Tharanga Rathnayaka RM, Seneviratne DMKN, Jianguo W, Arumawadu HI. A hybrid statistical approach for stock market forecasting based on artificial neural network and ARIMA time series models. In: 2015 International Conference on Behavioral, Economic and Socio-Cultural Computing (BESC). Nanjing, China; 2015. pp. 54-60. DOI: 10.1109/BESC.2015.7365958

[30] Struckov A, Yufa S, Visheratin AA, Nasonov D. Evaluation of modern tools and techniques for storing time-series data. Procedia Computer Science. 2019:19-28. DOI: 10.1016/j.procs.2019.08.125

[31] Christ M, Braun N, Neuffer J, Kempa-Liehr AW. Time series FeatuRe extraction on basis of scalable hypothesis tests (tsfresh – A python package). Neurocomputing [Internet]. 2018;**307**:72-77. Available from: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85047079994&doi=10.1016%2Fj.neucom.2018.03.067&partnerID=40&md5=188fb4f1101d239a1429d804fdcee66b

[32] Chen G, Wang K. Current advances of time series analysis in information science: Tasks, processes and problems. Documentation, Informaiton & Knowledge [Internet]. 2023;**40**(6):89-97. Available from: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85179605202&doi=10.13366%2Fj.dik.2023.06.089&partnerID=40&md5=eb6d51a1a43b84aacdd4a213d226b9eb

[33] Xiong O, Li S. Methods of nonlinear time series cycle analysis in big data environment and IoT application. Wireless Communications and Mobile Computing [Internet]. 2022:8. Available from: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85138422494&doi=10.1155%2F2022%2F6127601&partnerID=40&md5=9f3a2023a3d455347cf82fc5f0359eef

[34] Mehrmolaei S, Keyvanpour MR. TsP-SA: Usage of time series techniques

on healthcare data. International Journal of Electronic Healthcare [Internet]. 2018;**10**(3):190-230. Available from: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85051719544&doi=10.1504%2FIJEH.2018.094044&partnerID=40&md5=fa849ead3770e7692dbdc2a7047c3276

[35] Klepac G, Kopal R, Mršić L. REFII model as a base for data mining techniques hybridization with purpose of time series pattern recognition. Studies in Computational Intelligence [Internet]. 2016;**611**:237-270. Available from: https://www.scopus.com/inward/record.uri?eid=2-s2.0-84940179961&doi=10.1007%2F978-81-322-2544-7_8&partnerID=40&md5=f604cc91d246104bdda35f06be196795

[36] Wang S. Research on data mining and investment recommendation of individual users based on financial time series analysis. International Journal of Data Warehousing and Mining [Internet]. 2020;**16**(2):64-80. Available from: https://ideas.repec.org/a/igg/jdwm00/v16y2020i2p64-80.html

[37] Winters PR. Forecasting sales by exponentially weighted moving averages. Management Science. 1960;**6**(3):324-342

[38] Box GEP, Jenkins GM, Reinsel GC, Ljung GM. Time Series Analysis: Forecasting and Control. Hoboken, New Jersey: John Wiley & Sons; 2015

[39] Shiskin J. The X-11 Variant of the Census Method II Seasonal Adjustment Program. US Department of Commerce, Bureau of the Census; 1967. Available from: https://www.census.gov/library/working-papers/1967/adrm/shiskin-01.html

[40] Dagum EB. A new method to reduce unwanted ripples and revisions in trend-cycle estimates from X-11-ARIMA. Survey Methodology. 1996;**22**:77-84

[41] Yang H, Huang K, King I, Lyu MR. Localized support vector regression for time series prediction. Neurocomputing. 2009;**72**(10-12):2659-2669

[42] Chen T, Yin H, Chen H, Wu L, Wang H, Zhou X, et al. Tada: Trend alignment with dual-attention multi-task recurrent neural networks for sales prediction. In: 2018 IEEE International Conference on Data Mining (ICDM). Singapure, Singapure: IEEE; 2018. pp. 49-58. DOI: 10.1109/ICDM.2018.00020

[43] Haggan V, Ozaki T. Modelling nonlinear random vibrations using an amplitude-dependent autoregressive time series model. Biometrika. 1981;**68**(1):189-196

[44] Chen S, Wang XX, Harris CJ. NARX-based nonlinear system identification using orthogonal least squares basis hunting. IEEE Transactions on Control Systems Technology. 2007;**16**(1):78-84

[45] Frigola R, Rasmussen CE. Integrated pre-processing for Bayesian nonlinear system identification with Gaussian processes. In: 52nd IEEE Conference on Decision and Control. Firenze, Italy: IEEE; 2013. pp. 5371-5376. DOI: 10.1109/CDC.2013.6760734

[46] Bertsimas D, Boussioux L. Ensemble modeling for time series forecasting: an adaptive robust optimization approach [Internet]. 2023. Available from: cs.LG/2304.04308

[47] Khalil RA. Comparison of four neural network learning methods based on genetic algorithm for non-linear dynamic systems identification. AL Rafdain Engineering Journal. 2012;**20**(1):122-132

[48] Taylor JG. Univariate and multivariate time series predictions. In: Shadbolt J, Taylor JG, editors. Neural

Networks and the Financial Markets. Perspectives in Neural Computing. London: Springer; 2002. pp. 11-22. DOI: 10.1007/978-1-4471-0151-2_2

[49] Coyle D, Prasad G, McGinnity M. Faster self-organizing fuzzy neural network training and improved autonomy with time-delayed synapses for locally recurrent learning. In: Turgay T, editor. System and Circuit Design for Biologically-Inspired Intelligent Learning. Hershey, PA: IGI Global; 2011. pp. 156-183. DOI: 10.4018/978-1-60960-018-1.ch008

[50] Bengio Y. Learning deep architectures for AI. Found trends® Machine Learning. 2009;**2**(1):1-127

[51] Malekmohamadi Faradonbe S, Safi-Esfahani F, Karimian-kelishadrokhi M. A review on neural Turing machine (NTM). SN Computer Science [Internet]. 2020;**1**(6):333. DOI: 10.1007/s42979-020-00341-6

[52] Zhou H, Zhang S, Peng J, Zhang S, Li J, Xiong H, et al. Informer: Beyond efficient transformer for long sequence time-series forecasting. Proceedings of the AAAI Conference on Artificial Intelligence. 2021;**35**(12):11106-11115. DOI: 10.1609/aaai.v35i12.17325

[53] Zeng A, Chen M, Zhang L, Xu Q. Are transformers effective for time series forecasting? Proceedings of the AAAI Conference on Artificial Intelligence. 2023;**37**(9):11121-11128. DOI: 10.1609/aaai.v37i9.26317

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

[55] Cho K, van Merriënboer B, Bahdanau D, Bengio Y. On the properties of neural machine translation: Encoder{-}decoder approaches. In: Wu D, Carpuat M, Carreras X, Vecchi EM, editors. Proceedings of {SSST}-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation [Internet]. Doha, Qatar: Association for Computational Linguistics; 2014. pp. 103-111. Available from: https://aclanthology.org/W14-4012

[56] Graves A, Schmidhuber J. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. Neural Networks. 2005;**18**(5-6):602-610

[57] Sutskever I, Vinyals O, Le QV. Sequence to sequence learning with neural networks. In: Ghahramani Z, Welling M, Cortes C, Lawrence N, Weinberger KQ, editors. Advances in neural information processing systems. 2014;**27**:9. ISBN: 9781510800410. Available from: https://proceedings.neurips.cc/paper_files/paper/2014

[58] Gehring J, Auli M, Grangier D, Yarats D, Dauphin YN. Convolutional sequence to sequence learning. In: International Conference on Machine Learning. Vol. 70. PMLR; 2017. pp. 1243-1252. Available from: https://proceedings.mlr.press/v70/gehring17a.html

[59] Firat O, Cho K, Sankaran B, Yarman Vural FT, Bengio Y. Multi-way, multilingual neural machine translation. Computer Speech & Language [Internet]. 2017;**45**:236-252. Available from: https://www.sciencedirect.com/science/article/pii/S0885230816301097

[60] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, et al. Attention is all you need. Advances in Neural Information Processing Systems. 2017;**30**:11

Chapter 2

Time Series Classification: A Review of Algorithms and Implementations

Johann Faouzi

Abstract

Time series classification is a subfield of machine learning with numerous real-life applications. Due to the temporal structure of the input data, standard machine learning algorithms are usually not well suited to work on raw time series. Over the last decades, many algorithms have been proposed to improve the predictive performance and the scalability of state-of-the-art models. Many approaches have been investigated, ranging from deriving new metrics to developing bag-of-words models to imaging time series to artificial neural networks. In this review, we present in detail the major contributions made to this field and mention their most prominent extensions. We dedicate a section to each category of algorithms, with an intuitive introduction on the general approach, detailed theoretical descriptions and explicit illustrations of the major contributions, and mentions of their most prominent extensions. At last, we dedicate a section to publicly available resources, namely data sets and open-source software, for time series classification. A particular emphasis is made on enumerating the availability of the mentioned algorithms in the most popular libraries. The combination of theoretical and practical contents provided in this review will help the readers to easily get started on their own work on time series classification, whether it be theoretical or practical.

Keywords: time series, classification, machine learning, data sets, open-source software

1. Introduction

A time series is a time-ordered sequence of values. Because of their unstructured nature, time series can be found in numerous fields. With the increase in data gathering through sensors and Internet activity, the amount of time series data keeps growing. From historic fields such as econometrics and finance to more recent ones such as marketing through churn prediction and recommender system, the possible applications are almost endless.

In machine learning, classification is defined as the process of assigning a label to an unlabeled observation by exploiting patterns of the training data. Classification of time series data can address several real-world problems such as household device classification to reduce carbon footprint [1] and disease detection using electrocardiogram data [2]. However, standard machine learning classification algorithms, which are designed for structured data, are not always well suited for unstructured data such as time series. In particular, the order of the values is an essential part of time series, and consecutive time points are likely to be highly correlated. One typically unsuited algorithm is the Naive Bayes algorithm, which assumes conditional independence between each feature given the label.

Many different categories of algorithms have been investigated to tackle time series classification. In order to compare time series, specific metrics and kernels have been proposed. Some algorithms consist in extracting features from time series that can then be used as input of a standard machine learning classifier, while other algorithms work on raw time series. Bag-of-words models relying on discretizing time series are popular, with many algorithms being developed. Transforming time series into images has also been studied.

Time series classification has been applied in many fields, with numerous organized open competitions and publicly available data sets. The 2008 IEEE World Congress on Computational Intelligence presented a challenge based on a common problem in the automotive industry: detecting whether a certain symptom (defect) is present in an automotive subsystem based on a sequence of measurements [3]. The PhysioNet Computing in Cardiology Challenge 2004 was an open competition with the objective of developing automated methods for predicting spontaneous termination of atrial fibrillation [4]. Food spectrographs, represented as time series, are used in chemometrics to classify food types, with applications in food safety and quality assurance [5]. Automatic assessment of surgical skills based on kinematic data, instead of manual feedback from experienced surgeons, which is subjective and time-consuming, was proposed to improve surgical practice [6–8].

There exist two kinds of time series: univariate and multivariate time series. At a given time point, the value of a univariate time series is a single element (typically a real number), while the value of a multivariate time series is a vector (of real numbers typically). The large majority of the time series classification literature is focused on univariate time series classification. Therefore, unless otherwise specified, time series are assumed to be univariate.

In this review, we present the major contributions to time series classification. As the number of publications on this topic is very large, the aim of this review is not to provide an exhaustive list, but rather to present the main contributions in detail, with explicit illustrations. The following sections present these contributions, grouped by categories. Finally, a section highlighting publicly available resources, namely data sets and software, is provided so that the readers can easily get started with their own applications.

2. Nearest-neighbor classification with dynamic time warping

Nearest-neighbor methods are one of the most intuitive algorithms for supervised learning: The prediction for a new sample is based on the target value of similar samples. A key element of nearest-neighbor algorithms is the metric, that is, the mathematical function defining the (dis)similarity between any pair of samples.

Although the Euclidean distance is the most common metric, it is not well suited to compare time series for two main reasons: (i) It is only defined for two vectors with the same length, whereas the time series of a given data set often have different lengths, and (ii) it compares the values of both time series at each time point independently, whereas the values of time series are correlated. Considering the minimum of the Euclidean distances between the smaller time series and the subsequences of the same length from the larger time series may not be optimal. A concrete example from automatic speech recognition is to consider a given sentence, with one being pronounced slower than the other one. The corresponding time series will not only have different lengths, but the Euclidean distance between the smaller time series and any subsequence of the same length with consecutive time points of the larger time series will not be small, even though the same sentence was pronounced and a relevant metric should yield a small value. Another concrete example could consist of sequences of geolocations, with several people walking the same route but at different speeds.

Dynamic time warping (DTW) is a metric for time series that addresses both limitations of the Euclidean distance [9]. The one-nearest neighbor classifier with the DTW metric is often considered the baseline algorithm for time series classification. The next subsections introduce dynamic time warping and several of its variants.

2.1 Dynamic time warping

Let $X = (x_1, \dots, x_n)$ and $Y = \left(y_1, \dots, y_m\right)$ be two time series of length n and m, respectively. The cost matrix, denoted by C, is a $n \times m$ matrix consisting of the cost between each pair of values in both time series:

$$\forall i \in \{1, \dots, n\}, \forall j \in \{1, \dots, m\}, C_{ij} = f\left(x_i, y_j\right) \tag{1}$$

where f, often called *local divergence*, is a function evaluating the cost between any pair of real numbers; f is usually the squared difference function (and more generally the squared Euclidean distance for multivariate time series):

$$\forall x \in \mathbb{R}, \forall y \in \mathbb{R}, f(x, y) = (x - y)^2 \tag{2}$$

A warping path is a sequence $p = \left(p_1, \dots, p_L\right)$ such that:

- *Value condition:* $\forall l \in \{1, \dots, L\}, p_l = \left(i_l, j_l\right) \in \{1, \dots, n\} \times \{1, \dots, m\}$
- *Boundary condition:* $p_1 = (1, 1)$ and $p_L = (n, m)$
- *Step condition:* $\forall l \in \{1, \dots, L-1\}, p_{l+1} - p_l \in \{(0, 1), (1, 0), (1, 1)\}$

The cost associated with a warping path, denoted by C_p, is the sum of the elements of the cost matrix that belong to the warping path:

$$C_p(X, Y) = \sum_{l=1}^{L} C_{i_l, j_l} \tag{3}$$

The dynamic time warping score is defined as the minimum cost among all the warping paths:

$$DTW(X, Y) = \min_{p \in P} C_p(X, Y) \tag{4}$$

where P is the set of warping paths. Instead of computing the costs for all the warping paths, a more efficient computation is possible using dynamic programming. Let $X_{:i} = (x_1, \dots, x_i)$ and $Y_{:j} = \left(y_1, \dots, y_j\right)$ be two subseries of the original time series X and Y, respectively, with $i \in \{2, \dots, n\}$ and $j \in \{2, \dots, m\}$. Since the step condition constrains the difference between two consecutive elements of a warping path, the DTW score between two time series can be computed based on the cost matrix and the DTW scores of the subseries in which the last element has been removed:

$$DTW\left(X_{:i}, Y_{:j}\right) = C_{i,j} + min\left(DTW\left(X_{:i-1}, Y_{:j-1}\right), DTW\left(X_{:i-1}, Y_{:j}\right), DTW\left(X_{:i}, Y_{:j-1}\right)\right) \tag{5}$$

This recurrence equation motivates the introduction of the accumulated cost matrix, denoted by D and defined as:

$$\forall i \in \{1, \dots, n\}, \forall j \in \{1, \dots, m\}, D_{i,j} = DTW\left(X_{:i}, Y_{:j}\right) \tag{6}$$

The accumulated cost matrix can be computed by initializing its first row and column, for which the values are simply the cumulative sums of the first row and column of the cost matrix C respectively, and by using the recurrence equation:

$$\forall j \in \{1, \dots, m\}, D_{1,j} = \sum_{k=1}^{j} C_{1,k} \tag{7}$$

$$\forall i \in \{1, \dots, n\}, D_{i,1} = \sum_{k=1}^{i} C_{k,1} \tag{8}$$

$$\forall i \in \{2, \dots, n\}, \forall j \in \{2, \dots, m\}, D_{i,j} = C_{i,j} + min\left(D_{i-1,j-1}, D_{i-1,j}, D_{i,j-1}\right) \tag{9}$$

The last entry of the accumulated cost matrix is the DTW score between X and Y:

$$DTW(X, Y) = D_{n,m} \tag{10}$$

When the cost function f is the squared difference, the square root of $D_{n,m}$ is usually defined as the DTW score so that the DTW score has the same "unit" as the time series, as it is done for the Euclidean distance and more generally for L_p (norms).

2.2 Limitations and variants of dynamic time warping

Despite its advantages over the Euclidean distance to compare time series, dynamic time warping has several important limitations.

First, its algorithmic complexity is $O(nm)$, where n and m are the lengths of both time series, which is high. Second, it is not a distance because it does not satisfy not only the separation property (the DTW score between two different time series can be zero) but more importantly the triangle inequality, meaning that efficient nearest-neighbor search algorithms such as the K-dimensional tree [10] and the ball tree [11] structures cannot be used. Both limitations make nearest-neighbor classification with DTW a computationally intensive algorithm. Third, DTW allows for very large time warps, which may be undesired. Fourth, DTW is not differentiable, making it difficult to use with machine learning algorithms that rely on minimizing an objective function with gradient descent or a variant thereof.

Several variants of DTW have been proposed to address one or several limitations of its original version.

A common approach consists in limiting the time warps by using a *constraint region*: The set of warping paths is restricted to the set of warping paths such that all their elements belong to the constraint region. This approach also decreases the computational complexity of the cost and accumulated cost matrices since only the entries belonging to the constraint region have to be computed, but adds the computational cost of the constraint region. Two commonly used constraint regions are the Sakoe-Chiba band [9] and the Itakura parallelogram [12]. The Sakoe-Chiba band limits the time warps to be no greater than half the bandwidth, while the Itakura parallelogram limits the time warps to be no greater than a variable value, this value being larger in the middle of the time series than at the starting and ending time points. The Sakoe-Chiba band and the Itakura parallelogram do not depend on the values of the time series, but only on their lengths. Other constraint regions depending on the values of the time series have been proposed [13, 14], relying on the optimal warping path for downsampled versions of the original time series. **Figure 1** illustrates the optimal warping path for the original DTW algorithm and three of its variants with constraint regions.

Another variant of DTW, called soft-DTW [15], replaces the min function, which is not differentiable, with a smooth minimum function, namely the LogSumExp function, which is differentiable. The soft-DTW function, being differentiable, can be used as a loss function for machine learning algorithms, in particular artificial neural networks.

3. Kernel methods

Kernel methods are popular machine learning algorithms allowing for nonlinear transformations or decision functions, among which support vector machines are probably the most famous ones and have been successfully used in numerous applications.

Kernel methods rely on a *kernel* function measuring similarity between any pair of inputs. A key necessary assumption of kernel methods is that the kernel is *positive-definite*. However, as mentioned in the previous section, DTW is not a distance because it does not satisfy the triangle inequality, implying that DTW cannot be used to define a positive-definite kernel. Although DTW has been used with kernel methods in several publications with some tricks, the fact that the theoretical assumptions are not satisfied is an important limitation.

A true positive-definite kernel for time series, called the *global alignment kernel*, was proposed [16]. The global alignment kernel, denoted k_{GA}^{γ}, is defined as the sum of all the negatively exponentiated costs over all the possible warping paths:

$$k_{GA}^{\gamma}(X, Y) = \sum_{p \in P} exp\left(-C_p(X, Y)/\gamma\right) \tag{11}$$

where $C_p(x, y)$, defined in Eq. (3), is the cost associated with the warping path p, P is the set of all the warping paths, and $\gamma > 0$ is a smoothing parameter.

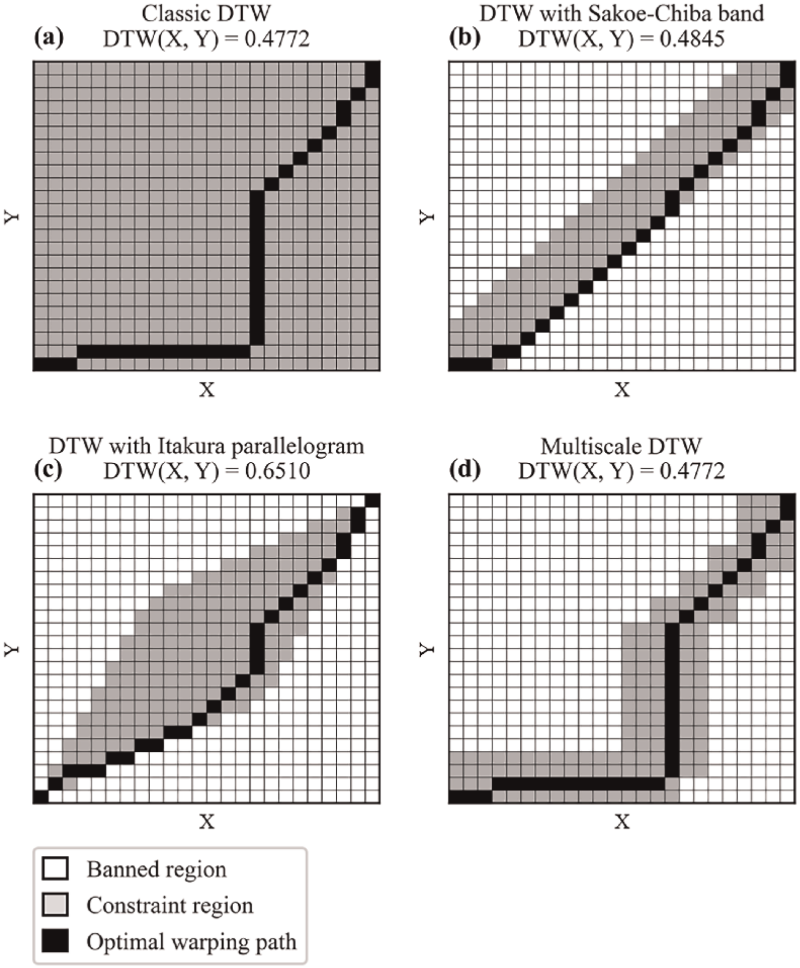


Figure 1.
Dynamic time warping. Illustration of the classic dynamic time warping (DTW) algorithm (a) and three of its variants with a constraint region: DTW with Sakoe-Chiba band (b), DTW with Itakura parallelogram (c), and multiscale DTW (d). For each algorithm, the set of admissible alignments and the optimal warping path is highlighted, and the corresponding score is computed. Multiscale DTW, by computing a constraint region specific to the input time series, is able to retrieve the same warping path as classic DTW with no constraint region.

It is important to note that the global alignment kernel is not a true kernel for every local divergence f. However, it can be proven that, if $1/(1 + exp\,(f))$ is a positive-definite kernel, then k^{γ}_{GA} is a kernel. In particular, this condition is satisfied when f is the squared difference function (and more generally the squared Euclidean distance for multivariate time series).

The global alignment kernel has the same computational complexity as DTW, that is, $O(nm)$, because the score between two time series can be computed using a recurrence equation similar to Eq. (5). Constraint regions such as the Sakoe-Chiba band and the Itakura parallelogram can also be used with global alignment kernels.

Support vector machines with the global alignment kernel have been shown to yield better predictive performances than with other pseudo kernels based on DTW for several multivariate time series classification tasks [16].

4. Shapelet-based algorithms

A shapelet is defined as a subsequence of consecutive observations from a time series. In some use cases, specific shapelets can be characteristic of the classes and thus helpful at discriminating them. Several algorithms rely on shapelets, either by extracting the best shapelets from the training data set or by directly learning them.

4.1 Shapelet transform

Lines and colleagues proposed an algorithm, called Shapelet Transform, that extracts the best shapelets from a data set [17]. Let $X = (x_1, \dots, x_n)$ be a time series of n real-valued observations and $S = (s_1, \dots, s_l)$ be a shapelet of l real numbers, with $l \leq n$. The distance between the shapelet S and the time series X, denoted $d(S, X)$, is defined as the minimum of the squared Euclidean distances between S and all the shapelets of length l from X:

$$d(S, X) = \min_{j \in \{0, \dots, n-l\}} \sum_{i=1}^{l} \left(s_i - x_{j+i}\right)^2 \tag{12}$$

The algorithm extracts all the shapelets whose length belongs to a range, the range being a hyperparameter of the algorithm, and selects the k best shapelets, k being another hyperparameter of the algorithm. This process can be seen as univariate feature extraction, where each feature is the distance between a given shapelet and all the time series in the data set. The shapelets are ranked based on the F-statistics of the analysis of variance test that compares the between- and within-class variabilities. In order to extract features (i.e., shapelets) that are not highly correlated, self-similar shapelets are removed, with any pair of shapelets being considered self-similar if they are from the same time series and have any overlapping indices.

When the k best shapelets have been identified and the corresponding features have been generated, any standard machine learning classifier can be applied to this new data set. Lines and colleagues investigated eight classifiers, including one-nearest neighbor classifier, support vector machine with a linear kernel, and random forest. In their experiments, the support vector machine with a linear kernel yielded the best results on average.

One limitation of this algorithm is its computational complexity. For a data set of N time series of length n, there are $N \times (n - l + 1)$ shapelets of length l. Since many values of $l \in \{1, \dots, n\}$ are investigated, the maximal computational complexity is $O(N \times n^2)$. Moreover, the definition of self-similarity for shapelets does not take into account the values of the shapelets, meaning that two shapelets very similar in terms of Euclidean distance but extracted from two different time series are not considered self-similar, even though they yield very similar features.

Several modifications to the algorithm have been proposed such as using other criteria to rank shapelets [18], in particular for multi-class classification tasks [19], and other classifiers built on top of the transformation [18].

4.2 Learning shapelets

In order to address the limitations of the Shapelet Transform algorithm, another algorithm relying on learning shapelets, instead of extracting them, was proposed [20].

The distance between a shapelet and a time series defined in Eq. (12) relies on the minimum function, which is not differentiable. Similar to the soft-DTW variant of DTW, the minimum function is replaced with a smooth minimum function, namely the LogSumExp function, which is differentiable. The logistic regression algorithm is used as the machine learning classifier built on top of the transformation. **Figure 2** illustrates two learned shapelets and the distances between both shapelets and time series, highlighting that each shapelet is characteristic of one class.

Since both the transformation and classification functions are differentiable, the chain rule allows for computing the gradients of the objection function with respect to the shapelets and the logistic regression coefficients, respectively, thus minimizing the objective function can be attempted to be solved by gradient descent.

Learning shapelets instead of extracting them has several advantages. First, it may lead to shapelets that are not from the data set but are discriminative of the classes. Second, it does not require going through the whole data set, and thus may be faster to train, especially with stochastic variants of gradient descent.

Nonetheless, learning shapelets also comes with drawbacks. As both the shapelets and the logistic regression coefficients need to be learned, the objective function is not convex (one can see the analogy with some clustering algorithms, such as k-means and Gaussian mixture models, where both the parameters and the members of the clusters need to be learned). Therefore, the optimization algorithm may converge to a bad local minimum. It also leads to more hyperparameters as the optimization process is a key component of the algorithm. Finally, as learning shapelets is embedded into the

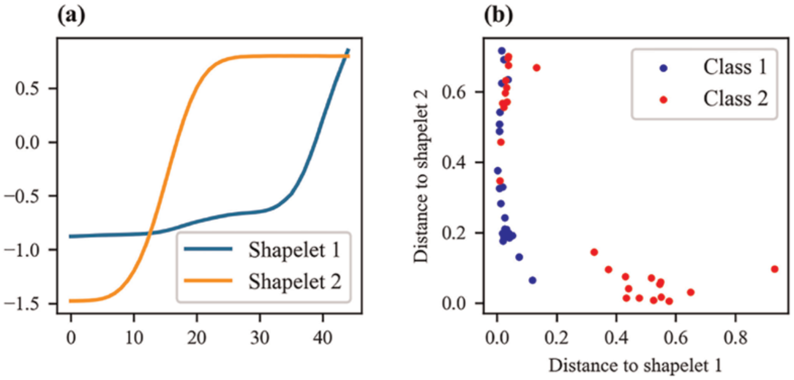


Figure 2.
Shapelets. Two shapelets have been learned from a training data set (a) and the distances between both shapelets and the training time series are plotted (b). The first shapelet is really specific of the second class, while the second shapelet is present in all the time series belonging to the first class and in some time series belonging to the second class.

algorithm, it may not be optimal to try other classifiers than logistic regression later on, whereas the Shapelet Transform algorithm is independent of the machine learning classifier, and thus, the transformation step can be computed only once and many classifiers can be built on top of it to find the best performing classifier.

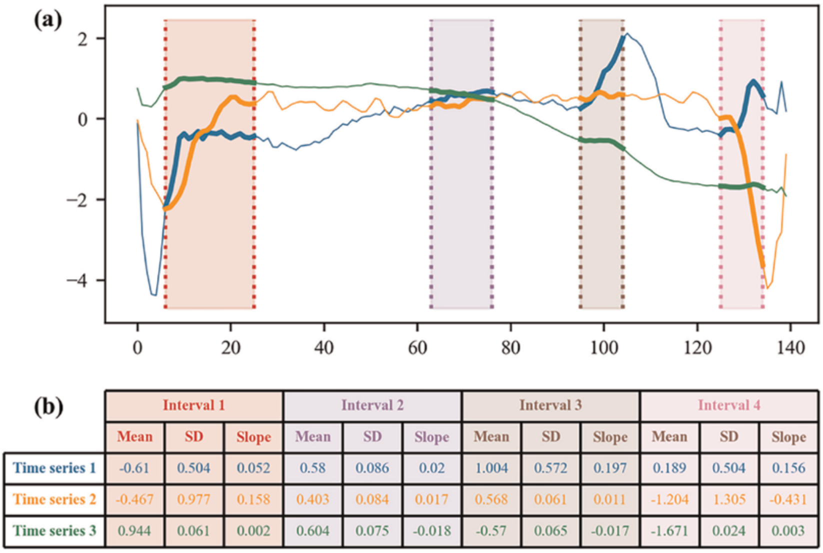
5. Tree-based algorithms

Standard tree-based machine learning algorithms, such as random forest [21] and extremely randomized trees [22], are popular algorithms that have proven to be powerful. The use of such algorithms for time series classification has been investigated, ranging from extracting features used as input to modifying the construction of each tree in the ensemble.

5.1 Time series forest

One of the first proposed algorithms based on the random forest algorithm is called time series forest [23] and is relatively simple. The algorithm considers information from subsequences of the time series. Given a minimum length for the subsequences, which is a hyperparameter, random intervals are generated, with the start indices, end indices, and lengths of all the intervals being all randomly generated. For a given time series and a given interval, the corresponding subsequence is the ordered set of values from the time series belonging to the interval. From each subsequence, three features are extracted: the mean, the standard deviation, and the slope. **Figure 3** illustrates this feature extraction.

The total number of extracted features is thus three times the number of considered intervals. A random forest classifier is then trained on these extracted features.



	Interval 1			Interval 2			Interval 3			Interval 4		
	Mean	SD	Slope	Mean	SD	Slope	Mean	SD	Slope	Mean	SD	Slope
Time series 1	-0.61	0.504	0.052	0.58	0.086	0.02	1.004	0.572	0.197	0.189	0.504	0.156
Time series 2	-0.467	0.977	0.158	0.403	0.084	0.017	0.568	0.061	0.011	-1.204	1.305	-0.431
Time series 3	0.944	0.061	0.002	0.604	0.075	-0.018	-0.57	0.065	-0.017	-1.671	0.024	0.003

Figure 3.
Time series forest. Random intervals are generated and the corresponding subsequences from each time series are extracted (a). Three features are derived from each subsequence: The mean, the standard deviation (SD), and the slope (b).

Predictions for new time series are obtained in the same manner: Given the already generated intervals, the three features are extracted from each subsequence, then the fitted random forest classifier outputs its prediction.

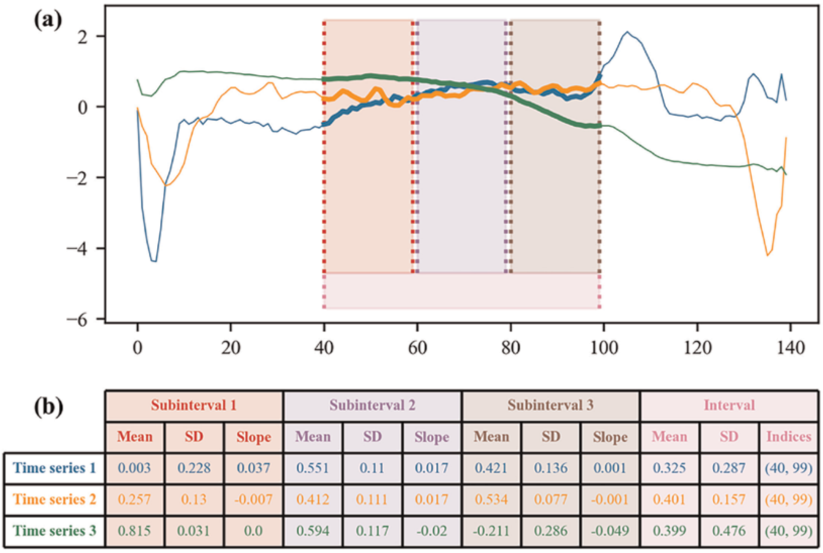
5.2 Time series bag-of-features

Time series bag-of-features [24] is a more advanced algorithm also based on features extracted from subsequences and random forest.

First, time series bag-of-forest also randomly generates intervals. However, this algorithm extracts more features than time series forest. Each interval is divided into non-overlapping subintervals, and the same three features (mean, standard deviation, and slope) are extracted from the subsequences corresponding to each subinterval. Moreover, four features from each interval are also extracted: the mean and the standard deviation of the subsequence corresponding to this interval, as well as the start and end indices of this interval. **Figure 4** illustrates this feature extraction.

A new data set is created, whose samples are the subsequences extracted from the time series for every interval, and whose features are the aforementioned extracted features. In this new data set, the number of samples is thus the number of time series times the number of intervals, while the number of features is equal to four plus three times the number of subintervals (four features for the interval and three features for each subinterval in the interval). The class of each subsequence is defined as the class of the time series from which the subsequence was extracted.

Then, a first random forest classifier is trained on this new data set, then outputs the probabilities to belong to each class for each subsequence. During the training phase, out-of-bag probabilities are actually used to have unbiased estimates of probabilities; that is, only trees that were built on bootstrap samples that did not contain the



	Subinterval 1			Subinterval 2			Subinterval 3			Interval		
	Mean	SD	Slope	Mean	SD	Slope	Mean	SD	Slope	Mean	SD	Indices
Time series 1	0.003	0.228	0.037	0.551	0.11	0.017	0.421	0.136	0.001	0.325	0.287	(40, 99)
Time series 2	0.257	0.13	-0.007	0.412	0.111	0.017	0.534	0.077	-0.001	0.401	0.157	(40, 99)
Time series 3	0.815	0.031	0.0	0.594	0.117	-0.02	-0.211	0.286	-0.049	0.399	0.476	(40, 99)

Figure 4.
Time series bag-of-features. Random intervals are generated and each window is split into subintervals, from which subsequences are extracted (a). Three features are derived from the subsequences of each subinterval (the mean, the standard deviation (SD), and the slope) and four features are derived from the subsequences of each interval (the mean, the standard deviation (SD), and the start and end indices) (b).

subsequence are used to compute the probabilities. Then, the probabilities are binned in order to summarize the distribution of the probabilities over all the subsequences; that is, the histogram of probabilities is computed to identify, for each class, how many subsequences were given high probabilities to belong to this class. More precisely, for each time series and for each class, the (out-of-bag) probabilities of belonging to the given class for all the subsequences extracted from the given time series are binned. For each time series and for each class, the mean probability over all the subsequences is also computed. Performing this operation for each time series and each class creates a new data set whose samples are the time series and whose features are the mean and binned probabilities for each class over all the subsequences.

Finally, a second random forest classifier is trained on this new data set during the training phase and outputs the predicted class for an unseen time series during the inference phase.

5.3 Proximity forest

In contrast to the time series forest and time series bag-of-features algorithms that rely on extracting features from time series and then build a random forest, the proximity forest algorithm [25] works directly with raw time series and is inspired by the extremely randomized trees algorithm.

To better understand the proximity forest algorithm, we briefly recall the main concepts of the extremely randomized trees algorithm. Like in a random forest, several trees are independently trained. The major difference between both algorithms comes from the splitting criterion used to split a node into child nodes. In a random forest, only a random subset of the features is considered, and the best (feature, threshold) pair is chosen over all the possible (feature, threshold) pairs. In extremely randomized trees, randomness goes one step further: Instead of considering all the possible thresholds for each feature from the random subset of features, a single threshold is randomly generated for each feature, and the best (feature, threshold) pair is chosen. The node splitting process is thus much faster in extremely randomized trees since much fewer (feature, threshold) pairs are considered. Another consequence is that a single tree from extremely randomized trees usually performs worse than a single tree in a random forest, but the extremely randomized trees are less correlated than the trees of a random forest, thus benefitting more from averaging the predictions of each tree.

The proximity forest algorithm is heavily inspired by the extremely randomized trees algorithm, the only difference being the splitting criterion that we now describe. In standard decision trees, the splitting criterion is a (feature, threshold) pair that splits a set of samples into two subsets: The subset of samples whose values for the given feature are greater than the threshold, and the subset of samples whose values for the given feature are lower than the threshold. In a proximity forest, the splitting criterion is a (metric, set of exemplars) pair: The metric allows for measuring similarity between any pair of time series, and the set of exemplars is a set containing one exemplar of each class from all the time series belonging to the given node. The number of (metric, set of exemplars) considered at each splitting node is a hyperparameter of the algorithm. Like in extremely randomized trees, the metric and the set of exemplars are randomly chosen: The metric is chosen uniformly at random from a set of 11 metrics for time series, and the exemplar for each class is chosen uniformly at random from all the time series belonging to this class and this node.

Because only a small set of (metric, set of exemplars) pairs is considered at each splitting node and because the decision tree growing process exponentially decreases the sample size at each depth of the tree, the proximity forest is highly scalable to large data sets of time series (in terms of both the number of time series and the number of time points). Because the trees of the proximity forest tend not to be highly correlated, the proximity forest benefits from aggregating the predictions of each tree by decreasing the variance of the final model and has proven to give a good predictive performance on average.

6. Bag-of-words (dictionary-based) approaches

Bag-of-words approaches, also known as dictionary-based approaches, consists in discretizing time series into sequences of symbols, then extracting words from these sequences with a sliding window, and finally counting the number of words for all the words in the dictionary. These approaches are split into two groups: the ones based on discretizing raw time series, and the other ones based on discretizing Fourier coefficients.

6.1 Approaches based on discretizing raw time series

The most commonly used algorithm to discretize raw time series is called Symbolic Aggregation approXimation (SAX) and simply maps each real-valued observation of the time series to its corresponding bin [26]. Several strategies to compute the bin edges are possible: quantiles of the standard normal distribution if the time series was standardized (zero mean, unit variance), uniform bins based on the extreme values of the time series, or quantiles of the time series. **Figure 5** illustrates the SAX discretization with different strategies to compute the bins.

Based on the SAX discretization of standardized time series with quantiles of the standard normal distribution, the bag-of-patterns algorithm [27] uses a sliding

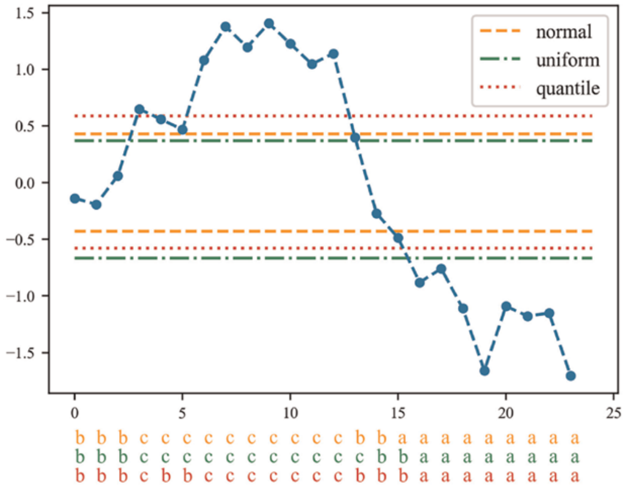


Figure 5.
Symbolic aggregate approXimation. A time series is discretized using bin edges. Different strategies to compute the bin edges are possible: Quantiles of the standard normal distribution, uniform bins based on the extreme values of the time series or quantiles of the time series. The resulting sequences of symbols for each strategy are shown at the bottom.

window to extract words from the discretized time series and computes the corresponding histogram; that, is the frequency of each word is computed, resulting in a new data set in which each feature is a word from the dictionary and each value is the number of occurrences of the given word in a given time series. Since consecutive subsequences are likely to be very similar and thus lead to identical words, it was proposed to count a single occurrence of identical consecutive words, this process being called *numerosity reduction*. Finally, a nearest-neighbor classifier was built on top of this transformation to perform classification.

Another algorithm, called Symbolic Aggregation approXimation in Vector Space Model (SAX-VSM), was later proposed with two major differences from the bag-of-patterns algorithm [28]. First, the order of the preprocessing steps is changed. The sliding window is applied to the raw time series to extract subsequences. Each subsequence is standardized and discretized using the SAX algorithm, resulting in an ordered sequence of words, to which numerosity reduction is usually applied. Second, classification is based on a simple numerical statistic used in natural language processing: the term frequency—inverse document frequency (TF-IDF) matrix. The idea is to identify words that are specific to each class. After computing the frequency of each word in the dictionary for each time series (resulting in a vector for each time series), all the vectors corresponding to time series belonging to the same class are summed in order to obtain the frequency of each word for each class. This process results in the term frequency matrix, whose rows represent the words in the dictionary, the columns represent the classes in the data set, and each entry is the frequency of the given word for the given class. Each row of this matrix is then normalized by the number of classes in which the word is present so that words that are specific to a small number of classes are more heavily weighted than words present in many classes. This normalized matrix is the so-called TF-IDF matrix. Classification is performed using the cosine similarity between the word frequency vector of a new time series and each column of the TD-IDF matrix, the predicted class being the one yielding the highest cosine similarity.

6.2 Methods based on discretizing Fourier coefficients

Instead of discretizing raw time series, other methods rely on discretizing Fourier coefficients. The most commonly used algorithm to do so is called Symbolic Fourier Approximation (SFA) and is a two-stage algorithm [29]. In the first stage, the discrete Fourier transform of the time series is computed and a subset of the Fourier coefficients is kept. In unsupervised learning, this subset is usually the set of first coefficients (the ones corresponding to the lowest frequencies) since they represent the trend of the time series. In supervised learning, univariate feature selection can be applied in order to select the more highly ranked coefficients based on statistics such as the F-statistics returned by one-way analysis of variance tests. Importantly, the same Fourier coefficients must be selected for all the time series. This transformation results in a matrix whose rows are time series and whose columns (i.e., features) are Fourier coefficients. In the second step, each column of this matrix is independently discretized. In unsupervised learning, the bin edges are usually computed so that the bins are uniform (i.e., the bin edges are based on the extreme values of the Fourier coefficients) or the number of Fourier coefficients falling in each bin is the same (i.e., the bin edges are based on the quantiles of the Fourier coefficients). In supervised learning, the bin edges can be computed to minimize an impurity criterion such as

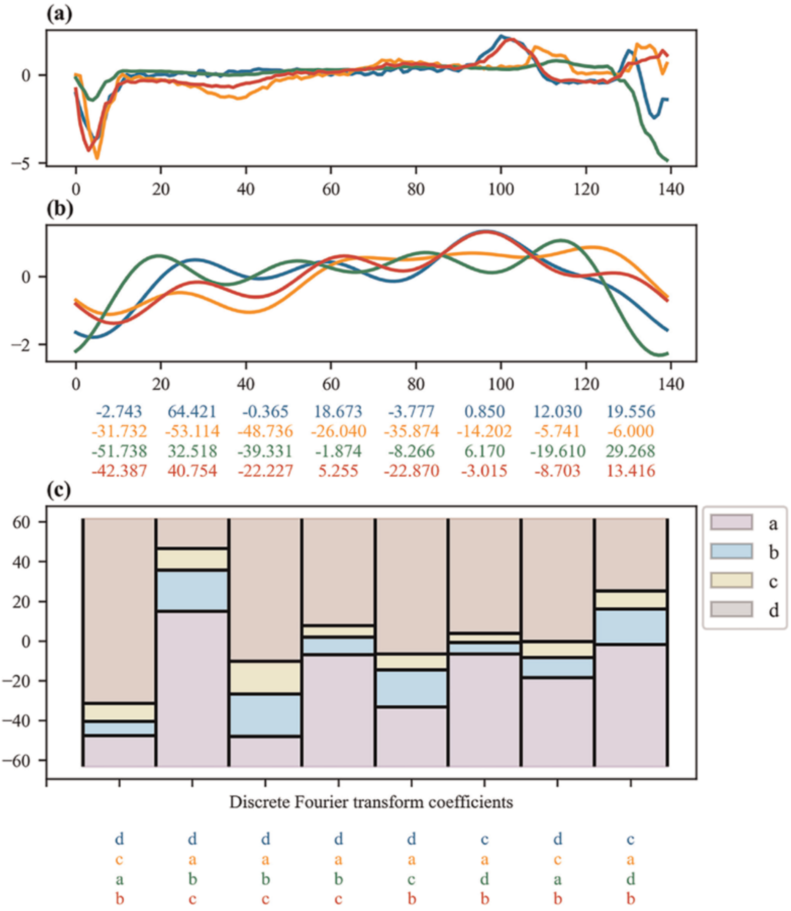


Figure 6.
Symbolic Fourier approximation. Raw time series (a) are approximated using a subset of the coefficients of the discrete Fourier transform, with the values of the coefficients for each time series being displayed at the bottom (b). The coefficients are discretized using bin edges that are computed either as uniform bins based on the extreme values of the Fourier coefficients or as quantiles of the Fourier coefficients, and the resulting sequences of symbols are displayed at the bottom (c).

entropy. Therefore, the SFA algorithm transforms a time series into a single sequence of symbols (i.e., a single word). **Figure 6** illustrates the SFA transformation.

Based on the SFA transformation, the Bag-of-SFA-Symbols (BOSS) algorithm was proposed [30]. First, subsequences of a time series are extracted with a sliding window and the SFA transformation is applied to each subsequence, resulting in an ordered sequence of words. Numerosity reduction is often applied to this sequence to avoid outweighing stable sections of time series. The frequency of each word is computed to obtain the word histogram of the time series. **Figure 7** illustrates these stages of the BOSS algorithm. Finally, classification is performed using the nearest-neighbor algorithm with the BOSS metric, which is a variant of the squared Euclidean distance that does not take into account the words that are not present in the histogram of the first time series. An ensemble of BOSS classifiers with sliding windows of different lengths is usually built to capture patterns of different lengths.

Several extensions of the BOSS algorithm have been proposed.

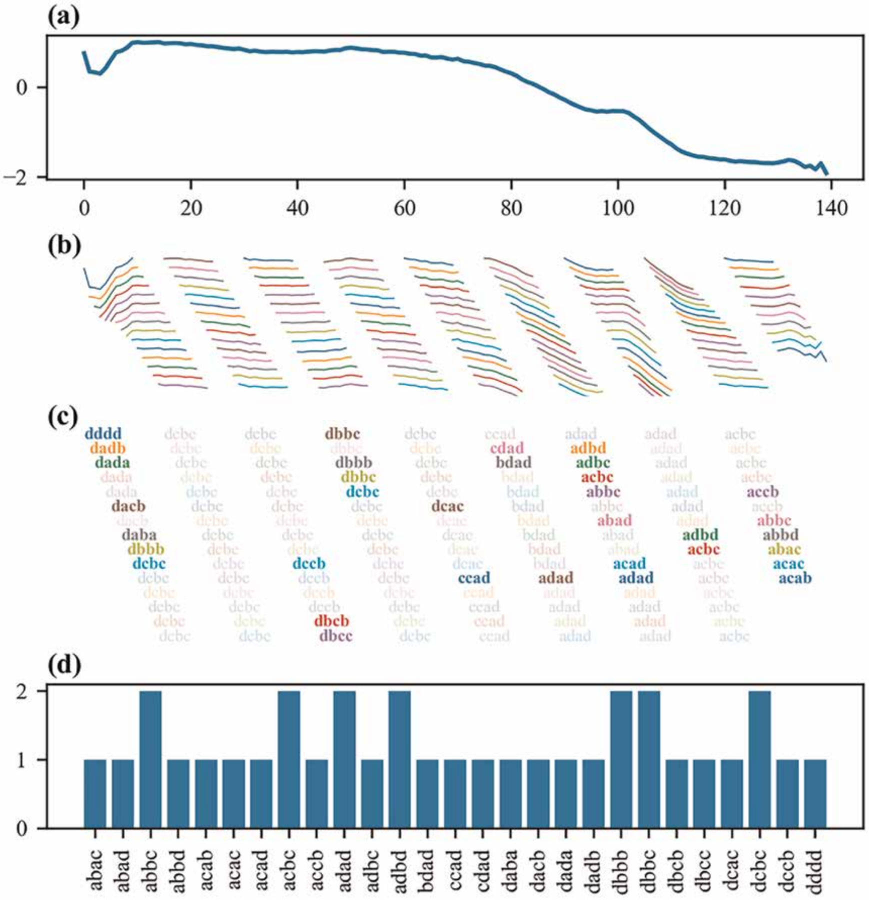


Figure 7.
Bag-of-SFA symbols. From a raw time series (a), a sliding window is applied to extract subsequences (b). Each subsequence is transformed into a word using the symbolic Fourier approximation (SFA) algorithm, and only the first occurrence of identical back-to-back words is kept (c). Finally, the histogram of the bag of words (i.e., the frequency of each word) is computed (d).

The Bag-of-SFA-Symbols in Vector Space (BOSSVS) combines the BOSS and vector space models [31]. Similar to some modifications of the bag-of-patterns algorithm in SAX-VSM, a single histogram is computed for each class (instead of a histogram for each time series) and the TF-IDF matrix is computed. Classification is performed by finding the class yielding the highest cosine similarity between the histogram of the time series and the TF-IDF vector of each class.

Randomized BOSS (RBOSS) introduces randomization in the choice of the lengths of the sliding windows and uses a more sophisticated aggregation of the predictions of the base BOSS classifiers, leading to lower computational time while maintaining similar performance [32].

BOSS with Spatial Pyramids (SP-BOSS) makes use of spatial pyramids, a method commonly used in computer vision, in order to combine temporal and phase independent features, and replaces the BOSS metric with the histogram intersection metric [33].

The Word Extraction for Time Series Classification (WEASEL) algorithm extends BOSS by integrating the sliding windows of different lengths inside the transformation, thus before the classification step [34]. To decrease the size of the resulting dictionary, only non-overlapping subsequences are extracted for each sliding window and the chi-squared test is applied to filter in the most relevant features. Since the constructed input matrix of word counts may have many features and be very sparse, logistic regression is built on top of the WEASEL transformation as this algorithm can handle both characteristics. WEASEL plus Multivariate Unsupervised Symbols and Derivatives (WEASEL+MUSE) is an extension of WEASEL to multivariate time series classification [35].

The Temporal Dictionary Ensemble (TDE) algorithm combines design features of four of these algorithms (BOSS, RBOSS, SP-BOSS, and WEASEL) with a novel mechanism of base classifier model selection based on an adaptative form of Gaussian process modeling of the parameter space [36].

7. Imaging time series

In order to investigate temporal correlations between all the pairs of observations, several methods relying on transforming time series (i.e., vectors) into images (i.e., matrices) have been proposed. In this section, we focus more on presenting these transformations than the classification algorithms built on top of them. Since these classifiers often belong to the class of deep learning algorithms, they will be more detailed in the next section.

7.1 Recurrence plot

A recurrence plot is an old technique that was originally introduced to visually inspect time constancy in dynamical systems through *trajectories* [37]. A trajectory is defined as a subsequence of equally spaced values:

$$\forall i \in \{1, \dots, n-(m-1)\tau\}, \overrightarrow{x_i} = \left(x_i, x_{i+\tau}, \dots, x_{i+(m-1)\tau}\right) \tag{13}$$

where m is the *length* of the trajectory and τ is the *time delay*, that is, the time gap between back-to-back time points in the trajectory.

A recurrence plot, denoted by R, is a matrix consisting of the binarized pairwise distances between all the pairs of trajectories from a time series:

$$\forall i,j \in \{1, \dots, n-(m-1)\tau\}, R_{i,j} = \Theta\left(\varepsilon - \left\|\overrightarrow{x_i} - \overrightarrow{x_j}\right\|_2\right) \tag{14}$$

where ε is the threshold used to binarize the distance and Θ is the Heaviside step function, which is 1 for positive arguments and 0 for non-positive arguments. Visually, as a black-and-white image, a pixel is black if and only if the distance between the two considered trajectories is smaller than the threshold (**Figure 8**).

Recurrence plots have been used as a preprocessing step to classify time series. Silva and colleagues used a k-nearest neighbor classifier with the Campana-Keogh distance [38] to measure the similarity between recurrence plots [39]. Tamura and colleagues used the recurrence plots computed from moving average convergence divergence histograms to train an artificial neural network consisting of stacked autoencoders [40]. Hatami and colleagues trained a 2D convolutional neural network

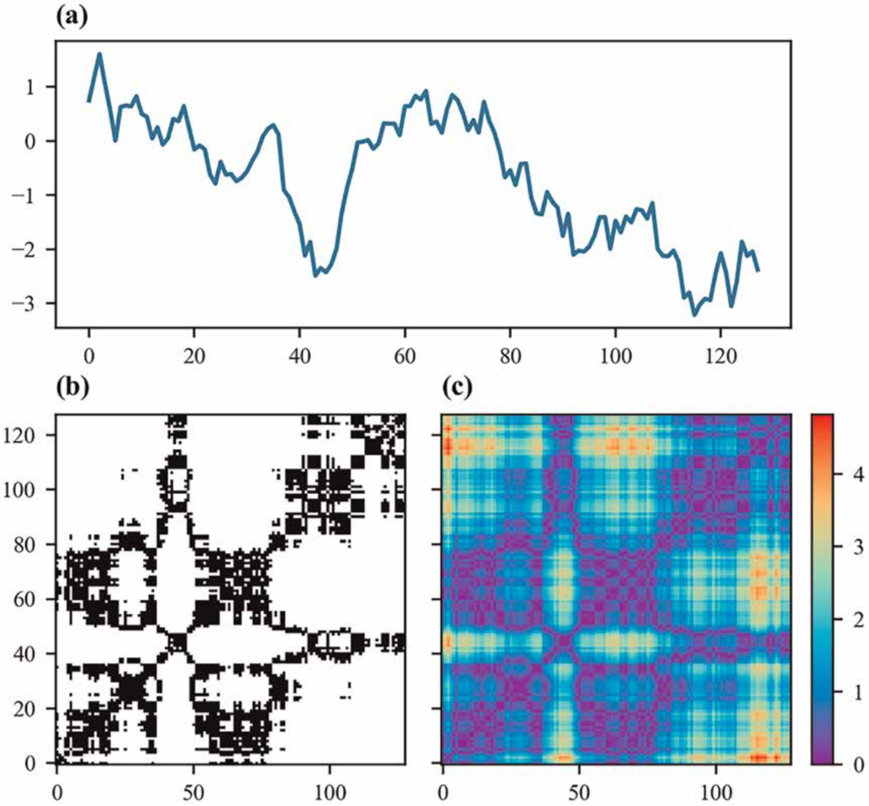


Figure 8.
Recurrence plot. Starting from a time series (a), the Euclidean distances between all the pairs of trajectories are computed and binarized using a threshold (b). The thresholding step is sometimes skipped and the raw pairwise Euclidean distances are considered (c).

using as input non-binarized recurrence plots by skipping the thresholding step, resulting in grayscale texture images [41].

7.2 Gramian angular field

While recurrence plots consider phase space trajectories, another method, called the Gramian angular field and based on the polar coordinate representation of time series, was proposed [42].

A time series $X = (x_1, \ldots, x_n)$ of real-valued observations is first linearly rescaled into the range $[a, b]$ with $-1 \leq a < b \leq 1$:

$$\forall i \in \{1, \ldots, n\}, \tilde{x_i} = a + (b - a) \times \frac{x_i - min\,(x)}{max\,(x) - min\,(x)} \tag{15}$$

The values of a and b may depend on the time series if the scale of the time series is important. Otherwise, the range is usually $[a, b] = [-1, 1]$ or $[a, b] = [0, 1]$.

The rescaled time series $\tilde{X}$ is then represented in polar coordinates with the radii depending on the time points and the angles depending on the values of the rescaled time series:

$$\forall i \in \{1, \dots, n\}, r_i = \frac{i}{n} \tag{16}$$

$$\forall i \in \{1, \dots, n\}, \phi_i = arccos\left(\tilde{x}_i\right) \tag{17}$$

Only the angles are considered since the radii do not depend on the values of the time series.

A Gramian angular field measures temporal correlation by computing the trigonometric sum or difference between all the pairs of angles. When the trigonometric sum is applied, the Gramian angular field is called a *Gramian angular summation field*, while it is called a *Gramian angular difference field* when the difference is applied. Let *GASF* be the matrix of a Gramian angular summation field and then *GADF* the matrix of a Gramian angular difference field. The entries of both matrices are computed using the following equations:

$$\forall i,j \in \{1, \dots, n\}, GASF_{i,j} = cos\left(\phi_i + \phi_j\right) \tag{18}$$

$$\forall i,j \in \{1, \dots, n\}, GADF_{i,j} = sin\left(\phi_i - \phi_j\right) \tag{19}$$

Figure 9 summarizes the whole process to generate Gramian angular fields and illustrates both the Gramian angular summation and difference fields. By default, a Gramian angular field is an $n \times n$ matrix, which can be excessively large for large n. A common approach consists in first downscaling the time series from n points to m points, with m being the desired size of the Gramian angular fields. It should also be noted that the computation of Gramian angular fields can be simplified using trigonometric identities:

$$\forall i,j \in \{1, \dots, n\}, GASF_{i,j} = \tilde{x}_i\tilde{x}_j - \sqrt{\left(1 - \tilde{x}_i\right)\left(1 - \tilde{x}_j\right)} \tag{20}$$

$$\forall i,j \in \{1, \dots, n\}, GADF_{i,j} = \tilde{x}_j\sqrt{1 - \tilde{x}_i} - \tilde{x}_i\sqrt{1 - \tilde{x}_j} \tag{21}$$

Gramian angular fields have been used for time series classification as a preprocessing step to generate images used as input of a tiled convolutional neural network [42] and of the pre-trained Inception v3 model followed by a multilayer perceptron [43].

7.3 Markov transition field

Another method consists in assimilating a time series, after discretization, as a Markov chain, and is called Markov transition field [42].

A time series $X = (x_1, \dots, x_n)$ of real-valued observations is first discretized based on its quantile bins; that is, each x_i is assigned to its corresponding bin q_j with $j \in \{1, \dots, Q\}$ and Q being the number of quantile bins, resulting into a discretize-valued time series of length n. By considering this discretize-valued time series as observations of a first-order Markov chain, one can compute the number of occurrences of pairs of back-to-back bins for every pair of bins, resulting in a $Q \times Q$ matrix. This matrix is then normalized to transform the frequencies into probabilities,

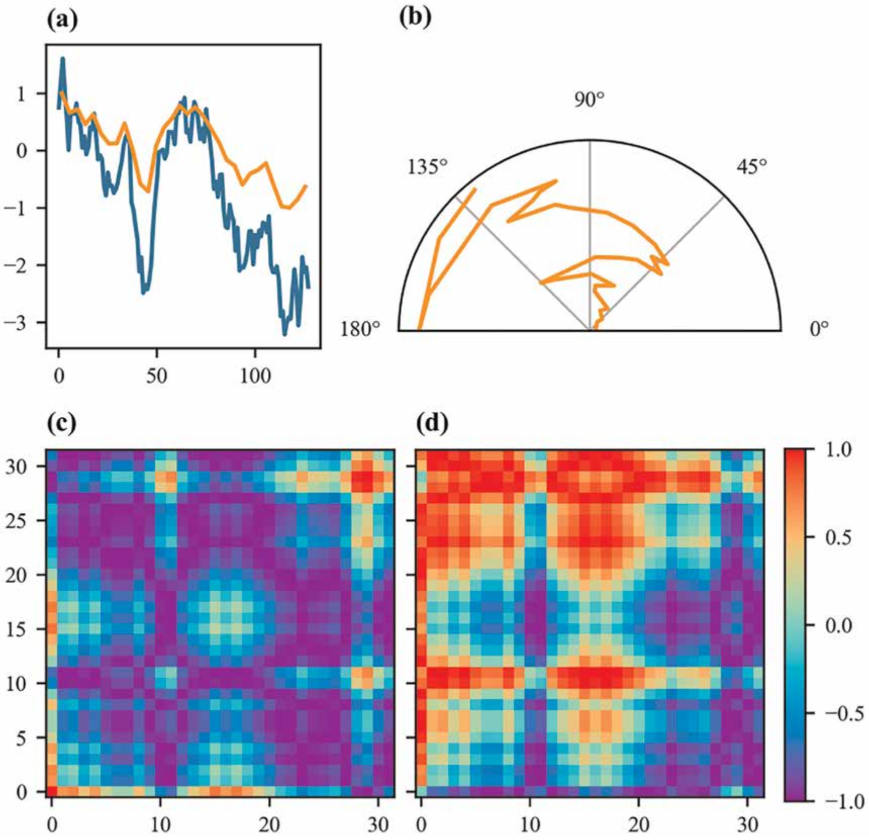


Figure 9.
Gramian angular fields. A raw time series is normalized in range [−1, 1] (a) and is represented in polar coordinates (b). The Gramian angular fields are computed either as the cosine of the sum (c) or the sine of the difference (d) of all the pair of angles.

leading to the Markov transition matrix, whose entries give the probabilities of going from q_j to q_k for every pair $\left(q_j, q_k\right)$ of bins.

The Markov transition matrix is insensitive to the temporal distribution of the time series X since it only captures the frequencies of the transition, but not at which time points they occurred. Moreover, its size depends on the number of bins and not the length of the time series, although larger time series may allow for a larger number of bins. To overcome these issues, the Markov transition matrix is projected onto a $n \times n$ matrix that is called the *Markov transition field*.

Let MTF be a Markov transition field and q be the function that maps the real-valued observations of the time series into their bins. Each entry of the Markov transition field is an entry of the Markov transition matrix, and thus a transition probability. $MTF_{i,j}$, that is, the Markov transition field entry for the pair $\left(x_i, x_j\right)$, is the probability of going from the bin associated to x_i, that is, $q(x_i)$, to the bin associated to x_j, that is, $q\left(x_j\right)$:

$$\forall i, j \in \{1, \ldots, n\}, MTF_{i,j} = \mathbb{P}\left(q\left(x_j\right) | q(x_i)\right) \tag{22}$$

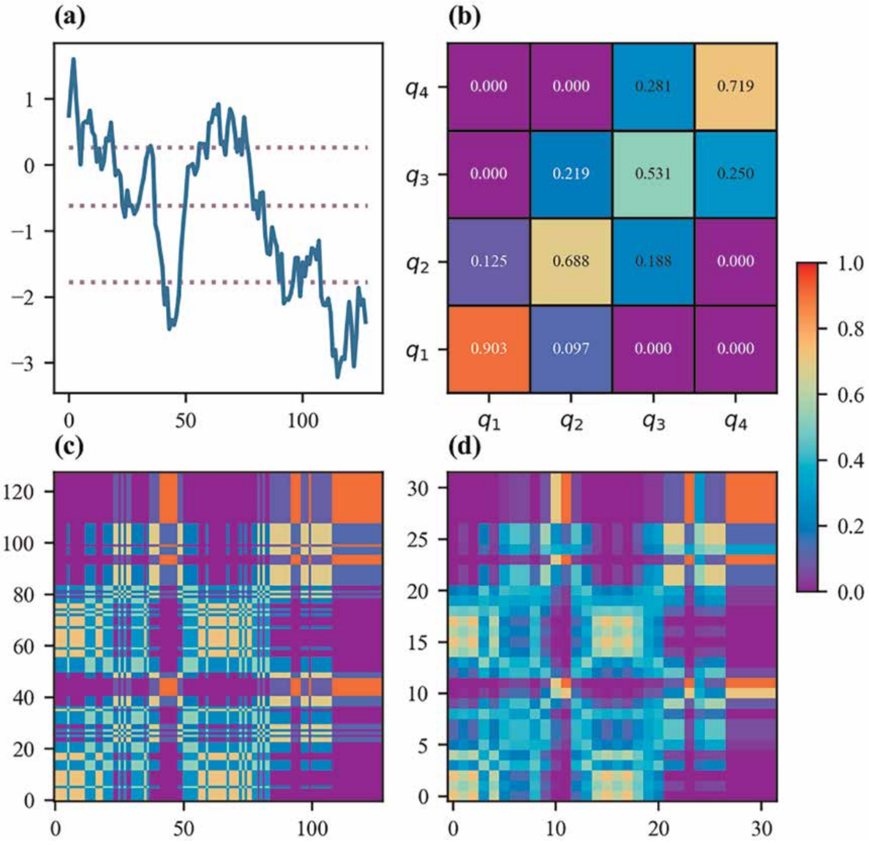


Figure 10.
Markov transition field. A raw time series is discretized with bin edges being defined as quantiles of the time series (a). The discretized time series is seen as a first-order Markov chain and the corresponding Markov transition matrix, representing the probability of going from a given bin to another given bin, is computed (b). The Markov transition matrix is spread through time to obtain the Markov transition field (c). The size of the Markov transition field can be reduced by taking the mean value inside each submatrix (d).

Figure 10 summarizes the whole process to generate Markov transition fields from raw time series. By default, a Markov transition field is an $n \times n$ matrix, which can be excessively large for large n. There are two common approaches to address this issue. The first approach consists in first downscaling the time series from n points to m points, with m being the desired size of the Markov transition fields, as it is done for Gramian angular fields. The second approach consists in downscaling the Markov transition field itself, from $n \times n$ to $m \times m$, by taking the mean value of each submatrix, which is commonly referred to as average pooling in the deep learning literature.

Like Gramian angular fields, Markov transition fields have been used for time series classification as a preprocessing step to generate images then train a tiled convolutional neural network [42].

8. Deep learning

Over the past decade, deep learning has led to many breakthroughs in several fields such as computer vision and natural language processing. Deep learning has also

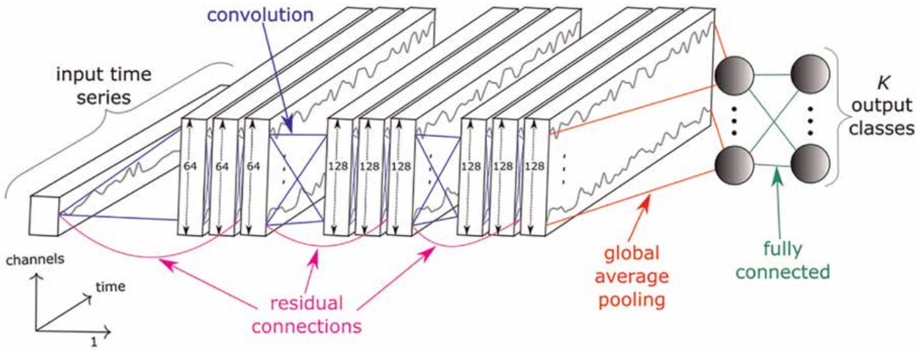


Figure 11.
InceptionTime architecture. The InceptionTime artificial neural network consists of several inception modules with residual connections, followed by a global average pooling layer and a fully connected layer. Reproduced from Ref. [50].

been recently investigated for time series classification. We refer the readers to an exhaustive review on this specific topic [44] and present only the major contributions in this review.

Ismail Fawaz and colleagues [45] proposed an ensemble of 60 neural network models to perform time series classification. The 60 models come from 6 architectures: a multi-layer perceptron [46], a fully convolutional neural network [46], a residual network [46], an encoder [47], a multi-channel deep convolutional neural network [48], and a time convolutional neural network [49]. Each architecture is used to train 10 different models with different initial weight values.

InceptionTime [50] is probably the main deep learning model for time series classification. InceptionTime is a neural network ensemble consisting of five Inception networks. **Figure 11** illustrates the architecture of each Inception network, consisting of blocks of three Inception modules (6 blocks by default), followed by a global averaging pooling layer and a fully-connected layer with the softmax activation function. Each Inception module consists of convolutions with kernels of several sizes followed by batch normalization and the rectified linear unit activation function.

Other investigations of deep learning for time series classification include transfer learning [51], data augmentation [52], adversarial attacks [53], and neural architecture search [54].

9. Random convolutions

Convolutional neural networks, containing several convolutional layers, have been investigated for time series classification. The values of the convolutional layers are trainable parameters that are optimized by stochastic gradient descent or a variant thereof. Convolutional neural networks have a large number of trainable parameters in comparison with more classic algorithms such as logistic regression or support vector machines, thus usually requiring a large sample size to find good values for the trainable parameters.

Based on this observation, the Random Convolutional Kernel Transform (ROCKET) algorithm was proposed [55]. This algorithm extracts features from time series using a large number of random convolutional kernels, meaning that all the parameters of all the kernels (length, weights, bias, dilation, and padding) are randomly generated from fixed distributions. Instead of extracting a single feature for each kernel, such as the maximum or the mean, as it is usually performed in convolutional neural networks, two features are extracted: the maximum and the proportion of positive values.

The classifier built on top of the transformation is responsible for selecting the most relevant features to perform classification. A ridge regression classifier was originally proposed for several reasons. First, it is highly efficient when the number of classes is high, because the multiclass classification task is treated as a multi-output regression task, with the predicted class corresponding to the output with the highest value; thus, the projection matrix needs to be computed only once. Second, the optimization of the λ parameter (controlling the amount of regularization) using leave-one-out cross-validation is also highly efficient [56]. Logistic regression was rather used for data sets in which the number of training time series was much larger than the number of extracted time series due to the better scalability of logistic regression solved with stochastic gradient descent for large numbers of training samples.

The ROCKET algorithm combined with a linear classifier has a much lower computational complexity than the best-performing time series classification algorithms while having a comparable performance. Its reported performance is actually higher on average than the ones of convolutional neural networks on the commonly benchmarked data sets. Given its high predictive performance and low computational time, ROCKET is one of the most prominent transformation algorithms for time series classification.

Several recent extensions have been proposed. MiniROCKET [57] reduces the randomness of the parameters of the kernels by using a fixed value or sampling from smaller distributions. Moreover, it only extracts the proportion of positive values for each kernel. These modifications also allow for more optimization and lead to a much lower computational complexity while maintaining similar performance. MultiROCKET [58] extends MiniROCKET by extracting possibly several features, leading to a slightly higher computational time but better accuracy. In particular, the authors found that the proportion of positive values and the longest period of consecutive positive values are the most effective features to be extracted from time series convolutional outputs.

10. Ensemble models

Averaging the predictions of several independently trained models into a single prediction is a common approach to build a better final model by decreasing the variance of the predictions. In traditional ensemble methods, all the base classifiers belong to a given type of algorithms. For instance, in a random forest, all the base classifiers are decision trees. However, using a single type of algorithm limits the upsides and downsides of the final model to the ones of the base classifier. On the other hand, using several types of algorithms allows for learning a more diverse representation of the data. In particular, for time series classification, ensemble models that combine different types of algorithms (bag-of-words approaches, shapelet-based algorithms, convolutions, etc.) have been developed. They often are state-of-the-art in terms of predictive performance, at the cost of high computational complexity.

The Collective of Transformation-Based Ensembles (COTE) algorithm was the first proposed ensemble classifier [59]. The most effective ensemble strategy was found to combine all the classifiers into a flat hierarchy and the corresponding model is often referred to as Flat-COTE [60]. Flat-COTE combines 35 classifiers over four data representations: 11 classifiers based on whole series similarity measures, 8 classifiers based on shapelet-transform, 8 based on autocorrelation features, and 8 based on power spectrum.

The Hierarchical Vote Collective of Transformation-Based Ensembles (HIVE-COTE) algorithm is an extension of COTE with significant modifications [60], including a new type of spectral classifier called Random Interval Spectral Ensemble, two more classifiers (BOSS and Time Series Forest), and a hierarchical voting procedure, defined as a weighted average of the probabilities returned by each classifier, with the weights being proportional to the classification accuracy estimated through cross-validation. HIVE-COTE is often updated based on newly published algorithms, with versions 1.0 [61] and 2.0 [62] being recently published.

The Time Series Combination of Heterogeneous and Integrated Embedding Forest (TS-CHIEF) algorithm is another ensemble model rivaling with HIVE-COTE in terms of predictive performance while having a substantially lower runtime [63]. TS-CHIEF builds a random forest of decision trees whose splitting functions are time series specific and based on similarity measures, dictionary (bag-of-words) representations, and interval-based transformations.

11. Public data sets and open-source software

So far, we have presented the mathematical aspects of time series classification algorithms. However, in practice, implementations and evaluations of these algorithms are equally important. In this section, we briefly present the commonly used data sets to assess the performance of time series classification algorithms, and popular software making available implementations of such algorithms.

11.1 Public data sets

The main resource used to benchmark time series classification algorithms is the UCR Time Series Classification Archive [64, 65], providing public access to univariate time series classification data sets that are already split into training and test sets, leading to a consistent benchmarking between algorithms. The current version of the archive contains 128 data sets from various domains (audio, medicine, motion, sensor, simulation, spectroscopy, etc.). **Figure 12** presents the distribution of several variables (number of time series in the training and test sets, number of time points, and number of classes) over the 128 data sets.

For multivariate time series classification algorithms, the main benchmark resources are the UEA Multivariate Time Series Classification Archive [66] and a publicly available archive [67], with some data sets being available in both resources.

A website [68] provides useful information about the data sets, the algorithms, and the performance of these algorithms on all the data sets, as well as download links for the data sets in the UCR & UEA Time Series Classification Repository.

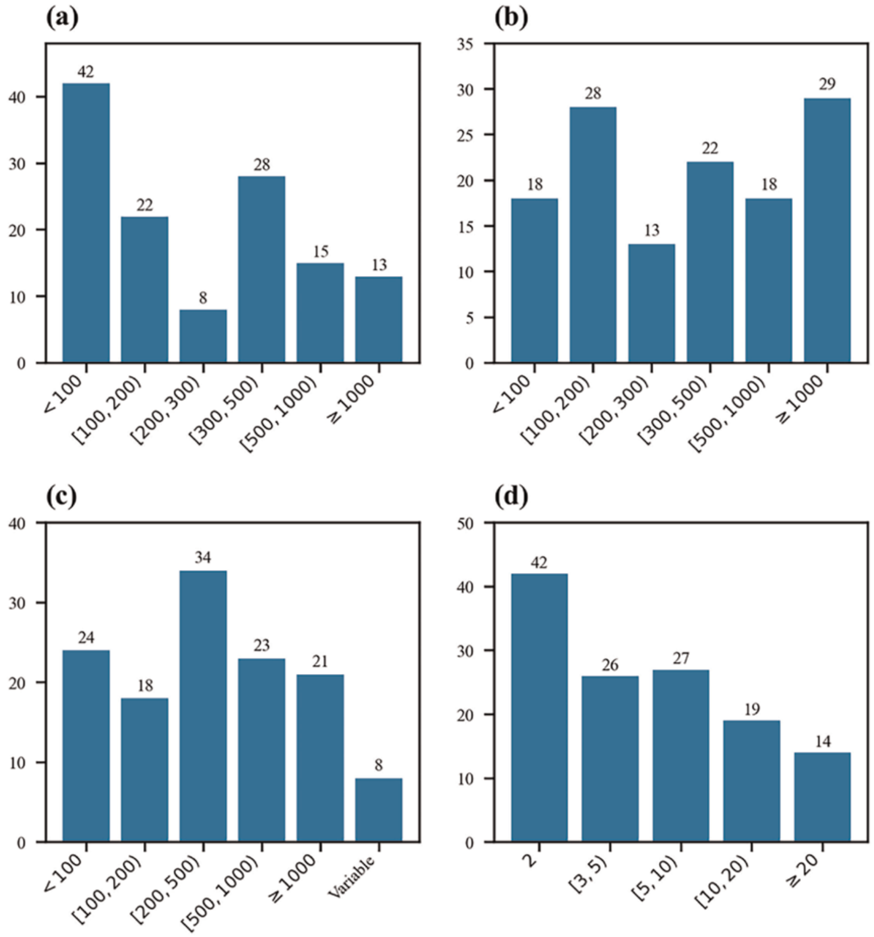


Figure 12.
Descriptive statistics of the data sets in the UCR time series classification archive. The UCR time series classification archive provides 128 univariate time series classification data sets. The histograms of four features of these data sets are plotted: The number of training time series (a), the number of test time series (b), the number of time points (c), and the number of classes (d).

11.2 Open-source software

Another key element of scientific research in machine learning is code availability. Although the code of most of the algorithms presented in this review is provided by their original authors, it is not trivial, for a new user, to build their work on top of it for several reasons. First, the code is usually only organized in order to reproduce the experiments (i.e., obtaining the same performance for all the data sets included in the analysis). Second, the code is usually neither commented nor documented. Third, different authors may code in different programming languages. For instance, among the algorithms presented in this review, the programming languages used in their original implementations included Java, Python, and R. All of these reasons make, for a given user, *reusing the provided code* and *comparing different algorithms* difficult.

Open-source software aims at providing, under the same application programming interface, a variety of tools, including implementations of specific algorithms, preprocessing tools, data set fetching and loading utilities, and visualization tools, all being documented, with the source code being publicly available. Popular open-source libraries for general machine learning already exist, such as scikit-learn [69] in Python and caret [70] in R. Several open-source libraries, focused on time series classification to different extents, have been developed, most of them being Python packages, which is not surprising as Python has become one of the most popular programming languages for machine learning.

Pyts [71] is a Python package entirely dedicated to time series classification, providing implementations of many algorithms presented in this review. Sktime [72] and tslearn [73] are more general machine learning toolboxes for time series, providing tools for other types of machine learning such as forecasting and clustering. Nonetheless, sktime also provides implementations for a lot of time series classification algorithms.

Table 1 enumerates the time series classification algorithms made available in the following libraries: pyts, sktime, and tslearn. Pyts and sktime separate themselves from tslearn by the numerous implemented algorithms in both Python packages. Research on deep learning approaches is recent, and the fact that it is developed in other libraries and that it is computationally intensive while not being the current state-of-the-art may explain that the development of dedicated libraries is lacking behind.

Category	**Algorithm**	**pyts**	**sktime**	**tslearn**
Metrics	k-nearest neighbors classifier with dynamic time warping	✓	✓	✓
Kernels	Support vector machines with global alignment kernel	✗	✗	✓
Shapelets	Shapelet transform	✓	✓	✗
	Learning shapelet	✓	✗	✓
Tree-based	Time series forest	✓	✓	✗
	Time series bag-of-features	✓	✗	✗
	Proximity forest	✓	✓	✗
Bag-of-words	Bag-of-patterns	✓	✗	✗
	SAX-VSM	✓	✗	✗
	BOSS	✓	✓	✗
	BOSSVS	✓	✗	✗
	WEASEL	✓	✓	✗
	WEASEL+MUSE	✓	✓	✗
	Randomized BOSS	✗	✓	✗
	BOSS with spatial pyramids	✗	✗	✗
	Temporal dictionary ensemble	✗	✓	✗
Image	Recurrence plot	✓	✗	✗
	Gramian angular field	✓	✗	✗
	Markov transition field	✓	✗	✗

Category	Algorithm	pyts	sktime	tslearn
Deep learning	Multilayer perceptron	✗	✓	✓
	Residual network	✗	✓	✗
	InceptionTime	✗	✓	✗
Random convolutions	ROCKET	✓	✓	✗
	MiniROCKET	✗	✗	✗
	MultiROCKET	✗	✗	✗
Ensemble	COTE	✗	✗	✗
	HIVE-COTE	✗	✗	✗
	HIVE-COTE version 1.0	✗	✓	✗
	HIVE-COTE version 2.0	✗	✓	✗
	TS-CHIEF	✗	✗	✗

For each algorithm, the availability of an implementation in the main libraries dedicated to time series classification is provided. Pyts and sktime, in addition to their documentation and unit tests, provide implementations of many algorithms under a unified application programming interface. Versions of the libraries: pyts (0.13.0), sktime (0.24.1), and tslearn (0.6.3).

Table 1.
Availability of time series classification algorithms in open-source libraries.

Other noteworthy software for time series classification includes the Python packages seglearn [74], tsfresh [75], and cesium [76], although these libraries do not provide implementations of many specific time series classification algorithms published in the literature, as well as the Java library tsml [77], which provides a lot of benchmarking results, but is mostly oriented toward academic research as it is neither tested nor documented yet.

12. Conclusion

Over the last decades, a lot of research on time series classification has led to major improvements in terms of predictive performance and scalability. Many approaches have been investigated, ranging from specific metrics to simple and complex feature extraction and transformations. More recently, open-source libraries dedicated to time series classification have been developed in order to provide implementations of these algorithms under a unified application programming interface. All this research on time series classification has been applied to real-life problems in various fields.

In this review, we presented in detail the main contributions and mentioned their most prominent extensions. We also presented the availability of implementations of these algorithms in the most prominent libraries dedicated to time series classification. We hope that the theoretical and practical contents provided in this review will allow the readers to easily get started in their own work on time series classification.

Author details

Johann Faouzi
Univ Rennes, Ensai, CNRS, CREST – UMR, Rennes, France

*Address all correspondence to: johann.faouzi@ensai.fr

References

[1] Lines J, Bagnall A, Caiger-Smith P, Anderson S. Classification of household devices by electricity usage profiles. In: Yin H, Wang W, Rayward-Smith V, editors. Intelligent Data Engineering and Automated Learning - IDEAL 2011, Lecture Notes in Computer Science. Berlin, Heidelberg: Springer; 2011. pp. 403-412

[2] Olszewski R. Generalized Feature Extraction for Structural Pattern Recognition in Time-Series Data [PhD thesis]. Pittsburgh, PA, USA: School of Computer Science, Carnegie Mellon University; 2001

[3] Adeodato PJL, Arnaud AL, Vasconcelos GC, Cunha RCLV, Gurgel TB, Monteiro DSMP. The role of temporal feature extraction and bagging of MLP neural networks for solving the WCCI 2008 ford classification challenge. In: 2009 International Joint Conference on Neural Networks, Atlanta, GA, USA. IEEE; 2009. pp. 57-62. Available from: https://ieeexplore.ieee.org/document/5178965

[4] Moody GE. Spontaneous termination of atrial fibrillation: A challenge from physionet and computers in cardiology 2004. In: Computers in Cardiology, 2004, Chicago, IL, USA. IEEE; 2004. pp. 101-104. Available from: https://ieeexplore.ieee.org/document/1442881

[5] Al-Jowder O, Kemsley EK, Wilson RH. Detection of adulteration in cooked meat products by mid-infrared spectroscopy. Journal of Agricultural and Food Chemistry. 2002;**50**(6): 1325-1329

[6] Forestier G, Petitjean F, Senin P, Despinoy F, Huaulmé A, Fawaz HI, et al. Surgical motion analysis using discriminative interpretable patterns. Artificial Intelligence in Medicine. 2018;**91**: 3-11

[7] Ismail Fawaz H, Forestier G, Weber J, Idoumghar L, Muller PA. Evaluating surgical skills from kinematic data using convolutional neural networks. In: Frangi AF, Schnabel JA, Davatzikos C, Alberola-López C, Fichtinger G, editors. Medical Image Computing and Computer Assisted Intervention – MICCAI 2018, Lecture Notes in Computer Science. Cham: Springer International Publishing; 2018. pp. 214-221

[8] Ismail Fawaz H, Forestier G, Weber J, Petitjean F, Idoumghar L, Muller PA. Automatic alignment of surgical videos using kinematic data. In: Riaño D, Wilk S, ten Teije A, editors. Artificial Intelligence in Medicine, Lecture Notes in Computer Science. Cham: Springer International Publishing; 2019. pp. 104-113

[9] Sakoe H, Chiba S. Dynamic programming algorithm optimization for spoken word recognition. IEEE Transactions on Acoustics, Speech, and Signal Processing. 1978;**26**(1):43-49

[10] Bentley JL. Multidimensional binary search trees used for associative searching. Communications of the ACM. 1975;**18**(9):509-517

[11] Omohundro SM. Five Balltree Construction Algorithms. Technical Report. Berkeley, CA: International Computer Science Institute; 1989. Available from: https://www.icsi.berkeley.edu/icsi/node/2291

[12] Itakura F. Minimum prediction residual principle applied to speech recognition. IEEE Transactions on Acoustics, Speech, and Signal Processing. 1975;**23**(1):67-72

[13] Müller M, Mattes H, Kurth F. An efficient multiscale approach to audio synchronization. In: Proceedings of the Sixth International Conference on Music Information Retrieval, Victoria, BC, Canada. 2006. pp. 192-197

[14] Salvador S, Chan P. Toward accurate dynamic time warping in linear time and space. Intelligent Data Analysis. 2007; **11**(5):561-580

[15] Cuturi M, Blondel M. Soft-DTW: A differentiable loss function for time-series. arXiv:170301541 [stat] [Internet]. 2018. Available from: http://arxiv.org/abs/1703.01541 [Accessed: Mar 28, 2020]

[16] Cuturi M. Fast global alignment kernels. In: Proceedings of the 28th International Conference on International Conference on Machine Learning (ICML'11). Madison, WI, USA: Omnipress; 2011. pp. 929-936

[17] Lines J, Davis LM, Hills J, Bagnall A. A Shapelet transform for time series classification. In: Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'12) [Internet]. New York, NY, USA: ACM; 2012. pp. 289-297. DOI: 10.1145/2339530.2339579 [Accessed: Jul 8, 2019]

[18] Hills J, Lines J, Baranauskas E, Mapp J, Bagnall A. Classification of time series by shapelet transformation. Data Mining and Knowledge Discovery. 2014;**28**:851-881. Available from: https://link.springer.com/article/10.1007/s10618-013-0322-1

[19] Bostrom A, Bagnall A. Binary shapelet transform for multiclass time series classification. In: International Conference on Big Data Analytics and Knowledge Discovery. 2015. Available from: https://link.springer.com/chapter/10.1007/978-3-319-22729-0_20

[20] Grabocka J, Schilling N, Wistuba M, Schmidt-Thieme L. Learning time-series Shapelets. In: Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '14 [Internet]. New York, New York, USA: ACM Press; 2014. pp. 392-401. Available from: http://dl.acm.org/citation.cfm?doid=2623330.2623613 [Accessed: 2019 Jul 8]

[21] Breiman L. Random forests. Machine Learning. 2001;**45**(1):5-32

[22] Geurts P, Ernst D, Wehenkel L. Extremely randomized trees. Machine Learning. 2006;**63**(1):3-42

[23] Deng H, Runger G, Tuv E, Vladimir M. A time series forest for classification and feature extraction. Information Sciences. 2013;**239**:142-153

[24] Baydogan MG, Runger G, Tuv E. A bag-of-features framework to classify time series. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2013;**35**(11):2796-2802

[25] Lucas B, Shifaz A, Pelletier C, O'Neill L, Zaidi N, Goethals B, et al. Proximity Forest: An effective and scalable distance-based classifier for time series. Data Mining and Knowledge Discovery. 2019;**33**(3):607-635

[26] Lin J, Keogh E, Wei L, Lonardi S. Experiencing SAX: A novel symbolic representation of time series. Data Mining and Knowledge Discovery. 2007; **15**(2):107-144

[27] Lin J, Khade R, Li Y. Rotation-invariant similarity in time series using bag-of-patterns representation. Journal of Intelligent Information System. 2012; **39**(2):287-315

[28] Senin P, Malinchik S. SAX-VSM: Interpretable time series classification

using SAX and vector space model. In: 2013 IEEE 13th International Conference on Data Mining, Dallas, TX, USA. IEEE; 2013. pp. 1175-1180. Available from: https://ieeexplore.ieee.org/document/6729617

[29] Schäfer P, Högqvist M. SFA: A symbolic fourier approximation and index for similarity search in high dimensional datasets. In: Proceedings of the 15th International Conference on Extending Database Technology – EDBT'12 [Internet]. Berlin, Germany: ACM Press; 2012. p. 516. Available from: http://dl.acm.org/citation.cfm?doid=2247596.2247656 [Accessed: Feb 18, 2019]

[30] Schäfer P. The BOSS is concerned with time series classification in the presence of noise. Data Mining and Knowledge Discovery. 2015;**29**(6): 1505-1530

[31] Schäfer P. Scalable time series classification. Data Mining and Knowledge Discovery. 2016;**30**(5): 1273-1298

[32] Middlehurst M, Vickers W, Bagnall A. Scalable dictionary classifiers for time series classification. In: Yin H, Camacho D, Tino P, Tallón-Ballesteros AJ, Menezes R, Allmendinger R, editors. Intelligent Data Engineering and Automated Learning – IDEAL 2019, Lecture Notes in Computer Science. Cham: Springer International Publishing; 2019. pp. 11-19

[33] Large J, Bagnall A, Malinowski S, Tavenard R. On time series classification with dictionary-based classifiers. Intelligent Data Analysis. 2019;**23**(5): 1073-1089

[34] Schäfer P, Leser U. Fast and accurate time series classification with WEASEL. In: Proceedings of the 2017 ACM on Conference on Information and Knowledge Management - CIKM'17. New York, NY, USA: Association for Computing Machinery; 2017. pp. 637-646

[35] Schäfer P, Leser U. Multivariate Time Series Classification with WEASEL +MUSE. arXiv:171111343 [cs] [Internet]. 2017. Available from: http://arxiv.org/abs/1711.11343 [Accessed: Jun 17, 2019]

[36] Middlehurst M, Large J, Cawley G, Bagnall A. The temporal dictionary ensemble (TDE) classifier for time series classification. ECML/PKDD. 2020;**12457**: 660-676

[37] Eckmann JP, Kamphorst SO, Ruelle D. Recurrence plots of dynamical systems. EPL. 1987;**4**(9):973-977

[38] Campana BJL, Keogh EJ. A compression-based distance measure for texture. Statistical Analysis and Data Mining: The ASA Data Science Journal. 2010;**3**(6):381-398

[39] Silva DF, Souza VMAD. Batista GEAPA. Time series classification using compression distance of recurrence plots. In: 2013 IEEE 13th International Conference on Data Mining, Dallas, TX, USA. IEEE; 2013. pp. 687-696. Available from: https://ieeexplore.ieee.org/document/6729553

[40] Tamura K, Ichimura T. Time series classification using MACD-histogram-based recurrence plot. International Journal of Computational Intelligence Studies [Internet]. 2018 [Accessed: Jul 30, 2021]. DOI: 10.1504/IJCISTUDIES.2018.096188

[41] Hatami N, Gavet Y, Debayle J. Classification of time-series images using deep convolutional neural networks. In: Tenth International Conference on Machine Vision (ICMV 2017) [Internet], Vienna, Austria. International Society for

Optics and Photonics; 2018. p. 106960Y. Available from: https://icmv.org/photo2017.html. DOI: 10.1117/12.2309486.short [Accessed: Jul 30, 2021]

[42] Wang Z, Oates T. Imaging time-series to improve classification and imputation. In: Proceedings of the 24th International Conference on Artificial Intelligence (IJCAI'15); 25-31 July 2015; Nuenos Aires, Argentina. AAAI Press; 2015. pp. 3939-3945

[43] Karimi-Bidhendi S, Munshi F, Munshi A. Scalable classification of univariate and multivariate time series. In: 2018 IEEE International Conference on Big Data (Big Data), Seattle, WA, USA. IEEE; 2018. pp. 1598-1605. Available from: https://ieeexplore.ieee.org/document/8621889

[44] Ismail Fawaz H, Forestier G, Weber J, Idoumghar L, Muller PA. Deep learning for time series classification: A review. Data Mining and Knowledge Discovery. 2019;**33**(4):917-963

[45] Ismail Fawaz H, Forestier G, Weber J, Idoumghar L, Muller PA. Deep neural network ensembles for time series classification. In: 2019 International Joint Conference on Neural Networks (IJCNN), Budapest, Hungary. IEEE; 2019. pp. 1-6. Available from: https://ieeexplore.ieee.org/document/8852316

[46] Wang Z, Yan W, Oates T. Time series classification from scratch with deep neural networks: A strong baseline. In: 2017 International Joint Conference on Neural Networks (IJCNN), Anchorage, AK, USA. IEEE; 2017. pp. 1578-1585. Available from: https://ieeexplore.ieee.org/document/7966039

[47] Serrà J, Pascual S, Karatzoglou A. Towards a universal neural network encoder for time series. arXiv:180503908 [cs, stat] [Internet]. 2018. Available from: http://arxiv.org/abs/1805.03908 [Accessed: Aug 10, 2021]

[48] Zheng Y, Liu Q, Chen E, Ge Y, Zhao JL. Time series classification using multi-channels deep convolutional neural networks. In: Li F, Li G, Won HS, Yao B, Zhang Z, editors. Web-Age Information Management, Lecture Notes in Computer Science. Cham: Springer International Publishing; 2014. pp. 298-310

[49] Zhao B, Lu H, Chen S, Liu J, Wu D. Convolutional neural networks for time series classification. Journal of Systems Engineering and Electronics. 2017;**28**(1):162-169

[50] Ismail Fawaz H, Lucas B, Forestier G, Pelletier C, Schmidt DF, Weber J, et al. InceptionTime: Finding AlexNet for time series classification. Data Mining and Knowledge Discovery. 2020;**34**(6):1936-1962

[51] Ismail Fawaz H, Forestier G, Weber J, Idoumghar L, Muller PA. Transfer learning for time series classification. In: 2018 IEEE International Conference on Big Data (Big Data), Seattle, WA, USA. IEEE; 2018. pp. 1367-1376. Available from: https://ieeexplore.ieee.org/document/8621990

[52] Fawaz HI, Forestier G, Weber J, Idoumghar L, Muller PA. Data augmentation using synthetic data for time series classification with deep residual networks. arXiv:180802455 [cs] [Internet]. 2018 [Accessed: Aug 10, 2021]. Available from: http://arxiv.org/abs/1808.02455

[53] Ismail Fawaz H, Forestier G, Weber J, Idoumghar L, Muller PA. Adversarial attacks on deep neural networks for time series classification. In: 2019 International Joint Conference on Neural Networks (IJCNN), Budapest, Hungary. IEEE; 2019. pp. 1-8. Available from:

https://ieeexplore.ieee.org/document/8851936

[54] Rakhshani H, Ismail Fawaz H, Idoumghar L, Forestier G, Lepagnot J, Weber J, et al. Neural architecture search for time series classification. In: 2020 International Joint Conference on Neural Networks (IJCNN), Glasgow, UK. IEEE; 2020. pp. 1-8. Available from: https://ieeexplore.ieee.org/document/9206721

[55] Dempster A, Petitjean F, Webb GI. ROCKET: Exceptionally fast and accurate time series classification using random convolutional kernels. Data Mining and Knowledge Discovery. 2020; **34**(5):1454-1495

[56] Rifkin RM, Lippert RA. Notes on Regularized Least Squares. Technical Report. Cambridge, MA, USA: Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology; 2007. Available from: https://dspace.mit.edu/handle/1721.1/37318 [Accessed: Aug 10, 2021]

[57] Dempster A, Schmidt DF, Webb GI. MiniRocket: A very fast (almost) deterministic transform for time series classification. In: Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining [Internet]. New York, NY, USA: Association for Computing Machinery (KDD'21); 2021. pp. 248-257. DOI: 10.1145/3447548.3467231 [Accessed: Aug 19, 2021]

[58] Tan CW, Dempster A, Bergmeir C, Webb GI. MultiRocket: Effective summary statistics for convolutional outputs in time series classification. ArXiv. 2021

[59] Bagnall A, Lines J, Hills J, Bostrom A. Time-series classification with COTE: The collective of transformation-based ensembles. IEEE Transactions on Knowledge and Data Engineering. 2015; **27**(9):2522-2535

[60] Lines J, Taylor S, Bagnall A. Time series classification with HIVE-COTE: The hierarchical vote collective of transformation-based ensembles. ACM Transactions on Knowledge Discovery from Data. 2018;**12**(5):52:1-52:35

[61] Bagnall A, Flynn M, Large J, Lines J, Middlehurst M. On the usage and performance of the hierarchical vote collective of transformation-based ensembles version 1.0 (HIVE-COTE v1.0). In: Lemaire V, Malinowski S, Bagnall A, Guyet T, Tavenard R, Ifrim G, editors. Advanced Analytics and Learning on Temporal Data, Lecture Notes in Computer Science. Cham: Springer International Publishing; 2020. pp. 3-18

[62] Middlehurst M, Large J, Flynn M, Lines J, Bostrom A, Bagnall A. HIVE-COTE 2.0: A new meta ensemble for time series classification. arXiv: 210407551 [cs] [Internet]. 2021. Available from: http://arxiv.org/abs/2104.07551 [Accessed: Aug 9, 2021]

[63] Shifaz A, Pelletier C, Petitjean F, Webb GI. TS-CHIEF: A scalable and accurate forest algorithm for time series classification. Data Mining and Knowledge Discovery. 2020;**34**(3):742-775

[64] Bagnall A, Lines J, Bostrom A, Large J, Keogh E. The great time series classification bake off: A review and experimental evaluation of recent algorithmic advances. Data Mining and Knowledge Discovery. 2017;**31**(3):606-660

[65] Dau HA, Keogh E, Kamgar K, Yeh CCM, Zhu Y, Gharghabi S, et al. The UCR Time Series Classification Archive

[Internet]. 2018. Available from: https://arxiv.org/abs/1810.07758 [Preprint]

[66] Bagnall A, Dau HA, Lines J, Flynn M, Large J, Bostrom A, et al. The UEA multivariate time series classification archive, 2018. arXiv:181100075 [cs, stat] [Internet]. 2018 [Accessed: May 28, 2019]. Available from: http://arxiv.org/abs/1811.00075

[67] Baydogan MG. Multivariate Time Series Classification Datasets [Internet]. 2017. Available from: http://www.mustafabaydogan.com/

[68] Bagnall A, Lines J, Vickers W, Keogh E. The UEA & UCR time series classification repository [Internet]. Available from: http://www.timeseriesclassification.com/ [Accessed: Aug 26, 2019]

[69] Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, et al. Scikit-learn: Machine learning in python. Journal of Machine Learning Research. 2011;**12**:2825-2830

[70] Kuhn M. Building predictive models in R using the caret package. Journal of Statistical Software. 2008;**28**(1):1-26

[71] Faouzi J, Janati H. Pyts: A python package for time series classification. Journal of Machine Learning Research. 2020;**21**(46):1-6

[72] Löning M, Bagnall A, Ganesh S, Kazakov V, Lines J, Király FJ. Sktime: A Unified Interface for Machine Learning with Time Series. arXiv:190907872 [cs, stat] [Internet]. 2019 [Accessed: Jul 30, 2021]. Available from: http://arxiv.org/abs/1909.07872

[73] Tavenard R, Faouzi J, Vandewiele G, Divo F, Androz G, Holtz C, et al. Tslearn, a machine learning toolkit for time series data. Journal of Machine Learning Research. 2020;**21**(118):1-6

[74] Burns DM, Whyne CM. Seglearn: A python package for learning sequences and time series. Journal of Machine Learning Research. 2018;**19**(83):1-7

[75] Christ M, Braun N, Neuffer J, Kempa-Liehr AW. Time series FeatuRe extraction on basis of scalable hypothesis tests (tsfresh – A python package). Neurocomputing. 2018;**307**:72-77

[76] Naul B, van der Walt S, Crellin-Quick A, Bloom J, Pérez F. Cesium: Open-Source Platform for Time-Series Inference Austin, Texas. 2016. pp. 27-35. Available from: https://conference.scipy.org/proceedings/scipy2016/brett_naul.html [Accessed: Aug 26, 2019]

[77] Bagnall A, Lines J, Vickers W, Keogh E. The UEA & UCR Time Series Classification Repository [Internet]. Available from: www.timeseriesclassification.com

Section 2

Biomedical Time Series Analysis

Chapter 3

A Heartbeat-Interval Time Series Analysis, Modified Detrended Fluctuation Analysis, mDFA, Distinguishes between Stressed- and Happy-Heartbeats: From Invertebrate Animals to Humans

Toru Yazawa

Abstract

This study provides evidence that a time series analysis (modified detrended fluctuation analysis, mDFA) is practically distinguish happy- and stressed hearts. This endures that the scaling exponent (scaling index, SI, or alpha, α) can characterize the state of heartbeats. We learned from various challenges of case studies; for example, the Wolff–Parkinson–White syndrome yields a high SI (way surpass 2.0) while feeling sick condition, but the same heart exhibits a healthy SI (~1.0) when the heartbeats return to normal. Meantime, a healthy SI (~1.0) goes down to a low SI (0.7) when truly enjoying meal. It seems that SI can represent invisible internal world. The complex interaction between the cardiac rhythm and the autonomic brain command becomes perceptible by the SI. Our observations confirm the state of the heart is measurable quantitatively. A time series analysis of mDFA can help holistic understanding of the brain-heart axis.

Keywords: heartbeat, time series analysis, scaling exponent, invertebrate hearts, stressed hearts, healthy hearts

1. Introduction

The cardiovascular system includes the heart, intrinsic nerve ganglia, and afferent/efferent extrinsic nerves. It maintains brain-heart health [1–3]. When it goes wrong, we get sick. It is a life-threatening cardiovascular disease (CVD) in the worst-case scenario.

CVD is caused by changes in the qualitative dynamics of physiological control systems [4]. Earlier detection of a sign of changes at the beginning of a disease (i.e., silent onset of a disease) is ideal if we oppose it. The changes often lead to alterations of rhythm, from normal one to pathological ones. Actually, it is said that beat-to-beat variations of heart rate reflect the sign of changes [5].

In the 1990s, Peng et al. [6] proposed the idea that heartbeat time series analysis helps reduce CVD. In clinical medicine today, however, can we decode the fluctuations in physiological rhythms to better diagnose human disease? The answer to the question is, No. Leon Glass [7] mentioned: The mathematical analyses of temporal properties of physiological rhythms have not yet led to medical advances. Others also said that the mathematical rhythm analysis is still far away from clinical medicine, and clinical utility is not established [8].

In short, a mathematical understanding of the complex fluctuations of cardiac rhythms has been unsuccessful in predicting a heart attack. In fact, the US Preventive Task Force (USPSTF) recommended against screening with the electrocardiogram (EKG) to prevent CVD events in asymptomatic adults at low risk of CVD events (see the latest USPSTF in 2022 as well as in 2018 [9, 10]).

However, since Einthoven's invention of EKG, EKG has become a recognized electrophysiological technique (see PQRST EKG wave, **Figure 1A**). Today, EKG is a canonical method of testing/diagnosing/understanding the diseased heart.

By observing abnormal configurations of EKGs (**Figure 1**), medical doctors can get diagnostic ideas: for example, Wolff–Parkinson–White syndrome (WPW syndrome, blue dotted line) and ST segment issues (elevation or depression, black dotted line). The doctors can measure the abnormality of the Q-T interval, if any. So, EKG is a useful diagnostic tool. Despite this, the USPSTF recommended against "screening" with EKG, as mentioned above.

Now, questions arise. What is wrong with EKG (i.e., heartbeat data) for screening? Does heart rate variability (HRV) analysis, which is based on EKGs, has any problem with screening? Our answer is: problems might be data-collection- or data analysis methods, which would blur the boundary/distinction between the healthy and pathological data.

As neurobiologists, we consider that sophisticated HRV researchers (e.g., physicists or mathematicians) use EKGs not collected by themselves. So, they might not be sure precisely the circumstances when the EKGs were taken. We decided that we must collect EKGs by ourselves, and we did so in the present project (which started around 2000).

We have already shown the distinction between the healthy and pathological data using model animals, crustaceans [11, 12]. We reported that heartbeat time series constructed from EKG contribute to understanding the cardiovascular system [11, 12].

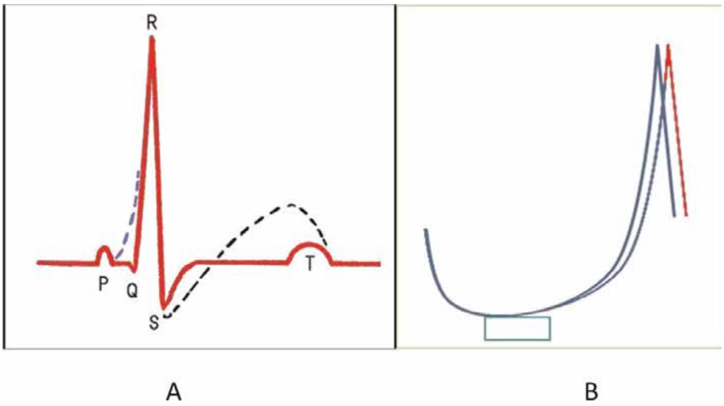


Figure 1.
(A) Diagrammatic representation of EKG waves (PQRST) showing normal (red line) and abnormal (dotted lines in blue and black) states. (B) Diagrammatic representation of action potentials (AP, intracellular membrane potential) of excitable cells such as myocardial cells and nerve cells. Note the unstable peak time of AP. Depolarizing stimulation slightly above the threshold (green square period) induces two superimposed action potentials (blue and red).

Although the USPSTF says that EKG has no benefits in the assessment of future CVD risk, we feel sure that EKG carries invisible information to understand the cardiovascular regulatory system. For the practical use of EKG in medicine, we report that reliable data and reliable data analysis will not blur the distinction between healthy and pathological data. We thus propose an idea regarding the time series analysis of heartbeat data. This has been conceived from neurobiological experiments.

2. Method

2.1 Accuracy and fidelity in sampling raw EKG

2.1.1 Sampling frequency: sampling rate

As shown in **Figure 1B**, neurobiologists have experiences: the period length from the onset of stimulation to the peak of the action potential is not stable but changeable. The peak time of the action potential fluctuates each time after the same stimulation. In particular, the fluctuation can be seen when being stimulated just above threshold depolarization. In the brain-heart axis, this fluctuation might cause fluctuation in heartbeat intervals. Therefore, heartbeat-interval time series involves complex fluctuations due to constantly changing subsystems in the complex cardioregulatory mechanisms, including neurotransmitters, ion channels, excitation–contraction coupling of myocardial cells, etc.

In a neuronal cell, the minimum fluctuation ranges 1–10 ms (e.g., Figure 2A shown in Ref. [13]). In a myocardial cell, the fluctuation ranges up to 50 ms (e.g., Figure 4 of Widemann's paper [14]).

From the above consideration, we determined our sampling rate 1 KHz (every 1 ms) to capture high-fidelity fluctuations.

2.1.2 EKG amplifier and input time constant (τ)

Internationally, the time constant for EKG amplifier is determined as a few seconds (3.2 sec in Japan Industrial Standard). Thanks to this "slow" time constant, we can easily recognize the appearance of abnormal traces (relatively slowly changing voltage traces) in EKG, such as **Figure 1A** blue and black lines. However, if a patient moves during EKG recording, strong movement generates a large noise (see white arrows in **Figure 2**) that induces scaling out from the screen of PC (see **Figure 2A**). In such occasions, a part of EKG's voltage traces are not recorded. In short, a few heartbeats are not registered (see **Figure 2A**, heartbeats numbered 3, 4, 5, and 6, shown in black). Thus, slow time constant machines lead to imperfect data acquisition if individuals move. Then, we fail to construct an accurate heartbeat time series.

For that reason, we determined our EKG amplifier has a small time constant (0.1 s), as shown in **Figure 2B**. Here, a large movement noise does not induce a large scaling out phenomenon (white arrow in **Figure 2B**). **Figure 3** shows our EKG amplifier's electric circuit, no other fancy circuit such as a hum-filter was added. The capacitance of the input capacitor is 0.1 μF and the resistance of the input resistor is 1 MΩ. Therefore, input time constant τ is 0.1 s ($C \times F = 0.1 \times 1$) (**Figure 3**).

Figure 4 shows how to record EKG. Three disposable EKG electrodes (R, G, and Y) (Vitrode V, Nihonkoden, Tokyo, Japan) are attached to the body, and lead cables (carbon fiber) are connected to the EKG amplifier. The EKG amplifier is put in a pocket of individuals, and EKG signal is sent through a long cable (5 m) to the logger (PowerLab,

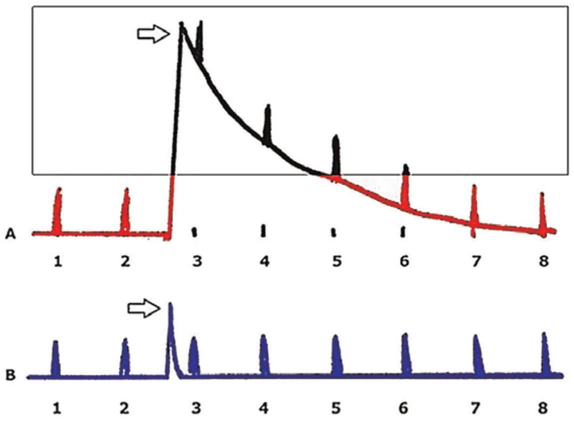


Figure 2.
Diagrammatic explanation for the time constant (τ) of input of an EKG amplifier. A, τ = 5 s, B, τ = 0.1 s. arrows indicate noise abruptly induced by the body movement. Square area data is not registered due to scaling out.

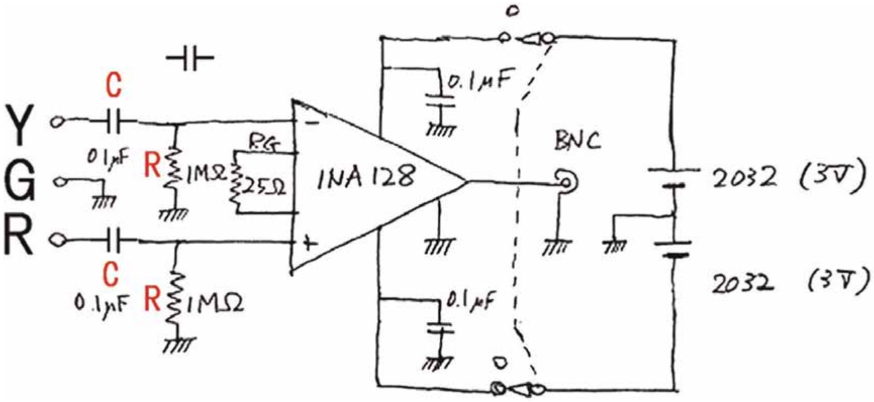


Figure 3.
EKG amplifier circuit, diagrammatically demonstrating the input time constant (τ). Here, τ = C × R = 0.1 s. (C, 0.1 μF. R, 1 MΩ). Courtesy of prof. Dr. Yukio Shimoda, Tokyo Women's medical university. See ***Figure 4*** *for the contact pins Y, G, and R.*

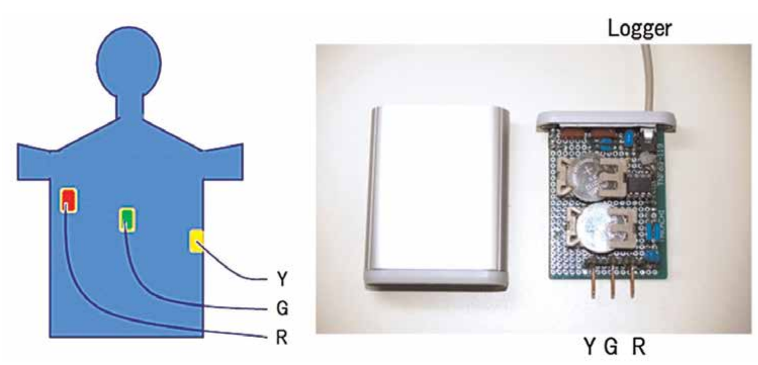


Figure 4.
EKG setup: Commercially available electrodes and self-made amplifier.

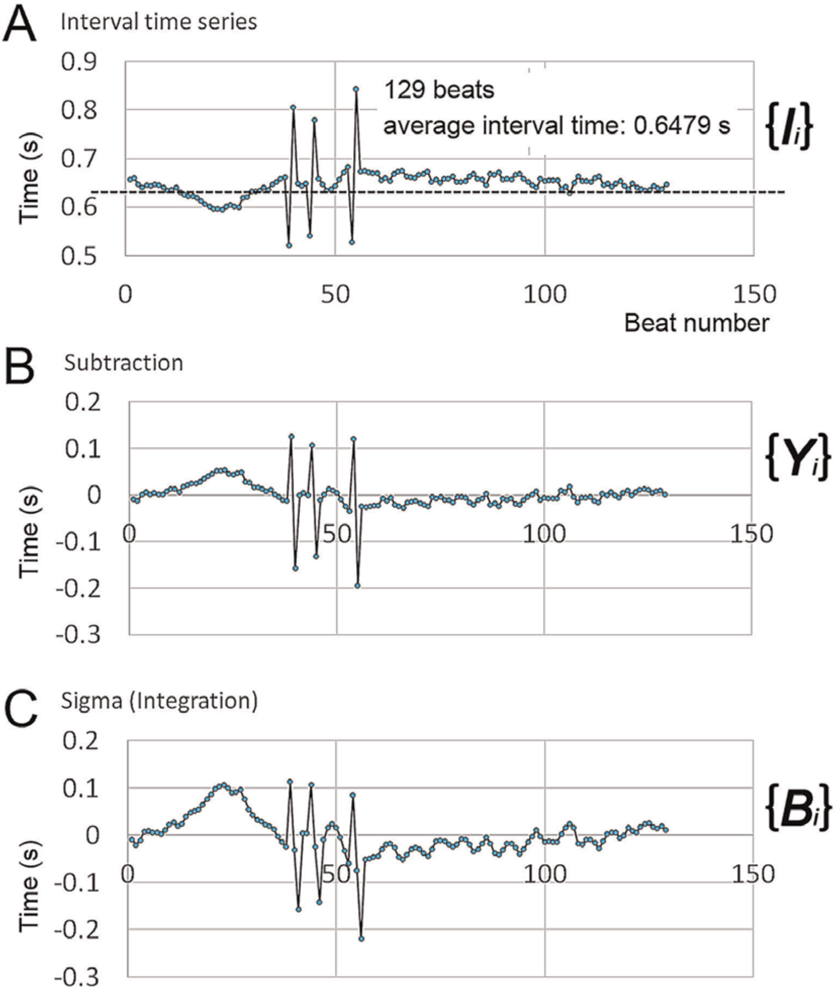


Figure 5.
Preprocessing for mDFA (see text). Volunteer: German in her 50s. Three arrhythmic heartbeats, premature ventricular contractions (PVCs) are included (volunteer subjects put their sign on a written certificate. The statement of mutual agreement describes that EKGs are used only for research and hold out no traceability of personal information. The human heartbeats were recorded outside of a hospital, in for example university laboratories and convention halls - the innovation Japan exhibition. All subjects were treated as per the ethical control regulations of the universities, Tokyo Metropolitan University; Tokyo Women's medical university).

AD Instruments, Australia). Therefore, individuals can freely walk around across 10 m circle area (**Figure 4**). Output from the digital logger is stored in PC.

From these methods, we can capture accurate and high-fidelity EKG data. To date, we met over 400 different individuals to record EKGs, all of them outside the hospital. As for invertebrate EKG recordings, we already have over 1000 data. All EKG collections were conducted according to the ethical code of the university (see also **Figure 5**).

2.2 Construction of the heartbeat-interval time series

Figure 6A shows EKG data recorded by PowerLab. We format EKG as Text, that is, a voltage vs. time data set by the PowerLab program. Then, we transfer it to our

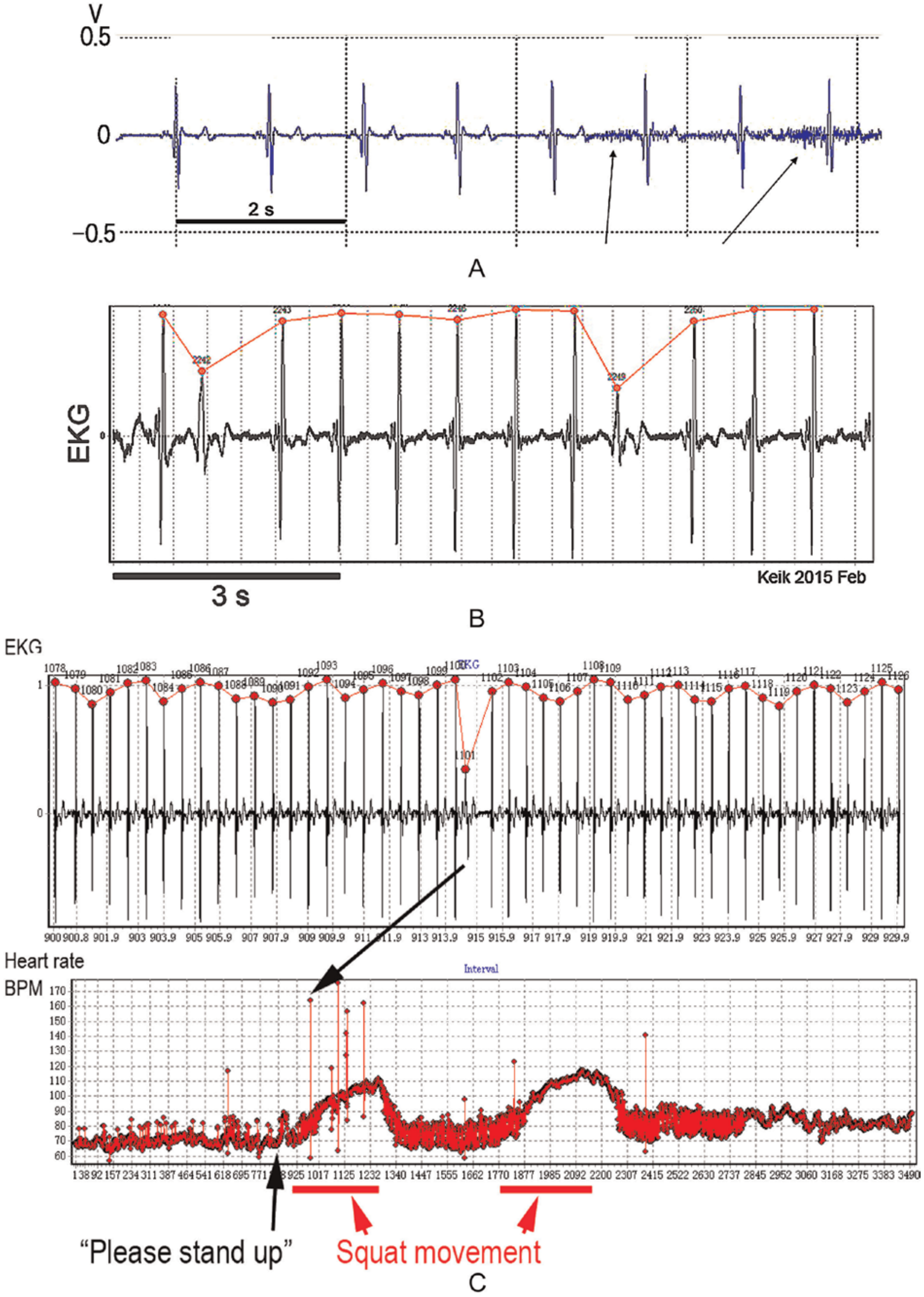


Figure 6.
(A) A raw EKG recorded by the amplifier with input τ = 0.1 s. note, despite the subject raised hands up and down, the EKG baseline is stable with very small noises (arrows). (B) R-peak detections with our program (points in red). (C) Calculation of peak-to-peak interval time, constructing the heart rate time series (beats per min, BPM) or interval time series (not shown). Note that a subject is asked to stand up from a chair and do two sets of 50-time squatting.

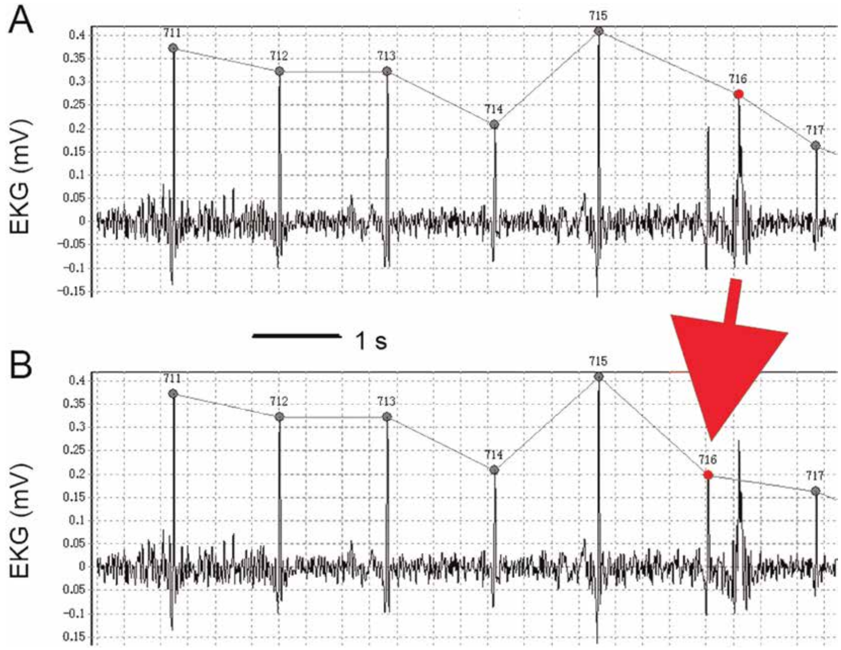


Figure 7.
Correction of a false peak (an arrow). (A) False. (B) Correct. We do all corrections manually by eye observation on the PC screen.

own program, which captures R-peaks (**Figure 6B**), and construct heartbeat-interval time series (**Figure 6C**). If the program captures an inaccurate R-peak, such as noses, we make changes by eyes (**Figure 7**). We do this on every EKG. Thus, the present study accurately captured all of the biological R-peaks. Thus, the authenticity of the time series is ensured.

2.3 Time series analysis

2.3.1 Crustacean model animals

On the spiny lobster heartbeats, we have reported that the isolated heart (natural brain control is lost) and the intact heart (neuronal and neurohumoral control are intact) are distinguishable from each other [11, 12]. In the report [12], we used a method, detrended fluctuation analysis (DFA) proposed by Peng et al. [6, 15], and found that DFA is a good method to distinguish between a healthy heartbeat and an abnormal heartbeat of a model animal. But, unfortunately, as mentioned above, heartbeat rhythm analysis is still not remarkably contributing to clinical medicine [7–10].

After studying the details of the computation procedures of DFA (see [6, 15] for a full explanation of DFA), we made a new method with a different concept based on neurobiological experiments.

DFA deals with “critical” phenomena existing in nature. We understand why Peng et al. [6] focused on “criticality” of a diseased heart [6, 15] because “sudden” heart attack is a big health issue, in cardiac medicine. However, for normal people, it is rare that the heartbeats suddenly encounter a critical moment. We created a different method, modified detrended fluctuation analysis (mDFA) [16, 17] with neurobiological thinking.

2.3.2 mDFA: data preparation

Each EKG recording is a long-lasting continuous recording, spanning over 30 min to 10 hr. or more in our project. The peak checking on the PC screen is formidable work, but the peak accuracy is necessary condition (**Figure 7**). Peak-capture failure surely messes up the "statistical" conclusions. Many results have been reported by "highly sophisticated physical" methods. We want to avoid inaccurate data preparation.

We only use the heartbeat-interval time series constructed in our lab. mDFA would otherwise ultimately yield a monstrous result. Perfect logging and accurate peak-to-peak sampling are ideal.

2.3.3 mDFA and DFA: algorithm

There are important conceptual differences in the algorithm between DFA and mDFA that were already mentioned elsewhere in 2015 and 2019 [16, 17]. Here, we shall explain it briefly.

Figure 5A shows the time series $\{I\text{i}\}$. This is the set of times for peak-to-peak intervals. **Figure 5A** involves 129 interval data, from I_1 to I_{129}.

The dotted line is the average of interval time from 129 intervals. Here, the average value $< I >$ is 0.6479 s.

Figure 5B shows the set $\{Y\text{i}\}$. This is the set of times of "real" fluctuations calculated as: $\{Y\text{i}\} = < I > - I\text{i}$. Either ($Y\text{i} = I\text{i} - < I >$) or ($Y\text{i} = < I > - I\text{i}$) can be usable. But if squared, $(< I > - I\text{i})^2 = (I\text{i} - < I >)^2$, both became equal.

Next, **Figure 5C** shows the integrated time series, that is, $\{B\text{i}\} = \sum_{\text{k}=1}^{129} Y\text{k}$. Here, the integration process, adding each value of $\{Y\text{i}\}$ successively, is plotted.

Interestingly, it is remarkable that a time segment of "persistent acceleration" or "persistent inhibition (deceleration)" is intensified by the integration (see beat numbers ranging over 1–40, **Figure 5C**).

In contrast, sporadically appearing events, that is, PVCs, are not intensified (see three PVCs, **Figure 5C**). Therefore, persistently occurring events (autonomic acceleration or deceleration) and sporadically occurring events (arrhythmic heartbeats) may have different effects on mDFA outcomes. But, without time series analysis, little is transparently noticeable when watching EKG alone. The time series data is so great and of interest.

The "sigma or integration" method is the most important step of mDFA. The trajectory of the set $\{B\text{i}\}$ is more dynamic than $\{Y\text{i}\}$ (**Figure 5B** and **C**). Again, constructing the set $\{B\text{i}\}$ is the key step for mDFA (and DFA as well). It is a random walk-like analysis concept.

2.3.4 mDFA: fitting

In mDFA, the integrated time series {Bi} includes 2000 sets of times. Practically, it takes at least 30 min for EKG recording.

In DFA, not mDFA, the integrated time series {Bi} is divided into boxes of equal length n, and a least squares line is fit to the data.

However, in mDFA, {Bi} is NOT divided into boxes of equal length n before the fitting to the data. In mDFA, at first, the biquadratic fitting line is made, then divided into box of equal length n. There are n samples in a box.

We have tested linear, quadratic, cubic, fourth-degree polynomial, fifth, sixth, etc. The results of linear, quadratic, and cubic fittings were scattered. At a greater than fourth degree polynomial fitting, the resulting scaling exponent is converged to a number. In mDFA, therefore, a biquadratic curve fitting is used (**Figure 8**).

2.3.5 mDFA: detrending

Computation of a least squares line fit (LSLF) to the data {*B*i} reveals the trend line (**Figure 9A**). In mDFA, the trend line is a biquadratic line, but here we will use linear fitting for the sake of convenience. We detrend the interval data {*B*i} by subtracting the trend (**Figure 9B**).

In the DFA algorithm, LSLF is made in each box, but in the mDFA algorithm, LSLF is done upon entire interval data {*B*i}. Why do we do so? Because we consider that the heartbeat change is a continuous phenomenon. Past incidences always affect future incidences to a lesser or greater degree. So, the first heartbeat in a time series has an influence on following heartbeats to a lesser or greater degree. Therefore, when drawing a fitting line (LSLF), mDFA computes the entire trend at once, not inside a box, as DFA does (representing the trend in that box). This notion of mDFA is a totally different concept from Peng's DFA concept.

2.3.6 mDFA: root mean square and box size

Imagine, in the eq. $Y = a \times X$, a determines the slope in the X vs. Y graph. A linear relationship exists between Y and X. Also, $\log Y = a \times \log X$, where a also gives the slope. This equation is compatible to the eq. $Y = X^a$. It is a double-log plotting graph. A linear relationship on a double-log graph indicates the presence of scaling, that is, $F(n) = n^{\alpha}$. The scaling index α can be determined from a log-log graph. (Abbreviation for the scaling index; we use *SI* or α).

In **Figure 9B**, the box size n is 21. The differen*ce*s between {*Bi*} and fit line are, $A_{21,1} - B_{21,1}$, $A_{21,2} - B_{21,2}$, $A_{21,3} - B_{21,3}$, — — —, that is, the length of vertical lines in **Figure 9B** (see vertical straight line segments in **Figure 9B**).

If A > B, A – B gives a positive value, and if A < B, A – B returns a negative value. Therefore, when we consider the average value of (A – B), we use the root mean square, which is $F^2 = \langle (A - B)^2 \rangle$, as a matter of practical convenience. This calculation is what Peng's DFA program does. This computation is repeated over all of the box sizes to provide a relationship between F and n.

In contrast to the DFA algorithm, mDFA algorithm does not use A – B. Instead, mDFA uses subtractions $B_{21,1} - B_{21,21}$ in one box, where the box size is 21 (see the dashed line in **Figure 9B**). So, $B_{21,1}$ is the first data of the box (*Entrance*), and $B_{21,21}$ is the last data of the box (*Exit*). Therefore, mDFA computes the subtraction (*Entrance – Exit*), or the difference between the value of *Entrance* and the value of the *Exit*. This computation is repeated over all of the box sizes to provide a relationship between F and n.

Why do we do so? The answer is: we want to illuminate the phenomena more slowly, varying with time. Vertical subtraction A – B focuses on the critical moment of relevant issues, that is, cross-section of time. In contrast, the subduction (*Entrance – Exit*) is focusing on phenomena varying with time, that is,

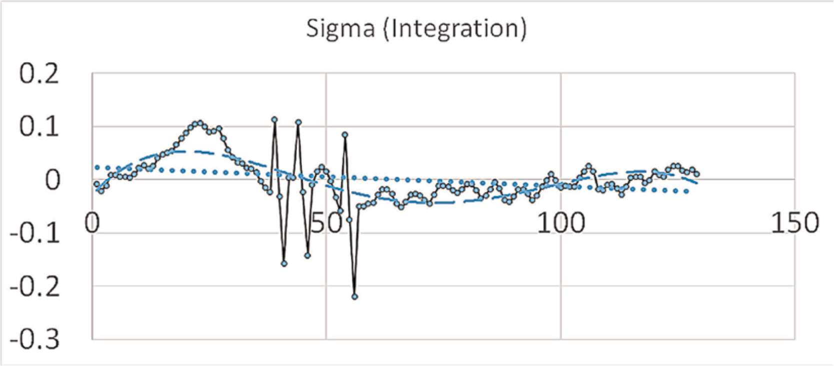


Figure 8.
An example least squares line fit to the data. Fitted curves: Dotted line (…), linear fitting, and dashed line (— — —), biquadratic fitting. Here, the number of heartbeats shown is only 129 beats. In mDFA, biquadratic fitting line is made from 2000 data, instead of within a box. Peng's DFA, however, makes the fitting line within individual boxes (see Peng et al. [6] for the details of "box").

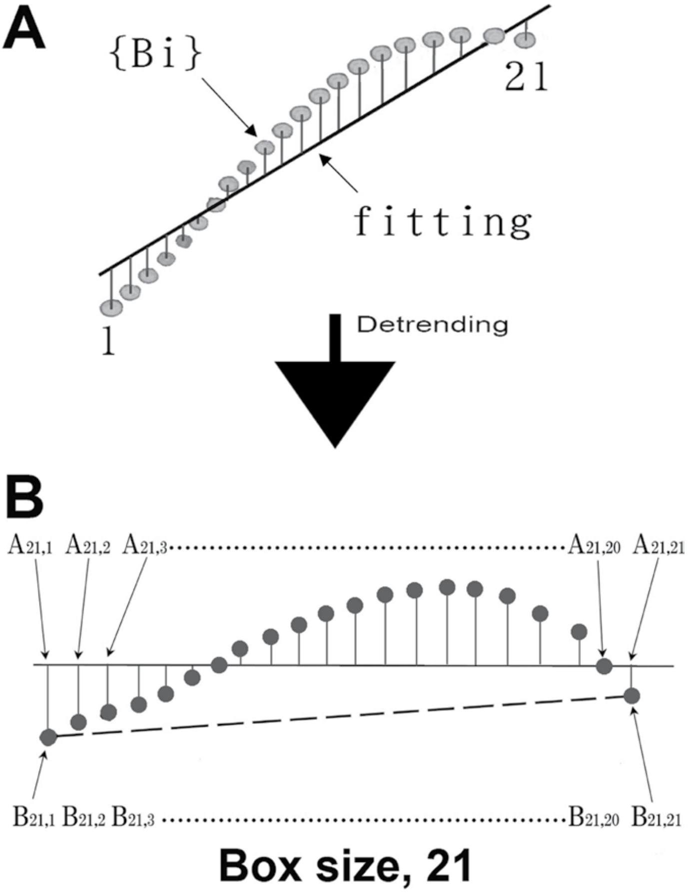


Figure 9.
Diagrammatic explanation of detrending. (A) the set {Bi} is a set of times. A fitting straight line is linear fitting. Although mDFA uses biquadratic fitting instead of linear fitting, we here use linear fitting for the sake of convenience. An arrow from A to B, the subtractions, "{Bi} – Fitting" that is a detrending process.
(B) Diagrammatic representation of the difference between mDFA and DFA (see text).

focusing on segment of time. This is one of the biggest algorithm differences between mDFA and DFA.

2.3.7 mDFA: the scacling law does not govern the entire domain (window size, domain size, margin, or boundary)

Peng and Goldberger et al. [6, 15] mention that DFA works on time series data that has fractal-like characteristics or self-similar structure. They also mention that they prefer to use a longer data. The longer, the better. They used 24 hr-length data with a Holter EKG monitor [6, 15].

Our body system is never run stable for 24 hr. It rather changes momentarily. In our everyday life, we can concentrate on a thing (one object, song, program, etc.) for a limited period. Our mind changes dynamically from one thing to the next. The stability of the brain-heart axis might last only for a short period. Imagine humans can keep continuing a steady state: a boxing-fighting game consists of 3 min for one round; we wait 3 min before a pot noodle becomes ready to eat; a popular music hit-the-charts, for example, "Take Me Home, Country Road" lasts for a perfect (not too long and not too short) length "three" min; and so forth.

In physics, ideally, the scaling law holds across a whole length of data set in the steady state condition. But, in biology, we consider, scaling will appear in a limited period length. In the current experiments, we wanted to find the period length, that is, window size or domain size, or margin or boundary, which is autonomously regulated. Within this length, we see the steady state of the body system.

From our experiences, for capturing "biologically true scaling index provided by the steady state of the body," we found a 2000 consecutive heartbeat data is adequate in length. In mDFA, the box size length (l) is fixed as: 10, 11, 12, − 18, 19, 20, 21, 22, − 99, 100, 110, − 190, 200, 210, 220, − 290, 300, 310, 320, − 480, 490, 500, 600, 700, 800, 900, and 1000 (BPM).

After that, the scaling index is calculated using a restricted window size of log–log graph, that is, box size length stretching out from 30 to 270, depicted as [30; 270], or the domain of definition $30 < l < 270$, in our projects (see [6, 15]).

In the regulation of heart rate, scaling characteristics of heartbeats will appear in the domain around "three" min. This size is in line with two decade in logarithmic scale, that is, between 10 BPM and 1000 BPM (BPM, beats per min). The scaling index is computed from a double logarithmic graph [6]. We, therefore, studied heartbeats during these two decade.

In the textbook definition, in healthy human individuals, heart rate at rest ranges from 50 to 100 BPM. So, the number of three-minute heartbeats corresponds to 150–300. The window size [30; 270] does not miss the point in terms of neurobiological considerations (see the graph **Figure 10B** for the scaling computation, that is, section length of linear fitting straight line).

In conclusion, the best window size, that is, the domain $30 < l < 270$, ensures that mDFA gives a reliable and satisfactory useful scaling index, as shown in the results section below. Our mind might be quantified by the scaling index [18, 19]. We can peer into the mind through a window. The size/length will be a period of approximately 3 min. We applied mDFA in various disciplines, including the heart, material vibration, and earthquake [6, 15]. The window size [30; 270], that is, the domain $30 < l < 270$, is universal. As a practical

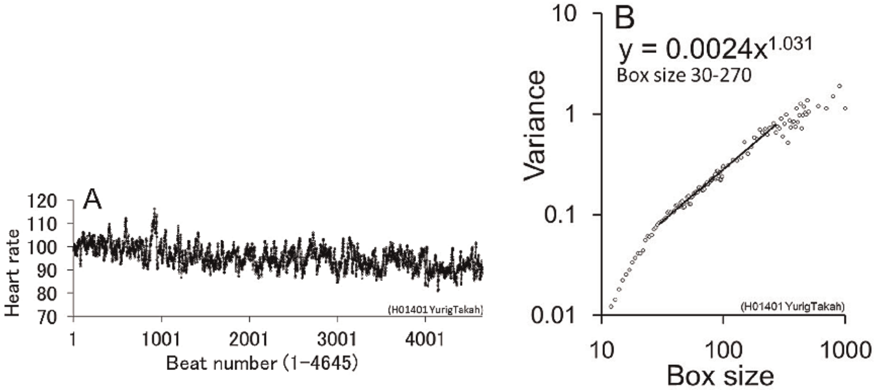


Figure 10.
mDFA's log–log graphic demonstration (see [16–18] for detail). (A) Time series of 4645 beats, from a healthy happy individual in her 50s (no skipping or arrhythmic beats, laughing, and talkative while recording this EKG). (B) Log–log plotting in order to calculate the slop. A fitting line is made from box size 30 to box size 270, that is, frame size [30; 270], not beyond this size. We reached a conclusion: This window size best captures the scaling exponent that fits the biological explanation regarding the state of the heart. Here, SI = 1.031. So, mDFA works at the restrictive span of dimension [30; 270].

tool for time series analysis, mDFA's window size is important to guarantee the reliability of mDFA.

2.3.8 mDFA: coda

What Peng's DFA characterizes is conceptually different from what mDFA characterizes. The former characterizes criticality phenomenon, the latter characterizes phenomena more time-change dependent than DFA does. Every happening in the world has causation. The causation is often hidden or merely cannot be seen or hardly detected by human sensory-brain ability. Some causalities are invisible like the silent onset of disease. Some causalities are explosive and appear suddenly. Peng's DFA can see critical solutions. mDFA may see the relationship between the past and the present life. The heart could not be a good field for criticality physics, but the brain is the most appropriate field because we often experience "Aha!" or eureka moment [20].

3. Results

3.1 Case study: unhappy heart, WPW syndrome

A recording set shown in **Figure 3** was used. Tachycardia (fast heartbeat) and arrhythmia are two of the many common heart problems today.

If a fast heartbeat occurs acutely and suddenly, the heart becomes an inefficient pump. Enough blood is not pumped to the brain, making the person feel sick. It is also life-threatening.

We recorded a fast heartbeat EKG from a girl who suffered from WPW syndrome (Wolff-Parkinson-White syndrome). She is approximately 10 years old. The family, parents, and children have moved to United States from Cuba to receive medical

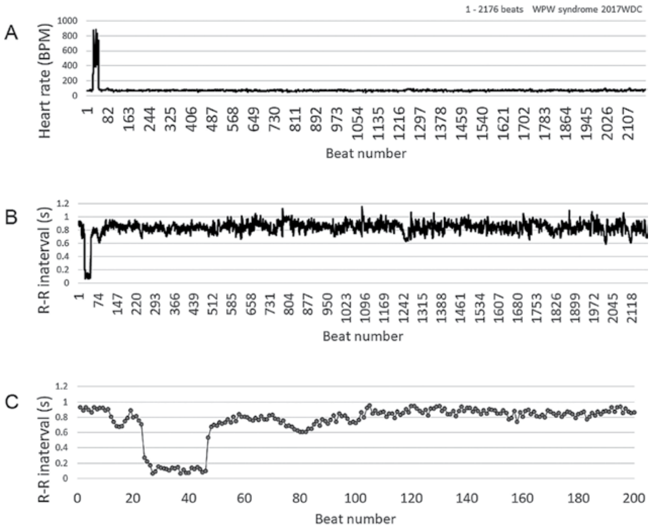


Figure 11.
Time series constructed from WPW syndrome of a young girl. About 10 years of age. Abrupt seizure is recorded in the initial part of the time series. (A) Heart rate time series, 1–2176 beats. (B) Converted to the peak-to-peak interval. (C) Initial part of the interval time series, 1–200 beats.

care for heart problems. We met them a few times in a town Great Falls near the Potomac River.

Her mother explained that the girl feels ill when a tachyarrhythmia seizure occurs. We saw her running into a washroom when she had an unpleasant sensation.

Figure 11 shows the interval time series of the young girl. In **Figure 11A** and **B**, the seizure is recorded. She did not tell me about feeling funny at that time, although her face looked sick to me. Later, when we checked the data, we understood that the seizure did not last long. It was only approximately 20-heartbeats (**Figure 11C**).

Figure 12 shows mDFA results. The seizure increases SI (scaling index, scaling exponent) value significantly. Her mother told me that her daughter is expected to receive catheter ablation next week. It will be the third time. In the two previous treatments, nothing was effective. She asked: “Do you think should I take her to the hospital again?”

I replied: “My wife’s friend has a husband who has a similar condition as the girl. He, the husband, fell over in the road before, and an ambulance saved him. He already received twenty (20) catheter ablations. Unfortunately, it never worked. He went to see a new doctor in a different hospital where a celebrated cardiologist was working. And then, the latest catheter ablation totally removed the arrhythmia. Please do not abandon hope.”

One month after my mDFA recording, the girl’s tachycardia disappeared after the third catheter ablation, of which I was notified by an e-mail.

For more results with this method, please consult [18], such as “Bundle Branch Block, Problems in the Electrical Conduction System of the Heart”.

3.2 Case study: healthy happy heart—working-eating-exercising

A recording set shown in **Figure 3** was used. **Figure 13** is an example of a cyclically computed mDFA upon a volunteer subject in her 60s. **Figure 13** shows a working-eating-exercising mDFA. She kindly offered herself to the EKG recording twice a

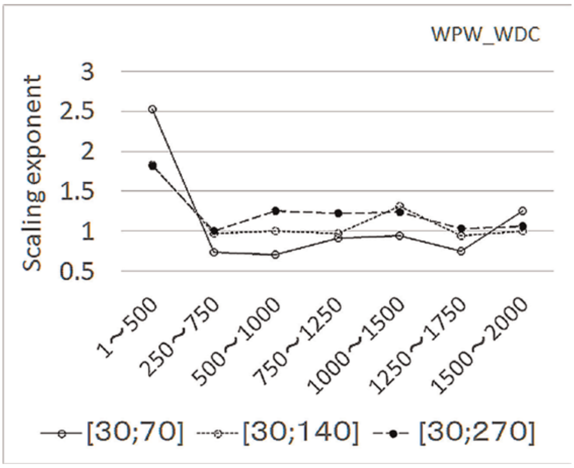


Figure 12.
mDFA results on the WPW syndrome data shown in ***Figure 11****. The data length for mDFA computation is 500 beats. Computation was displaced in sequence every 250 beats. Note: The seizure (the period 1–500) evidently and significantly increases SI (scaling exponent) values. Three different box sizes [30; 70], [30; 140], and [30; 270] are shown here, although generally, we use [30; 270].*

week, and for years when she worked in the laboratory. Her time series exhibits many PVCs. These PVCs are benign. While she is working on PC (1–3000), she has a low exponent, which means she is concentrating on the job. It is seen that she is working earnestly. When I showed her **Figure 13**, she told me that mDFA is scary because someone indeed can look at my internal world, my mind. During meal time (lunch, 6001–9000), the exponent noticeably decreased. Exercise (9000–15,000) elevated the scaling exponents significantly. This test supports the hypothesis that mDFA can quantify the mind.

In conclusion, concentration on the job lowers the scaling exponent from normal healthy exponent 1.0 down to 0.9 or lower. This observation does not contradict previous results which were found by employees at the Indonesian University [18]: People who are the president of the University, Vice president of the University, and the Dean of faculty all of them had a low exponent (SI = 0.72–0.84). They are relatively seriously talking to me with dignity during the EKG recording. In turn, ordinary employees (teaching only professors who were happily talking while EKG recording) had a healthy exponent (SI = 1.0) [18].

3.3 Case study: passenger's mDFA. EKG monitoring during in flight

A recording set shown in **Figure 14** was used. In the flight frying from Tokyo, Japan to Washington DC (WDC), Virginia, USA, mDFA found: (1) Wine service significantly decrease the scaling exponent (**Figure 15**, data number 4). (2) Sleepy conditions lower the scaling exponent (**Figure 15**, data numbers 7, 8, and 9). (3) A boring movie (i.e., sleeping) also decreased the scaling exponent, but an interesting movie does not (**Figure 15**, data numbers 12, 13, and 14). Fundamentally the same results were obtained by the returning flight and the next Tokyo-WDC flights too (data not shown). In other "mobile EKG—mDFA" confirmed that the happily eating condition (such as **Figure 13**, the value at 6001–9000) and sleepy condition lowers the scaling exponent (data not shown).

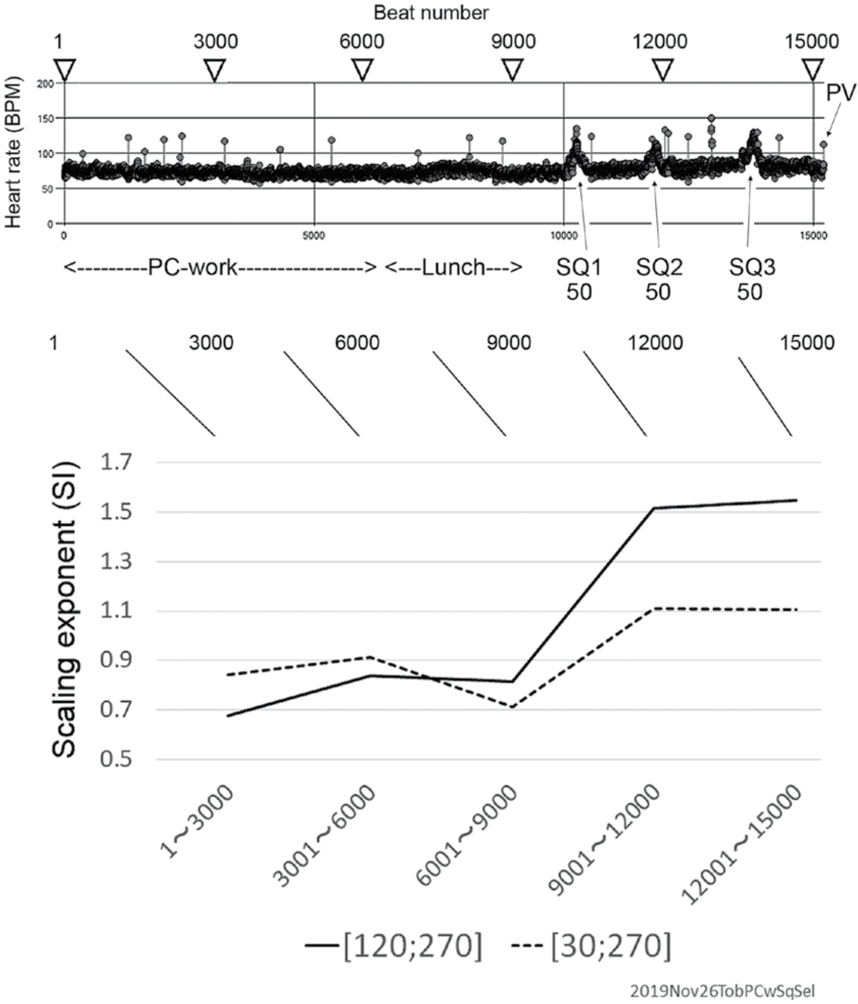


Figure 13.
An employee's EKG recording and mDFA. Volunteer subject, in her 60s. Inset time series shows 15,000-beat data while working on PC, having lunch meals, and squatting (SQ) three times, each 50-squattings. Many PVCs are occurring. These PVCs are benign – 60 percent people over age 40 have this PVCs (premature ventricular contractions). For every 3000 beats, SI values were computed. Working time, a low exponent. During meal time (lunch), the exponent noticeably decreased. Exercise elevated the scaling exponents significantly.

We have also tested "mobile EKG—mDFA" in the passenger aircraft from Orland, Florida, to WDC. It was of interest that, in the landing shaky aircraft in bad weather, feeling-scared states lowered the scaling exponent, but the scaling exponent recovered autonomously to a normal exponent (i.e., SI = approximately 1.0) after touching down on the runway (see Figure 63 of [18]).

4. Discussion

4.1 Accuracy and fidelity of the time series construction

In the present communication, we present how to record high-fidelity EKG and how to construct accurate heartbeat-interval time series (**Figures 4**–**7**). For

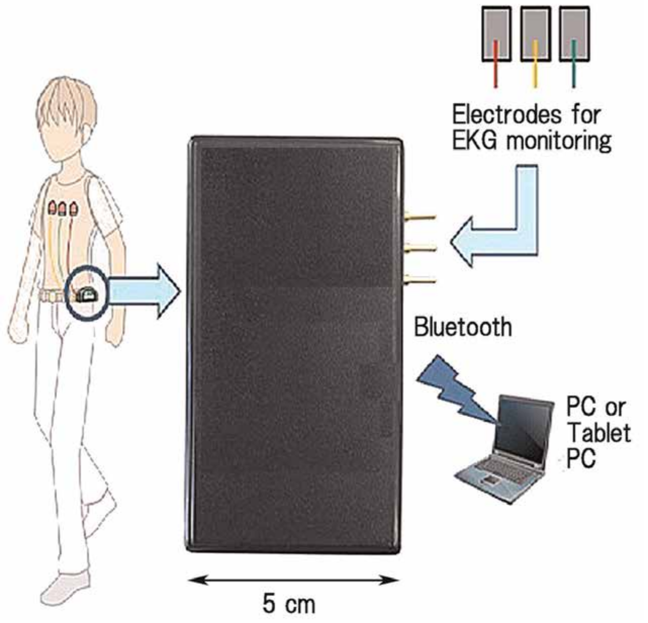


Figure 14.
An EKG monitoring set using the Bluetooth radio wave. Individuals can carry an EKG monitor in their pocket. The monitor sends authentic EKG waves to PC. EKG electrodes and EKG amplifier are the same, as shown in ***Figure 4****.*

	Flight from Tokyo to Washington DC	
	Note	Scaling exponent
1	Waiting lounge	1.06
2	Embarkation	0.96
3	Taxi and take-off	1
4	Wine service	0.72
5	Meal	0.96
6	After meal	1
7	Feeling sleepy	0.89
8	Sleepy and yawning	0.75
9	Sleepy	0.81
10	Shaky and waking up	0.98
11	Movie but drowsy	0.93
12	Movie Discovery Channel	0.95
13	The same movie Sleepy	0.83
14	Movie and falling asleep	0.58
15	Running out of battery	–

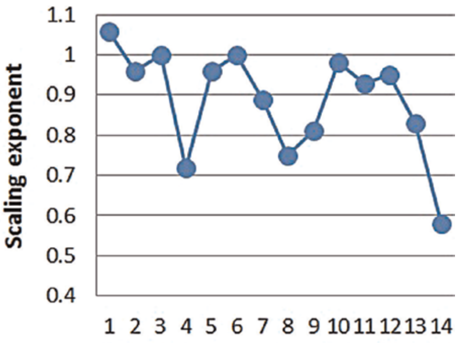


Figure 15.
EKG-mDFA cyclical challenge during a flight. EKG monitoring method is shown in ***Figure 14****. mDFA computation was done after the flight. From 1 to 14, each data has 2000 heartbeats. Note, record numbers 5 and 6: Having a meal but enjoying conversation with people next seat; SI does not decrease. Record number 15: Experiments failed due to running out of amplifier battery.*

the steady progress of clinical medicine, especially for diagnosing and managing patients, people like sophisticated physicists (e.g., [5, 21]) require to analyze the accurate time series, instead of easily available data (e.g., [22]). In the [22], authors mentioned that "a gap resulting from artifacts, noise, and exclusions of ectopic beats was considered equal to the R-R interval subsequent to the gap." And they continued by mentioning that "the length of interpolated gaps relative to the total length of the recording was 0.024%." We are, however, apprehensive about the "interpolation" of the gap. Such artificial data processing might break up authentic phenomena. It is not true data. In preparing data, false or true is a matter. We cannot ignore artificial alteration, even though it is very little, 0.024%. Our time series, it is so real.

In the article [15], similar modifications of missing heartbeats were done (the gap was supplemented [15], or the gap was removed [23]). To enhance trust in clinical medicine, especially for diagnosing and managing asymptomatic patients (i.e., early detection is the best way), high-tech physicists must fully fill a requirement: constructing high-fidelity time series data.

4.2 Our past experiences influence our current perceptions

Both DFA and mDFA use a subtraction, that is, (B – A) or (Q – P), respectively, as I already explained in **Figure 9B**. The former corresponds to DFA algorithm, and the latter corresponds to mDFA argorithm.

Both methods calculate a "mean value". The mean value is not a simple average but the mean-square, that is, $(B - A)^2$ or $(Q - P)^2$, respectively. Due to the usage of mean-square, this scaling analysis uses a graph of the double-log plotting, as mentioned. Both methods repeat computations cyclically over and over all the box sizes. Finally, the relation between F(n) and n is plotted on the double-log graph, as mentioned. This provides a relationship between F(n) and n. The slope of the graph determines the scaling exponent in both DFA and mDFA.

However, the concepts behind computations are different each others between DFA and mDFA. DFA calculates (B – A), but mDFA calculates (Q – P) instead. The former's concept suggests criticality or tipping point phenomena. In contrast, the latter seeks the random walk-like concept, that is, "how many steps proceeded in a box." This involves, neurobiologically, an idea: "Our past experiences influence our current perceptions." This is the main point of the computational concept of mDFA (**Figure 16**) (see [18]).

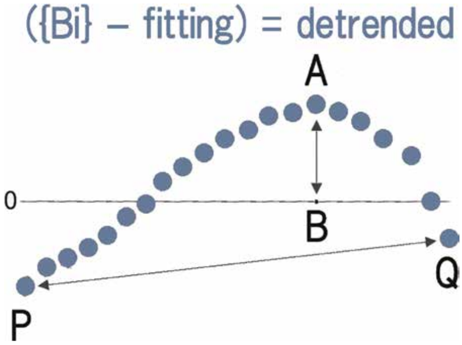


Figure 16.
Diagrammatic representation regarding the difference between DFA and mDFA. Modified from ***Figure 9B****. DFA uses subtraction (B – A). mDFA uses the subtraction (Q – P).*

4.3 Fractal the self-similar structure but restrictive span of dimensions

Heartbeat series shows self-similar structures and self-similar fluctuations [24]. Likewise, the tree-like spatial fractal has self-similar branchings: The small-scale structure resembles the large-scale form [24] (see A, B, and C in **Figure 17**). In the heart regulation, a fractal temporal process may generate fluctuations on different time scales that are statistically self-similar [24]. But, in biology, self-similarity appears only in restrictive span of dimensions, as shown in **Figure 17**.

mDFA looks at restrictive span of dimension or restrictive frame size or box size, that is, confined window size [30; 270] or the domain $30 < l < 270$ (see **Figure 10B**). A linear fitting line (see **Figure 10B**) does not extend beyond the framework [30; 270]. mDFA do not consider boxes smaller than 30 and larger than 270 when determining the scaling exponent in its computation. We actually tried various set of “frameworks” before reaching this conclusion. We finally found that, in this restrictive window size, mDFA is best fit to distinguishing various heart conditions, from healthy happy hearts to other hearts such as stressed- and diseased hearts. Thus, we temporarily conclude that mDFA is successful and acceptable as an efficient tool in the neurobiology of the cardio-regulatory system. Using mDFA, we can tell the distinction between happy hearts and stressed hearts, or between the hearts of natural death and unpredictable death [16–18].

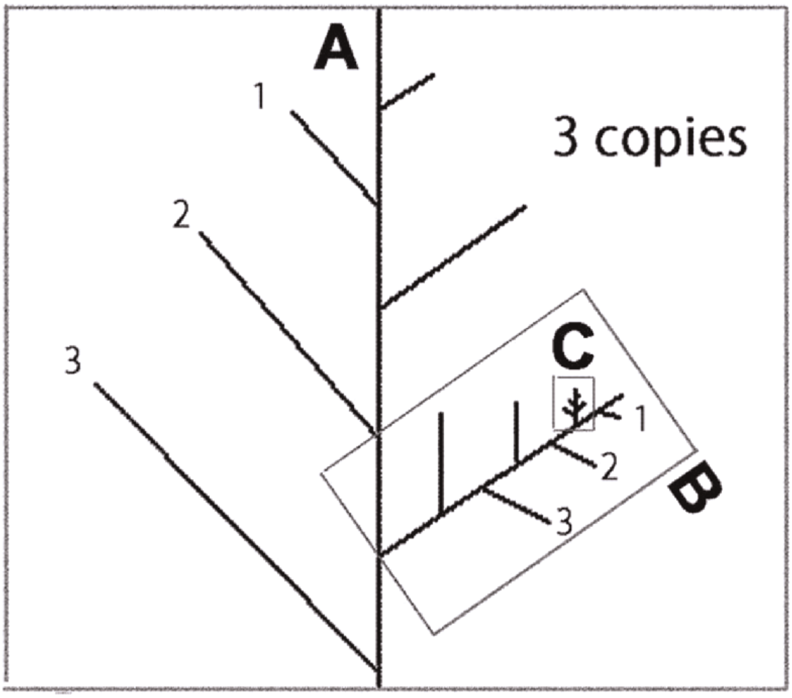


Figure 17.
Diagrammatic representation of tree-like self-similar branchings. The small-scale structure resembles the large-scale form (A, B, and C). In a real country field, there are NO trees larger scale than A, and NO smaller scale than C. In biology, scaling appears in a limited size; meantime, the size can grow up to infinity in ideal physics. Dimension should be adjusted to real size (window size, or tree size) to measure the scaling index in the self-similar real data.

4.4 mDFA identifies risky and unhappy hearts

From our experimental evidence, mDFA seems to work over a wide range of the scaling exponent, from the value near zero (which can be seen on a smoothly running electric motor) to the value surpass 2.0 (which can be seen on the heartbeats at unpleasant psychological state caused by the WPW syndrome, where the brain cannot receive enough oxygen supply from the heart) (**Figures 11** and **12**) [16–18]. Furthermore, the scaling exponent of a girl suffering from WPW syndrome quickly returns to a normal value, around 1.0, after abnormal heartbeat disappeared, that is, coming back to a normal heart condition (**Figures 11** and **12**). So, it appears that mDFA ensures high fidelity response to the behavior of internal complex system, which is changing its performance moment by moment.

Higher scaling exponents greater than 1.2 are of interest: Surprisingly, higher values indicate that the system is "approaching danger." This was proven by the nonheart experiments. We applied mDFA on the material vibration, recorded from aluminum rod experiments, where a rod irreversibly bent (snapping off) by the externally applied force. Here, mDFA proved that when the scaling exponent exceeded ca. 1.2, a rod was completely broken (see [17] for details).

Furthermore, we have already reported (Figure 5–22 in [16], p. 62) that terminal patients show high scaling exponents (around 1.5) before dying: The patients suffered from, (No. 1) hepatic cirrhosis, (No. 2) type 2 diabetes plus brain infarction, and (No. 3) colorectal cancer [16]. Meantime, No. 4 terminal patient maintained a lower exponent (~ 0.5) all the time before dying. She suffered from senile weakness. So, three patients (No. 1–3) died "unpredictably." But, in the case of No. 4, the doctor's gut instincts can "sense" that "something" will happen sooner or later. (We are grateful to Dr. K. T. at Kojimachi, Yotsuya, Tokyo, for providing us EKG data of terminal patients. He equipped his special EKG monitoring machine on patients, and patient's EKGs were seamlessly sent to the PC in his clinic office through the mobile-phone internet network. He does not take the patient to the office; instead, his office has an uninterrupted EKG online connection to the patients at home [16]. Time series data were constructed by us, and mDFA also by us.)

In summary, evidence indicates that higher scaling exponent way exceeding 1.2 could be showing unhappy (risky) state, either in human heart or material structures.

4.5 mDFA during working, eating, and doing exercise

The heartbeats during a job exhibit a low scaling exponent (0.8–0.9 in **Figure 13**). This does not contradict with the previous report, which showed that serious employees of the university, the president, vice president, and faculty dean, all exhibited a low scaling exponent [16] as mentioned above.

The heartbeats during happy lunch exhibit a low scaling exponent (0.7, **Figure 13**; 0.7, **Figure 15** during wine service).

But having meal and enjoying conversation with people next seat does not decrease SI (**Figure 14**, record numbers 5 and 6). Here, the brain does not concentrate on meal, and the scaling exponent was not noticeably altered. The previous report also supports this interpretation of "having meal mDFA" (see **Figure 14** of previous publication [19]). If individuals were interested in not so meal but conversation, we found that the scaling exponent did not decrease significantly. In conclusion, it appears that mDFA looks at the brain-heart axis (whole body complex system).

The exercise, in turn, increases the scaling exponents (**Figure 13**). We have reported the same results previously [16]: Ergometer-exercise experiments performed at the Bandung Olympic Training Center Indonesia showed that four out of four athletes all exhibited a significant increase of the scaling exponents by exercise (Figures 5–7 in [16], p. 47). Especially, in two athletes out of four, the "exercising" exponents attained a value as high as 1.4–1.5. The two were found to be amateur athletes. As mentioned above, very high exponents indicate that "the individuals are staying at a hazardous state" like the patients in a terminal condition. So, we can hypothesize that strong exercise (heavy muscular load) leads to a risky state of the human body system, although this is still an open question. So far, we have any contradictory results regarding the "exercising" mDFA. (We are grateful to Professor Dean Dr. A. Hutapea at Bandung Indonesia. To date, we have investigated over 50 individuals who are Asian-game medalist-class athletes. In preparation for publication.)

4.6 mDFA, falling asleep

We have already found that sleeping state lowers the scaling exponent [25]. The results shown in **Figure 15** strongly support previous findings.

5. Concluding remarks

We would like to stress that mDFA technique is not too bad as a tool. It is beyond expectation. Construction of accurate time series could be important to provide clear results. The computing concept of mDFA is derived from the neurobiological consideration supported by careful observation of subjects' behavior and of individual EKG trace. It is scientifically acceptable.

We stick to mDFA as long as it does not contradict experiments or observation. And there is no chance of abandoning it. We have never seen this kind before to our knowledge. In the future, more tests might help advance this tool. Alteration of the box size is a candidate for alteration (see **Figure 13**, where results from [120; 270] and from [30; 270] are shown).

Acknowledgements

This work was supported by JSPS Grant No. 21 K12668.

Conflict of interest

Symphodia Phil Co., Ltd. has patents for mDFA programs.

Author details

Toru Yazawa
Tokyo Metropolitan University, Hachioji, Japan

*Address all correspondence to: tryazawa@gmail.com; yazawa-tohru@tmu.ac.jp

References

[1] Zhao B et al. Heart-brain connections: Phenotypic and genetic insights from magnetic resonance images. Science. 2023;**380**:eabn6598, 13 pages

[2] Zhang Y et al. Topographical mapping of catecholaminergic axon innervation in the flat-mounts of the mouse atria: A quantitative analysis. Scientific Reports. 2023;**13**:4850, 21 pages. DOI: 10.1038/s41598-023-27727-9

[3] Mohr V et al. Social interception: Perceiving events during cardiac afferent activity makes people more suggestible to other people's influence. Cognition. 2023;**238**:105502, 14 pages. DOI: 10.1016/j.cognition.2023.105502

[4] Glass L, Mackey MC. Pathological conditions resulting from instabilities in physiological control systems. Annals of the New York Academy of Sciences. 1979;**316**(1):214-235

[5] Saul JP. Beat-to-beat variations of heart rate reflect modulation of cardiac autonomic outflow. News in Physiological Science. 1999;**5**:32-37

[6] Peng C-K et al. Quantification of scaling exponents and crossover phenomena in nonstationary heartbeat time series. Chaos. 1995;**5**:82-87

[7] Glass L. Synchronization and rhythmic processes in physiology. Nature. 2001;**410**:277-284. DOI: 10.1038/35065745

[8] Huikuri HV et al. Measurement of heart rate variability by methods based on nonlinear dynamics. Journal of Electrophysiology. 2003;**36**:95-99

[9] Davidson KW et al. Screening for atrial fibrillation: US preventive services task force recommendation statement. JAMA. 2022;**327**(4):360-367. DOI: 10.1001/jama.2021.23732

[10] Curry SJ et al. Screening for cardiovascular disease risk with electrocardiography: US preventive services task force recommendation statement. JAMA. 2018;**319**(22): 2308-2314. DOI: 10.1001/jama.2018.6848

[11] Yazawa T, Katsuyama T. Spontaneous and repetitive cardiac slowdown in the freely moving spiny lobster, Panulirus japonicus. Journal of Comparative Physiology A. 2001;**187**:817-824. DOI: 10.1007/s00359-001-0252-z

[12] Yazawa T et al. Neurodynamical control systems of the heart of Japanese spiny lobster, Panulirus japonicus. Izvestiya VUZ Applied Nonlinear Dynamics. 2004;**12**(1-2):114-121

[13] Johnson AS, Winlow W. Does the brain function as a quantum phase computer using phase ternary computation? Frontiers in Physiology. 2021;**12**:572041, 12 pages. DOI: 10.3389/fphys.2021.572041

[14] Weidmann S. Shortening of the cardiac action potential due to a brief injection of KCl following the onset of activity. The Journal of Physiology. 1956; **132**:157-163

[15] Goldberger AL et al. Physio Bank, physio toolkit, and physio net: Components of a new research resource for complex physiologic signals. Circulation. 2000;**101**(23):e215-e220

[16] Yazawa T. Modified Detrended Fluctuation Analysis, mDFA. ASME Monograph. New York, USA: Momentum Press; 2015

[17] Yazawa T, Omata S. mDFA detects abnormality: From heartbeat to material vibration, chap 3. In: Noise and Vibration Control - from Theory to Practice. London, UK: IntechOpen; 2019. pp. 33-54

[18] Yazawa T. Quantifying the Mind by mDFA. Collaboration between Neurobiology and Statistical Physics. NY: NOVA Science Publishers; 2020. pp. 1-126

[19] Yazawa T. Anxiety, worry and fear: Quantifying the mind using EKG time series analysis. Chap 2. In: Mohamudally N, editor. Time Series Analysis and Applications. London, UK: IntechOpen; 2018. 16 p. DOI: 10.5772/intechopen.71041

[20] Yazawa T. Isolated crayfish stretch receptor neuron electrophysiology may explain a longstanding mystery of human brain functioning: Eureka moment. In: Pertinent and Traditional Approaches towards Fishery. London, UK: IntechOpen; 2023. DOI: 10.5772/intechopen.109732

[21] Kiyono K et al. Critical scale invariance in a healthy human heart rate. Physical Review Letters. 2004;**93**(17): 178103, 4 page. DOI: 10.1103/PhysRevLett.93.17803

[22] Sakata S et al. Aging and spectral characteristics of the nonharmonic component of 24-h heart rate variability. The American Journal of Phyisiology. 1999;**276**:R1724-R1731

[23] Flynn AC et al. Heart rate variability analysis: A useful assessment tool for diabetes associated cardiac dysfunction in rural and remote area. The Australian Journal of Rural Health. 2005;**13**:77-82

[24] Goldberger AL et al. Fractal dynamics in physiology: Alterations with disease and aging. PNAS. 2002;**99** (Suppl. 1):2466-2472

[25] Yazawa T, Shimoda Y, Hutapea AM. Evaluation of sleep by detrended fluctuation analysis of the heartbeat. In: Ao S-L, editor. IAENG Transaction on Engineering Technologies. AIP Conference Proceedings 1373. Vol. 6. Melville, New York, USA: AIP Publishing; 2011. pp. 199-210

Chapter 4

Use of the Gini Coefficient for the Analysis of Heart Rate Variability in Sick and Healthy Individuals

Moacir Fernandes de Godoy, Beatriz Arid Rudnick
and João Victor de Menezes Reichert

Abstract

The Gini Coefficient (GC) is a statistical tool widely used in Economics to quantify the degree of variation of values belonging to a group, ranging from zero to one. The closer to zero, the less unequal the country. We proposed to evaluate GC in the analysis of Heart Rate Variability in different groups of individuals hypothesizing that groups with more similar conditions and better stationarity (healthy individuals) should be the groups with lower differences among them and among the aged, diseased, or premature individuals. Time series of normal RR intervals were analyzed in eight groups of individuals: premature (G1) or healthy newborns (G2), healthy children (G3), healthy young adults (G4), elderly adults (G5), patients with chronic liver (G6) or kidney disease (G7) and individuals with a state of brain death or who died in the short term after the initial evaluation (G8). GC distinguished the less unequal groups (G2, G3 and G4) from all other groups. Was identified, graphically, a parabolic behavior with near similarity among equivalent homeostatic levels. GC is a useful tool for characterizing the stationarity and the homeostatic level of clinical groups (by lower GC values) and to interpret more adequately the results.

Keywords: Gini coefficient, heart rate variability, autonomic nervous system, temporal series, non-stationarity

1. Introduction

1.1 Autonomic nervous system

The autonomic nervous system (ANS) is the portion of the central nervous system that controls most of the body's visceral functions, such as breathing, blood circulation, body temperature, and digestion. It is activated by centers located in the spinal cord, brainstem, and hypothalamus, which transmit signals to lower centers such as baroreceptors, chemoreceptors, atrial receptors, ventricular receptors, the vasomotor system, the renin-angiotensin-aldosterone system, and the thermoregulatory system [1]. These efferent autonomic signals are transmitted to the visceral organs by two subdivisions: the sympathetic nervous system and the parasympathetic nervous system, which differ anatomically and physiologically, the first being organized in the form of a thoracolumbar paravertebral ganglionic chain, while the second consists of in a chain of craniosacral conformation.

The sympathetic autonomic nervous system is responsible for increasing the global activity of the heart, through the release of epinephrine and norepinephrine, favoring three mechanisms: increased heart rate (HR), increased conduction velocity, and increased myocardial contraction force. On the other hand, parasympathetic stimulation - from the release of acetylcholine in the sinoatrial (SA) and atrioventricular (AV) nodes - causes effects essentially opposite to those observed in sympathetic stimulation, decreasing the rhythm of the sinus node and the speed of transmission of impulses to the ventricles. Thus, when vagal inhibition occurs, there is a relative predominance of the sympathetic over the parasympathetic, with an increase in HR. The decrease in HR is related to the predominance of vagal (parasympathetic) activity. In this sense, in healthy individuals, the cardiac rhythm is constantly controlled by the sympathetic stimulation, accelerating it, and by the parasympathetic stimulation, decelerating it. This dynamic balance, throughout the day, undergoes variability according to breathing, hemodynamic changes, mental stress, metabolic changes, physical exercises, sleep, and orthostatism, among others.

The study of heart rate fluctuations throughout the day reflects the functioning of the ANS and is called Heart Rate Variability (HRV). In this sense, the ability of the heart to adapt to the body's needs is based on high variability and is directly related to the individual's health. Thus, healthy patients with good ANS functioning have high HRV, while individuals with low HRV denote insufficient and abnormal physiological functioning of this system.

HRV is usually analyzed using time and frequency domains. Its measurement is made from the oscillations of the distances between the consecutive peaks on the electrocardiogram, the normal R-R intervals (RRi), graphically represented by the tachograms below (**Figure 1**).

In **Figure 1**, the tachogram shows in [A] the HRV of a healthy adult, while in [B] the HRV of a healthy newborn is recorded.

In this context, it is known that changes in HRV have predictive and anticipated value in the analysis of an individual's health. Thus, the decrease in HRV is an important indicator both for the appearance of diseases in previously healthy patients and for the appearance of complications in individuals who already have alterations [2].

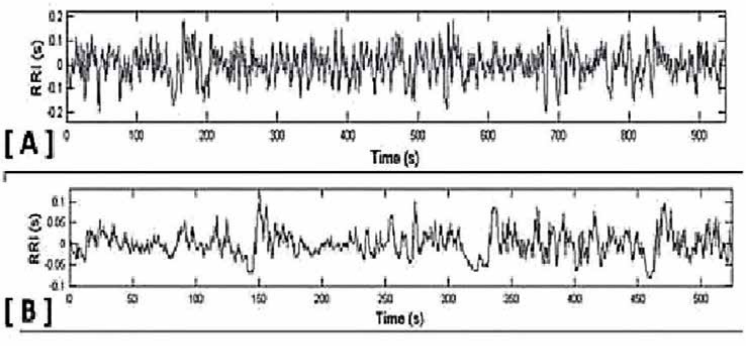


Figure 1.
Note the tachograms, in [A], of a healthy young adult, while, in [B], the tachogram of a healthy newborn. There is less variability in B relative to A.

This reduction in HRV is usually due to relative hyperactivity of the adrenergic system and a decrease in parasympathetic activity.

HRV quantification has been, in recent years, increasingly used, since it is easy to acquire data, it is a non-invasive method, easy to perform, reproducible, and, mainly, of wide clinical applicability. There is a large literature regarding the monitoring of various morbid conditions, such as coronary artery disease [3, 4], cardiomyopathies [5], systemic arterial hypertension [6, 7], renal failure [8], diabetes [9], among others.

1.2 Lorenz curve

The Lorenz curve is a graph used to study the relative distribution of a variable in a given domain. It is widely used in economics, where the "y-axis (ordinate)" corresponds to the income variability of a society, and the "x-axis (abscissa)" to the cumulative percentage of the population with that income [10].

In a perfectly egalitarian society, all citizens receive the same income, and the Lorenz curve would be represented as the black line in **Figure 2**, where y = x, is called the line of equality. However, most societies are not economically egalitarian, and the Lorenz curve is represented by the red line (**Figure 2**).

1.3 Gini coefficient

The Gini coefficient was created in 1931 by the Italian mathematician Conrado Gini to measure the degree of income concentration in a given society, based on the Lorenz curve. The coefficient consists of the difference between the equal distribution (black curve, **Figure 2**) and the Lorenz curve (red curve, **Figure 2**). The measurement of the Gini coefficient follows a scale that goes from 0 (when there is no inequality) to 1 (with maximum inequality), representing ideal extremes.

In this sense, a Gini coefficient equal to 1 would represent a society in which a single individual stores all income, and a Gini coefficient equal to 0, a society in which all people have the same income.

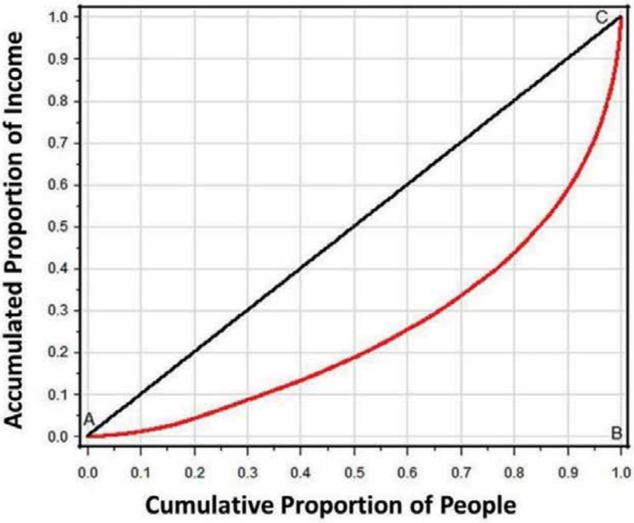


Figure 2.
Lorenz curve.

This coefficient has been used, however, not only in the economic area but also in social studies, such as the suicide rate related to income [11]; cell transport markers [12] and gene expression [13]; bacterial aggregation rates [14]; health access distribution rates [15]; screening for pathologies, such as central nervous system tumors in children [16] and perinatal mortality [17]. Therefore, it is understood that this coefficient can be used in any analysis of a variable that manifests itself through a time series.

The purpose of the present study is to use the Gini coefficient in a time series of heartbeats in healthy individuals and in those with different degrees of homeostatic impairment in a wide range of age groups, considering, that will be possible to characterize the variation degree of each group and, by extension, the variation degree relatively among groups hypothesizing that conditions with the worst homeostatic level should present higher coefficients than states with better homeostatic level.

2. Casuistic and method

Time series of normal RR intervals were analyzed in eight groups of individuals: 29 premature newborns admitted to the ICU (G1), 21 healthy newborns (G2), 31 healthy children (G3), 32 healthy young adults (G4), 27 elderly adults hospitalized in nursing homes (G5), 40 patients with chronic liver disease (G6), 22 patients with chronic kidney disease on hemodialysis program (G7) and 9 individuals with a state of brain death or who died in the short term after carrying out the initial evaluation (G8).

2.1 Gini coefficient formula

The Gini Coefficient (GC) was calculated from Brown's formula, based on the Lorenz curve (Equation):

$$G = \frac{2}{n^2 \overline{x}} \sum_{i=1}^{n} i\left(x_i - \overline{x}\right) \tag{1}$$

Where,

- G = Gini coefficient.
- P = accumulated proportion of the variable "population".
- R = accumulated proportion of the variable "income" (in this study "income" is understood as the numerical value of the RR interval in milliseconds.

2.2 Statistical analysis

Concomitant comparative analyses between the eight groups were carried out using the non-parametric Kruskal-Wallis test and, in case of detection of a statistically significant difference in the set, post-test comparisons were made of all pairs using the Conover-Iman test. An alpha error of 5% was admitted, with P values less than or equal to 0.05 being considered significant. The statistical software used was StatsDirect version 3.3.6 of May 23, 2023.

3. Results

Descriptive statistics values for groups G1 to G8 are shown in **Table 1**.

The graphic distribution by Box-Plot graphs is shown in **Figure 3**.

Table 2 shows the pairwise statistical comparisons.

Was identified, graphically, as a parabolic behavior with near similarity of values among equivalent homeostatic levels. (The lower the Gini Coefficient values, the greater the uniformity between the intragroup measures); **Figures 4** and **5**.

4. Discussion

Different behavior of the Gini coefficient was observed between the groups of patients studied. Firstly, when analyzing the Gini coefficients in the intragroup mode, that is, evaluating the values of the Gini coefficient in relation to the inequality of the RR intervals of each patient, it was found that the global means for the groups G2, G3, and G4 were the lowest (closer to zero) indicating similarity of behavior in these three groups. It is noteworthy that the three groups mentioned involve healthy individuals (newborns, children, and young adults). In addition, the groups corresponding to cases with potential for greater clinical impairment or lower homeostatic level (premature newborns, elderly hospitalized in nursing homes, chronic, liver and kidney disease, and, mainly, cases with ongoing brain death or in a state of imminent death), exhibited progressively higher mean values.

It is important to remember that the Gini Coefficient here is not assessing the heart rate variability, which is repeatedly lower in the presence of more important clinical impairment, but the intragroup variability of the GC values for the individual RR intervals. This can be translated as healthy individuals being more homogeneous among themselves. The groups composed of premature, senile, or severely ill individuals, on the other hand, contain individuals with more heterogeneous clinical

Variables	G1	G2	G3	G4	G5	G6	G7	G8
Valid data	29	21	31	32	27	40	22	9
Mean	0.0079	0.0048	0.0033	0.0031	0.0091	0.0065	0.0105	0.0265
SD	0.0097	0.0055	0.0033	0.0023	0.0122	0.0065	0.0130	0.0217
Maximum	0.0418	0.0224	0.0153	0.0108	0.0503	0.0277	0.0462	0.0713
UpperQuart	0.0089	0.0054	0.0046	0.0041	0.0115	0.0098	0.0146	0.0249
Median	0.0047	0.0026	0.0023	0.0032	0.0051	0.0049	0.0054	0.0191
LowerQuart	0.0019	0.0016	0.0010	0.0011	0.0024	0.0010	0.0021	0.0172
InterqRange	0.0070	0.0038	0.0036	0.0029	0.0091	0.0087	0.0125	0.0077
Minimum	0.0004	0.0003	0.0003	0.0000	0.0002	0.0000	0.0008	0.0037
Range	0.0414	0.0221	0.0150	0.0108	0.0501	0.0277	0.0454	0.0676

Legend. *G1 (29 premature newborns admitted to the ICU); G2 (21 healthy newborns); G3 (31 healthy children); G4 (32 healthy young adults); G5 (27 elderly adults hospitalized in nursing homes); G6 (40 patients with chronic liver disease); G7 (22 patients with chronic kidney disease on hemodialysis program); G8 (9 individuals with a state of brain death or who died in the short term after carrying out the initial evaluation).*

Table 1.
Descriptive statistics of the intragroup (G1 to G8) Gini coefficient values.

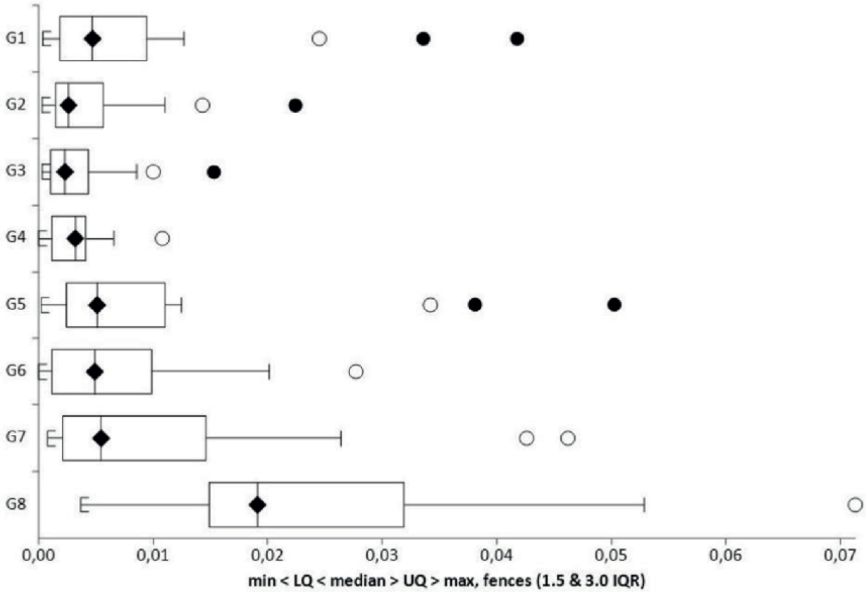


Figure 3.
Box-plot graph distribution of Gini coefficient descriptive values in groups G1 to G8. G1 (29 premature newborns admitted to the ICU); G2 (21 healthy newborns); G3 (31 healthy children); G4 (32 healthy young adults); G5 (27 elderly adults hospitalized in nursing homes); G6 (40 patients with chronic liver disease); G7 (22 patients with chronic kidney disease on hemodialysis program); G8 (9 individuals with a state of brain death or who died in the short term after carrying out the initial evaluation).

	G1	**G2**	**G3**	**G4**	**G5**	**G6**	**G7**	**G8**
G1	XXX	0.1629	**0.0186**	**0.0228**	0.9280	0.6185	0.6308	**0.0023**
G2		XXX	0.4544	0.5069	0.1452	0.3007	0.0797	**<0.0001**
G3			XXX	0.9208	**0.0164**	**0.0413**	**0.0078**	**<0.0001**
G4				XXX	**0.0201**	0.0506	**0.0096**	**<0.0001**
G5					XXX	0.5589	0.6972	**0.0030**
G6						XXX	0.3327	**0.0005**
G7							XXX	**0.0090**
G8								XXX

Legend*.* *G1 (29 premature newborns admitted to the ICU); G2 (21 healthy newborns); G3 (31 healthy children); G4 (32 healthy young adults); G5 (27 elderly adults hospitalized in nursing homes); G6 (40 patients with chronic liver disease); G7 (22 patients with chronic kidney disease on hemodialysis program); G8 (9 individuals with a state of brain death or who died in the short term after carrying out the initial evaluation). Statistical test: Kruskal-Wallis: all pairwise comparisons (Conover-Iman); significant P-values are in bold.*

Table 2.
Pair-to-pair statistical comparisons (P values) were performed using the Kruskal-Wallis non-parametric test and the Conover-Iman post-test.

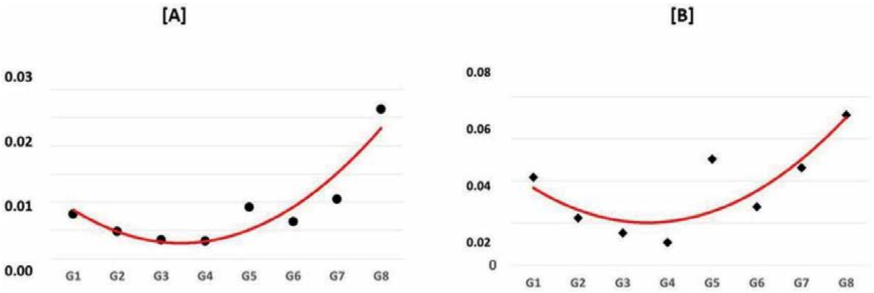


Figure 4.
Distribution of mean global Gini coefficient values [A] and maximum global Gini coefficient values [B], according to the clinical groups. G1 (29 premature newborns admitted to the ICU); G2 (21 healthy newborns); G3 (31 healthy children); G4 (32 healthy young adults); G5 (27 elderly adults hospitalized in nursing homes); G6 (40 patients with chronic liver disease); G7 (22 patients with chronic kidney disease on hemodialysis program); G8 (9 individuals with a state of brain death or who died in the short term after carrying out the initial evaluation). Y-axis [A] = mean values of intragroup Gini Coefficients; Y-axis [B] = maximum values of intragroup Gini Coefficients.

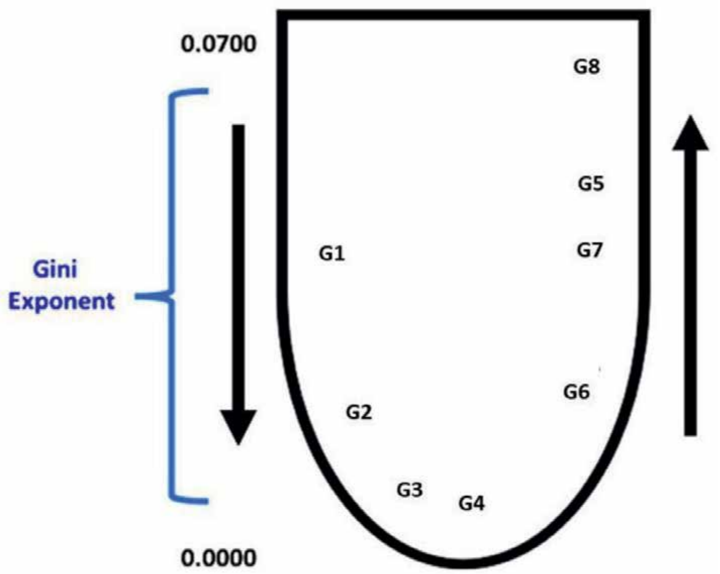


Figure 5.
Parabolic behavior with near similarity of values among equivalent homeostatic levels of the clinical groups. G1 (29 premature newborns admitted to the ICU); G2 (21 healthy newborns); G3 (31 healthy children); G4 (32 healthy young adults); G5 (27 elderly adults hospitalized in nursing homes); G6 (40 patients with chronic liver disease); G7 (22 patients with chronic kidney disease on hemodialysis program); G8 (9 individuals with a state of brain death or who died in the short term after carrying out the initial evaluation).

characteristics, hence the GINI coefficients have higher mean values. The same behavior was seen when considering the median and maximum values for the GC (**Figures 4** and 5).

This clinical heterogeneity, within each group, is based on the concept of non-stationarity. Stationarity means that the statistical properties of the signal remain the same throughout the period of recording and refers to the invariance of its distributional characteristics over time, being essential for HRV measurements in the frequency domain [18–20].

Nonstationarity translates into the presence of values that are more different from each other and leads, consequently, to higher Gini Coefficient values. In agreement, in groups G1, G5, G6, G7, and G8, higher overall values were found.

It is convenient to emphasize again that high or low heart rate variability may be independent of higher or lower CG values, as it all depends on the degree of present stationarity.

The allocation in parabolic distribution shown in **Figure 5** is quite illustrative as it is consistent with the behavior of homeostatic levels, which are still reduced in view of the present autonomic immaturity (G1), become higher from healthy newborns to children and to young adults when it reached the point of greatest autonomic maturity and continuous regression begins with the concurrence of progressively more serious illnesses until death occurs [21].

It is concluded that the Gini Coefficient, in addition to being a useful tool in the sense of assessing the inequality within a time series of any HRV variable, was also useful in characterizing the parabolic spatial distribution of the different groups studied, suggesting that in premature infants, elderly individuals and patients with severe diseases there is a significantly greater presence of non-stationarity.

Acknowledgements

This work has been supported by the Brazilian research agency (FAPESP).

The second author was funded by the grant 2021/04472-0 Sao Paulo Research Foundation.

Author details

Moacir Fernandes de Godoy[1,2]*, Beatriz Arid Rudnick[1]
and João Victor de Menezes Reichert[2]

1 Sao Jose do Rio Preto Medical School – Famerp, São Paulo, Brazil

2 Transdisciplinary Nucleus for Chaos and Complexity Studies, São Paulo, Brazil

*Address all correspondence to: mf60204@gmail.com

References

[1] Guyton AC, Hall JE. Guyton and Hall Textbook of Medical Physiology. Twelfth ed. Saunders: Elsevier Ed; 2011

[2] dos Reis AF, Bastos BG, Mesquita ET, Romêo Filho LJ, da Nóbrega AC. Parasympathetic dysfunction, heart rate variability and cholinergic stimulation after acute myocardial infarction. Arquivos Brasileiros de Cardiologia. 1998;**70**(3):193-199

[3] Carney RM, Freedland KE, Stein PK, Miller GE, Steinmeyer B, Rich MW, et al. Heart rate variability and markers of inflammation and coagulation in depressed patients with coronary heart disease. Journal of Psychosomatic Research. 2007;**62**(4):463-467

[4] Carnethon MR, Liao D, Evans GW, Cascio WE, Chambless LE, Rosamond WD, et al. Does the cardiac autonomic response to postural change predict incident coronary heart disease and mortality? American Journal of Epidemiology. 2002;**155**(1):48-56

[5] Limongelli G, Miele T, Pacileo G, Di Salvo G, Calabro P, Ancona R, et al. Heart rate variability is a weak predictor of sudden death in children and young patients with hypertrophic cardiomyopathy. Heart. 2007;**93**(1): 117-118

[6] Karas M, Larochelle P, LeBlanc RA, Dubé B, Nadeau R, Champlain J. Attenuation of autonomic nervous system functions in hypertensive patients at rest and during orthostatic stimulation. Journal of Clinical Hypertension (Greenwich, Conn.). 2008;**10**(2):97-104

[7] Terathongkum S, Pickler RH. Relationships among heart rate variability, hypertension, and relaxation techniques. Journal of Vascular Nursing. 2004;**22**(3):78-82

[8] Furuland H, Linde T, Englund A, Wikström B. Heart rate variability is decreased in chronic kidney disease but may improve with hemoglobin normalization. Journal of Nephrology. 2008;**21**(1):45-52

[9] Javorka M, Trunkvalterova Z, Tonhajzerova I, Javorkova J, Javorka K, Baumert M. Short-term heart rate complexity is reduced in patients with type 1 diabetes mellitus. Clinical Neurophysiology. 2008;**119**(5):1071-1081

[10] Van Mierlo T, Hyatt D, Ching AT. Employing the Gini coefficient to measure participation inequality in treatment-focused digital health social networks. Network Modeling Analysis in Health Informatics and Bioinformatics. 2016;**5**(1):32

[11] Veisani Y et al. Income inequality by Gini-coefficient on suicide death in Iran: A review of national data. Iranian Journal of Public Health. 2019;**48**(8):1512

[12] O'Hagan S et al. GeneGini: Assessment via the Gini coefficient of reference "housekeeping" genes and diverse human transporter expression profiles. Cell Systems. 2018;**6**(2):230-244

[13] Muelas MW et al. The role and robustness of the Gini coefficient as an unbiased tool for the selection of Gini genes for normalising expression profiling data. Scientific Reports. 2019;**9**(1):1-21

[14] Cai Y-m et al. A novel application of Gini coefficient for the quantitative measurement of bacterial aggregation. Scientific Reports. 2019;**9**(1):1-12

[15] Erdenee O et al. Distribution of health care resources in Mongolia using the Gini coefficient. Human Resources for Health. 2017;**15**(1):56

[16] Aubert AE, Seps B, Beckers F. Heart rate variability in athletes. Sports Medicine. 2003;**33**(12):889-919

[17] Malik M et al. Heart rate variability: Standards of measurement, physiological interpretation, and clinical use. Task Force of the European Society of Cardiology and the North American Society of Pacing and Electrophysiology. Circulation. 1996;**93**(5):1043-1965

[18] Peng CK, Havlin S, Stanley HE, Goldberger AL. Quantification of scaling exponents and crossover phenomena in nonstationary heartbeat time series. Chaos. 1995;**5**(1):82-87. DOI: 10.1063/1.166141

[19] Gao J, Gurbaxani BM, Hu J, Heilman KJ, Emanuele Ii VA, Lewis GF, et al. Multiscale analysis of heart rate variability in non-stationary environments. Frontiers in Physiology. 2013;**4**:119. DOI: 10.3389/fphys.2013.00119

[20] Catai AM, Pastre CM, Godoy MF, Silva E, Takahashi ACM, Vanderlei LCM. Heart rate variability: Are you using it properly? Standardisation checklist of procedures. Brazilian Journal of Physical Therapy. 2020;**24**(2):91-102. DOI: 10.1016/j.bjpt.2019.02.006

[21] Fernandes de Godoy M, Lima GM. Heart rate variability as a marker of homeostatic level. Autonomic nervous system – special interest topics. London, UK: IntechOpen; 2022. DOI: 10.5772/intechopen.102500

Chapter 5

Handling Missing Data in the Time-Series Data from Wearables

Jay Darji, Nupur Biswas, Lawrence D. Jones and Shashaanka Ashili

Abstract

Wearable technology is being used for tracking continuous events in various sectors of our lives. Wearables contain different types of sensors which can acquire movement data, blood pressure, blood sugar, temperature, and other physiological parameters. These parameters are recorded in the form of seamless univariate or multivariate time-series data. Very often, however, the data contains missing datum which disrupts the continuity of the data making it difficult to analyze the data. The missing part of the data needs to be imputed to make the remaining available data applicable. Choosing the proper imputation method is crucial for fruitful analysis and extracting underlined features from the data. In this context, this chapter discusses sensors associated with wearable technology which generate the time-series data, missing data in the wearables' time-series data, and various imputation methods being used for imputing the missing data.

Keywords: missing data, imputation, time-series data, wearables, digital health

1. Introduction

Wearable technology refers to the electronic devices which users can wear. Wearables are available in different manifestations like watches, eyeglasses, tattoos, strips, gloves, belts, footwears, clothes, and can be associated with different parts of the body like wrist, eyes, skin, chest, waist, and foot [1, 2]. These wearables are often called smart devices as they are remotely connected by different wireless communication technologies like the internet and Bluetooth. The data is also stored on cloud platforms and consequently shared. It helps to synchronize wearables with mobile phones and laptops. This chapter is focused on healthcare applications. The major purpose of wearing these smart devices is to track real-time physiological information from the body. The information gathered by wearables is further analyzed by different algorithms for extracting features relevant to the users' health. Finally, it is shared with users, and users' healthcare providers. This information is utilized for the purpose of self-management, monitoring health conditions, clinical decision-making, and preventive measures for different health conditions [3, 4].

The popularity of wearables is astonishing with a forecast of 20% annual growth rate for its market [5]. The major reason for the popularity of wearables is that it collects data in a non-invasive or a minimal invasive manner. For many devices, no intervention of clinicians is required. This advantage has turned wearable devices into smart consumer devices. Among the diverse wearable devices and gadgets, smartwatches are the most popular. Interested users can buy it from the market and use it for self-monitoring. The advancement of technology has turned it into a helpful supplementary in healthcare and disease monitoring for clinicians [6]. Smartwatches can primarily track heart rate, measure calories burnt, count steps, and monitor sleep. Moreover, the use of smartwatches has extended beyond self-monitoring. Smartwatches are being used for routine remote monitoring, diagnostics purposes, predicting disease risk factors and even for treatments. Smartwatch-derived data is being used for diagnosing cardiovascular diseases [7, 8], neurological disorders [9], fatty liver diseases [10], Parkinson's disease [11], and metabolic disorders including diabetes [12].

Data collection has become feasible due to technological advancement in developing various types of sensors. Wearable devices contain inertial sensors, optical sensors, temperature sensors, barometric sensors, and others. This is a rapidly growing area of research as improving the sensitivity of sensors, incorporating multiple sensors in a limited space, energy efficiency of the sensors, connectivity of the sensors, and lighter weight materials are the associated challenges. Sensors are also being specially designed for specific purposes [13].

Due to the presence of different types of sensors and their continuous operations, wearable devices generate a huge amount of multivariate data often considered as "big data" which is personalized also [14, 15]. Data consists of values of different parameters recorded continuously maintaining a regular interval of time. The time-series data from smartwatches contain heart rate, step counts, elevation, sleep duration, and sleep quality. Wearables also provide blood pressure, blood oxygen level, electrocardiogram (ECG), and body temperature. The cardiac data from wearables are being used for diagnosing and monitoring several disease conditions like coronary artery diseases [16], atrial fibrillation [17–19], and atrial arrhythmias [20]. The gyroscopes used in the wearables provide time series data of rotation along three mutually perpendicular directions. This data can further be analyzed to identify different activities and body postures like walking, running, sitting, moving up or down along the staircase, and standing [21]. It also detects tremors, and bradykinesia in case of users having Parkinson's disease [22].

However, very often due to several reasons time series data generated by wearables contain missing data. The reasons may include but not limited to that the device was not used for a certain interval of time, the device was used but data was not transmitted, or the recorded data was noisy and hence removed [23]. The presence of missing data or missing values of any recorded parameter downgrades the data quality. Wearable data are very often analyzed by artificial intelligence-based machine learning algorithms, for which continuous data is preferred [24]. Ignoring the timestamps of missing data is not a good choice as it is a complete loss of data. Also, multivariate data may have missing values only for a particular parameter. Imputing the missing data is a necessity for making the entire dataset useful. Hence, missing values are substituted with predicted values. Moreover, there are different aspects of missing data that need to be considered during imputation. Data may or may not be missing at random fashion. Data may be missing continuously or in a discrete manner [25].

Researchers have concentrated on developing effective ways of imputing missing values in time-series data. This chapter is focused on wearable technology and frequently used imputation methods in imputing missing values of time-series data in the context of wearables.

2. Wearable technology

Wearable technology comprising portable sensors is being used to collect various types of information in a continuous seamless fashion. It has appeared as a huge source of time-series data. Data is collected from biofluids like sweat, tears, saliva, and interstitial fluid.

Wearable devices are loaded with different types of sensors which include accelerometers, gyroscopes, magnetometers, heart rate sensors, oximetry sensors, barometric pressure sensors, ambient temperature sensors, skin conductance sensors, etc.

The most often used sensors are inertial sensors like accelerometers and gyroscopes. Such sensors can detect the motion of body parts like hands and limbs. The acceleration and angular velocity of the associated organ are measured in three mutually perpendicular directions using combinations of three accelerometers and gyroscopes. The acceleration is measured by measuring the displacement of a test mass attached to a suspension system [26]. This information has been analyzed to successfully detect different daily activities like walking, sitting, running, and moving up/down through steps [21]. It can even predict the risk of possible falls [27] and the user and even the caregivers can be alerted by remote monitoring. Such sensors are particularly useful for patients having gait-related disorders. Patients suffering from Parkinson's disease are also benefitted [28]. Magnetometers are also associated with motion sensors. It measures the earth's magnetic field to detect the direction of the motion.

Wrist-worn wearable devices, smartwatches, use optical sensors to measure heart rate by the photoplethysmography (PPG) method. They use green LED light which is absorbed by blood as red-colored blood absorbs green light most. The photosensors detect light reflected from blood and measure absorbance which is then quantified to the volume of the blood. The synchronous cardiac beats change blood volume within the vessel in each beat. Heart rate is extracted from the volume change measured from the changes in absorbance [29]. Apart from measuring heart rates, smartwatches like KickLL, are also designed to measure respiratory rates using the PPG method [30].

Smartwatches also measure peripheral blood oxygen saturation levels by measuring light absorbed by oxygenated and non-oxygenated hemoglobin in blood vessels. The oxygenated and non-oxygenated hemoglobin molecules absorb light of different wavelengths (940 nm and 660 nm, respectively) [31]. Smartwatches use red and infrared lights and have inbuilt sensors for measuring blood oxygen levels only when the user is not physically active. Blood with higher oxygen levels will absorb more red light compared to infrared light [32, 33].

Wearables can provide an estimate of body temperature. It is based on the epidermal measurement and hence does not provide the core body temperature of the user. However, long-term, continuous, and accurate measurement is possible [34]. Apart from monitoring regular health conditions, the recorded temperature plays an important role in detecting diseases like epilepsy [9] and COVID-19 [35].

Bioimpedance (BIA) is the response to the externally applied electric signal. Smartwatches are equipped with bioimpedance sensors. These sensors are used to measure body fat, composition, blood pressure, and body glucose levels [36]. A recent study, involving 75 participants reports that the measurements of smartwatch BIAs is comparable with laboratory measures for measuring body fat [37]. BIA sensors are also being used for blood pressure measurements [38]. The advantage of wearable sensors is that they can continuously measure blood pressure with a comfortable cuffless setup. Kireev et al. have designed self-adhesive grapheme-made electronic tattoos for measuring blood pressure [39]. BIA sensors measure pulse transit time and further machine learning based analysis estimates both diastolic and systolic blood pressures [40].

The wearables are often coupled with biosensors. A typical biosensor has two parts receptor and transducer. The receptor may consist of a specific enzyme, antibody, DNA or cell which identifies target analytes by bio-reaction. The transducer translates the bio-reaction into a measurable electrical signal. Depending on the bio-reaction, the transducer can be electrochemical, optical, thermal, or piezoelectric type [1].

3. Missing data

Time series data in biology is generated when information on a physiological parameter or a phenomenon is recorded with time. The presence of missing values can lead to the loss of information and statistical power [41], making typical data analysis techniques ineffective or challenging to use and can inject bias into estimates obtained from a statistical model [42, 43]. Missing data typically falls into one of the following three categories according to Little et al. [44]:

- Missing completely at random (MCAR): In cases where missing data is said to be missing completely at random (MCAR), the missing values are independent of both the observed and unobserved data. This means that there is no systematic relationship between the existing data and the missing values [44, 45].

- Missing at random (MAR): In situations where missing data is said to be missing at random (MAR), the missing values are systematically related to the observed data, but not the unobserved data. In other words, although they cannot be predicted, the missing values have some relationship with other variables in the dataset that can be used to explain why they are missing [44, 45].

- Not missing at random (NMAR): Missing data is considered missing not at random (MNAR) when the missing values are systematically related to the unobserved data, which cannot be explained by other variables in the dataset. This means that there is a pattern to the missing data that is not accounted for by the available information [46].

High data quality criteria are essential for machine learning (ML) applications to provide reliable prediction performance and appropriate use of automated decision-making. Eliminating the missing data is one approach to solving this issue. However, if we just delete the data, we run the danger of losing important information. To impute the missing data would be a preferable course of action. In other words, we must extrapolate those missing numbers from the data that is already available. This chapter goes into further detail on the substantial work that the statistics community has done on the imputation of missing data. Techniques for handling missing data have been divided into three groups by Song et al. [47]:

1. Delete missing data: With this approach, the row with the missing data is simply deleted. However, due to its ease of use, this strategy is popular. Due to the possibility of leaving out important details, it can only be utilized when the missing data are modest in number as compared to the available data.

2. Tolerate missing data: In contrast to the missing data deletion approach, this method uses the whole dataset for the analysis and replaces the missing data points with the special value NULL. This approach cannot be used when the goal is to forecast the data point.

3. Impute missing data: The process of imputing missing data entails replacing estimated values for missing data points and then evaluating the entire dataset as if the imputed data were original. This approach is considered to be better when we have to deal with limited data, which is usually the case in most practical scenarios. Let us discuss some techniques used to impute missing data in the following section.

4. Imputation methods

Missing data imputation refers to replacing missing values with approximated ones. The current portion of this book chapter will go through various missing data imputation techniques. Statistical procedures, such as substituting missing values with the mean, median, or mode of the column containing missing values, are one of the most used approaches for dealing with missing data. While it is simple and quick, it alters the statistical character of the data set. This not only skews our histograms, but it also understates the variation in our data because we are making several values the same whereas, in reality, they evidently would not be the same [48]. Time series data can be seasonal or cyclical in nature, and they can follow a pattern. The standard statistical data imputation techniques compromise these components of time series data. Therefore, while this is a popular practice, it should be avoided while handling time-series data.

When mean imputation is not the best fit for the missing values, interpolation is used. Interpolation is a technique for estimating missing values between two known positions. An equally-spaced continuous time-series data can be represented as $y = \{y_t | \, t = t_0, t_0 + a, \dots, t_0 + Na\}$

$$y_{t_0}, y_{t_0+a}, y_{t_0+2a}, \dots, y_{t_0+Na} \tag{1}$$

where the series starts at time t_0, y_t is the measured value of the variable at time t, the series continues for N timestamps, and a is the interval between two consecutive observations, also called "sampling interval". If data for n continuous timestamps are missing, the series looks like

$$y_{t_0}, y_{t_0+a}, y_{t_0+2a}, \dots, y_{t_0+(i-1)a}, nan, nan, \dots, nan, nan, y_{t_0+(i+n+1)a}, \dots, y_{t_0+Na} \tag{2}$$

Here we consider data is missing continuously from time $t_0 + ia$ to time $t_0 + (i + n)a$, for n number of timestamps. The missing values are represented as *nan* (not a number).

The following are the most often used algorithms for imputing missing values in time series.

4.1 Linear interpolation

Linear interpolation is the simplest imputation method. In this method, missing data points are estimated considering the linearity between the data points available before and after missing data points. The missing period is replaced by [49, 50]:

$$y_t = y_{t_0+(i-1)a} + m(t - (t_0 + (i-1)a)) \tag{3}$$

Where, $m = \left(y_{t_0+(i-1)a} - y_{t_0+(i+n+1)a}\right)/(n+2)a$ is the slope of the straight line. This method is simple and fast; however, it is not useful if the data is seasonal. It misses the time-dependent variation in the data.

4.2 Exponential weighted moving average (EWMA)

The Exponential Weighted Moving Average (EWMA) method estimates the missing values as a weighted average of the historical data points, where the weighting factors decrease exponentially for the older timestamps. The timestamps in the near past have more weightage as compared to the timestamps in the distant past [51]. EWMA relates the predicted value and error in prediction.

$$EWMA = y_{t+1}^{*} = y_t^{*} + \lambda\left(y_t - y_t^{*}\right) \tag{4}$$

Where, y_t^{*} is the estimation of y at time t, λ $(0 < \lambda < 1)$ is the weight of the data at the time t and y_t is the data point at time t. It is a recursive process where the weighting constant λ determines the memory of the process [51].

Intuitively, having a weightage of λ implies how much "older" data is considered. λ = 1 means only the most recent data is considered. A higher value of λ means recent data is given weightage, whereas a lower value of λ imposes more weightage on older data. An optimum choice of λ is necessary. J. Stuart Hunter recommends a choice of λ between 0.1 and 0.3 [51].

According to Wijesekara et al. [52], EWMA outperformed mean imputation and interpolation in data imputation. EWMA resulted in an MSE of 0.297 when imputing 5% of missing data. Some advantages of using EWMA for data imputation include its simplicity and flexibility, making it suitable for a wide range of datasets with different patterns of missing data. Additionally, EWMA is known to preserve trend and seasonality in time-series data. However, there are some limitations to EWMA as an imputation method. For example, EWMA is also sensitive to parameter selection, which can significantly affect the imputed values.

4.3 k-nearest neighbor (kNN)

Another approach to imputing missing values in data is the k-nearest neighbor (kNN) method. This method is used when we have limited knowledge about the distribution of the data [53]. This technique is derived from the nearest neighbor (NN) approach, which involves using the value of the closest data point to fill in a missing value. However, the NN approach can sometimes be affected by outliers and result in overfitting. The kNN method addresses this issue by using the average or median of the target values from the k nearest neighbors to make a prediction, rather than just relying on a single nearest neighbor. This can help to reduce the risk of overfitting and improve the generalizability of the model [54, 55]. Using the kNN technique categorical missing values can be imputed using the majority rule among k neighbors, whereas missing numerical values can be handled by imputing the average value of k – closest neighbors. This is known as the majority/mean rule. kNN first finds the k nearest neighbors and imputes the missing points with the weighted average of these neighbor points. The number of nearest neighbors k to be considered appears as a hyperparameter and is to be tuned in the experiments. For a set of k nearest neighbors, $D_k = \left\{ \left(t_j, y_j, 1 \right) \right\}$ where $j = 1, 2, \dots, k$, the kNN estimator predicts [55].

$$y = arg\ max_v \left\{ \sum_{\left(t_j, y_j, 1\right) \in D_k} \mathbf{1}\left(y_j = v \right) \right\} \tag{5}$$

when y is categorical. Here v is a value in the domain of target feature y. $\mathbf{1}\left(y_j = v \right)$ is an indicator function that returns 1 when $v \in y_j$ and returns 0 otherwise. For numerical outcomes, the kNN estimator predicts [55]

$$y = \frac{1}{k} \sum_{j=1}^{k} y_j \tag{6}$$

The advantages of the kNN imputation method include simplicity and model-free approach. However, there are challenges associated with the proper choice of k and the subsequent selection of k nearest neighbors. Lall et al. recommend $k = \sqrt{N}$ for $N > 100$, N being total number of observations or timestamps [56]. Minkowski distance is a commonly used approach to define the k nearest neighbors. Generally, this method gives good accuracy but it requires a lot of calculation and memory to predict the missing values [55]. kNN works well for randomly missing values; however, wearables data are often correlated over time. Lagged kNN method combined with Fourier transform imputation has shown better imputation for biomedical time series data [57]. Several modified kNN methods have been applied to impute ECG signal and hence can be applied to wearables data [58].

Kenyhercz et al. [59] conducted a study on data imputation using various methods such as KNN, mean imputation, hot deck imputation, and iterative robust model, with 25% and 50% data missing. The study used craniometric data from 352 individuals representing four population groups, which were publicly available in Howell's dataset. The results showed that KNN performed better than mean imputation and hot deck imputation, and iterative robust model.

4.4 Expectation maximization (EM)

The EM method is an iterative process that comprises two steps i.e., estimation step (E-step) and maximization step (M-step). The E-step calculates expected values with the help of complete data points. In M-step, the parameters are optimized for best estimates [60]. The process repeats itself until the changes between expected values in each iteration become negligible or reaches convergence.

If we consider our time series data set as $y = \{y_{obs}, y_{mis}\}$ where y_{obs} and y_{mis} represent observed and missing data set, respectively. The data set can be described by probability or density function $p(y|\theta)$ governed by a set of parameters. Here $p(y|\theta)$ is a function of y for a given θ. The likelihood function can be defined from the density function $L(\theta|y) = p(y|\theta)$ where $L(\theta|y)$ is a function of parameter θ for data y. The initial estimate parameter is θ^0. The E-step finds the objective function $Q(\theta|\theta^t)$ which provides the expected value of the observed data log-likelihood ($l(\theta|y)$) for the given observed data and current parameters

$$Q(\theta|\theta^t) = E\left[l(\theta|y)|y^0, \theta^t\right] \tag{7}$$

The M-step determines parameter vector θ^{t+1} which maximizes the log likelihood of the imputed data. At each iteration, the observed data likelihood increases but due to imposed upper bound, convergence is achieved. For all θ, the objective function satisfies,

$$Q\left(\theta^{t+1}|\theta^t\right) \geq Q(\theta|\theta^t) \tag{8}$$

EM is deemed to be superior then many substitution algorithms such as mean, median, and mode. EM is assumed to produce an unbiased estimation of values MCAR and little biased estimation for values MAR. The limitation of the algorithm is that, while it delivers reliable and correct estimation for missing data, the standard error is low, resulting in inaccuracy in several statistical tests (e.g., t-test). Thus, this method can only be used when the standard error is not crucial like in the case of factor analysis, which does not have the p-values [61].

EM algorithm has been applied to impute missing data of fetal heart rate data [62] as well as to ischemic heart rate data [63]. Aljuaid et al. [64] used the EM (Expectation-Maximization) algorithm to impute missing data in datasets obtained from the UCI machine learning repository. The study compared the performance of the EM algorithm with other imputation methods such as mean imputation, hot deck, kNN, and C5.0. The results showed that the EM algorithm produced the lowest RMSE (Root Mean Square Error) value, with an approximate value of 0.36. These findings suggest that the EM algorithm is a highly effective method for imputing missing data, and can outperform other popular imputation methods in terms of accuracy.

4.5 Kalman prediction

In 1960 R. E. Kalman proposed a method of predicting a signal from its past observations [65]. It is a linear model for discrete data and is widely used for predicting time-series data [66]. It estimates observations based on a set of measurements taken over time that contains noise. This algorithm consists of two processes: a prediction stage and a correction stage. The state and error covariance of the next data point is predicted with the help of the current state and error covariance in the prediction stage. The Kalman filter can be formally expressed as:

Step 1: Prediction process

It predicts *a priori* state estimate $\hat{y}_{k+1}^{-}$ and *a posteriori* state estimate $\hat{y}_k$ by measurement equation,

$$\hat{y}_{(k+1)}^{-} = A_k \hat{y}_k \tag{9}$$

Here A is a $n \times n$ matrix that relates state $\hat{y}_k$ at time step k with the state of the next time step $(k + 1)$. It further estimates the next state *a posteriori* estimate error covariance P_{k+1}^{-} based on the current state covariance P_k considering the noise covariance Q by transition equation

$$P_{k+1}^{-} = AP_k A^T + Q \tag{10}$$

Step 2: Correction process

The prediction is further corrected by Kalman gain K (a $n \times m$ matrix) which reflects confidence between the predicted result and actual measurement z_k.

$$K_{k+1} = P_{k+1}^{-} H^T \left(HP_{k+1}^{-} H^T + R \right)^{-1} \tag{11}$$

Here R is the measurement noise covariance matrix. The matrix H ($m \times n$ matrix) relates with the state of the measurement z_k. The gain matrix updates the estimation by

$$\hat{y}_{k+1} = \hat{y}_{k+1}^{-} + K_k \left(z_k - H\hat{y}_{k+1}^{-} \right) \tag{12}$$

The covariance is also updated,

$$P_{k+1} = (I - K_{k+1} H) P_{k+1}^{-} \tag{13}$$

We observe that there is a feedback relation between *a priori* estimate and *a posteriori* estimate. In Kalman gain, noise in measurement determines the weightage on the predicted values [67]. Being a recursive relation, the Kalman filter is used for the imputation of missing values by measurement and transition equations [68].

Kalman prediction is used for various types of physiological data like heart rate variability and body weight variability [69–71] as well as for other time series data [72]. Luis Alfonso et al. [73] used data imputation techniques on air quality data and evaluated the results using RMSE values. The study compared the performance of the Kalman smoothing algorithm with other imputation methods such as kNN and RF (Random Forest). The results showed that the Kalman smoothing algorithm performed well and outperformed the other methods in terms of accuracy, as evidenced by the lower RMSE values obtained.

The Kalman filter is a popular method for data imputation in time series data. It can incorporate temporal relationships between observations for more accurate imputations. However, it can be computationally intensive and require knowledge of system dynamics and parameters, which can be challenging to estimate. Despite these limitations, the Kalman filter remains effective for time series data imputation.

5. Conclusion

We observe that the wearables generate a massive amount of personalized multivariate time-series data. This data is generated by different types of sensors which include gyroscopes, accelerometers, magnetometers, light sensors, and temperature sensors. The technological advances have improved the performance of different types of sensors which are also energy efficient, and occupy lesser space. The advancement of computational efficiency has made the analysis of these data feasible. Data is processed in real-time and further analyzed by software which is often powered by artificial intelligence. Apart from real-time monitoring of patients, data extracted from wearable devices have successfully been used for monitoring and diagnosing different diseases. The diseases include cardiological disorders, metabolic disorders, neurological disorders, sleep quality, and others. The robust analysis of this data is a requisite to make the data insightful and beneficiary to the users. The analysis demands the availability of complete data without any missing parts. However, in reality, data contains missing data, compensation for which requires imputation. There are various types of missing data, depending on the randomness of missing. The choice of imputation method is critical as subsequent analysis depends on the imputed data. Several imputation methods are being used whose success depends on the type, trend, and amount of available data as well as the missing data. We have observed that instead of using entire days' data, use of data binned around the missing period generates better imputed data [74]. Finally, we can conclude that the data which is missing is cardinal along with the available time-series data and hence should be imputed in a worthwhile way.

Conflict of interest

The authors declare no conflict of interest.

Author details

Jay Darji[1], Nupur Biswas[1]*, Lawrence D. Jones[2] and Shashaanka Ashili[2]

1 Rhenix Lifesciences, Hyderabad, India

2 CureScience, San Diego, USA

*Address all correspondence to: nupur@rhenix.org; nupurbiswas@gmail.com

References

[1] Kim J, Campbell AS, de Ávila BEF, Wang J. Wearable biosensors for healthcare monitoring. Nature Biotechnology. 2019;**37**(4):389-406

[2] Rodrigues JJPC, De Rezende Segundo DB, Junqueira HA, Sabino MH, Prince RMI, Al-Muhtadi J, et al. Enabling technologies for the Internet of health things. IEEE Access. 2018;**6**:13129-13141

[3] Lee JH, Lee KH, Kim HJ, Youk H, Lee HY, Lee JH, et al. Effective prevention and management tools for metabolic syndrome based on digital health-based lifestyle interventions using healthcare devices. Diagnostics. 2022;**12**(7):1730

[4] Dunn J, Runge R, Snyder M. Wearables and the medical revolution. Persian Medicine. 2018;**15**(5):429-448

[5] Ometov A, Shubina V, Klus L, Skibińska J, Saafi S, Pascacio P, et al. A survey on wearable technology: History, state-of-the-art and current challenges. Computer Networks. 2021;**193**:108074

[6] Sharma A, Badea M, Tiwari S, Marty JL. Wearable biosensors: An alternative and practical approach in healthcare and disease monitoring. Molecules. 2021;**26**(3):748

[7] Wang YC, Xu X, Hajra A, Apple S, Kharawala A, Duarte G, et al. Current advancement in diagnosing atrial fibrillation by utilizing wearable devices and artificial intelligence: A review study. Diagnostics. 2022;**12**(3):689

[8] Torres-Soto J, Ashley EA. Multi-task deep learning for cardiac rhythm detection in wearable devices. NPJ Digital Medicine. 2020;**3**(1):116

[9] Tang J, El Atrache R, Yu S, Asif U, Jackson M, Roy S, et al. Seizure detection using wearable sensors and machine learning: Setting a benchmark. Epilepsia. 2021;**62**(8):1807-1819

[10] Schneider CV, Zandvakili I, Thaiss CA, Schneider KM. Physical activity is associated with reduced risk of liver disease in the prospective UK Biobank cohort. JHEP Reports. 2021;**3**(3):100263

[11] Ancona S, Faraci FD, Khatab E, Fiorillo L, Gnarra O, Nef T, et al. Wearables in the home-based assessment of abnormal movements in Parkinson's disease: A systematic review of the literature. Journal of Neurology. 2022;**269**:100

[12] Chakrabarti S, Biswas N, Jones LD, Kesari S, Ashili S. Smart consumer wearables as digital diagnostic tools : A review. Diagnostics. 2022;**12**(9):2110

[13] Vijayan V, Connolly J, Condell J, McKelvey N, Gardiner P. Review of wearable devices and data collection considerations for connected health. Sensors. 2021;**21**(16):5589

[14] Dai H, Younis A, Kong JD, Puce L, Jabbour G, Yuan H, et al. Big data in cardiology: State-of-art and future prospects. Frontier in Cardiovascular Medicine. 2022;**9**:844296

[15] Chen S, Qi J, Fan S, Qiao Z, Yeo JC, Lim CT. Flexible wearable sensors for cardiovascular health monitoring. Advanced Healthcare Materials. 2021;**10**(17):e2100116

[16] Ukil A, Bandyopadhyay S, Puri C, Pal A, Mandana K. Cardiac condition monitoring through

photoplethysmogram signal denoising using wearables: Can we detect coronary artery disease with higher performance efficacy? In: IEEE Conference Publication | IEEE Xplore. In: IEEE Computing in Cardiology Conference. Vancouver, BC, Canada; 2016

[17] Tison GH, Sanchez JM, Ballinger B, Singh A, Olgin JE, Pletcher MJ, et al. Passive detection of atrial fibrillation using a commercially available smartwatch. JAMA Cardiology. 2018; **3**(5):409-416

[18] Bashar SK, Han D, Hajeb-Mohammadalipour S, Ding E, Whitcomb C, McManus DD, et al. Atrial fibrillation detection from wrist photoplethysmography signals using smartwatches. Scientific Reports. 2019; **9**(1):15054

[19] Inui T, Kohno H, Kawasaki Y, Matsuura K, Ueda H, Tamura Y, et al. Use of a smart watch for early detection of paroxysmal atrial fibrillation: Validation study. JMIR Cardiology. 2020;**4**(1):e14857

[20] Fedorin I, Slyusarenko K. Consumer smartwatches as a portable PSG: LSTM based neural networks for a sleep-related physiological parameters estimation. In: Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS. Mexico: Institute of Electrical and Electronics Engineers Inc; 2021. pp. 849-852

[21] Nemati E, Liaqat D, Rahman MM, Kuang J. A novel algorithm for activity state recognition using smartwatch data. In: 2017 IEEE Healthcare Innovations and Point of Care Technologies, HI-POCT 2017. Bethesda, MD, US. 2017

[22] Khwaounjoo P, Singh G, Grenfell S, Özsoy B, MacAskill MR, Anderson TJ, et al. Non-contact hand movement analysis for optimal configuration of smart sensors to capture Parkinson's disease hand tremor. Sensors (Basel). 2022;**22**(12):4613

[23] Wu X, Mattingly S, Mirjafari S, Huang C, Chawla NV. Personalized imputation on wearable-sensory time series via knowledge transfer. International Conference on Information and Knowledge Management, Proceedings. 2020;**10**:1625-1634

[24] Emmanuel T, Maupong T, Mpoeleng D, Semong T, Mphago B, Tabona O. A survey on missing data in machine learning. Journal of Big Data. 2021;**8**(1):1-37

[25] Mack C, Su Z, Westreich D. Types of missing data. In: Managing Missing Data in Patient Registries: Addendum to Registries for Evaluating Patient Outcomes: A User's Guide. Third ed. Maryland, US: Agency for Healthcare Research and Quality (US); 2018

[26] Sigcha L, Pavón I, Arezes P, Costa N, De Arcas G, López JM. Occupational risk prevention through smartwatches: Precision and uncertainty effects of the built-In accelerometer. Sensors. 2018; **18**(11):3805

[27] Mauldin TR, Canby ME, Metsis V, Ngu AHH, Rivera CC. SmartFall: A smartwatch-based fall detection system using deep learning. Sensors. 2018; **18**(10):3363

[28] Powers R, Etezadi-Amoli M, Arnold EM, Kianian S, Mance I, Gibiansky M, et al. Smartwatch inertial sensors continuously monitor real-world motor fluctuations in Parkinson's disease. Science Translational Medicine. 2021;**13**:579

[29] Allen J. Photoplethysmography and its application in clinical physiological

measurement. Physiological Measurement. 2007;**28**(3):R1

[30] Hoilett OS, Twibell AM, Srivastava R, Linnes JC. Kick LL: A smartwatch for monitoring respiration and heart rate using Photoplethysmography. In: Annual International Conference of the IEEE Engineering in Medicine and Biology Society. Honolulu, HI, USA: NIH Public Access; 2018. p. 3824

[31] Spaccarotella C, Polimeni A, Mancuso C, Pelaia G, Esposito G, Indolfi C. Assessment of non-invasive measurements of oxygen saturation and heart rate with an Apple smartwatch: Comparison with a standard pulse oximeter. Journal of Clinical Medicine. 2022;**11**(6):1467

[32] How do I track blood oxygen saturation (SpO_2) with my Fitbit device? [Internet]. Available from: https://help.fitbit.com/articles/en_US/Help_article/2459.htm [Accessed: November 2, 2022]

[33] How to use the Blood Oxygen app on Apple Watch – Apple Support (IN) [Internet]. Available from: https://support.apple.com/en-in/HT211027 [Accessed: November 2, 2022]

[34] Magno M, Salvatore GA, Mutter S, Farrukh W, Troester G, Benini L. Autonomous smartwatch with flexible sensors for accurate and continuous mapping of skin temperature. In: IEEE International Symposium on Circuits and Systems. Montreal, QC, Canada: Institute of Electrical and Electronics Engineers Inc.; 2016. pp. 337-340

[35] Gadaleta M, Radin JM, Baca-Motes K, Ramos E, Kheterpal V, Topol EJ, et al. Passive detection of COVID-19 with wearable sensors and explainable machine learning algorithms. NPJ Digital Medicine. 2021;**4**(1):166

[36] Bertemes-Filho P, Morcelles KF. Wearable bioimpedance measuring devices. In: Simini F, Bertemes-Filho P, editors. Medicine-Based Informatics and Engineering. Switzerland: Springer Science and Business Media Deutschland GmbH; 2022. pp. 81-101

[37] Bennett JP, Liu YE, Kelly NN, Quon BK, Wong MC, McCarthy C, et al. Next generation smartwatches to estimate whole body composition using bioimpedance analysis: Accuracy and precision in a diverse multiethnic sample. The American Journal of Clinical Nutrition. 2022;**116**(5):1418-1429

[38] Huynh TH, Jafari R, Chung WY. A robust bioimpedance structure for smartwatch-based blood pressure monitoring. Sensors. 2018;**18**(7):2095

[39] Kireev D, Sel K, Ibrahim B, Kumar N, Akbari A, Jafari R, et al. Continuous cuffless monitoring of arterial blood pressure via graphene bioimpedance tattoos. Nature Nanotechnology. 2022;**17**(8): 864-870

[40] Ibrahim B, Jafari R. Continuous blood pressure monitoring using wrist-worn bio-impedance sensors with wet electrodes. In: IEEE Biomedical Circuits and Systems Conference, BioCAS 2018. Cleveland, OH, USA: Institute of Electrical and Electronics Engineers Inc.; 2018

[41] Kim J, Curry J. The treatment of missing data in multivariate analysis. Sociological Methods & Research. 1977; **6**(2):215-240

[42] Rubin DB. In: Rubin DB, editor. Multiple Imputation for Nonresponse in Surveys. First ed. Hoboken, NJ, USA: John Wiley & Sons, Inc.; 1987 (Wiley Series in Probability and Statistics)

[43] Becker WE, Walstad WB. Data loss from Pretest to Posttest as a sample selection problem. The Review of Economics and Statistics. 1990;**72**(1): 184-188

[44] Little RJA, Rubin DB. Statistical Analysis with Missing Data. 1st ed. New Jersey, US: Wiley; 2019. pp. 1-449

[45] Bennett DA. How can I deal with missing data in my study? Aust N Z J Public Health. 2001;**25**:464-469

[46] Mack C, Su Z, Westreich D. Managing missing data in patient registries: addendum to registries for evaluating patient outcomes: A user's guide. 2018. Available from: https://europepmc.org/article/med/29671990 [Accessed: 2023 May 17]

[47] Song Q, Shepperd M. Missing data imputation techniques. International Journal of Business Intelligence and Data Mining. 2007;**2**(3):261-291

[48] Little RJA. Regression with missing X's: A review. Journal of the American Statistical Association. 1992;**87**(420): 1237

[49] Junninen H, Niska H, Tuppurainen K, Ruuskanen J, Kolehmainen M. Methods for imputation of missing values in air quality data sets. Atmospheric Environment. 2004;**38**(18): 2895-2907

[50] Zhang Z. Missing data imputation: Focusing on single imputation. Annals of Translational Medicine. 2016;**4**(1):9

[51] Hunter JS. The exponentially weighted moving average. Journal of Quality Technology. 1986;**18**(4):203-210

[52] Wijesekara W, Liyanage L. Comparison of imputation methods for missing values in air pollution data: Case study on Sydney air quality index. In: Advances in Information and Communication: Proceedings of the 2020 Future of Information and Communication Conference (FICC). Vol. 2. San Francisco, US. 2020. pp. 257-269

[53] Parvin H, Alizadeh H, Minati B. A modification on K-nearest neighbor classifier. Global Journal of Computer Science and Technology. 2010; **10**(14):37

[54] Malarvizhi MR, Selvadoss TA. K-nearest neighbor in missing data imputation. International Journal of Engineering Research and Development. 2012;**5**(1):5-07

[55] Zhang S. Nearest neighbor selection for iteratively kNN imputation. Journal of Systems and Software. 2012;**85**(11): 2541-2552

[56] Lall U, Sharma A. A nearest neighbor bootstrap for resampling hydrologic time series. Water Resources Research. 1996; **32**(3):679-693

[57] Rahman SA, Huang Y, Claassen J, Heintzman N, Kleinberg S. Combining Fourier and lagged k-nearest neighbor imputation for biomedical time series data. Journal of Biomedical Informatics. 2015;**58**:207

[58] Yang F, Du J, Lang J, Lu W, Liu L, Jin C, et al. Missing value estimation methods research for arrhythmia classification using the modified kernel difference-weighted KNN algorithms. BioMed Research International. 2020; **2020**:7141725

[59] Kenyhercz MW, Passalacqua NV. Missing data imputation methods and their performance with biodistance analyses. In: Biological Distance

Analysis. Amsterdam, Netherlands: Elsevier; 2016. pp. 181-194

[60] Dempster AP, Laird NM, Rubin DB. Maximum likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society Series B (Methodological). 1977;**39**(1):1-38

[61] Molenberghs G, Verbeke G. Multiple imputation and the expectation-maximization algorithm. In: Models for Discrete Longitudinal Data. New York, NY: Springer; 2005. pp. 511-529

[62] Nokas G, Koutras A, Christoyannis I, Georgoulas G, Stylios CH, Groumpos P. Prediction of missing data in Cardiotocograms using the expectation maximization algorithm. In: Scattering and Biomedical Engineering. Singapore: World Scientific Pub Co Pte Lt; 2002. pp. 354-362

[63] Cenitta D, Vijaya Arjunan R, V PK. Engineered science ischemic heart disease multiple imputation technique using machine learning algorithm. Engineered Science. 2022;**19**:262-272

[64] Aljuaid T, Sasi S. Proper imputation techniques for missing values in data sets. In: 2016 International Conference on Data Science and Engineering (ICDSE). Vol. 1. 2016. p. 5

[65] Kalman RE. A new approach to linear filtering and prediction problems. Transactions of the ASME–Journal of Basic Engineering. 1960;**82**:35-45

[66] Sarkka S, Vehtari A, Lampinen J. Time series prediction by Kalman smoother with cross-validated noise density. In: IEEE International Joint Conference on Neural Networks. Budapest, Hungary: Institute of Electrical and Electronics Engineers (IEEE); 2004. pp. 1653-1657

[67] Zhang J, Welch G, Bishop G, Huang Z. A two-stage Kalman filter approach for robust and real-time power system state estimation. IEEE Transactions on Sustainable Energy. 2014;**5**(2):629-636

[68] Durbin J, Koopman SJ. Time Series Analysis by State Space Methods. Second ed. Oxford; 2012

[69] Turicchi J, O'Driscoll R, Finlayson G, Duarte C, Palmeira AL, Larsen SC, et al. Data imputation and body weight variability calculation using linear and nonlinear methods in data collected from digital smart scales: Simulation and validation study. JMIR Mhealth Uhealth. 2020;**8**(9):e17977

[70] Tarvainen MP, Georgiadis SD, Ranta-Aho PO, Karjalainen PA. Time-varying analysis of heart rate variability signals with a Kalman smoother algorithm. Physiological Measurement. 2006;**27**(3):225

[71] Lin S, Wu X, Martinez G, Chawla NV. Filling missing values on wearable-sensory time series data. In: Proceedings of the 2020 SIAM International Conference on Data Mining (SDM). Ohio, US: Society for Industrial and Applied Mathematics Publications; 2020. pp. 46-54

[72] Xie C, Huang C, Zhang D, He W. BiLSTM-I: A deep learning-based long interval gap-filling method for meteorological observation data. International Journal of Environmental Research and Public Health. 2021;**18**(19): 10321

[73] Menéndez Garcia LA, Menéndez Fernández M, Sokoła-Szewioła V, de Prado L, Ortiz Marqués A, Fernández López D, et al. A method of pruning and random replacing of known values for comparing missing data imputation

models for incomplete air quality time series. Applied Sciences. 2022;**12**(13): 6465

[74] Chakrabarti S, Biswas N, Karnani K, Padul V, Jones LD, Kesari S, et al. Binned data provide better imputation of missing time series data from wearables. Sensors. 2023;**23**(3):1454

Section 3

Methodological Approaches in Time Series Analysis

Chapter 6

Variability Analysis of Observational Time Series: An Overview of the Decomposition Methods for Non-Stationary and Noisy Signals

Olivier Delage, Hassan Bencherif, Thierry Portafaix, Alain Bourdier, René Tato Loua and Damaris Kirsch Pinheiro

Abstract

The analysis of observational data sequences in Geophysics consists of characterizing the underlying dynamics. An important preliminary step aims to analyze the variability related to the observed dynamic. The specific objectives related to this step are to remove noise, to determine the overall trend of the observational time series and to identify the relevant components contributing significantly to the original time series variability knowing that their number determines the dimensionality of the observed dynamics. Most of the observational time series have characteristics of non-stationarity and present fluctuations at all-time scales. In this context, variability analysis consists in representing time series in the time-frequency space and requires the development of specific numerical signal decomposition methods. The most commonly used techniques are adaptive and data-driven and among the most cited in the literature are the empirical mode decomposition, the empirical wavelet transform, and singular spectrum analysis. In this work, we describe all of these techniques and evaluate their ability to remove noise and to identify components corresponding to the physical processes involved in the evolution of the observed system and deduce the dimensionality of the associated dynamics. Results obtained with all of these methods on experimental total ozone columns and rainfall time series will be discussed and compared.

Keywords: time series analysis, non-stationary signals, complex systems, noise removal, empirical mode decomposition, wavelet decomposition, singular Spectrum analysis, underlying dynamics

1. Introduction

The notion of complexity is a characteristic present in most of systems of the real world. Intuitively, complexity constitutes a measure of the organization, the structure and the dynamics of a system i.e. its behavior over time. In the physical sciences, a complex system can be defined as composed of several entities interacting with each other on several scales of time and space. The complexity is created by the superimposition of the individual behavior of each entity with their interactions and makes the overall behavior of the system difficult to characterize and predict. Therefore, the characterization of the system dynamic is carried out from the global properties of the system which cannot be reduced to the sum of the characteristics of the entities of which it is composed. Many studies that aim to gain a better understanding of the complex system dynamics are based on the analysis of observational measurements data sequence. Most of the observational data sequences in geophysics derive from complex dynamics resulting from the superimposition of the individual behaviors of the physical processes involved in the evolution of the overall system with the interactions they have between them. As a consequence, time series are non-stationary and have fluctuations at all scales of time. Then, the analysis methodology consists of analyzing the variability of the time series. The method used is to decompose the observed time into the sum of noise, physically meaning full components and trends.

$$TS = noise + \sum\nolimits_i c_i + Trend, TS = \{x_k = x(k\delta t), k = 1, N\} \tag{1}$$

Where TS is the original times series composed of N measurement values x_k of the signal x with a sample time δt and c_i are the components that contribute significantly to the variability of the original signal. Each of c_i can be identified with a physical process involved in the overall evolution of the system. To implement such decomposition, two approaches have been considered: The first approach consists in getting a representation of TS in the time-frequency space which amounts to classify the fluctuations according to the temporal resolution at which they occur. Among the most commonly used techniques enabling to obtain a representation of time series in time frequency space, the empirical mode decomposition (EMD) is very popular. EMD is a relatively new method proposed by Huang et al. in 1998 well suited to non-stationary signals that decompose a time series into a finite number of components called "Intrinsic Mode Function" (IMFs) [1]. IMF captures the repeating behavior of the signal at some particular time scales. Unlike the Fourier or wavelet transforms, EMD enables to decompose a time signal into a set of basis signals derived from the data itself. The biggest advantage of the EMD method is to be totally adaptive and data-driven without the need for a-priori basis function selection for signal decomposition. As EMD acts in the time domain, another advantage of this method is to be close to the observed dynamics. Moreover, the EMD acts as a bank of bandpass filter [2] in the frequency domain and as a result, the main limiting factor of EMD is the frequency resolution which when it becomes too small can induce the mode-mixing phenomenon where the spectral content of some IMFs overlaps each other [3]. Although several techniques exist to overcome this problem [4–6], Gilles proposed an alternative entitled empirical wavelet transform (EWT) [7], which operates in the frequency domain and consists of elaborating a segmentation of the original signal's Fourier spectrum and building an appropriate wavelet filter bank on the segmented Fourier spectrum. The EWT allows a better frequency resolution and thus overcomes the mode-mixing problem by partitioning the spectrum of the original signal into separate spectral bands. However, although the EWT technique enables to detect the relevant frequencies involved in the original signal, it does not allow to associate to a specific mode of variability as EMD does. Because the variability modes provided by the EMD technique are closer to the observed dynamics than EWT, O. Delage [8] proposed an optimization of the EMD technique entitled Empirical Adaptive Wavelet Decomposition (EAWD) which combines the advantages of both EMD and EWT techniques. The heart of the EWT method lies in the segmentation of the original signal's Fourier spectrum. The main idea is to use the local maxima of the IMFs' spectral support returned by EMD to design the Fourier spectrum segmentation required to initiate the EWT method. IMFs involved in the Fourier spectrum segmentation are those selected to be relevant i.e. whose contribute more than 1% of the variability of the original signal. The final step consists in applying the EWT method on the segmented Fourier spectrum.

Another technique called Singular Spectrum Analysis (SSA) using a different approach is described in the scope of this chapter. Such a technique can decompose a non-stationary signal into a sum of independent components and also be able to reconstruct the underlying dynamics.

This document is structured around three sections: the first section is devoted to the description of all the signal decomposition techniques mentioned above. In the second section, three observational time series are analyzed by using EMD, EAWD and SSA techniques and the corresponding results are presented. In the last section, the results obtained respectively with the EMD, EAWD and SSA techniques are compared and discussed.

2. Review of the signal decomposition techniques

The methodology related to the signal variability analysis is composed of three main steps: (1) express the original signal as a sum of a finite number of components, (2) identify the components dominated by noise, (3) select the relevant components that contribute significantly to the signal variability with the objectives of determining the dimensionality of the underlying dynamics and identifying the relevant components to physical processes involved in the underlying dynamics.

2.1 The empirical mode decomposition (EMD)

2.1.1 EMD basics

In 1998, Huang et al. proposed an original method called the Empirical Mode Decomposition (EMD) whose purpose is to adaptively decompose any signals into oscillatory contributions. The EMD technique can be summarized as an iterative method where the signal can be decomposed into a local average m called trend and a strongly oscillating part called details. The trend is related to low frequencies while the details characterized by strong fluctuations are related to high frequencies. At each iteration, the high-frequency fluctuations part is separated from the low-frequencies trend and are reinjected as a new signal in the next iteration. During the iterations of the algorithm [1], the "details" related to the high frequencies are successively separated from the low-frequency part by using a procedure called "Sifting process". The main steps of the sifting process consist in: identify the signal local extrema; the local maxima and minima are then interpolated by using cubic splines to form respectively the upper and lower envelopes of the signal; the mean envelope is then determined by calculating the half sum of the upper and lower envelopes; the mean envelope is subtracted from the initial signal. The same procedure is reiterated on the remainder until the mean envelope is close to zero everywhere and the resultant signal is designated as the first IMF. The criterion for the sifting process to stop is generally set as:

$$0.2 \le SD = \sum_{i=1}^{T} \left[\frac{|h_{k-1}(t) - h_k(t)|^2}{h_{k-1}{}^2} \right] \le 0.3 \tag{2}$$

Where h_k is the result after k iterations of the sifting process. The higher order IMFs are iteratively extracted by using the above sifting process procedure until the remaining signal cannot be assimilated to an IMF knowing that a signal can be called IMF if it satisfies the two following criteria: the numbers of local maxima and local minima must differ by at most one; the half sum of its upper and lower envelopes is locally close to zero. The original signal can be finally expressed as:

$$TS = \sum_{m=1}^{M} IMF_m + R \tag{3}$$

Where R is the residual trend. EMD decomposes the data into M fundamental components with distinct time scales where the first component has the smallest time scale. The interesting fact about this algorithm is that it is adaptive and data-driven and is able to extract non-stationary parts of the original signal at different time scales as multi-resolution analysis does. In this context, Flandrin et al. [9] described the EMD as behaving as a dyadic filter bank as those involved in the multi-resolution analysis. Consequently, the maximum number of IMFs that can be extracted from the original time series is Log(N)/Log (2) if N is the time series size.

2.1.2 Noise removing-Noisy components identification

Generally, the original signal is corrupted by noise. As a consequence, a few IMFs may be the oscillations of the noise-free signal and the others correspond to noise. To determine which IMFs are noise-related components, a robust threshold is required. In this chapter, a "Detrended Fluctuations analysis" (DFA) technique is used to obtain such a threshold. The basic principle of the DFA [10, 11] is to compute how the time series fluctuations around the local trend varies as a function of the time scale. When applied to a time series TS(t), the first step of the DFA technique is to compute the time series TSI(t) composed of the cumulative sums of TS after removing its mean:

$$TSI(k) = \sum_{i=1}^{k} \left[x(i\delta t) - \bar{TS} \right], 1 \le k \le N \tag{4}$$

Where $\bar{TS}$ states the average of TS time series over [1, N], TSI is then divided into time windows of size n samples each. For each time window, the estimated local trend $TSI_n(k)$ is determined by using least-square linear fitting. Finally, the average root mean square (RMS) of the fluctuations at a specific time scale n, F(n) may be written as:

$$F(n) = \sqrt{\frac{1}{N} \sum\nolimits_{k=1}^{N} [TSI(k) - TSI_n(k)]^2} \tag{5}$$

The method suggested to identify the noise-related IMFs is to use the slope α of the curve log[F(n)]/log(n) as a threshold. The method consists in excluding IMFs with α value less than a threshold θ. For DFA, an α value of 0.5 characterizes an uncorrelated white noise. The commonly threshold value θ taken in the literature is 0.5 with a 0.2 confidence interval that is θ = 0.7.

2.1.3 Relevant components selection

Although the EMD is a powerful tool for analyzing complicated datasets, many irrelevant IMF may appear in the decomposition. A relevant IMF may be defined as an IMF that retains most of the information content of the signal. As a consequence, relevant IMFs would have a good correlation with the original signal while irrelevant ones would have a poor correlation. For discriminating between relevant and irrelevant IMFs, we use the Pearson correlation coefficient between each IMF and the original time series, i.e.:

$$CORR(IMF_m, TS) = \frac{cov(IMF_m, TS)}{sd(IMF_m) * sd(TS)} \quad \mathrm{m} = 1, \mathrm{M} \tag{6}$$

Where cov(IMFm,TS) is the covariance between the m^{th} IMF and the original time series, $\mathrm{sd}(\mathrm{IMF_m})$ is the standard deviation of the m^{th} IMF while sd(TS) is the standard deviation of the original time series. A threshold s is required for selecting relevant IMF: if $\mathrm{CORR}(\mathrm{IMF_m},\mathrm{TS}) > \mathrm{s}$, keep the m^{th} IMF. Otherwise, eliminate the m^{th} IMF and add it to the residual. As the threshold is different for different signals, a suitable threshold must be selected according to the relativity between IMfs and the original signal. In general threshold s can be the ratio of the mean value between the maximum and the minimum of $\{CORR(IMF_m, TS) m = 1, M\}$ that is:

$$s = \frac{M + m}{2\eta} \tag{7}$$

Where η is a coefficient greater than 1, $\mathrm{M} = \max_{\mathrm{m}}(\mathrm{CORR}(\mathrm{IMF_m},\mathrm{TS})$, $\mathrm{m} = \min_{\mathrm{m}}(\mathrm{CORR}(\mathrm{IMF_m},\mathrm{TS})$.

So that the correlation coefficients between each of the IMFs and the original signal would be in the interval [0,1], all IMF and original signals will be normalized at first. In the literature, two different thresholds are proposed $s_1 = \frac{M+m}{20}$ [12] and $s_2 = \frac{M}{10M-3}$ [13]. The threshold proposed in this work depends on the relative position de M/m and 10. If we suppose that $0.7 \leq \mathrm{M} \leq 1$, and $\mathrm{M/m} > 10$, m could be less than 0.07 which characterizes a very weak correlation. In this case s has to be greater than m and from (Eq. (7)), $\eta > 5.5$ which is verified when, $\mathrm{s} = \mathrm{s_1}$. If on the contrary when $\mathrm{M/m} \leq 10$, m could be greater than 0.1 and s must be close to m which from (Eq. (7)) can be written:

$$1 - \varepsilon \leq \frac{s}{m} = \frac{1 + \frac{M}{m}}{2} . \frac{1}{\eta} \leq 1 + \varepsilon \tag{8}$$

Given that $5 \leq \mathrm{M/m} \leq 10$, then $3 \leq (1 + \mathrm{M/m})/2 \leq 5.5$ and (Eq. (8)) becomes

$$\frac{5.5}{1 + \varepsilon} \leq \eta \leq \frac{3}{1 - \varepsilon} \tag{9}$$

If we set ε = 0.052, (Eq. (9)) becomes $3 \leq \eta \leq 5.2$ which is verified when $\mathrm{s} = \mathrm{s_2}$.

In summary, s is defined as follows:

$$s = \begin{cases} \dfrac{M+m}{20} if \dfrac{M}{m} > 10 \\ \dfrac{M}{10M-3} if \dfrac{M}{m} \leq 10 \end{cases} \tag{10}$$

2.1.4 EMD limiting factors-IMF oscillation cycle

As the EMD technique acts as a bank of bandpass filters [2], each IMF is associated with a specific bandpass and the oscillation cycle corresponds to the prevailing frequency in the bandpass. One way to get the IMFs oscillation cycle is to calculate the spectral density for each IMF and identify the frequency for which the spectral density is maximum. The local maxima occurring in the IMFs spectrum characterize the frequencies involved in the corresponding EMD mode of variability. One of the major limiting factors of the EMD is the frequency resolution which can, when it is not sufficient, induce the mode-mixing phenomenon where the spectral content of some IMFs overlaps each other. The source at the origin of mode-mixing can be categorized in three main groups: (1) presence of noise, (2) presence of intermittency, (3) presence of closely spaced spectral components. To overcome the inherent mode-mixing problem of the EMD, Wu and Huang [14] proposed the Ensemble Empirical Mode Decomposition (EEMD). In EEMD, white noise signals $n_i(t)$ are added to the original signal x(t); Because the white noise spectrum is evenly distributed, the white noise signals will be automatically distributed to the appropriate reference scales. Moreover, because of its zero-mean characteristic, the white noise will cancel itself out after many rounds of averaging. The specific steps of the EEMD algorithm are as follows:

- Initialize the number of ensemble members M. - Compute M realizations of white noise with the different variances that are added to the original signal: $x_i(t) = n_i(t, \sigma_i) + x(t)i = 1, M$ where n_i is a white noise of variance $\boldsymbol{\sigma}_i$. – Use the EMD algorithm to decompose $x_i(t)$ into IMFs: $x_i(t) = \sum_{j=1}^{J} IMF_{i,j} + R_i$ i = 1,M - Calculate the ensemble means of the decomposed IMFs: $IMF_j(t) = \frac{1}{M}\sum_{i=1}^{M} IMF_{i,j}$

2.2 Wavelet approaches

Wavelets are commonly used to analyze the variability of a signal. In the temporal domain, a wavelet basis is defined as the mother wavelet $\boldsymbol{\Psi}$ of zero mean, dilated with a parameter s > 0 and translated by $u \in \mathfrak{R}$:

$$\psi_{u,s}(t) = \frac{1}{\sqrt{s}}\psi\left(\frac{t-u}{s}\right) \tag{11}$$

For the wavelet decomposition of a time series x(t), the most widely used case is the dyadic one, $s = 2^j$. Then the wavelet decomposition of x is obtained by computing the inner product $W_x(kj)$ as:

$$W_x(k,j) = \langle x, \psi_{k,j} \rangle with \psi_{k,j}(t) = \frac{1}{\sqrt{2^j}}\psi\left(\frac{t-k}{2^j}\right), k \in Z \tag{12}$$

where j represents the resolution level. The decomposition is then similar to a multi-resolution analysis carrying out successive projections of x on a sequence of nested subspaces $V_j \in L^2(\mathfrak{R})$ j = [0, n], which leads to increasingly coarse approximations of x as j increases. At each resolution level j, the information about the signal x, at resolution j-1 in the subspace V_{j-1}, is split into two parts named approximation and details corresponding, respectively to the low frequencies and the high frequencies contained in the signal x. The approximation of x results from the orthogonal projection of x onto the subspace V_j and the information of details results from the orthogonal projection of x onto the subspace W_j orthogonal to V_j such that $V_{j-1} = V_j \ W_j$, where ⊕ denotes the direct sum of vector subspaces. Wavelets $\left\{\psi_{k,j}(t), k \ Z\right\}$ form the basis of W_j. According to the definition of multi-resolution analysis, there exists a function $\boldsymbol{\varphi}(t)$, called scaling function, such that $\{\varphi(t-k), k \ Z\}$ form a basis of V_0 corresponding to the coarsest approximation of x. The reconstruction of x is obtained from:

$$x(t) = \langle x(t), \varphi(t) \rangle . \varphi(t) + \sum_{j=1}^{n} \langle x(t), \psi_{k,j}(t) \rangle . \psi_{k,j}(t) \tag{13}$$

where ⟨⟩ represents the inner product. The approximation coefficients corresponding to the coarsest resolution level are given by $\langle x, \varphi \rangle$ and the details coefficients corresponding to the successively decreasing resolution level $\Delta s_j = \frac{\Delta s_{j-1}}{2}$ are given by $\langle x, \psi_{k,j} \rangle$ as follows:

$$\langle x(t), \psi(t) \rangle = \int x(\tau). \overline{\varphi(\tau - t)} d\tau, \langle x(t), \psi_{k,j}(t) \rangle = \int x(\tau). \overline{\psi_{k,j}(\tau - t)} d\tau \tag{14}$$

2.2.1 Empirical wavelets-EWT

The essence of EMD is that the time domain functions into which a signal is decomposed have the same length as the original signal, allowing time-varying frequencies to be preserved. EMD is described [2, 9] as behaving as a dyadic filter bank as those involved in the multi-resolution analysis. In this context, the mode-mixing phenomenon specific to EMD can be interpreted as the presence of several filters of overlapping frequency content. As a result, the spectral content of some IMFs restituted by the EMD overlap each other. To overcome this problem, Gilles [7] proposed an alternative named the EWT. This method acts in the spectral domain and starts from the segmentation of the original signal's Fourier spectrum. The Fourier support [0,$\boldsymbol{\pi}$] is subdivided into N non-overlapping contiguous segments denoted $\boldsymbol{\Delta}_n = [\boldsymbol{\omega}_{n-1}, \boldsymbol{\omega}_n]$. An appropriate wavelet filter bank is then used to extract spectra relative to each Fourier segment. In the time domain, the components related to the original signal's decomposition are obtained by performing inverse Fourier transform on each Fourier spectrum segments.

The filter bank [15, 16] is defined by the empirical scaling function and the empirical wavelets on each $\boldsymbol{\Delta}_n$ through the following equations:

$$\varphi_n \hat{(\omega)} = \begin{cases} 1 if \lceil \omega \rceil \le (1-\gamma)\omega_n \\ cos\left[\frac{\pi}{2}\beta\left(\frac{1}{2\gamma\omega_n}(|\omega| - (1-\gamma)\omega_n)\right)\right] \\ if (1-\gamma)\omega_n \le |\omega| \le (1+\gamma)\omega_n \\ 0 otherwise \end{cases} \tag{15}$$

and

$$\hat{\psi}_n(\omega) = \begin{cases} 1 if (1+\gamma)\omega_n \le |\omega| \le (1-\gamma)\omega_{n+1} \\ cos\left[\frac{\pi}{2}\beta\left(\frac{1}{2\gamma\omega_{n+1}}(|\omega| - (1-\gamma)\omega_{n+1})\right)\right] \\ if (1-\gamma)\omega_{n+1} \le |\omega| \le (1+\gamma)\omega_{n+1} \\ sin\left[\frac{\pi}{2}\beta\left(\frac{1}{2\gamma\omega_n}(|\omega| - (1-\gamma)\omega_n)\right)\right] \\ if (1-\gamma)\omega_n \le |\omega| \le (1+\gamma)\omega_n \\ 0 otherwise \end{cases} \tag{16}$$

The function **β**(x) is an arbitrary $C^k([0,1])$ function defined as:

$$\beta(x) = \begin{cases} 0 if x \le 0 \\ \beta(x) + \beta(1-x) = 1 x \in [0,1] \\ 1 if x \ge 1 \end{cases} \tag{17}$$

Many functions satisfy this property and the one the most used in the literature [17] is:

$$\beta(x) = x^4\left(35 - 84x + 70x^2 - 20x^3\right) \tag{18}$$

The parameter **γ** is chosen to satisfy the following criterion:

$$\gamma < Min_n\left(\frac{\omega_{n+1} - \omega_n}{\omega_{n+1} + \omega_n}\right) \tag{19}$$

The details and approximation coefficients are calculated by using (Eqs. (15) and (16)) and are respectively given by inner products with the empirical wavelets $\boldsymbol{\psi}_n$ and the scaling function $\boldsymbol{\varphi}_1$:

$$W_x(n,t) = \langle x(t), \psi_n \rangle = IFFT \tag{20}$$

$$W_x(1,t) = \langle x(t), \varphi_1 \rangle = IFFT \tag{21}$$

where X is the Fourier transform of the original signal x,¯represents the complex conjugate, IFFT represents the inverse Fourier transform, and $\boldsymbol{\psi}_n$ and $\boldsymbol{\varphi}_1$ are the results of the inverse Fourier transforms of $\hat{\psi}_n$ and $\hat{\varphi}_1$ respectively. The segmentation of the original signal's Fourier spectrum is obtained from the detection of local maxima. Each segment is centered around one or a group of successive local maxima. The boundary between two contiguous segments is set as the nearest local minimum to the midpoint between two adjacent local maxima groups. Many of the detected local maxima are irrelevant as their contributions to the original signal variability are negligible. Selecting the relevant local maxima requires determining a threshold which is not always possible.

2.2.2 The empirical adaptive wavelet decomposition (EAWD)

The EMD technic enables an observational data sequence to be decomposed into multiple empirical modes of variability, each of them reflecting the observed dynamics at a specific timescale. As the spectral contents of the IMFs returned by the EMD are determined from the relative positions of the original signal's maxima, one of the major drawbacks of the EMD is the mode-mixing phenomenon resulting in a few IMFs to have overlapping spectral supports. On the contrary, the EWT has a more solid theoretical context using wavelets and therefore provides a better frequency resolution. On the other hand, the EWT does not allow to associate the detected frequencies to a specific mode of variability as the EMD technique does. The main idea of the proposed EAWD method is to combine EMD and EWT techniques by setting non-overlapping groups of local maxima from the spectral contents of the IMFs returned by the EMD technique. Each IMF local maximum group will be associated with a segment of the original signal Fourier spectrum segmentation. The boundaries of each of these segments will be set as the local minima the closest to the midpoint between local maximum groups of two consecutive IMFs. As the EMD method acts as a bank of dyadic band-pass filters, the result of each of these filters in the frequency domain is composed of a set of local maxima relative to a specific time scale in which the resolution is divided by 2 in comparison with the timescale immediately above it. Considering that a time scale is characterized by a set of values in the range $[2^n, 2^{n+1}]$, to carry out a segmentation of the Fourier spectrum, it is necessary to distribute the local maxima groups relatively to a grid $[2^i, 2^{i+1}]$, $i \in [2, J]$ with $J = \int\left(\frac{log\,(N)}{log\,(2)}\right) - 1$, where N represents the size of the original time series and int. is the integer part of a number. The proposed Fourier spectrum segmentation algorithm has three main steps: (1) calculation of the spectral content of each IMF based on spectral density and selection of significant local maxima whose energy contribution is >1%. (2) Compute the cutoff frequency between the spectral supports of two consecutive IMFs. (3) EWT technic is run from the Fourier transform obtained from step 2. To get more information on the segmentation algorithm, it is described in detail in Ref. [8].

2.3 Signal decomposition based on singular spectrum analysis

In time series analysis, singular spectrum analysis (SSA) combines elements of classical time series analysis, multivariate statistics and dynamical systems. SSA is used both to decompose time series into components each having a meaningful interpretation and to reconstruct the underlying dynamics from a single time series based on the embedding theorem [18]. In this paragraph, as the reconstruction of the underlying dynamics is beyond the scope of this work, we will describe the SSA technique and its use in signal decomposition.

2.3.1 Singular spectrum decomposition

2.3.1.1 The basic singular spectrum analysis method

As EMD and EWT, SSA is a data-adaptive and non-parametric method for time series decomposition which is suitable to non-stationary time series. The efficiency of such a technic has been recognized for its ability to provide meaningful results in a wide range of application fields, without making any assumption on the processed data. Generally, the components extracted by SSA can be identified as trends, periodic (possibly amplitude-modulated) components or noise components. Furthermore, the characteristics of the components provided by SSA which are to be independent make it particularly suitable for blind source separation. SSA consists of four stages: embedding, decomposition, grouping, and reconstruction [19–21].

Embedding: The starting point of SSA is to embed a time series TS = {x_k, k = [1,N]} of size N in a vector space of dimension L. L is a strictly positive integer named the embedding dimension or window length, 1 < L < N. The embedding procedure forms K=N-L + 1 lagged vectors $X_i = (x_i, \ldots .x_{i+L-1})^T$ with i = 1….N-L + 1. The trajectory matrix of the time series TS is then given by:

$$MT = [X_1, \ldots X_K] \tag{22}$$

MT is a Hankel matrix (constant cross-diagonals) of size (LxK). Hence, the embedding procedure builds a sequence of L-dimensional vectors from the original time series TS, by using a sliding window of size L.

Decomposition: The singular value decomposition (SVD) of the trajectory matrix MT is then computed, providing MT = U.D.V^T with U = (LxL) and V = (KxK) being orthogonal matrices containing respectively the left and right singular vectors. D = (LxK) is a matrix containing the singular values σ_I on the main diagonal and zero elsewhere (where $\sigma_i = \sqrt{K\lambda_i}$) with λ_I being the eigenvalues of C the covariance matrix of MT which can be expressed $C = \frac{1}{K}MT.MT^T$. The matrix MT is thus decomposed in a sum of rank-one matrices MT_i such that:

$$MT = \sum_{i=1}^{L} MT_i = \sum_{i=1}^{L} \sigma_i u_i v_i^T \tag{23}$$

where u_i is i^th^ column of matrix U, v_i is the i^th^ column of matrix V. In other word, the SVD of the trajectory matrix corresponds to an orthogonal matrix whose column vectors form an eigenbasis of the multidimensional space created by the embedding step. Relevant components of the signal can be obtained by projecting the data onto that eigenbasis and omitting noise-related components and finally reconstruct an improved (denoised) version of the original time series.

Grouping step: This step consists of splitting the set of elementary matrices MT_i into r disjoint groups and summing the matrices within each group. The result may be written as:

$$MT = \sum_{k=1}^{r} MT_{I_k} with MT_{I_k} = \sum\nolimits_{i \in I_k} MT_i \tag{24}$$

Thus MT_{I_k} is the resulting matrix of the group I_k, k = 1, …., r.

Diagonal averaging: The averaging along cross-diagonals of the matrix MT_{I_k} aims at solving the problem of finding the time series TS^k for which the trajectory matrix is the closest to MT_{I_k} in a least-squares sense. In other words, the diagonal averaging of MT_{I_k}={$x_{i,j}$, i = 1..L,j = 1… K} provide the elements of a time series $\left\{TS_n^k, n = 1, \ldots N\right\}$ as:

$$TS_n^k = \begin{cases} \frac{1}{n}\sum_{m=1}^{n} x_{m,n-m+1} for 1 \le n < L \\ \frac{1}{L}\sum_{m=1}^{L} x_{m,n-m+1} for L \le n \le K \\ \frac{1}{N-n+1}\sum_{m=n-K+1}^{L} x_{m,n-m+1} for K+1 \le n \le N \end{cases} \tag{25}$$

This cross-averaging process can also be applied to each MT_i matrix. The resulting time series can be assimilated to elementary components. This process finally provides an exact expansion of the original time series TS into r components that satisfies: $TS = \sum_{k=1}^{r} TS_n^k$.

2.3.1.2 Separability and choice of parameters

The very important question is how to choose parameters to build the proper decomposition of the observed time series. We need first to study the concept of separability which characterizes how well different components can be separated from each other. To do so, the so-called w-correlation matrix is computed. This is the matrix consisting of a weighted correlation between the reconstructed time series components. The weight reflects the number of entries of the time series terms into its trajectory matrix. Well separated components have a weak correlation while badly separated components have a large correlation. Therefore, looking at the w-correlation matrix, one can find groups of correlated elementary reconstructed series and use this information for the consequent grouping procedure. MATWCORR, the w-correlation matrix is expressed as follows:

$$MATWCORR_{k_1,k_2} = \frac{(TS_{k_1}, TS_{k_2})}{\|TS_{k_1}\|_w . \|TS_{k_2}\|_w} \tag{26}$$

where

$$\|TS_{k_i}\|_w = \sqrt{(TS_{k_i}, TS_{k_i})_w}, \left(TS_{k_i}, TS_{k_j}\right)_w = \sum_{l=1}^{N} w_l x_l^{k_i} x_l^{k_j} \tag{27}$$

with $w_l = \min(l, L, N-l)$ where $L \leq N/2$.

An important issue is the selection of window length L which is a main parameter to determine the rank of the trajectory matrix. The window length is in the range [2, N/2]. The construction of trajectory matrix is similar to the phase space reconstruction of a nonlinear time series. The most common phase space reconstruction method is the method of delays (MOD). Many techniques have been suggested to estimate the parameters of MOD [22–24], i.e. the time delay τ and the embedding dimension m. The embedding dimension refers to the samples (separated by a fixed time delay τ) from a time window length $L = (m-1)\tau$. The difference between the MOD and the trajectory matrix of SSA is that in MOD the m coordinates are samples separated by a fixed τ which can be greater than 1, while in the standard SSA, all the available samples in window length L are samples separated by a constant $\tau = 1$. For the selection of window length L in SSA, there are some typical methods available, e.g. Kugiumtzis [25] suggested to take the window length L proportional to the mean orbital period" (MP), which can be determined from the spectral density of the original time series:

$$MP = \frac{\sum_{i=1}^{M} DS(f)}{\sum_{i=1}^{M} f . DS(f)} + 1 \tag{28}$$

where M is the number of frequencies f contained in the spectral support of the original time series, DS(f) is the spectral density of the f frequency. Several approaches to determine an optimal window length L are proposed in the literature

[26, 27]. Unfortunately, those proposed methods are all case sensitive and based on the assumption that the measured time series is stationary. So far there does not exist any universal method to find an optimal window length L for an arbitrary time series.

3. Time series analysis and results

Numerical techniques mentioned above as EMD, EAWD, SSA have been applied to three monthly experimental time series of observation. The first time series is 22 years total ozone columns (in Dobson units) from January 1998 recorded in Natal (Brazil). The second time series is 42 years total ozone columns from 1978 recorded in Argentina. The third one is 60 years rainfall time series recorded in Conakry (Guinea).

3.1 Natal monthly total columns ozone time series analysis

The original time series is displayed in **Figure 1**, its size is N = 264:

An EEMD technique has been applied to the time series presented above. After having selected relevant IMFs and add irrelevant ones to the residue, 5 IMFs and the trend has been determined. Results are shown in the **Figure 2**.

The IMF cycle is determined by the dominant frequency of its spectrum. In **Figure 3**, the residual trend R returned by the EEMD is compared with the trend of the original signal obtained from a moving window with a size set at the maximum of the relevant IMF cycles, i.e. 135 months named Tmb.

The accuracy of the Trend returned by the EEMD is estimated by using the following expression:

$$AC_{EEMD} = \sqrt{\frac{\sum_{i=1}^{N}(R(i) - Tmb(i))^2/N}{\overline{Tmb}}} \tag{29}$$

where N is the size of the original time series and $\overline{...}$ represents the mean operator.

No mode-mixing has been detected in the results returned by the EEMD. However, because the EEMD suffers from a lack of theory, the spectral supports of IMFs are not totally disjoint. The EAWD method allows, while relying on the spectral content of the IMFs provided by the EEMD to restitute components whose spectral

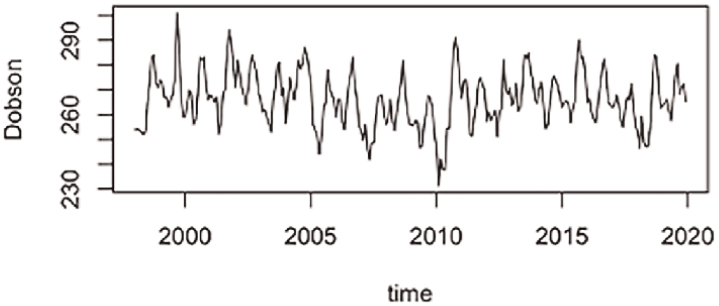
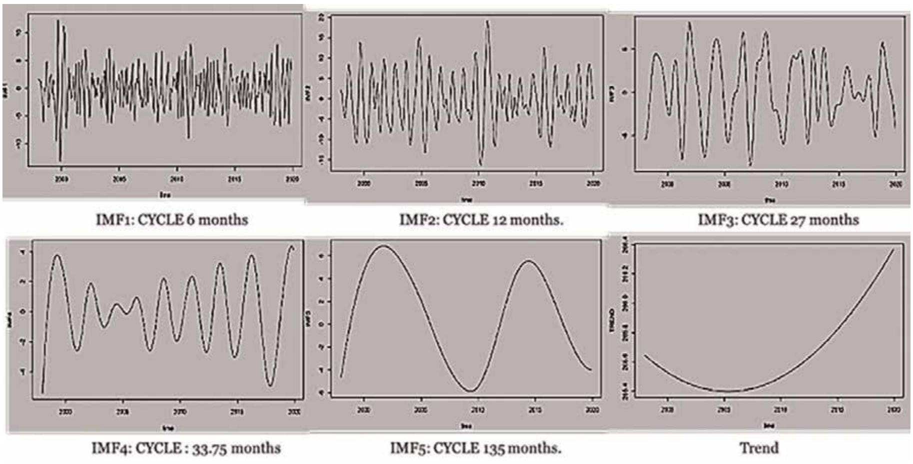


Figure 1.
Natal monthly total ozone columns.



IMF1: CYCLE 6 months
IMF2: CYCLE 12 months.
IMF3: CYCLE 27 months
IMF4: CYCLE : 33.75 months
IMF5: CYCLE 135 months.
Trend

Figure 2.
EEMD decomposition of the Natal total ozone columns time series.

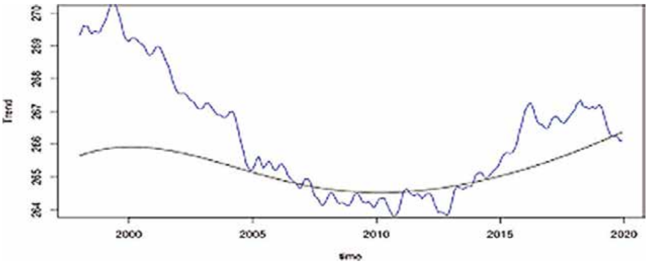
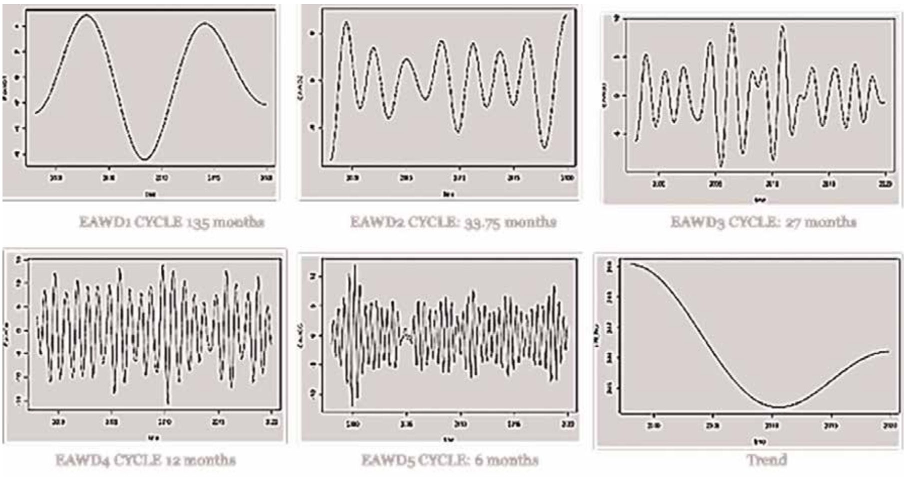


Figure 3.
Trend obtained by the EEMD technique (black curve). The trend of the original signal obtained from a moving window (blue curve) with a 135 months size.



EAWD1 CYCLE 135 months
EAWD2 CYCLE: 33.75 months
EAWD3 CYCLE: 27 months
EAWD4 CYCLE 12 months
EAWD5 CYCLE: 6 months
Trend

Figure 4.
EAWD decomposition of the Natal total ozone columns time series.

contents are disjoint. In this context, the EAWD technique has been applied to the Natal time series and the obtained results are displayed on **Figure 4**.

The accuracy of the trend returned by the EAWD has been estimated by using (Eq. (29)). $AC_{EAWD} = 0.2\%$.

The trend restituted by respectively EEMD and EAWD techniques have been superimposed with the trend obtained from the original signal by using a moving window of size 135 months as shown in **Figure 5**.

The SSA technique has been applied to the Natal total ozone columns. The mean orbit period MP has been estimated by using (Eq. (28)), MP = 11. The embedding window size L has been fixed to L = 110. After having removing noisy components and applying the SSA basic procedure, 37 principal components have been found including trend and relevant periodic components. As the EEMD, EWT, EAWD acts as a bank of dyadic filter, to compare the results provided by SSA with those returned by EEMD, EAWD techniques, cycles of the 37 components extracted from the original signal by SSA are distributed along a grid $\left[2^{\frac{log\,(Cymax)}{log\,(2)}}, 2^2\right]$. Thus, the SSA components whose cycles are in the interval $[2^i, 2^{i+1}]$ are summed to form a component that can be compared to those returned by EEMD, EAWD methods. The components thus obtained are displayed in the **Figure 6**.

Comparison between EAWD and SSA results are displayed on **Figure 7**.

Picture (a), (c) and (d) represents the comparison between SSA and EAWD for respectively the cycles 135 months, 12 months and 6 months. In picture (c) it's the superimposition of summed EAWD components 2 and 3 (cycles 33.75 months and 27 months) with SSA component 4 (cycle 27 months). Picture (c) is interesting from a climate forcing point of view. SSA2 component can be identified to the ENSO climate forcing (45 months) and on the graph of SSA2 (**Figure 6**), we can see that SSA2 energy contribution is very weak. SSA3 component can be identified to the QBO climate forcing (27 months). In **Figure 6(c)**, the red curve is the sum the EAWD components 2 and 3 of 33.75 months and 27 months respectively (**Figure 4**). The difference the two curves in picture (c) is due to the presence of a very weak contribution of ENSO climate forcing in red curve (EAWD results).

3.2 Argentina monthly total columns ozone time series analysis

The original time series is displayed in **Figure 8**, its size is N = 507.

EEMD results are displayed in **Figure 9**.

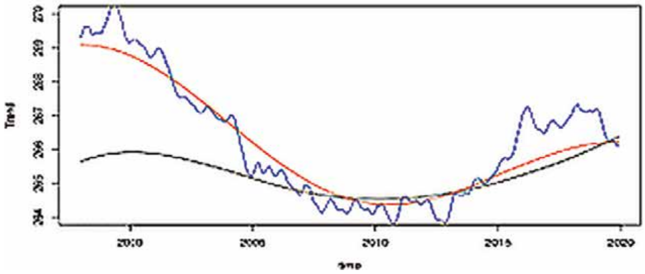
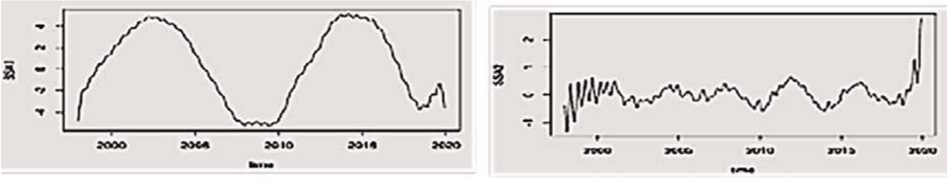
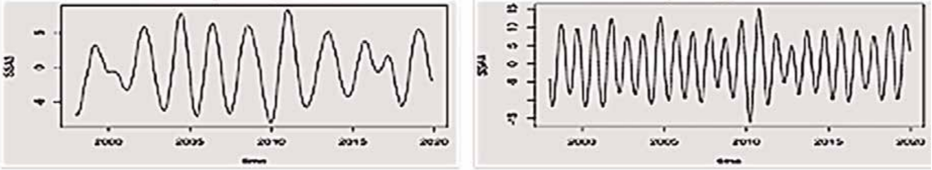


Figure 5.
Superimposition of the EAWD trend (red curve), EEMD trend (black curve) and the trend of the original signal obtained by using a moving window (blue curve).



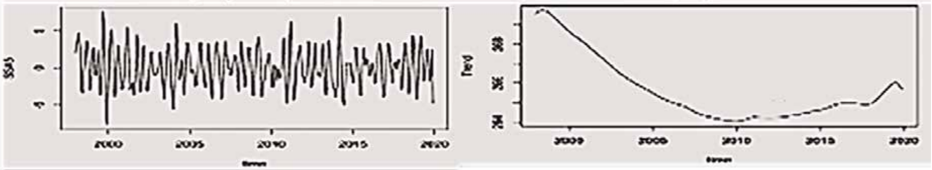
SSA1: Cycle 135 months

SSA2: Cycle 45 months.



SSA3: Cycle 27 months

SSA4: Cycle 12 months.



SSA5: Cycle 6 months

Trend

Figure 6.
SSA results on Natal ozone total columns time series.

Mode-mixings are detected between IMFs 2 and 3 and between IMFs 5 and 6 as their dominant frequency are the same. The accuracy of the Trend returned by the EEMD is estimated to be 0.6%. An EAWD technique has been applied then to overcome mode-mixings encountered when EEMD has been applied. Results are displayed in **Figure 10**.

Mode-mixings occurring in EEMD results have been removed. In EAWD results three new components have been appeared with cycles of respectively 28.5, 21.3 and 6 months. The IMF with cycle 2.4 months returned by EEMD disappeared. The trend restituted by respectively EEMD and EAWD techniques have been superimposed with the trend obtained from the original signal by using a moving window of size 170 months (**Figure 11**).

The accuracy of the trend returned by the EAWD (red curve) has been estimated to be equal to 0.2%. The SSA technique has been applied to the Argentina total ozone columns. The mean orbit period MP has been estimated by using (Eq. (28)), MP = 10. The embedding window size L has been fixed to L = 110. After having removing noisy components and applying the SSA basic procedure 28 principal components have been found including trend and relevant periodic components. SSA principal components have been distributed along a grid $\left[2^{\frac{\log(Cymax)}{\log(2)}}, 2^2\right]$. The components obtained are displayed in **Figure 12**.

Figure 13 displays the superimposition of the EAWD results (red curve) and SSA results (black curve). In pictures (a), (b) (d), (e) and (f) SSA results in black and EAWD results in red are shown for respectively the cycles 6, 12, 64, 170 months and trends. In picture (c) it's the superimposition of summed EAWD components 3 and 4 (cycles 28 months and 21 months) with SSA component 4 (cycle 28 months). Curves in picture (c) can be identified to QBO climate forcing. Curves in pictures (d) and (e) can be respectively identified to ENSO and solar cycle.

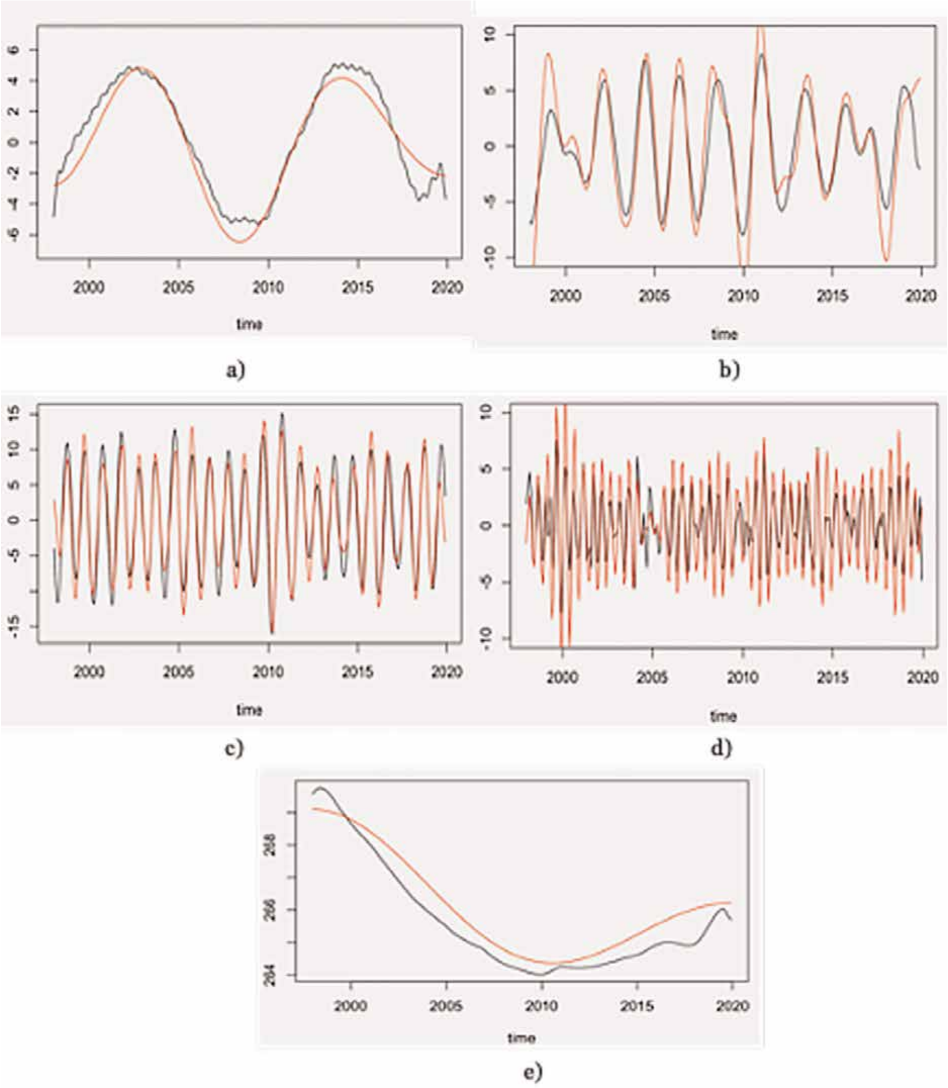


Figure 7.
Comparison between SSA results (black curve) and EAWD results (red curve).

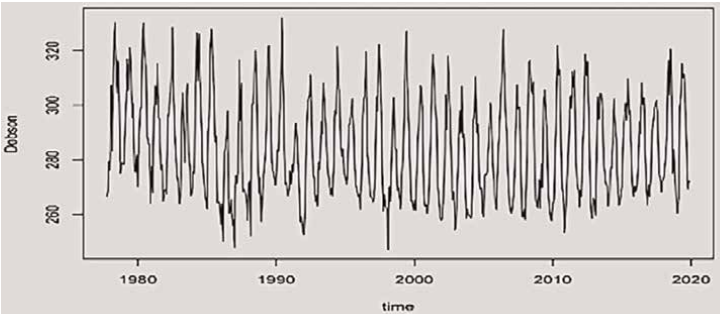
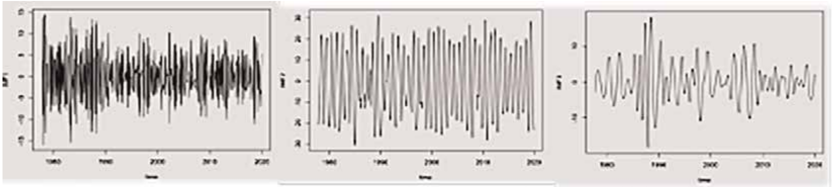
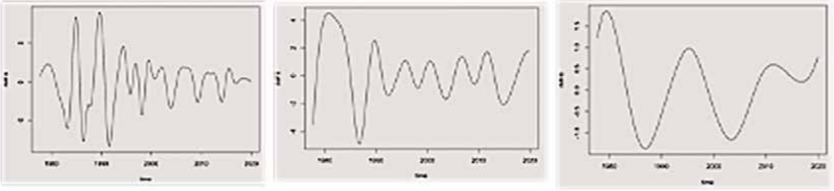


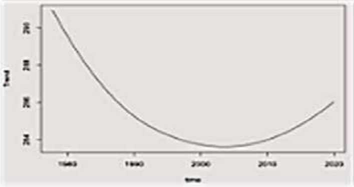
Figure 8.
Argentina monthly total ozone columns.



IMF1 CYCLE : 2.4 months. **IMF2 CYCLE: 12 months.** **IMF3 CYCLE: 12 months**

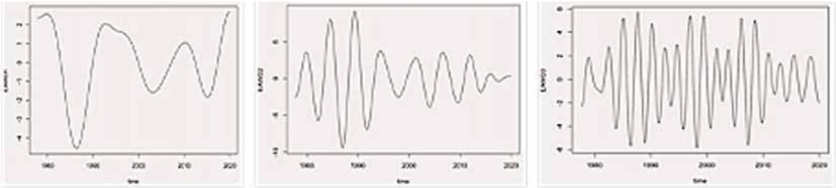


IMF4 CYCLE : 64 months. **IMF5 CYCLE: 170 months.** **IMF6 CYCLE: 170 months**

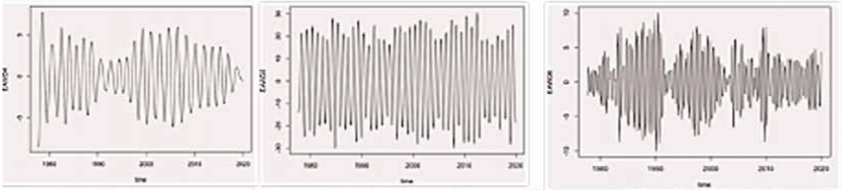


Trend

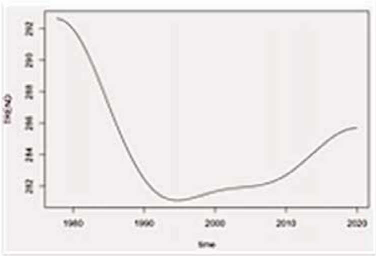
Figure 9.
EEMD results of Argentina total ozone columns.



EAWD1 CYCLE 170 months **EAWD2 CYCLE: 64 months** **EAWD3 CYCLE: 28.5 months**



EAWD4 CYCLE 21.33 months **EAWD5 CYCLE: 12 months** **EAWD6 CYCLE: 6 months**



Trend

Figure 10.
EAWD results of Argentina total ozone columns.

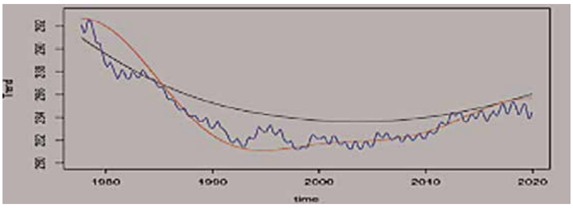
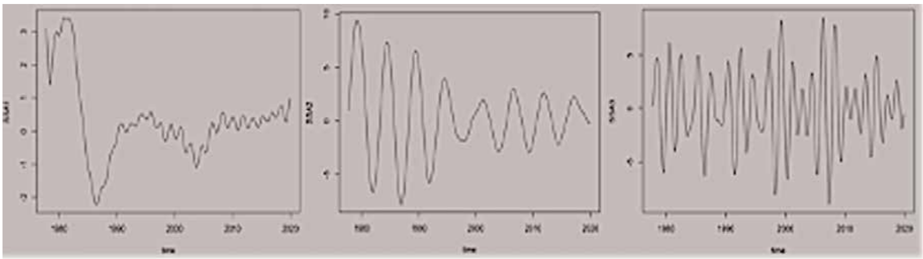
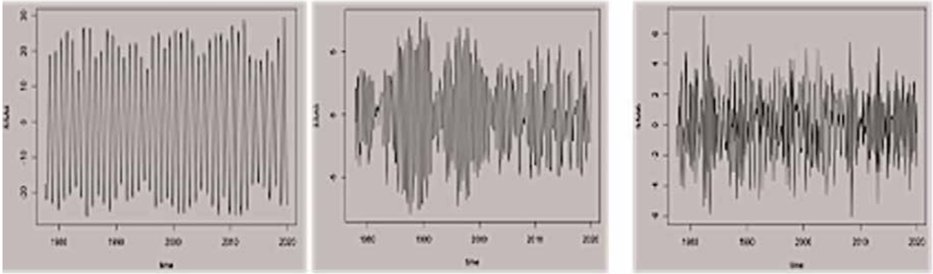


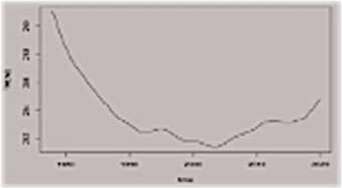
Figure 11.
Superimposition of the EAWD trend (red curve), EEMD trend (black curve) and the trend of the original signal obtained by using a moving window (blue curve).



SSA1: Cycle 170 months **SSA2: Cycle 64 months.** **SSA3 CYCLE: 28 months**



SSA4: Cycle 12 months **SSA5: Cycle 6 months.** **SSA3 CYCLE: 4 months**



Trend

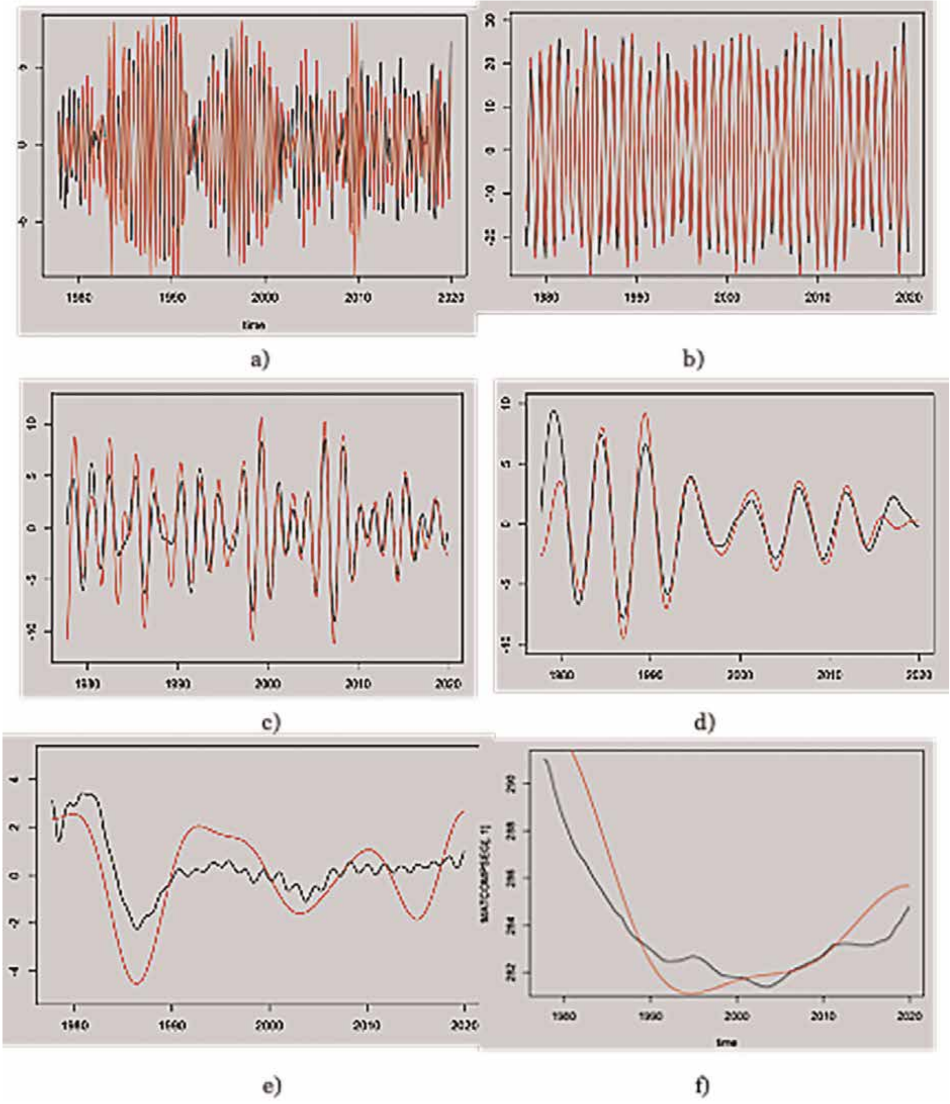
Figure 12.
SSA results on Argentina ozone total columns time series.

3.3 Conakry monthly rainfall time series analysis

The original time series is displayed in **Figure 14**, its size is N = 720:

EEMD results are displayed in **Figure 15**.

Mode-mixings are detected between IMFs 2 and 3 and between IMFs 5 and 6 as their cycles are the same (12 months). An EAWD technique has been applied then to



a)

b)

c)

d)

e)

f)

Figure 13.
Superimposition of SSA results (black curve) and EAWD results (red curve).

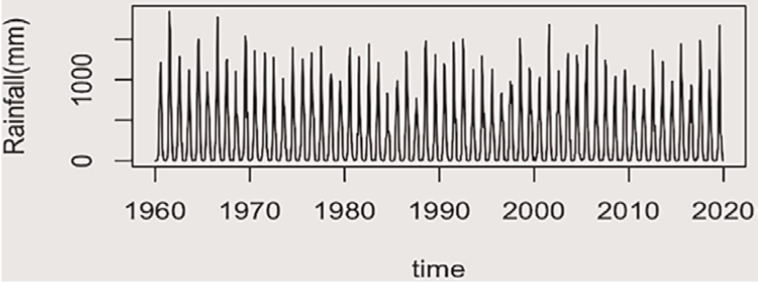
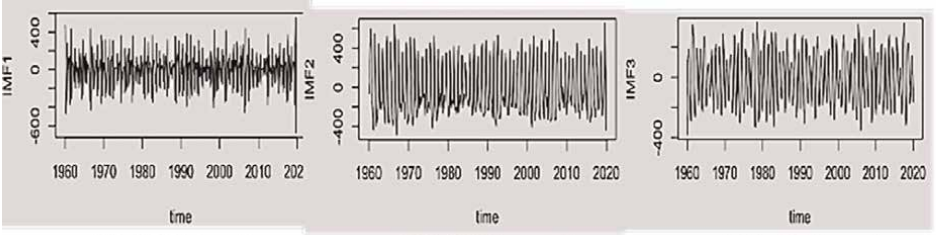
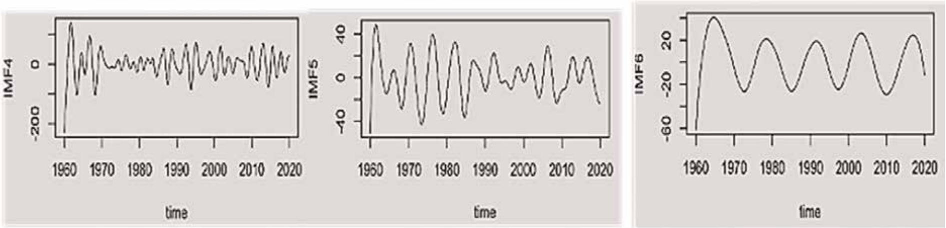


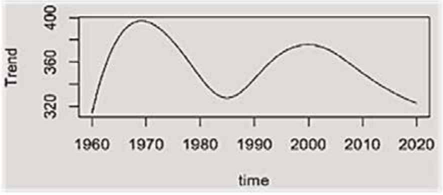
Figure 14.
Conakry monthly rainfall in millimeters.



IMF1 CYCLE: 6 months IMF2 CYCLE: 12 months IMF3 CYCLE: 12 mois



IMF4 CYCLE: 34.3 months IMF5 CYCLE: 72 months IMF6 CYCLE: 144 mois



Trend

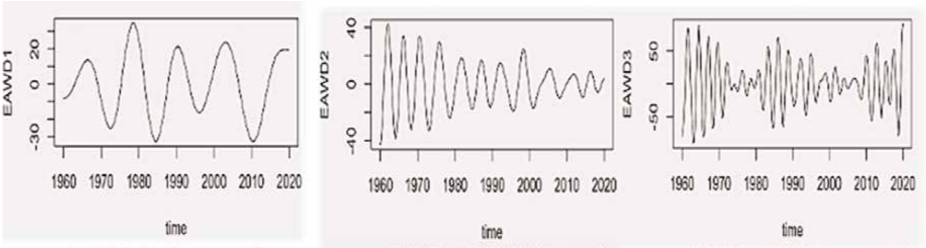
Figure 15.
EEMD results of Conakry rainfall.

overcome mode-mixings encountered when EEMD has been applied. Results are displayed in **Figure 16**.

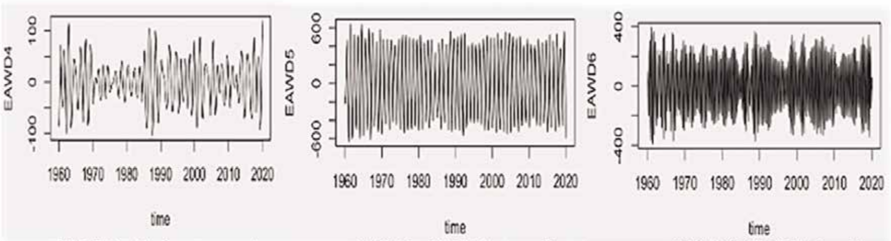
Mode-mixing occurring between IMFs 2 and 3 in EEMD results has been removed. The trend restituted by respectively EEMD and EAWD techniques have been superimposed with the trend obtained from the original signal by using a moving window of size 144 months (**Figure 17**).

The accuracy of the trend returned by the EAWD (red curve) has been estimated to be equal to 0.97%. The SSA technique has been applied to the Conakry rainfall time series. The mean orbit period MP has been estimated by using (Eq. (28)), MP = 9. The embedding window size L has been fixed to L = 108. After having removing noisy components and applying the SSA basic procedure 42 principal components have been found including trend and relevant periodic components. SSA principal components have been distributed along a grid $\left[2^{\frac{\log(Cymax)}{\log(2)}}, 2^2\right]$. The components obtained are displayed in **Figure 18**.

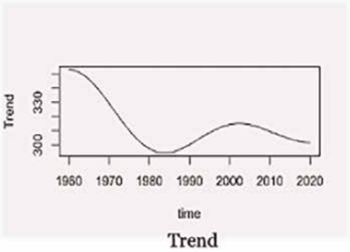
Figure 19 displays the superimposition of the EAWD results (red curve) and SSA results (black curve). In pictures (a), (b), (c), (d) (e), and (f) SSA results in black and EAWD results in red are shown for respectively the cycles 6, 12, 32.7, 65.5, 144 months, and trend. As the segmentation of the original signal spectrum is not identical in EAWD and SSA, a 21 months cycle component appears in the EAWD decomposition and a 9 months cycle component appears in the SSA decomposition.



EAWD1 Cycle: 144 mois **EAWD2 CYCLE: 65.5 mois** **EAWD3 CYCLE: 32.7 mois**



EAWD4 Cycle: 21.2 mois **EAWD5 CYCLE: 12 mois** **EAWD6 CYCLE: 6 mois**



Trend

Figure 16.
EAWD results of Conakry rainfall time series.

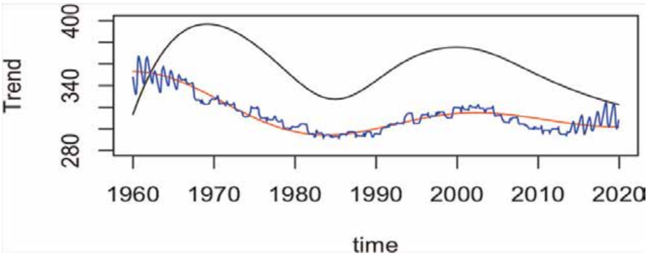


Figure 17.
Superimposition of the EAWD trend (red curve), EEMD trend (black curve) and the trend of the original signal obtained by using a moving window (blue curve).

4. Conclusion

A comparative analysis of signal decomposition methods has been performed. The methods include empirical mode decomposition (EMD), ensemble empirical mode decomposition (EEMD) which can be seen as an optimization of EMD against noise, empirical wavelet transform and empirical adaptive wavelet decomposition which are

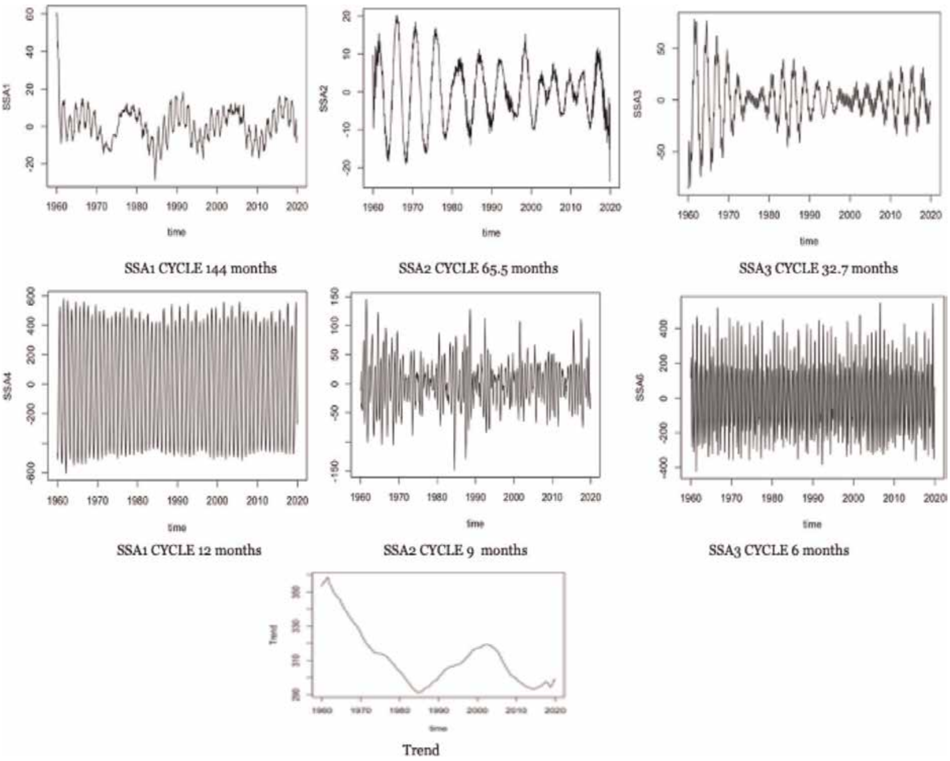
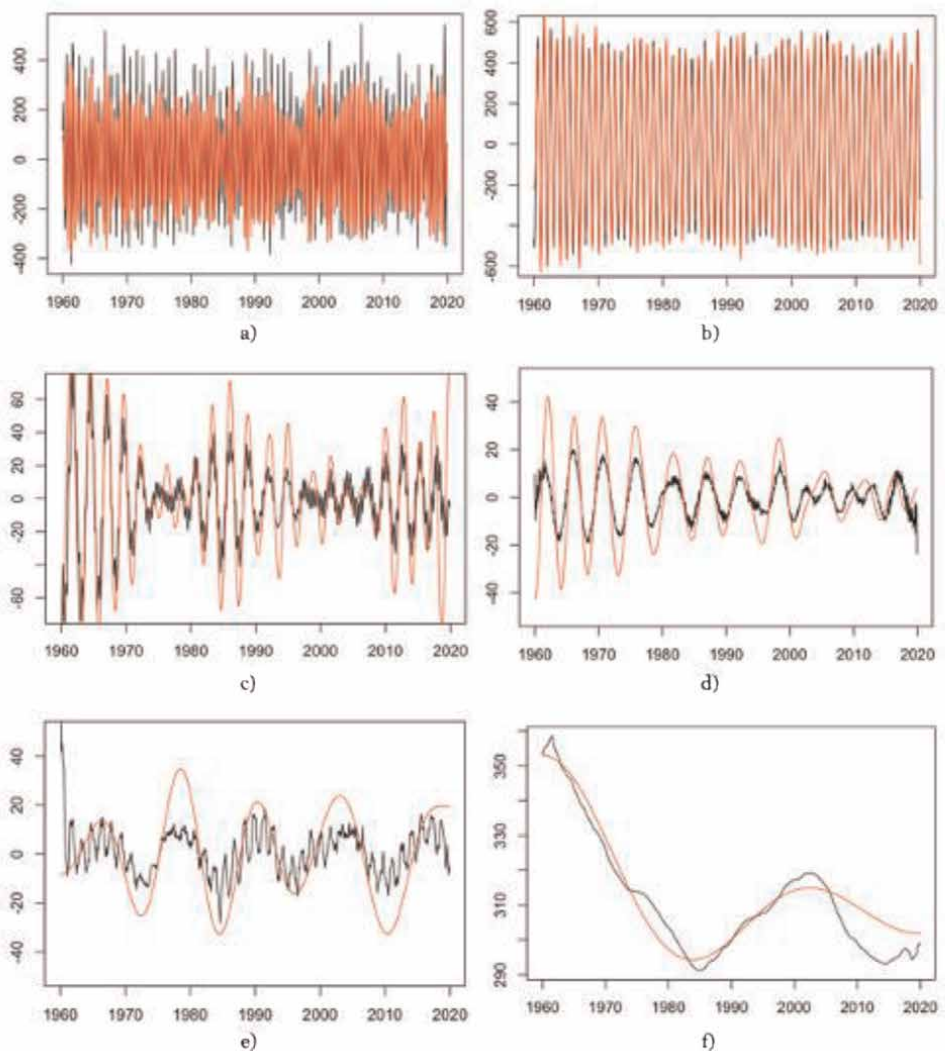


Figure 18.
SSA results on Conakry rainfall time series.

methods that allows the mode-mixing problem to be avoid, and finally the singular spectrum decomposition which acts as a bank of narrow band filters and requires grouping techniques to be compared with results obtained with EAWD. Assuming that the variability components returned by the EMD are close to the physics implicitly contained in the observation time series. The EAWD technique seems to be an advantageous technique as it combines the rigor of wavelets while taking into account the spectral content of the components returned by the EMD. Thus the mode-mixing is avoid and in addition if an EEMD technique is used instead EMD, this technique becomes more robust against noise. Moreover, such a technique does not require a priori specification of the number of decomposed components. However, the SSA technique requires setting parameters like the width of the embedding window L and selecting an appropriate grouping technique to obtain components similar to those returned by the EAWD. SSA technique was found to be sensitive to parameter changes and L determines how well the components are separated. Despite the fact that SSA requires an appropriate embedding window size L, SSA is close to the underlying dynamics to the time series. Moreover, as the components extracted by SSA are independent, such a technique can be very useful for blind source separation purpose.



a) b) c) d) e) f)

Figure 19.
Superimposition of SSA results (black curve) and EAWD results (red curve).

Acknowledgements

This study was led in the scope of the CNRS-LEFE-MANU SOLSTYCE project. Rainfall data were collected and qualified in the framework of the French South-African International Research Group IRP-ARSAIO (International Research Project - Atmospheric Research in Southern Africa and the Indian Ocean) supported by the NRF and CNRS, while the ozone time series were constructed within the French-Brazilian COFECUB-CAPES program.

Author details

Olivier Delage[1]*, Hassan Bencherif[1], Thierry Portafaix[2], Alain Bourdier[3],
René Tato Loua[4] and Damaris Kirsch Pinheiro[5]

1 Laboratoire de l'Atmosphère et des Cyclones, LACy UMR 8105, CNRS, Université de La Réunion, Météo-France, France

2 Research Institute for the Development, Antananarivo, Madagascar

3 University of New Mexico, Albuquerque, USA

4 National Agency of Meteorology of Guinea (ANM), Guinea

5 Federal University of Santa Maria (UFSM), Santa Maria, Brazil

*Address all correspondence to: olivier.delage@univ-reunion.fr

References

[1] Huang NE, Shen Z, Long SR, Wu MC, Shih HH, Zheng Q, et al. The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time-series analysis. Proceedings of the Royal Society of London A: Math., Physical and Engineering Sciences. 1998;**454**(1971): 903-995. DOI: 10.1098/rspa.1998.0193

[2] Flandrin P, Rilling G, Gonçalvés P. Empirical mode decomposition as a filter bank. IEEE Signal Processing Letters. 2004;**11**(2):112-114. DOI: 10.1109/LSP.2003.821662

[3] Xu B, Sheng Y, Li P, Cheng Q, Wu J. Causes and classification of EMD mode-mixing. Microengineering Procedia. 2019;**22**:158-164. DOI: 10.21595/vp.2018.20250

[4] Gao Y, Ge G, Sheng Z, Sang E. Analysis and solution to the mode mixing phenomenon in EMD. In: 2008 Congress on Image and Signal Processing, Sanya, China. Shangai, China: IEEE; 2008. pp. 223-227. DOI: 10.1109/CISP.2008.193

[5] Fosso OB, Molinas M. Method for mode mixing separation in empirical mode decomposition. September 2017. arXiv: 1709.05547v1 [stat. ME]

[6] Delage O, Portafaix T, Bencheriff H, Guimbretière G, Loua RT. Multi-scale variability analysis of time series in geophysics by using the empirical mode decomposition. In: Proceedings SAGA. Durban, South Africa: HAL; October 2019. Available from: https://hal.archives-ouvertes.fr/hal-02363170

[7] Gilles J. Empirical wavelet transform. IEEE Transactions on Signal Processing. 2013;**61**:3999-4010. DOI: 10.1109/TSP.2013.2265222

[8] Delage O, Portafaix T, Bencherif H, Bourdier A, Lagracie E. Empirical adaptive wavelet decomposition (EAWD): An adaptive decomposition for the variability analysis of observation time series in atmospheric science. Non-linear process in Geophysics. 2022;**29**(3): 265-277. DOI: 10.5194/npg-29-265-2022

[9] Rehman N, Mandic DP. Filter bank property of multivariate empirical mode decomposition. IEEE Transactions on Signal Processing. 2011;**55**(5):2421-2426. DOI: 10.1109/TSP.2011.2106779

[10] Mert A, Akan A. Detrended fluctuations analysis for empirical mode decomposition based denoising. In: 2014 22nd European Signal Processing Conference (EUSIPCO). Lisbon, Portugal: IEEE; November 2014. ISBN: 978-0-9928-6661-9

[11] Gonzales JS et al. Analyzing chaos systems and fine spectrum sensing using detrended fluctuations analysis algorithm, Hindawi publishing corporation. Mathematical Problems in Engineering. 2016;**2016**:1-18. DOI: 10.1155/2016/2865195

[12] Liu S, Ma R, Cong R, Wang H, Zhao H. A new approach for embedding dimension determination based on empirical mode decomposition. Kybernetes. 2012;**41**(9):1176-1184. DOI: 10.1108/03684921211275180

[13] Souza DB, Chanussot J, Favre AC. On selecting relevant intrinsic mode functions in empirical mode decomposition: An energy approach. In: 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). Florence, Italy: IEEE; July 2014. DOI: 10.1109/ICASSP.2014.6853611

[14] Wu Z, Huang NE. Ensemble empirical mode decomposition: A noise assisted data analysis method. Advances in Adaptive Data Analysis. 2009;**1**:1-41. DOI: 10.1142/SI1793536909000047

[15] Jaffard S, Meyer Y, Ryan RD. Wavelets: Tools for Science and Technology. SIAM; 2001. pp. 1-256. ISBN: 0898714486

[16] Meyer Y. Wavelets: Vibrations and Scaling. American Mathematical Society; 1997. ISBN: 9780821806852

[17] Daubechies I. Ten lectures on wavelets. In: CBMS-NSF Conference, Society for Industrial and Applied Mathematics. SIAM; 1992. ISBN: 978-0-89871-274-2

[18] Takens F. Dynamical systems and turbulence. In: Rand D, Young LS, editors. Lecture Notes in Mathematics. Vol. 898. New York: Scientific Research Publishing; 1981. pp. 366-381

[19] Golyandina N, Korobeynikov A. Basic singular spectrum analysis and forecasting with R. Computational Statistics and Data Analysis. 2014; **71**:934-954. DOI: 10.1016/j.csda.2013.04.009

[20] Bonizzi P, Bonizzi J, Karel MH. Singlar spectrum decomposition: A new method for time series decomposition. Advances in Adaptive Data Analysis. 2014;**6**(4):1-34. DOI: 10.1142/SI793536914500113

[21] Harmouche J, Fourer D, Auger F, Borgnat P, Flandrin P. The sliding singular spectrum analysis: A data-driven non-stationary signal decomposition tool. IEEE Transactions on Signal Processing. 2017;**66**(1): 251-263. DOI: 10.1109/TSP.2017.2752720

[22] Tan E, Algar S, Correa D, Small M, Stemler T, Walker D. Selecting embedding delays: An overview of embedding techniques and a new method using persistent homology. Chaos: An Interdisciplinary Journal of Nonlinear Science. 2023;**33**:1-28. DOI: 10.1063/5.0137223

[23] Cao L. Practical method for determining the minimum embedding dimension of a scalar times series. Physica D: Nonlinear Phenomena. 1997; **110**:43-50. DOI: 10.1016/SO167-2789(97) 00118-8

[24] Delage O, Bourdier A. Selection of optimal embedding parameters applied to short and noisy time series from Rössler system. Journal of Modern Physics. 2017;**8**:1607-1632. DOI: 10.4236/jmp.2017.89096

[25] Kugiumtzis D. State space reconstruction parameters in the analysis of chaotic time series-the role of the time window length. Physica D: Nonlinear Phenomena. 1996;**95**(1):13-28. DOI: 10.1016/0167-2789(96)00054-1

[26] Kim HS, Eykhold R, Salas JD. Non linear dynamics delay times, and embedding windows. Physica D. 1999; **127**:48-60

[27] Ma HG, Han CZ. Selection of embedding dimension and delay time in phase space reconstruction. Frontiers of Electrical and Electronic Engineering in China. 2006;**1**:111-114

Chapter 7

Segmentation of High-Dimensional Matrix-Variate Time Series

Zhaoxing Gao

Abstract

In this chapter, we introduce a new segmentation method for high-dimensional matrix-variate time series. Specifically, we look for linear transformations to segment the matrix into many small sub-matrices for which each of them is uncorrelated with the others both contemporaneously and serially, thus they can be analyzed separately, which will greatly reduce the number of parameters to be estimated in terms of modeling. To overcome the identification issue, we propose a two-step and more structured procedure to segment the rows and columns separately. When the dimension is large in relation to the sample size, we assume the transformation matrices are sparse and use threshold estimators for the (auto) covariance matrices. Unlike principal component analysis (PCA) for independent data, we cannot guarantee that the required linear transformation exists. When it does not, the proposed method provides an approximate segmentation, which may be useful for forecasting. The proposed method is illustrated with simulated data examples.

Keywords: high dimension, matrix-variate time series, segmentation, dimension reduction, eigen-analysis

1. Introduction

Modern scientific studies often gather data under combinations of multiple factors. For example, neuroimaging experiments record brain activity at multiple spatial locations, at multiple time points, and under a variety of experimental stimuli. Studies of social networks record social links of a variety of types from multiple initiators of social activity to multiple receivers of the activity. Data such as these are naturally represented not as lists or tables of numbers, but as multi-indexed arrays or tensors. As many types of such data are collected over time, it is natural to view them as tensor-valued time series. The matrix-valued time series is a sequence of second-order random tensors. For example, financial and economic studies often collect data from different countries with a number of economic indicators (e.g. GDP growth, unemployment rate, etc.) every quarter. Therefore, it is important and interesting to develop appropriate statistical methods to analyze such types of data. The most common approach to modeling such data is to stack the matrix into a large vector, and then apply the standard multivariate methods. However, such an approach will ignore the matrix structure of the data, which can lead to inefficient use of data and important patterns in the data being overlooked. For example, [1] pointed out that after vectorizing the matrices the resulting vectors have a Kronecker structure. Ignoring this structure means that a much larger number of parameters need to be estimated. Therefore, it is urgent to find an effective way to reduce the number of parameters, especially when the dimension is large.

When the data are independent and identically distributed (i.i.d.), [2] introduced dimension folding sufficient reduction for conditional mean functions, [3] proposed a dimension folding method for data with matrix-valued predictors, [4–6] extended the generalized linear models to the matrix- and tensor-valued predictors for analyzing image data. Ding and Dennis [7] studied the matrix-variate regression with a matrix-variate response. An incomplete list of publications also includes [8–12]. With temporal dependence, the matrix-valued time series has not been well studied in the literature; [13] handled this kind of data in signal and image processing, and [14, 15] proposed two versions of factor models for matrix-valued time series, which maintain and utilize the matrix structure to achieve the dimension reduction.

In this chapter, we extend the PCA approach of [16] to matrix-variate time series without stacking the matrix into a vector, and the structure can be preserved. Our goal is as follows: Let $\mathbf{Y}_t = \left(y_{ij}^t\right)$ be a $p \times q$ matrix-variate time series, that is. there are pq recorded values at each time, for example, p individuals and over q indices or variables. We assume $\mathbf{Y}_t$ can be represented as

$$\mathbf{Y}_t = \mathbf{B}\mathbf{W}_t\mathbf{A}^{\mathrm{T}}, \tag{1}$$

where $\mathbf{B} \in R^{p\times p}$, $\mathbf{A} \in R^{q\times q}$, and $\mathbf{W}_t$ is a latent $p \times q$ matrix in which the rows are divided into $p_1(\le p)$ groups, and there are no correlations across different groups at all time lags, and the columns are divided into $q_1(\le q)$ groups, and there are no correlations across different groups at all time lags either. With such a decomposition, we only need to model the small sub-matrices in $\mathbf{W}$ separately, and we can achieve substantial dimension reduction. As $\mathbf{B}$, $\mathbf{W}_t$, and $\mathbf{A}$ are all latent ones and the identification is a big issue. For example, even when $\mathbf{W}_t$ is observable, $(\mathbf{A}, \mathbf{B})$ can be replaced by $(\mathbf{A}/c, c\mathbf{B})$ for any nonzero constant c without changing the relationship of (1).

Instead of estimating them simultaneously, we propose in this paper a two-step and more structured approach; first, we seek a column transformation, that is. We transform linearly the columns of $\mathbf{Y}_t$ into q new variables, and ideally, those q new variables form q_1 uncorrelated groups with $q_1 \le q$. The second step applies the same segmentation method to the p rows of the obtained ones in the first step, and the transformation of the rows will not alter the uncorrelatedness of the column groups in the first step. In the end, this new matrix can be divided into several smaller sub-matrices, and those sub-matrices are uncorrelated with each other both contemporaneously and serially. Our method is a building block for modeling tensor-valued time series, and it turns out that all tensor-valued time series can be rearranged as a matrix time series by matricization, see [17].

2. Methodology

2.1 Setting and method

Let $\mathbf{Y}_t = \left(\mathbf{y}_1^t, \dots, \mathbf{y}_q^t\right)$ be an observable $p \times q$ matrix-valued time series with $\mathbf{y}_i^t \in R^p$. We assume $\mathbf{Y}_t$ admits a latent segmentation structure:

$$\mathbf{Y}_t = \mathbf{X}_t \mathbf{A}^{\mathrm{T}}, \tag{2}$$

where $\mathbf{X}_t$ is an unobservable $p \times q$ matrix-valued time series in which the q columns can be classified into q_1 (>1) groups, and any two groups are contemporaneously and serially uncorrelated, and $\mathbf{A} \in R^{q \times q}$ is an unknown constant matrix. Before we proceed further, we give the definitions of row- and column-covariance matrix between two random matrices.

Definition 1 Let $\mathbf{U}_t \in R^{s_1 \times r_1}$ and $\mathbf{V}_t \in R^{s_2 \times r_2}$. If $r_1 = r_2 = r$, the covariance matrix over the columns between $\mathbf{U}_t$ and $\mathbf{V}_t$ is defined as

$$\mathrm{Cov}_c(\mathbf{U}_t, \mathbf{V}_t) := \frac{1}{r} E(\mathbf{U}_t - E\mathbf{U}_t)(\mathbf{V}_t - E\mathbf{V}_t)^{\mathrm{T}}, \tag{3}$$

and if $s_1 = s_2 = s$, the covariance matrix over the rows is defined as

$$\mathrm{Cov}_r(\mathbf{U}_t, \mathbf{V}_t) := \frac{1}{s} E(\mathbf{U}_t - E\mathbf{U}_t)^{\mathrm{T}}(\mathbf{V}_t - E\mathbf{V}_t) = \mathrm{Cov}_c\left(\mathbf{U}_t^{\mathrm{T}}, \mathbf{V}_t^{\mathrm{T}}\right). \tag{4}$$

The variances $\mathrm{Var}_c(\mathbf{U}_t)$ and $\mathrm{Var}_r(\mathbf{U}_t)$ can be defined in a similar way. In particular, when $r = 1$ or $s = 1$, (3) or (4) reduces to the traditional case for two random vectors.

In model (2), we assume $\mathbf{Y}_t$ and $\mathbf{X}_t$ are both weakly stationary in the sense that the means and the autocovariances do not vary with respect to time for any fixed (p, q). The stationarity of $\mathbf{Y}_t$ can be inherited from $\mathbf{X}_t$ through (2), and a sufficient condition for this is to assume $\mathrm{Vec}(\mathbf{X}_t)$ and $\mathrm{Vec}(\mathbf{Y}_t)$ are stationary, where $\mathrm{Vec}(\cdot)$ is the vectorization of a matrix. Denote the segmentation of $\mathbf{X}_t$ by

$$\mathbf{X}_t = \left(\mathbf{x}_1^t, \ldots, \mathbf{x}_q^t\right) = \left(\mathbf{X}_t^{(1)}, \ldots, \mathbf{X}_t^{(q_1)}\right) \tag{5}$$

with $\mathrm{Cov}_r\left(\mathbf{X}_t^{(i)}, \mathbf{X}_s^{(j)}\right) = 0$ for all t, s and $i \neq j$. Therefore, all the autocovariances of $\mathbf{X}_t^{\mathrm{T}}$ are of the same block-diagonal structure with q_1 blocks, and $\mathbf{X}_t^{(1)}, \ldots, \mathbf{X}_t^{(q_1)}$ can be modeled or forecasted separately as far as their linear dynamic structure is concerned.

Now, we spell out how to find the segmentation transformation under (2) and (5). Without loss of generality, we assume

$$\mathrm{Var}_r(\mathbf{Y}_t) = \mathbf{I}_q \quad \text{and} \quad \mathrm{Var}_r(\mathbf{X}_t) = \mathbf{I}_q. \tag{6}$$

The first equation in (6) is implied by replacing $\mathbf{Y}_t$ by $\mathbf{Y}_t \widehat{\mathbf{S}}_{y,0}^{-1/2}$, where $\widehat{\mathbf{S}}_{y,0}$ is a consistent estimator of $\mathrm{Var}_r(\mathbf{Y}_t)$. The second equation is to conceptually replace $\mathbf{X}_t$ by $\mathbf{X}_t \widehat{\mathbf{S}}_{x,0}^{-1/2}$, where $\widehat{\mathbf{S}}_{x,0}^{-1/2}$ is a consistent estimator for $\mathrm{Var}_r(\mathbf{X}_t)$, and it will not alter the fact that there are no correlations across different groups. As both $\mathbf{A}$ and $\mathbf{X}_t$ are unobservable, (6) implies that we can view $\widehat{\mathbf{S}}_{y,0}^{-1/2} \mathbf{A} \widehat{\mathbf{S}}_{x,0}^{1/2}$ as $\mathbf{A}$. As a consequence of (6), the transformation matrix $\mathbf{A}$ in (2) is orthogonal. Let l_j be the number of columns of $\mathbf{X}_t^{(j)}$ with $l_1 + \cdots + l_{q_1} = q$. Write $\mathbf{A} = \left(\mathbf{A}_1, \ldots, \mathbf{A}_{q_1}\right)$, where $\mathbf{A}_j \in R^{q \times l_j}$. It follows from (2) and (5) that

$$\mathbf{X}_t^{(j)} = \mathbf{Y}_t \mathbf{A}_j, \quad j = 1, \ldots, q_1. \tag{7}$$

However, similar to that in [16], $\mathbf{A}$ and $\mathbf{X}_t$ are not uniquely identified in (2), even with additional assumption in (6). For example, let $\mathbf{H}_j$ be any $l_j \times l_j$ orthogonal matrix and $\mathbf{H} = \mathrm{diag}\left(\mathbf{H}_1, \dots, \mathbf{H}_{q_1}\right)$. Then $(\mathbf{A}, \mathbf{X}_t)$ in (2) can be replaced by $(\mathbf{AH}, \mathbf{X}_t\mathbf{H})$, while (5) still holds. In fact, only $\mathcal{M}(\mathbf{A}_1), \dots, \mathcal{M}\left(\mathbf{A}_{q_1}\right)$ are uniquely defined by (2), where $\mathcal{M}\left(\mathbf{A}_j\right)$ denotes the linear space spanned by the columns of $\mathbf{A}_j$. As a result, $\mathbf{Y}_t\Gamma_j$ can be taken as $\mathbf{X}_t^{(j)}$ for any $q \times l_j$ matrix Γ_j as long as $\Gamma_j^{\mathrm{T}}\Gamma_j = \mathbf{I}_{l_j}$ and $\mathcal{M}\left(\Gamma_j\right) = \mathcal{M}\left(\mathbf{A}_j\right)$. Thus, to estimate $\mathbf{A} = \left(\mathbf{A}_1, \dots, \mathbf{A}_{q_1}\right)$, it is sufficient to estimate the linear spaces $\mathcal{M}(\mathbf{A}_1), \dots, \mathcal{M}\left(\mathbf{A}_{q_1}\right)$.

To discover the latent segmentation, we introduce some notation first. We denote $\mathbf{y}_{i:}^t$ and $\mathbf{x}_{i:}^t$ the row vectors of $\mathbf{Y}_t$ and $\mathbf{X}_t$, respectively. For any integer k, let $\Sigma_y(k) = \mathrm{Cov}_r(\mathbf{Y}_{t+k}, \mathbf{Y}_t)$, $\Sigma_x(k) = \mathrm{Cov}_r(\mathbf{X}_{t+k}, \mathbf{X}_t)$, $\Sigma_{y,i,j}(k) = \mathrm{Cov}\left(\mathbf{y}_{i:}^{t+k}, \mathbf{y}_{j:}^t\right)$, and $\Sigma_{x,i,j}(k) = \mathrm{Cov}\left(\mathbf{x}_{i:}^{t+k}, \mathbf{x}_{j:}^t\right)$. By (6), we have $\Sigma_y(0) = \Sigma_x(0) = \mathbf{I}_q$. For a pre-specified integer k_0, define

$$\mathbf{W}_y = \sum_{k=0}^{k_0} \Sigma_y(k)\Sigma_y(k)^{\mathrm{T}} = \mathbf{I}_q + \sum_{k=1}^{k_0} \Sigma_y(k)\Sigma_y(k)^{\mathrm{T}} \tag{8}$$

and

$$\mathbf{W}_x = \sum_{k=0}^{k_0} \Sigma_x(k)\Sigma_x(k)^{\mathrm{T}} = \mathbf{I}_q + \sum_{k=1}^{k_0} \Sigma_x(k)\Sigma_x(k)^{\mathrm{T}}. \tag{9}$$

It follows from (2) and (5) that both $\Sigma_x(k)$ and $\mathbf{W}_x$ are block-diagonal and

$$\mathbf{W}_y = \mathbf{A}\mathbf{W}_x\mathbf{A}^{\mathrm{T}}. \tag{10}$$

Note that both $\mathbf{W}_y$ and $\mathbf{W}_x$ are positive definite matrices; therefore, we have the following decomposition

$$\mathbf{W}_x\Gamma_x = \Gamma_x\mathbf{D}, \tag{11}$$

where Γ_x is a $q \times q$ orthogonal matrix with the columns being the orthonormal eigenvectors of $\mathbf{W}_x$, and $\mathbf{D}$ is a diagonal matrix with the corresponding eigenvalues as the elements on the main diagonal. By (10) and (11), $\mathbf{W}_y\mathbf{A}\Gamma_x = \mathbf{A}\Gamma_x\mathbf{D}$ and hence the columns of $\Gamma_y := \mathbf{A}\Gamma_x$ are the orthonormal eigenvectors of $\mathbf{W}_y$. Consequently,

$$\mathbf{Y}_t\Gamma_y = \mathbf{Y}_t\mathbf{A}\Gamma_x = \mathbf{X}_t\Gamma_x, \tag{12}$$

where the last equality follows from (2). Let

$$\mathbf{W}_x = \mathrm{diag}\left(\mathbf{W}_{x,1}, \dots, \mathbf{W}_{x,q_1}\right), \tag{13}$$

where $\mathbf{W}_{x,j}$ is an $l_j \times l_j$ positive definite matrix, and the eigenvalues of $\mathbf{W}_{x,j}$ are also the eigenvalues of $\mathbf{W}_x$. Suppose that $\mathbf{W}_{x,i}$ and $\mathbf{W}_{x,j}$ do not share the same eigenvalues for any $i \neq j$. Then if we line up the eigenvalues of $\mathbf{W}_x$ (i.e. the eigenvalues of $\mathbf{W}_{x,1}, \dots, \mathbf{W}_{x,q_1}$ combining together) in the main diagonal of $\mathbf{D}$ according to the order of the blocks in $\mathbf{W}_x$, Γ_x must be a block-diagonal orthogonal matrix of the same shape as $\mathbf{W}_x$; see Proposition 1 (i). However, the order of the eigenvalues is latent, and any Γ_x defined by (11) is nevertheless a column-permutation (i.e. a matrix consisting of the same column vectors but arranged in a different order) of such a block-diagonal orthogonal matrix; see Proposition 1(ii). By Proposition 1(i), write $\Gamma_x = diag\left(\Gamma_{x,1}, \ldots, \Gamma_{x,q_1}\right)$, it follows from (5) and (12) that

$$\mathbf{Y}_t\Gamma_y = \mathbf{X}_t\Gamma_x = \left(\mathbf{X}_t^{(1)}\Gamma_{x,1}, \ldots, \mathbf{X}_t^{(q_1)}\Gamma_{x,q_1}\right), \tag{14}$$

Hence, $\mathbf{X}_t\Gamma_x$ does not alter the fact that there are no correlations between different groups in $\mathbf{X}_t$, and Γ_y can be regarded as $\mathbf{A}$ so long as the eigenvalues of $\mathbf{W}_x$ are ordered appropriately. As they are all latent, $\mathbf{Y}_t\Gamma_y$ can be taken as a permutation of $\mathbf{X}_t$, and Γ_y can be viewed as a column-permutation of $\mathbf{A}$; see the discussion below (7). This leads to the following three-step estimation for $\mathbf{A}$ and $\mathbf{X}_t$:

Step 1. *Let* $\widehat{\mathbf{S}}_{y,0}$ *be a consistent estimator for* $\mathrm{Var}_r(\mathbf{Y}_t)$. *Replace* $\mathbf{Y}_t$ *by* $\mathbf{Y}_t\widehat{\mathbf{S}}_{y,0}^{-1/2}$.

Step 2. *Let* $\widehat{\mathbf{S}}$ *be a consistent estimator for* $\mathbf{W}_y$. *Calculate an* $q \times q$ *orthogonal matrix* $\widehat{\Gamma}_y$ *with columns being the orthonormal eigenvectors of* $\widehat{\mathbf{S}}$.

Step 3. *The columns of* $\widehat{\mathbf{A}} = \left(\widehat{\mathbf{A}}_1, \ldots, \widehat{\mathbf{A}}_{q_1}\right)$ *are a permutation of the columns of* $\widehat{\Gamma}_y$ *such that* $\widehat{\mathbf{X}}_t = \mathbf{Y}_t\widehat{\mathbf{A}}$ *is segmented into* q_1 *uncorrelated sub-matrix series* $\widehat{\mathbf{X}}_t^{(j)} = \mathbf{Y}_t\widehat{\mathbf{A}}_j$, $j = 1, \ldots, q_1$.

In Steps 1 and 2, the estimators $\widehat{\mathbf{S}}_{y,0}$ and $\widehat{\mathbf{S}}$ should be consistent and will be constructed under various scenarios in Section 3 below. The permutation in Step 3 can be carried out by grouping the columns of $\widehat{\mathbf{Z}}_t := \mathbf{Y}_t\widehat{\Gamma}_y$.

We now state a proposition that demonstrates the assertion after (13); the proof is similar to Proposition 1 in [16], and we therefore omit it.

Proposition 1: (i) The orthogonal matrix Γ_x in (11) can be taken as a block-diagonal orthogonal matrix with the same block structure as $\mathbf{W}_x$. (ii) An orthogonal matrix Γ_x satisfied (11) if and only if its columns are a permutation of the columns of a block-diagonal orthogonal matrix described in (i), provided that any two different blocks $\mathbf{W}_{x,i}$ and $\mathbf{W}_{x,j}$ do not share the same eigenvalues.

From Proposition 1, we can see that the proposed method will not be able to separate $\mathbf{X}_t^{(i)}$ and $\mathbf{X}_t^{(j)}$ if $\mathbf{W}_{x,i}$ and $\mathbf{W}_{x,j}$ share one or more common eigenvalues. But, it does not rule out the possibility that each block $\mathbf{W}_{x,j}$ may have multiple eigenvalues.

2.2 Permutation

2.2.1 Permutation rule

According to the discussion in Section 2.1, $\widehat{\mathbf{A}}$ is a permutation of the columns of $\widehat{\Gamma}_y$, and the permutation can be carried out by grouping the columns of $\widehat{\mathbf{Z}}_t := \mathbf{Y}_t\widehat{\Gamma}_y$ into q_1 groups, where q_1 and the number of columns l_j $(1 \le j \le q_1)$ are unknown. Let $\widehat{\mathbf{Z}}_t = \left(\widehat{\mathbf{z}}_1^t, \ldots, \widehat{\mathbf{z}}_q^t\right)$, $\mathbf{Z}_t = \mathbf{Y}_t\Gamma_y = \left(\mathbf{z}_1^t, \ldots, \mathbf{z}_q^t\right)$, and $\Gamma_{i,j}(h)$ denote the covariance matrix between two series $\widehat{\mathbf{z}}_i^t$ and $\widehat{\mathbf{z}}_j^t$ at lag h, that is. $\Gamma_{i,j}(h) = \mathrm{Corr}\left(\widehat{\mathbf{z}}_i^{t+h}, \widehat{\mathbf{z}}_j^t\right)$. We say that $\widehat{\mathbf{z}}_i^t$ and $\widehat{\mathbf{z}}_j^t$ are connected if the multiple null hypothesis

$$H_0 : \Gamma_{i,j}(h) = \mathbf{0} \quad \text{for any} \quad h = 0, \pm 1, \pm 2, \dots, \pm m \tag{15}$$

Is rejected, where $m \geq 1$ is a prescribed integer. We should mention that the true $\Gamma_{i,j}(h)$ is not known since $\widehat{\mathbf{z}}_i^t$ is also one estimator for $\mathbf{z}_i^t$, but it will be asymptotically equivalent to $\text{Corr}\left(\mathbf{z}_i^{t+h}, \mathbf{z}_j^t\right)$ so long as $\widehat{\Gamma}_y$ is consistent to Γ_y. Given the structure of $\mathbf{W}_x$, this can be done under some regularity conditions, and therefore we also denote the true $\Gamma_{i,j}(h) = \text{Corr}\left(\mathbf{z}_i^{t+h}, \mathbf{z}_j^t\right)$, and the estimator $\widehat{\Gamma}_{i,j}(h) = \widehat{\text{Corr}}\left(\widehat{\mathbf{z}}_i^{t+h}, \widehat{\mathbf{z}}_j^t\right)$. The permutation in Step 3 in Section 2.1 can be performed as follows.

i. *Start with the q groups with each group containing one column of $\widehat{\mathbf{Z}}_t$ only.*

ii. *Combine two groups together if one connected pair is found.*

iii. *Repeat Step ii above until all connected pairs are within one group.*

We introduce below one way to identify the connected pairs of the transformed matrix $\widehat{\mathbf{Z}}_t$.

2.2.2 Maximum cross-correlation method

Similar to [16], one natural way to test hypothesis H_0 in (15) is to use the maximum cross-correlation over all elements of $\Gamma_{i,j}(h)$ and all the lags between $-m$ to m:

$$\widehat{L}_n(i,j) = \max_{|h| \leq m} \left| \widehat{\Gamma}_{i,j}(h) \right|_\infty, \tag{16}$$

where $\widehat{\Gamma}_{i,j}(h)$ is a sample correlation matrix between $\widehat{\mathbf{z}}_i^t$ and $\widehat{\mathbf{z}}_j^t$ at lag h when the dimension p and q are fixed, and it is a block-wisely thresholded sample correlation matrix when p and q are moderately high, and it will be constructed in the same way as that in [18]. We would reject H_0 for the pair $\left(\widehat{\mathbf{z}}_i^t, \widehat{\mathbf{z}}_j^t\right)$ if $\widehat{L}_n(i,j)$ is greater than an appropriate threshold value.

For the $q_0 = q(q-1)/2$ pairs of $\widehat{\mathbf{Z}}_t$, we propose a ratio-based method to those pairs for which H_0 will be rejected. We rearrange the q_0 pairs obtained $\widehat{L}_n(i,j)$'s in descending order: $\widehat{L}_1 \geq \cdots \geq \widehat{L}_{q_0}$. Define

$$\widehat{d} = \arg \max_{1 \leq j < c_0 q_0} \widehat{L}_j / \widehat{L}_{j+1}, \tag{17}$$

where $c_0 \in (0,1)$ is a prescribed constant. Similar ideas can be found in [16, 19].

3. Numerical results

3.1 Simulation

In this section, we illustrate the finite sample properties of the proposed methodology using simulated data. We only study the performance of the column transformation in (2) since the row transformation in the second stage is essentially the same. As the estimated, $\widehat{\mathbf{A}}$ is an orthogonal matrix for the normalized model in which $\mathrm{Var}_r(\mathbf{Y}_t) = \mathrm{Var}_r(\mathbf{X}_t) = \mathbf{I}_q$. We should use $\widehat{\mathbf{S}}_{y,0}^{-1/2}\mathbf{A}\widehat{\mathbf{S}}_{x,0}^{1/2}$ instead of $\mathbf{A}$ in computing estimation error defined as

$$D(\mathcal{M}(\mathbf{H}_1), \mathcal{M}(\mathbf{H}_2)) = \sqrt{1 - \frac{1}{\min(r_1, r_2)}\mathrm{tr}(\mathbf{P}_1\mathbf{P}_2)}, \tag{18}$$

where $\mathbf{H}_i$ is a $p \times r_i$ matrix with rank $(\mathbf{H}_i) = r_i$ and $\mathbf{P}_i = \mathbf{H}_i(\mathbf{H}_{i'}\mathbf{H}_i)^{-1}\mathbf{H}_{i'}$ for $i = 1, 2$. Let $\mathbf{A}^* = \{\Sigma_y(0)\}^{-1/2}\mathbf{A}\{\Sigma_x(0)\}^{1/2} \equiv \left(\mathbf{A}_1^*, \ldots, \mathbf{A}_{q_1}^*\right)$, and $\{\Sigma_y(0)\}^{-1/2}\mathbf{A} = \left(\mathbf{H}_1, \ldots, \mathbf{H}_{q_1}\right)$. Since $\{\Sigma_x(0)\}^{1/2}$ is a block-diagonal matrix, it holds that $\mathcal{M}(\mathbf{H}_j) = \mathcal{M}\left(\mathbf{A}_j^*\right)$ for $1 \le j \le q_1$. Therefore, we only need to replace $\mathbf{A}$ by $\widehat{\mathbf{S}}_{y,0}^{-1/2}\mathbf{A}$.

Since the goal is to specify (via estimation) the q_1 linear spaces $\mathcal{M}(\mathbf{A}_j)$, $j = 1, \ldots, q_1$, simultaneously, we first introduce the concept of a 'correct' specification. We call $\widehat{\mathbf{A}} = \left(\widehat{\mathbf{A}}_1, \ldots, \widehat{\mathbf{A}}_{\widehat{q}_1}\right)$ a *correct specification* for $\mathbf{A}$ if (i) $\widehat{q}_1 = q_1$, and (ii) rank $\left(\widehat{\mathbf{A}}_j\right) = \mathrm{rank}(\mathbf{A}_j)$ for $j = 1, \ldots, q_1$, after rearranging the order of $\widehat{\mathbf{A}}_1, \ldots, \widehat{\mathbf{A}}_{\widehat{q}_1}$ (we still denote the rearranged sub-matrices as $\widehat{\mathbf{A}}_1, \ldots, \widehat{\mathbf{A}}_{\widehat{q}_1}$ for simplicity in notation). When more than one $\mathbf{A}_j$ have the same rank, we pair each of those $\mathbf{A}_j$ with the $\widehat{\mathbf{A}}_j$ for which

$$D\left(\mathcal{M}(\mathbf{A}_j), \mathcal{M}\left(\widehat{\mathbf{A}}_j\right)\right) = \min_{rank\left(\widehat{\mathbf{A}}_i\right) = rank\left(\mathbf{A}_j\right)} D\left(\mathcal{M}(\mathbf{A}_j), \mathcal{M}\left(\widehat{\mathbf{A}}_i\right)\right). \tag{19}$$

Note that a correct specification for $\mathbf{A}$ implies a *structurally correct segmentation* for $\mathbf{X}_t$, which will be abbreviated as 'correct segmentation' hereafter. For a correct segmentation, we report the estimation error defined as

$$\overline{D}\left(\widehat{\mathbf{A}}, \mathbf{A}\right) = 1q_1 \sum_{j=1}^{q_1} D\left(\mathcal{M}(\mathbf{A}_j), \mathcal{M}\left(\widehat{\mathbf{A}}_j\right)\right). \tag{20}$$

In addition to the correct segmentations, we also report the proportions of the near-complete (NC) segmentations with $\widehat{q}_1 = q_1 - 1$ in all the following examples.

Example 1. We consider the model (2) with $p = 3$ and $q = 6$. The columns of $\mathbf{X}_t$ are generated as follows:

$$\mathbf{x}_i^t = \boldsymbol{\eta}_{t+i-1}^{(1)}\,(i = 1,2,3), \quad \mathbf{x}_i^t = \boldsymbol{\eta}_{t+i-4}^{(2)}\,(i = 4, 5), \quad \text{and} \quad \mathbf{x}_6^t = \boldsymbol{\eta}_t^{(3)}, \tag{21}$$

where

$$\boldsymbol{\eta}_t^{(j)} = \Phi^{(j)}\boldsymbol{\eta}_{t-1}^{(j)} + \varepsilon_t^{(j)} - \Theta^{(j)}\varepsilon_{t-1}^{(j)}, \quad j = 1,2,3. \tag{22}$$

The elements of $\Phi^{(j)}$ are drawn independently from $U(-3, 3)$, and then normalized by $0.9 \times \Phi^{(j)}/\|\Phi^{(j)}\|_2$ so that $\boldsymbol{\eta}^{(j)}$ is stationary, and the elements of $\Theta^{(j)}$ are drawn independently from $U(-1, 1)$. Meanwhile, the elements of the transformation matrix $\mathbf{A}$ are also drawn independently from $U(-3, 3)$. Thus $\mathbf{X}_t$ consists of three independent sub-matrices with, respectively, 3, 2, and 1 columns. In the experiments, we choose $c_0 = 0.75$ in (17) and $k_0 = 2$ in (8), and the other k_0 's give similar results. The sample sizes (n) are 100, 200, 300, 400, 500, 1000 and 1500, and the number of replications is 500 for each case. The proportions of the correct, incorrect, and near-complete (NC) segmentations are reported in **Table 1**. From **Table 1**, we can see that the proposed method improves as the sample size increases. With a dimension of $pq = 18$, we can see that the performance is reasonably well even for a small sample size, and the sum of the proportions of the complete and the near complete ones is more than 90% for a small sample size $n = 100$, from which we can see that we have achieved sufficient dimension reduction. We next study the estimation errors (20), and the box plots of the errors in the complete segmentations are shown in **Figure 1**. From **Figure 1**, we can see that the proposed method improves as the sample size increases.

As a concrete example, we report the correlogram of $\mathbf{Y}_t$ for one replication. We choose $m = 10$, and for $0 \le k \le m$ in **Figure 2**, the correlation between $\mathbf{y}_i^t$ and $\mathbf{y}_j^t$ are computed by

$$\widehat{\text{Corr}}\left(\mathbf{y}_i^{t+k}, \mathbf{y}_j^t\right) = \left| \text{diag}\left\{\widehat{\Sigma}_{y,i,i}(0)\right\}^{-1/2} \widehat{\Sigma}_{y,i,j}(k)\, \text{diag}\left\{\widehat{\Sigma}_{y,j,j}(0)\right\}^{-1/2} \right|_{\infty}. \tag{23}$$

From **Figure 2**, we can see that each of the columns of $\mathbf{Y}_t$ are highly correlated with the others. We then apply our method to $\mathbf{Y}_t$, and **Figure 3** depicts the cross

n	100	200	300	400	500	1000	1500
Correct segmentation	0.608	0.668	0.748	0.806	0.864	0.948	0.980
Incorrect segmentation	0.392	0.332	0.252	0.294	0.246	0.052	0.020
NC segmentation with $\hat{q}_1 = 2$	0.298	0.306	0.250	0.190	0.136	0.052	0.020

Table 1.
The proportions of correct and incomplete segmentations in Example 1. $p = 3, q = 6$.

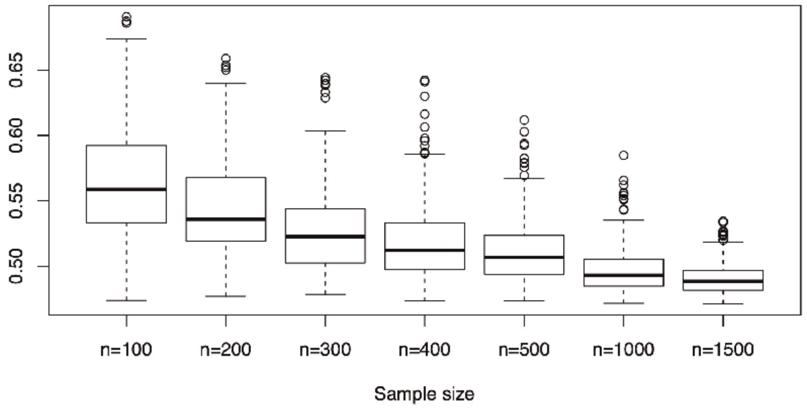


Figure 1.
The boxplots of estimation errors $\overline{D}\left(\hat{A}, A\right)$ in Example 1.

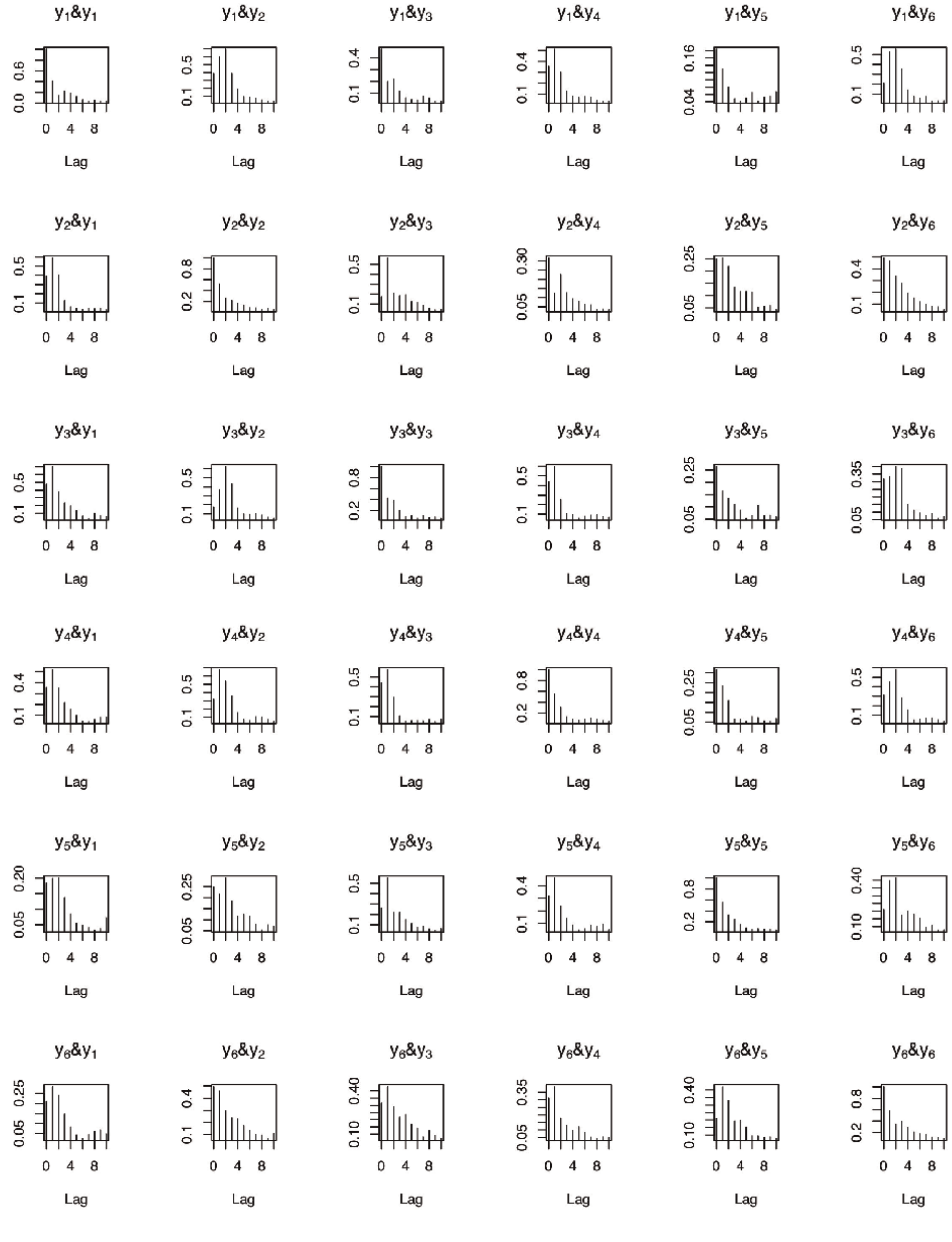


Figure 2.
Cross correlogram of the y_i^t in Example 1, $n = 1500$.

correlogram of the transformed matrix $\mathbf{Z}_t = \mathbf{Y}_t\widehat{\Gamma}_y$, and the scales of the y-axes are not the same. We can see from **Figure 3** that the columns of $\mathbf{Z}_t$ can be divided into three groups: $\{1,3,6\}$, $\{2, 4\}$, and $\{5\}$.

Example 2. In this example, we slightly increase the dimension as $p = q = 6$, and the data-generating processes are the same as those in Example 1. We observe that the performance of the proposed method is not as good as that in Example 1 when the dimension is higher with $pq = 36$. Nevertheless, similar conclusions can also be obtained from **Table 2** such as the proportion of the complete segmentation increases

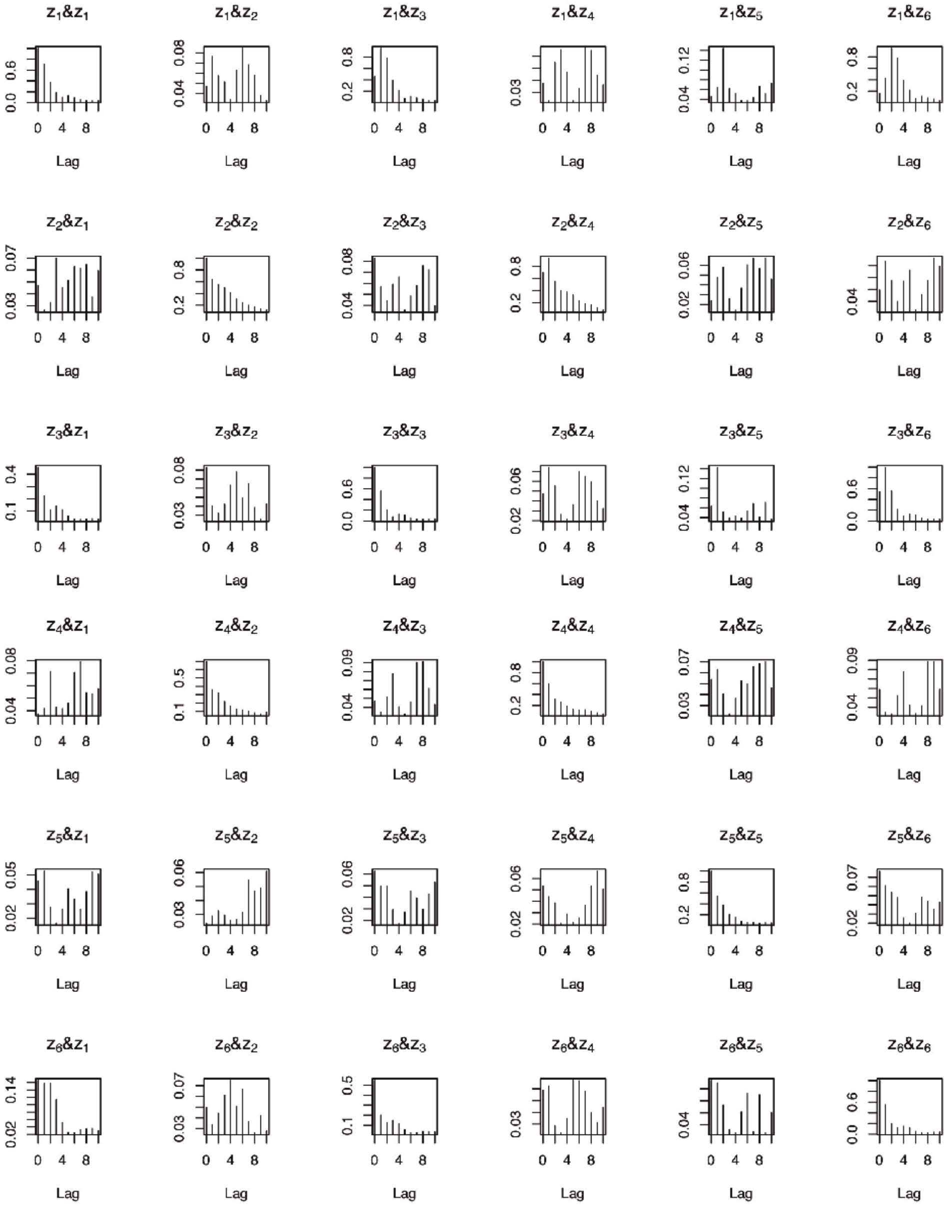


Figure 3.

Cross correlogram of the transposed series $\hat{z}_i^t = Y_t \hat{v}_i$ in Example 1, $n = 1500$. The components of $\hat{Z}_t$ can be segmented into 3 groups: {1,3,6}, {2,4}, and {5}.

n	100	200	300	400	500	1000	1500
Correct segmentation	0.504	0.620	0.648	0.670	0.732	0.766	0.780
Incorrect segmentation	0.496	0.380	0.352	0.330	0.268	0.234	0.220
NC segmentation with $\hat{q}_1 = 2$	0.354	0.320	0.312	0.308	0.264	0.230	0.210

Table 2.

The proportions of correct and incomplete segmentations in Example 2. $p = 6, q = 6$.

as the sample size becomes larger, and the sum of the proportions of the complete and the near complete ones is more than 90% even for a small sample size. The box plots of the estimation errors are similar to that in Example 1, and hence we do not report it here. From this example, we can see that when the dimension is higher, the proposed method without thresholding may not work well, and we will illustrate this point in the next example.

Example 3. We consider model (2) with $p = q = 10$, and the dimension is $pq = 100$. The columns of $\mathbf{X}_t$ are generated as follows:

$$\mathbf{x}_i^t = \boldsymbol{\eta}_{t+i-1}^{(1)}\,(i = 1,2,3,4), \mathbf{x}_i^t = \boldsymbol{\eta}_{t+i-5}^{(2)}\,(i = 5,6,7), \mathbf{x}_i^t = \boldsymbol{\eta}_{t+i-8}^{(3)}\,(i = 8, 9), \text{and} \quad \mathbf{x}_{10}^t = \boldsymbol{\eta}_t^{(4)}, \tag{24}$$

where

$$\boldsymbol{\eta}_t^{(j)} = \Phi^{(j)}\boldsymbol{\eta}_{t-1}^{(j)} + \varepsilon_t^{(j)} - \Theta^{(j)}\,\varepsilon_{t-1}^{(j)}, \quad j = 1,2,3,4, \tag{25}$$

and $\Phi^{(j)}$ and $\Theta^{(j)}$ are generated in the same way as Example 1. To fulfill a sparsity assumption, let

$$\mathbf{B}_i = \begin{pmatrix} \cos(\theta_i\pi) & \sin(\theta_i\pi) \\ -\sin(\theta_i\pi) & \cos(\theta_i\pi) \end{pmatrix}, \quad \mathbf{A} = \operatorname{diag}(\mathbf{B}_1, \mathbf{B}_2, \mathbf{B}_3, \mathbf{B}_4, \mathbf{B}_5), \tag{26}$$

where $\theta_1 = \pi/5$, $\theta_2 = \pi/6$, $\theta_3 = \pi/7$, $\theta_4 = \pi/8$, $\theta_5 = \pi/9$. Thus, $\mathbf{A}$ is a sparse orthogonal transformation matrix. Since the covariance of $\mathbf{X}_t$ and $\mathbf{Y}_t$ are all block-diagonal by the data generating process, hence $\widehat{\mathbf{S}}_{y,0}^{-1/2}\mathbf{A}\mathbf{S}_{x,0}^{1/2}$ is also block-diagonal and hence satisfies Assumption 3. If we apply our methodology to this model without thresholding the covariance matrices, the results are reported in **Table 3**. From **Table 3**, we can see that the proportions of the complete segmentations are pretty low for all the sample sizes, and it does not necessarily improve as the sample size increases. Now, we adopt the thresholding technique as that in [18], and the choice of the threshold is computed by a cross-validation method therein. **Table 4** presents the proportions of the correct, incorrect, and near-complete segmentations for model (2)

n	100	200	300	400	500	1000	1500
Correct segmentation	0.148	0.184	0.186	0.170	0.216	0.174	0.164
Incorrect segmentation	0.852	0.816	0.814	0.830	0.784	0.826	0.836
NC segmentation with $\hat{q}_1 = 3$	0.216	0.310	0.388	0.470	0.462	0.674	0.744

Table 3.
The proportions of correct and incomplete segmentations in Example 3 without thresholding. $p = 10$, $q = 10$.

n	100	200	300	400	500	1000	1500
Correct segmentation	0.440	0.572	0.654	0.666	0.840	0.910	0.936
Incorrect segmentation	0.560	0.428	0.346	0.334	0.260	0.080	0.064
NC segmentation with $\hat{q}_1 = 3$	0.490	0.294	0.282	0.286	0.084	0.062	0.042

Table 4.
The proportions of correct and incomplete segmentations in Example 3 by thresholding. $p = 10$, $q = 10$.

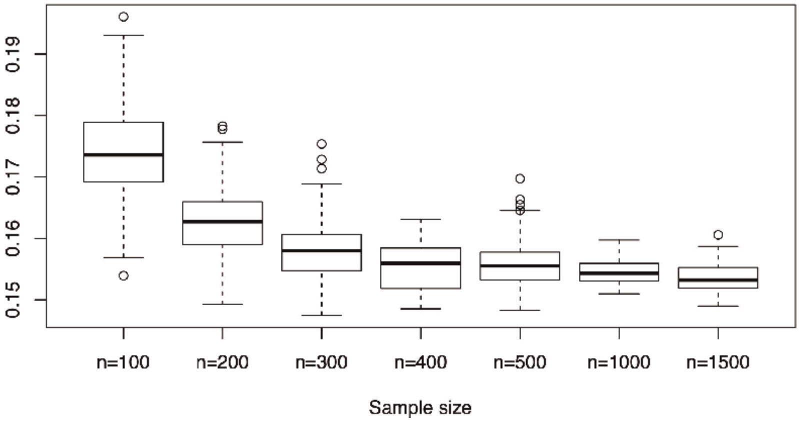


Figure 4.
The boxplots of estimation errors $\overline{D}\left(\hat{A}, A\right)$ *in Example 3.*

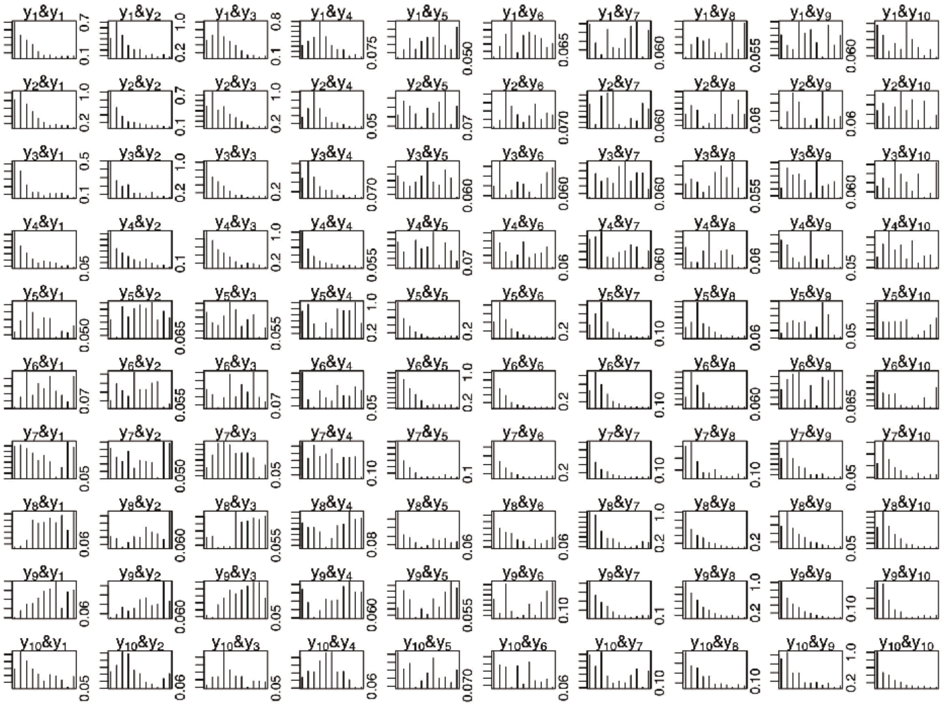


Figure 5.
Cross correlogram of the y_i^t *in Example 3,* $n = 1500$.

with $pq = 100$ dimensions. From **Table 4**, we can see that the thresholding technique works reasonably well for even small sample sizes. **Figure 4** reports the boxplots of the estimation errors, from which we can see that the estimation is very accurate, especially when the sample size is large. **Figure 5** displays the cross correlogram of $\mathbf{Y}_t$ for one instance of the 500 replications, and the correlations are calculated in the same way as that in Example 1 with thresholded estimators for the (auto) covariance

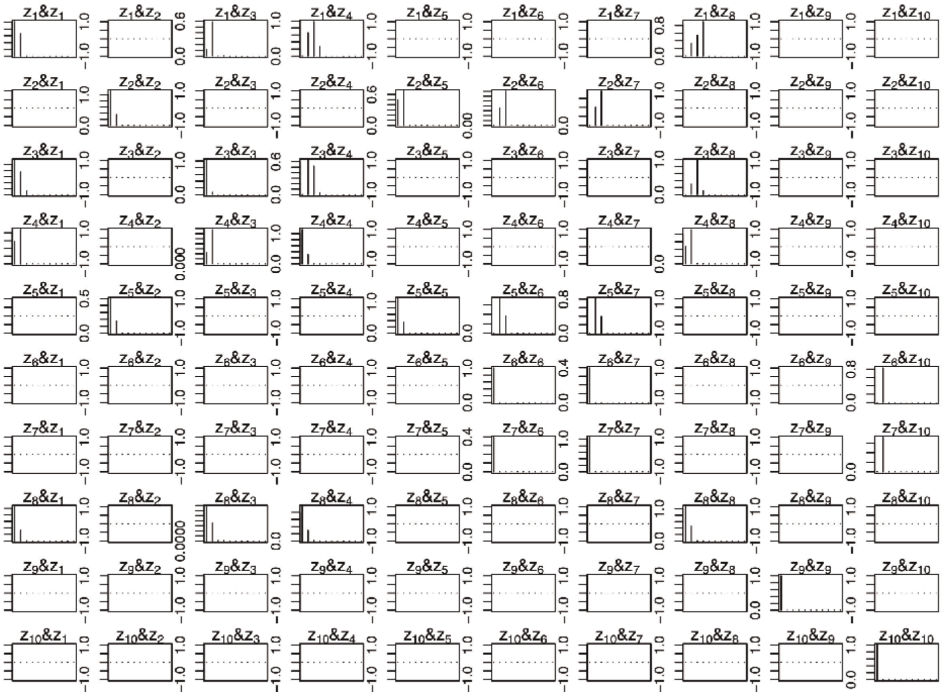


Figure 6.
Cross correlogram of the transposed series $\hat{z}_i^t = Y_t\hat{v}_i$ in Example 3, $n = 1500$. The components of $\hat{Z}_t$ can be segmented into 4 groups: {1,3,4,8}, {2,5,7}, {6,10}, and {9}.

matrices, and **Figure 6** gives the correlogram of the transformed data matrix $\mathbf{Z}_t = \mathbf{Y}_t\widehat{\Gamma}_y$. We can see from **Figure 5** that all the columns of $\mathbf{Y}_t$ are highly correlated, while those of $\mathbf{Z}_t$ can be separated into $_{\text{four}}$ groups: $\{1,3,4,8\}$, $\{2,5,7\}$, $\{6,10\}$, and $\{9\}$. The threshold of the correlation is 0.39 according to the rule in [18], and those pairs will be treated as uncorrelated ones if their maximal correlation is below this threshold level. Together with **Tables 3** and **4** and **Figures 4** and **5**, we can achieve substantial dimension reduction using our proposed method.

4. Conclusions

In this chapter, we proposed a new procedure to reduce the dimension of matrix-variate time series by looking for linear transformations to segment the matrix into many small sub-matrices, for which each of them is uncorrelated with the others both contemporaneously and serially, thus they can be analyzed separately, which will greatly reduce the number of parameters to be estimated in terms of modeling. We propose a two-step and more structured procedure to segment the rows and columns separately. When the dimensions are large in relation to the sample size, we assume the transformation matrices are sparse and use threshold estimators for the (auto) covariance matrices. Simulation studies suggest that our procedure works well in segmenting matrix-variate series, and it provides another option for researchers and practitioners to choose from the toolbox in matrix-variate time series modeling.

Acknowledgements

This research was supported in part by the Start-up Fund granted by the “100 Talents Program” of Zhejiang University and the National Natural Science Foundation of China (NSFC, 12201558).

Additional information

An earlier version of this chapter was previously published as a preprint in: Gao Z. Segmenting High-dimensional Matrix-valued Time Series via Sequential Transformations [Internet]. arXiv.org. 2020 [cited 2023 Sep 4]. Available from: https://arxiv.org/abs/2002.03382.

Author details

Zhaoxing Gao
Center for Data Science, Zhejiang University, Hangzhou, PR China

*Address all correspondence to: zhaoxinggao@zju.edu.cn; mazxgao@gmail.com

References

[1] Werner K, Jansson M, Stoica P. On estimation of covariance matrices with Kronecker product structure. IEEE Transactions on Signal Processing. 2008; **56**(2):478-491. DOI: 10.1109/TSP.2007.907834

[2] Xue Y, Yin X. Sufficient dimension folding for regression mean function. Journal of Computational and Graphical Statistics. 2014;**23**(4):1028-1043. DOI: 10.1080/10618600.2013.859619

[3] Li B, Kim MK, Altman N. On dimension folding of matrix-or array-valued statistical objects. The Annals of Statistics. 2010;**38**(2):1094-1121. DOI: 10.1214/09-AOS737

[4] Hung H, Wang CC. Matrix variate logistic regression model with application to EEG data. Biostatistics. 2012;**14**(1):189-202. DOI: 10.1093/biostatistics/kxs023

[5] Zhou H, Li L, Zhu H. Tensor regression with applications in neuroimaging data analysis. Journal of the American Statistical Association. 2013;**108**(502):540-552. DOI: 10.1080/01621459.2013.776499

[6] Zhou H, Li L. Regularized matrix regression. Journal of the Royal Statistical Society. 2014;**B76**(2):463-483. DOI: 10.1111/rssb.12031

[7] Ding S, Dennis CR. Matrix variate regressions and envelope models. Journal of the Royal Statistical Society Series B: Statistical Methodology. 2018;**80**(2):387-408. DOI: 10.1111/rssb.12247

[8] Gupta AK, Nagar DK. Matrix Variate Distributions. Boca Raton, FL: Chapman & Hall/CRC; 2000. DOI: 10.1201/9780203749289

[9] Leng C, Tang CY. Sparse matrix graphical models. Journal of the American Statistical Association. 2012; **107**(499):1187-1200. DOI: 10.1080/01621459.2012.706133

[10] Yin J, Li H. Model selection and estimation in the matrix normal graphical model. Journal of Multivariate Analysis. 2012;**107**:119-140. DOI: 10.1016/j.jmva.2012.01.005

[11] Zhao J, Leng C. Structured lasso for regression with matrix covariates. Statistica Sinica. 2014;**24**:799-814. DOI: 10.5705/ss.2012.033

[12] Zhou S. Gemini: Graph estimation with matrix variate normal instances. The Annals of Statistics. 2014;**42**(2): 532-562. DOI: 10.1214/13-AOS1187

[13] Walden A, Serroukh A. Wavelet analysis of matrix-valued time series. Proceedings: Mathematical, Physical and Engineering Sciences. 2017;**2002**(458): 157-179. DOI: 10.1098/rspa.2001.0866

[14] Wang D, Liu X, Chen R. Factor models for matrix-valued high-dimensional time series. Journal of Econometrics. 2019;**208**(1):231-248. DOI: 10.1016/j.jeconom.2018.09.013

[15] Gao Z, Tsay RS. A two-way transformed factor model for matrix-variate time series. Econometrics and Statistics. 2023;**27**:83-101. DOI: 10.1016/j.ecosta.2021.08.008

[16] Chang J, Guo B, Yao Q. Principal component analysis for second-order stationary vector time series. Annals of Statistics. 2018;**46**(5):2094-2124. DOI: 10.1214/17-AOS1613

[17] Kolda TG, Bader BW. Tensor decompositions and applications. SIAM

Review. 2009;**51**(3):455-500. DOI: 10.1137/07070111

[18] Bickel PJ, Levina E. Covariance regularization by thresholding. The Annals of Statistics. 2008;**36**(6): 2577-2604. DOI: 10.1214/08-AOS600

[19] Lam C, Yao Q. Factor modeling for high-dimensional time series: Inference for the number of factors. The Annals of Statistics. 2012;**40**(2):694-726. DOI: 10.1214/12-AOS970

Chapter 8

Neural Networks in Nonlinear Time Series: A Subsampling Model Selection Procedure

Michele La Rocca and Cira Perna

Abstract

In this chapter, the problem of model selection in neural networks for nonlinear time series data is addressed. A systematic review and an appraisal of previously published research on the topic are presented and discussed with emphasis on a complete strategy to select the topology of the model. The procedure attempts to explain the black box structure of a neural network by providing information on the complex structure of the relationship between a set of inputs and the output. The procedure combines a set of graphical and inferential statistical tools and allows to choose the number and the type of inputs, considered as explanatory variables, by using a formal test procedure based on relevance measures and to identify the hidden layer size by looking at the predictive performance of the neural network model. To obtain an approximation of the involved statistics, the approach heavily uses the subsampling technique, a computer-intensive statistical methodology. The results on simulated data show the good performance of the overall procedure.

Keywords: nonlinear time series, feed-forward neural networks, model selection, subsampling, statistical inference

1. Introduction

In the last decades, artificial neural networks have received considerable attention in the literature on time series analysis and forecasting, due to their flexibility in modeling complex nonlinear mapping functions for an output variable, given the input variables. They are mainly useful when the underlying relationships are complex and cannot be modeled through a known functional form.

In a time series context, single hidden layer feedforward neural networks are the most popular and widely used network paradigm in many applications [1] due to their clear advantages with respect to alternative non-parametric techniques. Firstly, this simple neural network structure allows handling short time series (typically available in many time series contexts, such as economics, banking, and insurance applications) where lack of information hardly justifies the use of more complex deep learning structures, such as long short-time memory or convolutional neural networks. Moreover, single hidden layer feedforward neural networks are able to provide an arbitrarily accurate approximation of an unknown function of interest with any desired accuracy [2, 3]. Finally, they deliver good predictive accuracy without suffering the problems that arise when working with high-dimensional data, the so-called "curse of dimensionality", which affects other alternative non-parametric techniques [4].

Along with the great success of neural networks, there is also growing concern about their black-box nature due to the lack of procedures, usually employed in statistical and econometric frameworks, able to provide explanations of their inner workings and input-output mappings. The problem can be traced back to the identification of a proper network topology which could improve the network interpretability. When dealing with feedforward NN, the identification of an "optimal" topology is related to the determination of the number and the type of inputs, the hidden layer size, and the activation function type.

While as regards the activation function, many studies agree on the choice of a well-behaved, i.e., bounded, monotonically increasing, and differentiable function (such as a sigmoidal function), the problem of input selection and hidden neurons identification, was and still is much debated and different solutions have been proposed.

The most widely used approaches are based on pruning (for a review see [5] and the references therein), regularization (see [6] and the recent survey in [7]), and stopped training rules (see [8]). Although these techniques can lead to good approximation results, they are based on criteria for reducing the model complexity and do not, however, frame the search for an appropriate model in a statistical perspective, capable of providing information on the plausibility of the model. From this standpoint, methods for eliminating insignificant weights based on their asymptotic distributions have been proposed [9, 10]. Focusing on single weights, which have no clear interpretation, does not give any information either on the most "significant" variables or on the hidden neurons, which is useful in any model-building strategy. Alternatively, statistical methods of model selection based on information criteria have been used [11], with particular emphasis on the Akaike information criterion, the Bayesian information criterion, and the Schwarz information Criterion. These criteria add a complexity penalty to the usual sample log-likelihood, and the model that optimizes this penalized log-likelihood is selected. However, these measures should be used carefully in a neural network context, since over-parameterized models, with heavy consequences on overfitting and poor ex-post forecast accuracy, could be selected [12]. Moreover, they do not take the uncertainty of the models into account and they do not explicitly give any information about the significant variables of the model.

In our opinion, an effective model selection strategy should be addressed in a broad statistical framework, relating it to the classic model-building approach. In a regression framework, in [13] a method to select a proper network strategy has been proposed; it is based on a comparison of the out-of-sample predictive ability of several different models through appropriate testing procedures. The same strategy has been implemented in [14] to identify the hidden layer size.

Alternatively, the problem can be addressed by pointing out the different roles in the model of the input and hidden layer neurons. The former are the explanatory variables of the model; the latter are related to the complexity of the model and have an effect on its forecast performance. In this perspective, a model selection strategy should highlight the role of input nodes useful for model identification and interpretation and should treat the hidden layer as a smoothing parameter capable of capturing the pattern in data and performing complex nonlinear patterns.

In this context, this chapter provides a systematic review and critical evaluation of an approach for selecting a neural network model for nonlinear time series proposed in [13–22]. The procedure focuses on a comprehensive strategy that combines a set of graphical, exploratory, and inferential statistical tools and allows to select the number and the type of inputs by using a formal test procedure based on relevance measures and identifying the hidden layer size by looking at appropriate measures of predictive risk.

The whole procedure makes extensive use of subsampling, a computer-intensive statistical methodology for estimating parameters of the sampling distribution of a statistic computed from a sample. In this context, it has been used for obtaining an approximation of the sampling distributions of the involved test statistics for the input nodes and to estimate the predictive risk for the hidden layer size. The choice of this resampling technique is due to its validity for dependent data, under quite general and weak assumptions [23]. Moreover, unlike the residual bootstrap, used in the context of neural networks in [24], it is robust against misspecified models, as the neural networks are.

The chapter is organized as follows. In Section 2, we briefly illustrate the subsampling scheme for hypothesis testing. In Section 3 we describe the data-generating process and the employed neural network model. In Section 4 we present and discuss the procedure for neural network model selection, focusing on the problem of inclusion of relevant variables, the omission of the irrelevant ones, and the identification of the hidden layer size. In Section 5, we illustrate how the whole procedure can be implemented. In Section 6, we report an illustrative example of a simulated data set, in order to evaluate the performances of the proposed procedure. Some concluding remarks, in Section 7 will close the chapter.

2. The subsampling technique for time series data

In this approach, blocks of consecutive observations are obtained from the original observed time series, in order to take into account the dependence structure of the observed time series. Therefore, each individual subseries of the observations is considered as a valid “time sub-series” in its own right. Each block is generated by true underlying data-generating process and so, information on the sampling distribution of a given statistic can be gained by evaluating the statistic on all subseries.

Following [23], let $\{Y_t, t \in \mathbb{Z}\}$ be a stationary and mixing process governed by a probability law P, assumed to belong to a certain class of laws $\mathcal{P}$. The goal is to construct an asymptotically valid test for the null hypothesis $H_0 : P \in \mathcal{P}_0$ versus the alternative $H_1 : P \in \mathcal{P}_1$ with $\mathcal{P}_0 \cup \mathcal{P}_1 = \mathcal{P}$. Given the observed time series $\{Y_1, Y_2, \dots, Y_T\}$, the test can be based on a test statistic such as

$$W_T = \delta_T w_T = \delta_T w_T(Y_1, Y_2, \dots, Y_T) \tag{1}$$

where δ_T is a fixed nonrandom normalizing sequence. Assume that there exists a constant $w(P)$ which satisfies $w(P) = 0$ under the null and $w(P) > 0$ under the alternative and that $w_T \to w(P)$ in probability.

Let $G_T(x, P) = \Pr_P\{\delta_T w_T \leq x\}$ the cumulative distribution function of the sampling distribution of the test statistic and assume that it converges in distribution to a given limit law at least for $P \in \mathcal{P}_0$. Naturally, as long as $\delta_T \to \infty$, this implies that $w_T \to 0$ in probability for $P \in \mathcal{P}_0$.

The subsampling scheme works as in Algorithm 1.

Algorithm 1. Subsampling for hypothesis testing.

```
Require: Fix the subseries length $b$ with $b < T$.
for $j = 1, \dots, T - b + 1$ do
    Determine $\mathbf{Y}_{b,j} = \{Y_j, Y_{j+1}, \dots, Y_{j+b-1}\}$, the subseries of $b$ consecutive observations.
    Evaluate the test statistic at the block of data $\mathbf{Y}_{b,j}$, obtaining $w_{T,b,j}$
end for
Approximate the sampling distribution of $W_T$ by
    $\hat{G}_{T,b}(x) = \frac{1}{T-b+1} \sum_{j=1}^{T-b+1} \mathbb{I}\{\delta_b w_{T,b,j} \le x\}$
    where $\mathbb{I}(.)$ is the indicator function.
Obtain the critical value for the test as the $1-\alpha$ quantile of $\hat{G}_{T,b}(\cdot)$, that is
    $g_{T,b}(1-\alpha) = \inf\{x : \hat{G}_{T,b}(x) \ge 1-\alpha\}$.
Reject the null $H_0$ at the nominal level $\alpha$ if and only if
    $W_T > g_{T,b}(1-\alpha)$.
```

A subsampling p-value can be computed as

$$PV_{T,b} = \frac{1}{T-b+1} \sum_{t=1}^{T-b+1} \mathbb{I}\{\delta_b w_{b,t} \ge \delta_T w_T\}$$

In this case, the nominal level α test rejects the null if and only if

$$PV_{T,b} < \alpha.$$

The subsampling method delivers consistent results under very general and minimal assumptions, valid for both linear and nonlinear processes. The scheme requires that $\frac{\delta_b}{\delta_T} \to 0$, $\frac{b}{T} \to 0$ and $b \to \infty$ as $T \to \infty$ and the existence of a limiting law for the sampling distribution of the test statistic. The normalizing sequence δ_T could also be unknown and can be consistently estimated by a preliminary subsampling cycle [25]. Moreover, subsampling does not require any knowledge of the specific structures of the time series other than its stationarity and strong mixing property. As a consequence, it provides a robust alternative to the bootstrap techniques, whose consistency may fail unless specific regularity conditions on the model hold. This property seems to be extremely suitable when using a neural network model which, by its own nature, can be considered as a "misspecified" model, being an approximation of the underlying functional relationship [26]. Finally, the subsampling can also be extended to heteroskedastic time series [27] and diverging statistics [28].

A major drawback to applying subsampling is that the block size b must be chosen in applications. This parameter is related to the amount of dependence assumed in the series. If the blocks are long, a poor estimate of the distribution of the statistic could be obtained whereas too short blocks could not preserve, in the resampled series, the original dependence in the data. However, note that, for small and moderate sample sizes, the block size could critically affect the subsampling accuracy but it seems to be not particularly critical for large sample sizes since asymptotic results are still valid for a broad range of values for b [23].

Various methods for choosing the block size have been proposed. These are calibration, minimum volatility, interpolation, and extrapolation methods (see Chapter 10 in [23]). In the following, we focus on the use of the minimum volatility method, which works under very general conditions.

The procedure is illustrated in Algorithm 2.

Algorithm 2. Minimum volatility method for selecting the block size b in subsampling.

```
Require: Fix $\alpha$.
Require: Fix the values $b_{small}$ and $b_{big}$.
1: for For any integer $b$: $b \in (b_{small}, b_{big})$ do
2:     Compute the subsampling quantile $g_{T,b}(1-\alpha)$ using Algorithm 1
3:     Fix a small integer $k$
4:     Compute $VI_b$ as a measure of variability of the values
         $\left\{g_{T,b-k}(1-\alpha), \dots, g_{T,b+k}(1-\alpha)\right\}$.
5: end for
6: Pick the value $b^*$ corresponding to the smallest volatility. That is
       $b^* = min_b VI_b$
```

The value $g_{T,b^*}(1-\alpha)$ can be used in line 7 of the Algorithm 1, as a critical value of the test.

As remarks, note that the range of b values, determined by b_{small} and b_{big}, is not very important, as long as it is not too narrow. Moreover, the integer k, introduced in line 3 of Algorithm 2 seems not to affect the determination of b; it can be fixed at $k = 2$ or $k = 3$ [23]. Finally, in line 4 of Algorithm 2, as a measure of variability standard deviation or other robust alternatives, such as the mean absolute deviation, can be employed.

3. The DGP and the neural network model

Let $\{Y_t, t \in \mathbb{Z}\}$, a process modeled as:

$$Y_t = g(\mathbf{X}_t) + \varepsilon_t \tag{2}$$

where $\{Y_t, \mathbf{X}_t'\}$ is a stationary, α-mixing sequence and $\mathbf{X}_t = (X_{1t}, \dots, X_{dt})'$ is a vector of d random variables possibly including explanatory variables, lagged explanatory variables and lagged values of Y_t. The unknown function $g(\cdot)$ is assumed to be a continuously differentiable function defined on a compact subset of $\mathbb{R}^d$. This data-generating process is very general, incorporating, in particular cases, regression models with lagged variables and dependent errors as well as autoregressive nonlinear models for time series.

When it is not possible to postulate a parametric model for the function $g(.)$, the use of non-parametric methods is required. In this context, neural networks have proved to be powerful tools due to their great flexibility and their capability of providing a model which fits any data with an arbitrary degree of accuracy.

A feed-forward neural network $NN(d, r)$ to approximate the function $g(.)$ can be defined as

$$f(\mathbf{x}_t, \theta) = \sum_{k=1}^{r} c_k \phi\left(\sum_{j=1}^{d} a_{kj} x_{jt} + a_k\right) + c_0 \tag{3}$$

where $\mathbf{x} = (x_1, \ \ldots, \ x_d)$ is the vector of the d input variables, a_{kj} is the weight of the connection between the j-th input neuron and the k-th neuron in the hidden level; $c_{(k)}, k = 1, \ldots, r$ is the weight of the link between the k-th neuron in the hidden layer and the output; a_{k0} and c_0 are respectively the bias term of the hidden neurons and of the output; $\phi(.)$ is the activation function of the hidden layer. We define $\boldsymbol{\theta} = \left(c_0, c_1, \ldots, c_r, \mathbf{a}_1', \mathbf{a}_2', \ldots, \mathbf{a}_r'\right)'$ where $\mathbf{a}_i' = (a_{i0}, a_{i1}, \ldots, a_{id})$ with $\theta \in \Theta \subset \mathbb{R}^{r(d+2)+1}$.

As usual in the neural network literature, a sigmoidal function, such as the logistic or hyperbolic tangent, is assumed for the hidden layer. In this case, a single-layer neural network can arbitrarily closely approximate the unknown function as well as its derivatives, up to a given order (provided that they exist), as measured by a proper norm [29].

Moreover, it is possible to use an approximation result [30] according to which, if $g(.)$ is differentiable on a compact set, feedforward networks with one layer of sigmoidal nonlinearities achieve an integrated squared error of order $O(1/r)$, being r the number of the hidden nodes. That is, there exists a parameter vector $\boldsymbol{\theta}^*$ such that:

$$\|g(\mathbf{x}) - f(\mathbf{x}, \boldsymbol{\theta}^*)\| \leq \frac{\left(2C_g\right)^2}{r} \tag{4}$$

where $C_g > 0$ is a proper chosen constant.

Once the neural network topology has been fixed, the estimation of the network weights (learning) has to be obtained. If $\mathcal{X} \equiv \{\mathbf{z}_i, i = 1, 2, \ldots, T\}$, with $\mathbf{z}_i' = \left(Y_i, \mathbf{x}_i'\right)$ is a random sample of size T, the estimate of the parameter vector θ can be obtained as:

$$\hat{\boldsymbol{\theta}}_T = \arg\min_{\boldsymbol{\theta} \in \Theta} \sum_{i=1}^{T} q(Y_i, f(\mathbf{x}_i, \boldsymbol{\theta})) \tag{5}$$

where Θ is the parameter space and $q(\cdot)$ is a proper chosen loss function.

Under general regularity conditions, if π denotes the joint distribution of the random vector $\mathbf{z}^T = \left(Y, \mathbf{x}^T\right)$, the vector $\hat{\boldsymbol{\theta}}_n$ exists and converges almost surely to $\boldsymbol{\theta}_0$, given by:

$$\boldsymbol{\theta}_0 = \arg\min_{\boldsymbol{\theta} \in \Theta} \int q(y, f(\mathbf{x}, \boldsymbol{\theta}) d\pi(\mathbf{z}), \tag{6}$$

if the integral exists and the optimization problem has a unique solution vector interior to Θ. The latter condition is required to avoid many distinct weight vectors produce identical network outputs, which could be a serious challenge when dealing with this type of model. However, sufficient conditions to guarantee the uniqueness of $\boldsymbol{\theta}_0$ a suitable parameter space Θ have been proposed in the specialized literature [31] and are generally assumed everywhere.

However, even if neural networks enjoy numerous properties, including that of being universal approximators, and many efficient estimation algorithms are available in the specialized literature, they face a challenging issue, related to the lack of consolidated techniques capable of interpreting the model.

The over-parameterized nature of these models, due to the parameters often exceeding the size of the training dataset, together with the lack of information on the relationship between the input level and the output, which does not provide any understanding of the structure of the function that is approaching, transforms neural networks into black boxes, greatly limiting their usefulness. These problems require making systems explainable and interpretable, transforming the black box into a white box and allowing for a clear explanation of the underlying relationships captured by the neural network models.

In our opinion, a possible solution could be to address the choice of a suitable network topology in the classical model selection approach. In this context, in the following, we review a strategy proposed in [13–21] for regression models and extend it to the case of time series data.

4. Network selection procedure

The procedure highlights the different roles of the input neurons with respect to the hidden ones. The latter are explanatory model variables and, as a consequence, are important for model identification and interpretation; the hidden layer size can be considered as a smoothing parameter that accounts for the trade-off between estimation bias and variability.

Following this standpoint, the problem of identifying the number and the type of inputs can be framed in a statistical and econometric perspective using a formal test procedure to evaluate the inclusion of irrelevant variables or the omission of relevant ones. This approach has been proposed in [32, 33] in the case of *iid* case and extended in [14] for dependent data. On the contrary, the hidden layer size can be determined by looking at procedures of the predictive performance of the model.

4.1 The selection of the input: inclusion of irrelevant variables

Removing the irrelevant variables is a crucial part of the model-building process, impacting the stability and the complexity of the final model. The most obvious reason is that some variables may not be related to the outcome and should be removed following the parsimony principle, obtaining an improvement of the interpretability of the final model and, by reducing the variance, a model with better predictive ability.

The approach we focus on is based on relevance measures for the input variables. In order to remove irrelevant variables we use a stepwise selection rule which involves the definition of a variable's relevance to the model, the estimation of the sampling distribution of the relevance measure and a testing procedure to verify the hypothesis that the variable is irrelevant.

Following [15, 16, 34], let $f_i(\mathbf{x}, \boldsymbol{\theta}^*)$ be the partial derivative of $f(\mathbf{x}, \boldsymbol{\theta}^*)$ with respect to x_i; that is, $\forall\mathbf{x}$:

$$f_i(\mathbf{x}; \boldsymbol{\theta}^*) = \frac{\partial f(\mathbf{x}; \boldsymbol{\theta}^*)}{\partial x_i}. \tag{7}$$

A measure of the relevance of a variable X_j to the model can be based on the expectation of some function of the derivatives of the neural network. Therefore, it can be defined as:

$$RM_i(\boldsymbol{\theta}^*) = \mathbb{E}\left[h\left[f_i(\mathbf{X}_t, \boldsymbol{\theta}^*)\right]\right] \tag{8}$$

where $h(\cdot)$ is a proper chosen function and $\mathbb{E}\{\cdot\}$ is the expected value w.r.t. the probability measure of the vector of the explanatory variables. We can derive the measures proposed by [33] by choosing for example the average derivative $(h(x) = x)$; the absolute average derivative $(h(x) = |x|)$; the square average derivative $(h(x) = x^2)$ or the maximum and minimum derivative $(h(x) = \max(x)$ and $h(x) = \min(x))$.

The logic of this proposal is the same as that used in linear models in which the relevance of a variable is measured by its coefficient, which is also the magnitude of the partial derivative of the dependent variable. However, it takes into account that in the nonlinear case, the partial derivative is not a constant but it varies through the range of the independent variables.

To select the set of variables to be tested as irrelevant, simple graphical exploratory tools could be used. They are based on plots of the derivatives and plots of the relevance measures for each single lag. Values of the derivatives and of the relevance measures close to zero designate the corresponding lag as a potential variable in the set of irrelevant ones.

A formal test can be implemented as follows.

Let $\mathcal{X}_0 = \{x_i, i \in I_0\}$ be the set of variables to be tested as irrelevant to the model and let:

$$m_i(x, \boldsymbol{\theta}) = h\left[f_i(x, \boldsymbol{\theta})\right] \tag{9}$$

The hypothesis that the variables in $\mathcal{X}_0$ are not relevant can be written as:

$$H_0 : m^* = \sum_{i \in I_0} \mathbb{E}[m_i(\mathbf{X}_t, \boldsymbol{\theta}^*)] = 0 \tag{10}$$

The null H_0 can be tested by using the statistic,

$$\hat{m}_T = T^{-1} \sum_{i \in I_0} \sum_{t=1}^{T} m_i\left(\mathbf{X}_t, \hat{\boldsymbol{\theta}}_T\right) = T^{-1} \sum_{t=1}^{T} m\left(\mathbf{X}_t, \hat{\boldsymbol{\theta}}_T\right) \tag{11}$$

where the parameter vector $\hat{\boldsymbol{\theta}}_T$ is a consistent estimator of the unknown parameter vector $\boldsymbol{\theta}^*$.

The asymptotic distribution of the test statistic is not one of the familiar tabulated distributions and it is very difficult to deal with. It can be approximated by using the subsampling defined in Algorithm 1, in which in line 5, it is:

$$W_T = \delta_T w_T = T\hat{m}_T \quad (12)$$

The effectiveness of this procedure has been studied in [15] where, under the null and under quite general assumptions, the consistency of the proposed subsampling test procedure has been formally proved and the good finite sample properties have been stated by simulations.

4.2 The selection of input: omission of relevant variables

Leaving out one or more relevant variables is another crucial issue in model building. Any bias could cause the model to attribute the effect of the missing variables to those that have been included.

In order to verify if there are any omitted variables, it can be used a procedure, proposed in [33] for the *iid* case and extended to the case of regression models in [20]. It is based on the comparison of competing neural network models.

Suppose that $f_1(x, \boldsymbol{\theta}_1^*)$ and $f_2(x, \boldsymbol{\theta}_2^*)$ are two competing neural network models which differ only in the inputs and are, therefore, nested. The idea is that if there are no omitted variables the network f_1 is capable of producing an output identical to that of the network f_2.

The omission of relevant variables can be verified by using a discrepancy measure between the outputs of the two competing neural network models $f_1(x, \boldsymbol{\theta}_1^*)$ and $f_2(x, \boldsymbol{\theta}_2^*)$ defined as:

$$m^* = \mathbb{E}\left[\left(f_1(x, \boldsymbol{\theta}_1^*) - f_2(x, \boldsymbol{\theta}_2^*)\right)^2\right] \quad (13)$$

Therefore, the hypothesis that the two models are equivalent and so there are no omitted variables, can be written as:

$$H_0 : m^* = 0 \quad (14)$$

and it can be tested by using the statistic

$$\hat{m}_T = \frac{1}{T}\sum_{t=1}^{T} \left(f_1\left(\mathbf{X}_t, \hat{\boldsymbol{\theta}}_{T1}\right) - f_2\left(\mathbf{X}_t, \hat{\boldsymbol{\theta}}_{T2}\right)\right)^2 \quad (15)$$

where $\hat{\boldsymbol{\theta}}_{T1}$ and $\hat{\boldsymbol{\theta}}_{T2}$ are consistent estimators of, respectively, $\boldsymbol{\theta}_1^*$ and $\boldsymbol{\theta}^*$. Again, the distribution of the test statistic can be consistently estimated by using the subsampling described in Algorithm 1 in which in line 5, it is:

$$W_T = \delta_T w_T = T\hat{m}_T \quad (16)$$

The validity of this subsampling test procedure can easily be proved along the same lines as in [15].

4.3 The selection of the hidden layer size

The selection of the hidden layer size plays an important role in neural network modeling. It is related to the complexity of the model and, in our opinion, it can be considered as a smoothing parameter governing the trade-off between estimation bias and variability. In particular, if a too-low number of hidden layer neurons is chosen, an underfitting situation can occur, with serious consequences on the ability of the network to approximate the unknown target function. In this case, the neural network model will have poor performance on the training data. On the contrary, over-parameterization of the network produces overfitting, also capturing the noise present in the data, and, as a consequence, the neural network model will perform well in approximating the data used for the parameter estimations, but it will have a reduced accuracy of ex-post forecasts.

However, determining r in such a way that the neural network model balances good fitting properties and simultaneously good predictive performances is a difficult task.

In the past, practitioners generally used to determine this parameter by trial and error methods and, in this context, some rules of thumb were suggested. They are based on the assumption that the hidden size is directly proportional to the number of inputs, for a suitable choice of the proportional constant (see [1]). Hybrid methods have also been proposed; they combine pruning, growing, or empirical formulas with other methods such as genetic algorithm (see [35]) or singular value decomposition [36]. None of these methods has the theoretical rigor of revealing optimal or at least near-optimal solutions in the context of time series analysis, being "data dependent" criteria.

More recently, since the number of hidden neurons has an effect on forecast performance, methods have been proposed that take into account the forecast ability of the model. In this perspective, statistical measures such as the Akaike and Bayesian information criteria can be implemented and, besides them, it has become common practice to use cross-validation techniques. Such methods, in their classical formulations, are also used in the literature to evaluate autoregressions, but their application is not straightforward because of the inherent serial correlation and potential non-stationarity of the data. Some extensions to dependent data have been proposed (see [37, 38]) but, again, no inferential procedures are available to discriminate among the alternative models. In this context, we propose to use a statistical procedure proposed in [39] in a general set-up and implemented in [20] in nonlinear regression functions approximated by neural networks. It is based on a measure of predictive risk estimated by using the subsampling.

Let $\hat{Y}_{T+1}$ be the one-step ahead predictor of Y_{T+1} obtained by using a neural network model $NN(d, r)$ and suppose that the input layer size d has been fixed. We propose as a measure of predictive risk the quantity:

$$\Delta_T(r) = \mathbb{E}\left[\left(\hat{Y}_{T+1} - Y_{T+1}\right)^2\right] \tag{17}$$

which, by construction, depends only on r. Therefore it is reasonable to determine r as the value $\hat{r}$ such that:

$$\hat{r} = \arg\min_r \Delta_{T,b}(r) \tag{18}$$

Since $\Delta_T(r)$ is unknown, an estimate is needed. It can be employed by using subsampling.

The procedure is implemented in Algorithm 3.

Algorithm 3. Predictive risk estimation of the hidden layer size by subsampling.

```
Require: Fix the value $r_{max}$.
Require: Fix the subseries length $b$
for $r = 1, \dots, r_{max}$ do
    for $j = 1, \dots, T - b$ do
        Determine $\mathbf{Y}_{b,j} = \{Y_j, \dots, Y_{j+b-1}\}$, and $\mathbf{X}_{b,j} = \left(\mathbf{X}'_j, \dots, \mathbf{X}'_{j+b-1}\right)'$ the subseries of $b$ consecutive observations.
        estimate of the neural network parameter vector $\hat{\theta}_{T,b,j}$ by using $Y_{b,j}$ and $\mathbf{X}_{b,j}$
        Obtain the one-step ahead predictor $\hat{Y}^{(t)}_{t+b}$ by using the model $\hat{\theta}_{T,b,t}$
    end for
    Get the subsampling estimate as $\hat{\Delta}_{T,b}(r) = \frac{1}{T-b+1}\sum_{t=0}^{T-b}\left(\hat{Y}^{(t)}_{t+b} - Y_{t+b}\right)^2$
end for
Pick the value $\hat{r}$ corresponding to the smallest value of $\hat{\Delta}_{T,b}(r)$. That is:
    $$\hat{r} = \arg\min_r \hat{\Delta}_{T,b}(r).$$
```

---

5. The three steps procedure for neural network topology selection

In this section, we illustrate how to implement the whole procedure for neural network model selection. In particular, we consider the case of a time series Y_t, $t = 1, \dots T$, which follows an autoregression model of order p, $(NAR(p))$:

$$Y_t = g\left(Y_{t-1}, \dots, Y_{t-p}\right) + \varepsilon_t \tag{19}$$

where $g(\cdot)$ is a possibly nonlinear function and the innovations ε_t are distributed as standard normal.

The starting point of the analysis is the implementation of nonlinearity tests on the original time series. Although in fact, neural networks are capable of approximating both linear and nonlinear functions, in the first case it is preferable to use the classic AR(p) models. The latter are much less complex and are equipped with well-established and consolidated model selection procedures. The rejection of linearity and the frequent impossibility of assuming a particular nonlinear functional form for the function $g(.)$ justifies the use of a neural network model, whose typology has to be suitably selected.

In Algorithm 4, we present the procedure which synthesizes the relevant steps to get the best neural network approximation.

Algorithm 4. Three step procedure for model selection.

---

1. Select the relevant variables

1.1 Fix $r = 1$ and $d = d^*$ where d^* is a proper chosen maximum number of lags.

1.2 Estimate the model $NN(d^*, r)$.

1.3 Plot the derivatives and the relevance measures to identify a candidate set I_0 of the irrelevant variables.
1.4 Test if the set of variables I_0 is irrelevant (by using the procedure in Section 4.1)
1.5 Determine the 'optimal' value $\hat{d}$.

2. Select the hidden layer size

2.1 Fix $r = r^*$, the maximum number of hidden units.
2.2 Estimate the models $NN\left(\hat{d}, 1\right)$, …, $NN\left(\hat{d}, r^*\right)$.
2.3 Compute the predictive risk for each model (by using the procedure in Section 4.3).
2.4 Choose $NN\left(\hat{d}, \hat{r}\right)$ such that the predictive risk is minimum.

3. Check for omitted variables

3.1 Identify a set of possibly omitted variables.
3.2 Estimate the new neural model including the new set of variables.
3.3 Test if the two models give the same output (by using the procedure in Section 4.2).
3.4 If the two models are equivalent, choose the most parsimonious one.

The first step concerns the selection of relevant variables which, in the case of a NAR(p) model, becomes the choice of the relevant lags. The procedure illustrated in the section is implemented, with an appropriate minor modification to take into account the autoregressive structure of the explicative variables. In this step, the number of hidden neurons has been fixed at one. This is not particularly influential; as we will show empirically in the illustrative application in Section 6, the choice of an optimal value d^* seems to be not much sensitive to the value fixed for r.

The second step deals with the choice of a value for the hidden layer size. As stated in Section 4.3, we implement the prediction risk estimated by subsampling.

The third step covers the problem of the omission of relevant variables. In this case, a comparison among the alternative models, presented in Section 4.2, is implemented. By the parsimony criterion, if two models exhibit the same performance, in terms of output, the one with fewer parameters is preferable.

Finally, the tests for neglected nonlinearity on the residuals from the identified optimal model, along with tests for normality, have to be performed to verify that the nonlinear structure of the data has been correctly modeled.

6. An illustrative example on simulated data

To illustrate how the proposed model selection procedure works, the results of an illustrative example of simulated data will be reported. The experimental setup is based on a dataset generated by an Exponential Autoregressive model of order 2, defined as:

$$Y_t = \left(0.5 + 0.9\exp\left(-Y_{t-1}^2\right)\right)Y_{t-1} - \left(0.8 - 1.8\exp\left(-Y_{t-1}^2\right)\right)Y_{t-2} + \varepsilon_t \qquad (20)$$

where the innovations ε_t are distributed as standard normal.

The choice of this nonlinear model is due to its great flexibility, allowing the generation of very different time series structures. Moreover, being the skeleton of the model defined by a function in the class of continuously differentiable functions, the universal approximation theorem in [30] applies. In addition, the EXPAR(2) process is geometrically ergodic, and so stochastically stable, and it is also strongly mixing with geometrically decreasing mixing coefficients [40].

To capture the nonlinear dynamical structure of the data we approximate the data-generating process by a properly chosen neural network model $NN(d, r)$ whose parameters are determined by applying the procedure reported in Algorithm 4.

Note that, although the time series is nonlinear by construction, some linearity tests have still been implemented. In particular, here we refer to Teraesvirta [41] and White [42] tests. Both of them seem well suited to our context, since they assume linearity in mean under the null, and use neural networks to derive an appropriate test statistic. Moreover, the classical Jarque-Bera test for normality has been also implemented, in order to verify if the observations can be considered as a realization of a Gaussian process. The results are shown in the first column of **Table 1** where it is evident that, as expected, all three tests reject the null hypothesis.

To select the set of variables to be tested as irrelevant, we have specified, in line 1.1 of Algorithm 4, a value $d^* = 6$, so starting with an initial tentative model $NN(6, 1)$. The plots of the derivatives for each single lag, reported in **Figure 1**, show that lags 1 and 2 can be identified as possible relevant variables.

This result seems to be confirmed by the plot of **Figure 2** where we report the values of the relevant measure.

It is worthwhile to observe that the identification of the relevant lags is not influenced by the number of units in the hidden layer. The six plots in **Figure 2**, where we have considered $r = 1, \dots, 6$, show again a clear cut between the first two lags and the others.

To confirm the exploratory identification of the first two lags as relevant, we have implemented the formal statistical test illustrated in Section 4.1 in which the subsampling has been used with an optimal block length identified as in Algorithm 2. By looking at **Figure 3**, in which the variance inflation indexes are reported, a value for b equal to 180 is reasonable.

The results of the identification procedure are reported in **Table 2**. Analyzing the p-values of the tests, estimated by subsampling, we identify a neural network model with the first two lags as relevant. Again, the decision about the number and type of relevant lags remains stable for different values of the hidden layer size.

Now we can proceed to select a proper hidden layer size by using the predictive risk measured by the mean square error of prediction. In **Figure 4** we report the distributions of the MSE estimated by subsampling, for different values of the subseries length, corresponding to different hidden layer sizes ranging from 1 to 8.

Clearly, there is no improvement in the performance by using more than 5 neurons in the hidden layer. So the optimal identified model is $NN(2, 5)$.

	Original data		**Residuals**	
Test	**Statistic**	**p-value**	**Statistic**	**p-value**
Teraesvirta	68.0038	0.0000	11.5667	0.1157
White	73.1060	0.0000	1.9058	0.3856
Jarque-Bera	84.4625	0.0000	1.1218	0.5707

Table 1.
Teraesvirta and White neural network tests for neglected nonlinearity on the original data and on the residuals from the "optimal" estimated neural network model $NN(2, 3)$ and alternative network models.

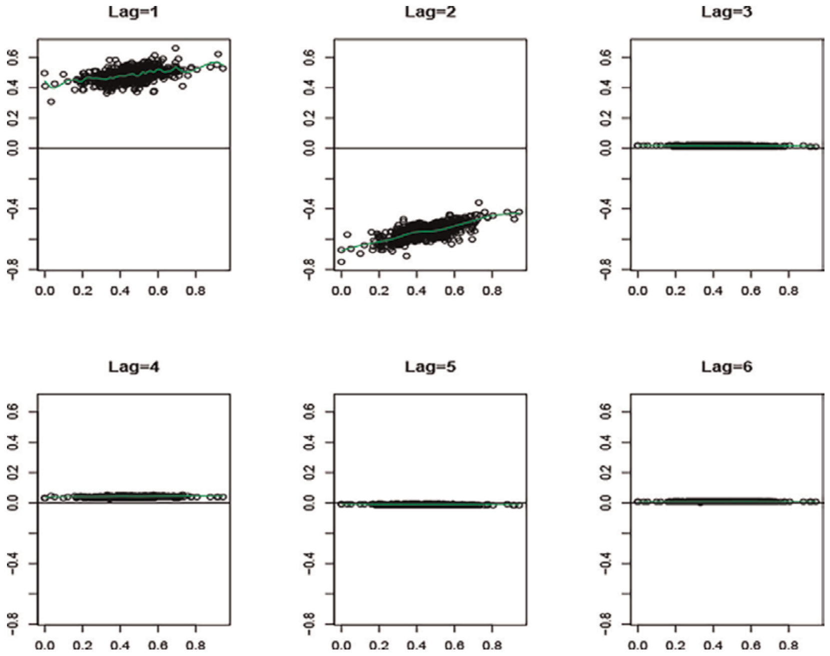


Figure 1.
Plots of the derivatives for different lags. Neural network model NN(6, 1).

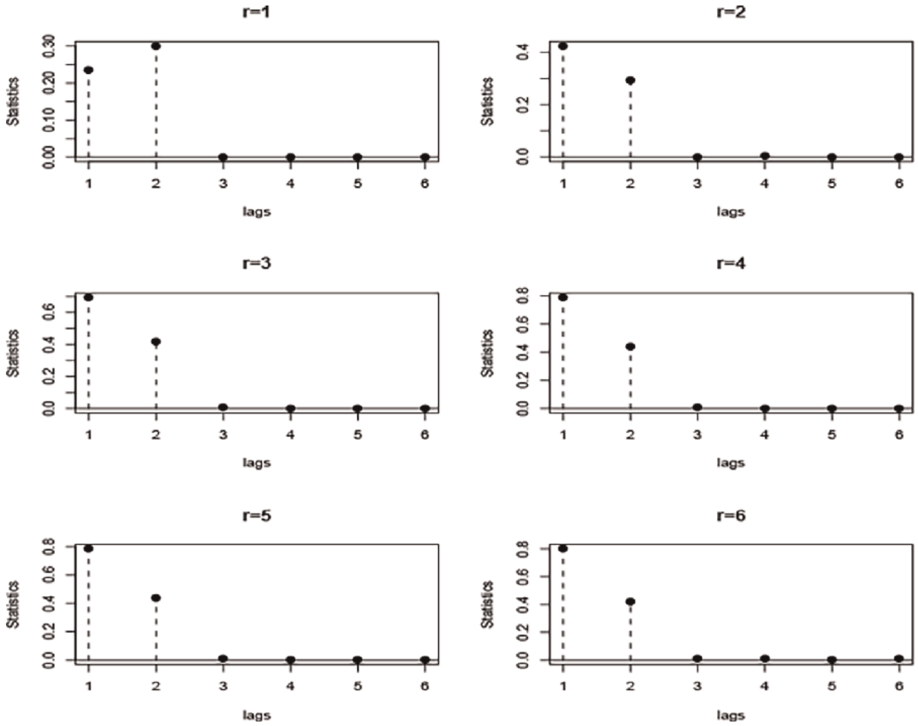


Figure 2.
Plots of the values of the relevance measure for different lags and different hidden layer sizes.

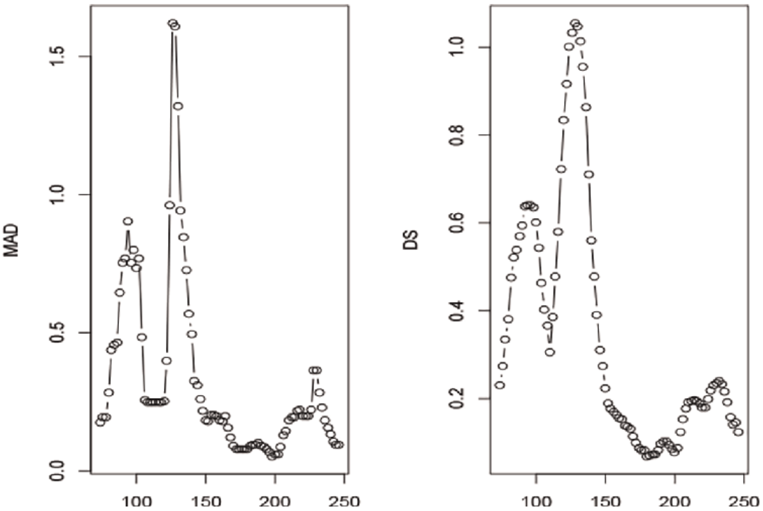


Figure 3.
Values of the VI measure defined in Algorithm 2 for different values of the block length.

r	I_0	{1}	{2}	{3}	{4}	{5}	{6}
1	Test Statistic	243.02	299.98	0.2859	1.8345	0.0842	0.0931
	p-value	*0.0000*	*0.0000*	*0.7698*	*0.2692*	*0.8453*	*0.6760*
2	Test Statistic	421.28	316.86	1.6937	6.1157	2.6152	0.1861
	p-value	*0.0000*	*0.0000*	*0.7393*	*0.2521*	*0.4604*	*0.9367*
3	Test Statistic	698.24	413.57	12.1104	2.6203	1.2834	1.0792
	p-value	*0.0000*	*0.0000*	*0.0804*	*0.6236*	*0.7540*	*0.7333*
4	Test Statistic	781.28	446.56	12.9795	3.9536	1.0705	1.3363
	p-value	*0.0000*	*0.0000*	*0.4994*	*0.7284*	*0.9622*	*0.9050*
5	Test Statistic	728.02	444.57	16.3791	3.7655	3.9889	2.8014
	p-value	*0.0000*	*0.0000*	*0.4848*	*0.8940*	*0.8758*	*0.8149*
6	Test Statistic	790.27	429.05	16.7634	10.2584	7.3156	10.9096
	p-value	*0.0000*	*0.0000*	*0.6821*	*0.6833*	*0.8161*	*0.7150*

Table 2.
Values of the test statistic and p-values (in italic) for different variables set $\mathcal{X}_0$, (Subseries length $b = 180$).

To evaluate if the model selection strategy has been able to correctly identify a neural network model for the data, the neural network tests for neglected nonlinearity by Teraesvirta and White on the residuals from the identified optimal model, along with the Jarque-Bera test for normality, are reported in **Table 1**. Clearly, the tests do not refuse the null and so the nonlinear structure of the data seems to be correctly modeled. Moreover, the residuals can be considered as the realization of a Gaussian process.

To check if there are omitted variables in the identified model, we also compared neural models with different lag structures. The results are reported in **Table 3**.

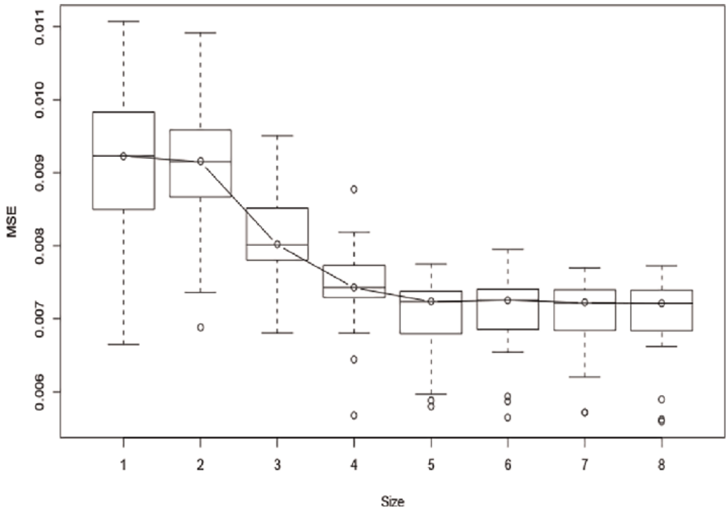


Figure 4.
Boxplots of the distributions of the predictive accuracy measure for different values of the subseries length. Each distribution refers to a given hidden layer size.

Model	$\mathrm{NN}(1,5)$	$\mathrm{NN}(2,5)$
$NN(2,5)$	5.1924 (0.0000)	—
$NN(3,5)$	5.2106 (0.0000)	0.0617 (0.3801)
$NN(4,5)$	5.0483 (0.0000)	0.0724 (0.8478)
$NN(5,5)$	5.2799 (0.0000)	0.1349 (0.6529)
$NN(6,5)$	5.2909 (0.0000)	0.1599 (0.8977)

Table 3.
Values of the test statistics and p-values (estimated by subsampling) in parenthesis for the hypotheses of equivalence of neural network models.

Clearly, neural network models with more than two neurons in the input layer seem to be equivalent to the chosen one. Therefore, there are no omitted lags. Observe that, on the contrary, if we had used a network including just one input neuron (corresponding to the first lag) there would have been better models obtained by including more lags.

7. Concluding remarks

In this chapter, a neural network model selection strategy has been reviewed and discussed. It highlights the different roles of the input variables and the hidden layer size. The former are selected using testing procedures based on the definition of a measure for the variable relevance to the model which allows to evaluate the inclusion of irrelevant variables as well as the omission of the relevant ones. The hidden layer size is determined by using a measure of the predictive risk. The whole procedure uses subsampling to obtain the distribution of the involved statistics. The procedure has been applied to an illustrative example of simulated data. It seems to be able to detect correctly the set of input variables and a proper hidden layer size. Clearly, joint usage of neural network models and subsampling is usually quite demanding from a computational point of view. In any case, once fixed the subseries length, the proposed procedure takes just a few seconds on a desktop PC.

Author details

Michele La Rocca† and Cira Perna*†
Department of Economics and Statistics, University of Salerno, Fisciano, SA, Italy

*Address all correspondence to: perna@unisa.it

† These authors contributed equally.

References

[1] Zhang G, Patuwo BE, Hu MY. Forecasting with artificial neural networks: The state of the art. International Journal of Forecasting. 1998;**14**(1):35-62

[2] Cybenko G. Approximation by superpositions of a sigmoidal function. Mathematics of Control, Signals and Systems. 1989;**2**(4):303-314

[3] Hornik K. Some new results on neural network approximation. Neural Networks. 1993;**6**(8):1069-1072

[4] Poggio T, Mhaskar H, Rosasco L, Miranda B, Liao Q. Why and when can deep-but not shallow-networks avoid the curse of dimensionality: A review. International Journal of Automation and Computing. 2017;**14**(5):503-519

[5] Blalock D, Gonzalez Ortiz J, Frankle J, Guttag J. What is the state of neural network pruning? Proceedings of Machine Learning and Systems. 2020;**2**: 129-146

[6] Reed R. Pruning algorithms-a survey. IEEE Transactions on Neural Networks. 1993;**4**(5):740-747

[7] Tessier H. Neural network pruning 101: All you need to know not to get lost Towards Data Science. 2021. Available from: https://towardsdatascience.com/neural-network-pruning-101-af816aaea61

[8] Prechelt L. Early stopping-but when? In: Orr GB, Miller K-R, editors. Neural Networks: Tricks of the Trade. Berlin, Heidelberg: Springer; 2002. pp. 55-69

[9] Cottrell M, Girard B, Girard Y, Mangeas M, Muller C. Neural modeling for time series: A statistical stepwise method for weight elimination. IEEE Transactions on Neural Networks. 1995; **6**(6):1355-1364

[10] Anders U, Korn O. Model selection in neural networks. Neural Networks. 1999;**12**:309-323

[11] Panchal G, Ganatra A, Kosta YP, Panchal D. Searching most efficient neural network architecture using Akaike's information criterion (AIC). International Journal of Computer Applications. 2010;**1**(5):41-44

[12] Qi M, Zhang G. An investigation of model selection criteria for neural network time series forecasting. European Journal of Operational Research. 2001;**132**(3):666-680

[13] La Rocca M, Perna C. Resampling techniques and neural networks: some recent developments for model selection. In: Atti della XLIII Riunione Scientifica SIS; 14–16 Giugno 2006. Torino(IT); 2006. pp. 14-16

[14] La Rocca M, Perna C. Modelling complex structures by artificial neural networks. In: Proceedings of Knowledge Extraction and Modelling (KNEMO06), IASC INTERFACE IFCS Workshop; 4–6 September 2003. Capri(IT); 2003. pp. 4-6

[15] La Rocca M, Perna C. Variable selection in neural network regression models with dependent data: A subsampling approach. Computational Statistics & Data Analysis. 2005;**48**(2): 415-429

[16] La Rocca M, Perna C. Neural network modeling by subsampling. In: Cabestany J, Prieto A, Sandoval F, editors. Computational Intelligence and Bioinspired Systems. Berlin, Heidelberg: Springer; 2005. pp. 200-207

[17] La Rocca M, Perna C. A multiple testing procedure for input variable selection in neural networks. In: Proceedings of the 13th European Symposium on Artificial Neural Networks (ESANN2005); 27–29 April 2005. Bruges, Belgium; 2005. pp. 173-178

[18] La Rocca M, Perna C. Neural network modelling with applications to euro exchange rates. In: Kontoghiorghes E, Rustem B, Winker P, editors. Computational Methods in Financial Engineering: Essays in Honour of Manfred Gilli. Springer: Berlin-Heidelberg; 2008. pp. 163-189

[19] La Rocca M, Perna C. A two-step procedure for neural network modeling. Quaderni di Statistica. 2012;**14**:145-148

[20] La Rocca M, Perna C. Designing neural networks for modeling biological data: A statistical perspective. Mathematical Biosciences & Engineering. 2013;**11**(2):331-342

[21] La Rocca M, Perna C. Model selection for neural network models: A statistical perspective. In: Dehmer M, Emmert-Streib F, Pickl S, editors. Computational Network Theory: Theoretical Foundations and Applications. New Jersey: Wiley-Blackwell; 2015. pp. 1-27

[22] La Rocca M, Perna C. Opening the black box: Bootstrapping sensitivity measures in neural networks for interpretable machine learning. Stat. 2022;**5**(2):1-18

[23] Politis DN, Romano JP, Wolf M. Subsampling. New York: Springer Science & Business Media; 1999. p. 347

[24] Fildes R. A new bootstrapped hybrid artificial neural network approach for time series forecasting. Computational Economics. 2020;**2020**:1355-1383

[25] Bertail P, Politis DN, Romano JP. On subsampling estimators with unknown rate of convergence. Journal of the American Statistical Association. 1999; **94**(446):569-579

[26] White H. Estimation, Inference and Specification Analysis. Cambridge: Cambridge University Press; 1996. p. 380

[27] Politis DN, Romano JP, Wolf M. Subsampling for heteroskedastic time series. Journal of Econometrics. 1997; **81**(2):281-317

[28] Bertail P, Haefke C, Politis DN, White H. Subsampling the distribution of diverging statistics with applications to finance. Journal of Econometrics. 2004;**120**(2):295-326

[29] Hornik K, Stinchcombe M, White H, Auer P. Degree of approximation results for feedforward networks approximating unknown mappings and their derivatives. Neural Computation. 1994;**6**(6):1262-1275

[30] Barron AR. Universal approximation bounds for superpositions of a sigmoidal function. IEEE Transactions on Information Theory. 1993;**39**(3):930-945

[31] Ossen A, Rüger SM. An analysis of the metric structure of the weight space of feedforward networks and its application to time series modeling and prediction. In: Proceedings of the 4th European Symposium on Artificial Neural Networks (ESANN1996); 24–26 April 1996. Bruges, Belgium; 1996. pp. 315-322

[32] Baxt WG, White H. Bootstrapping confidence intervals for clinical input variable effects in a network trained to identify the presence of acute myocardial infarction. Neural Computation. 1995;7(3):624-638

[33] White H, Racine J. Statistical inference, the bootstrap, and neural-network modeling with application to foreign exchange rates. IEEE Transactions on Neural Networks. 2001; **12**(4):657-673

[34] Giordano F, La Rocca M, Perna C. Input variable selection in neural network models. Communications in Statistics-Theory and Methods. 2014; **43**(4):735-750

[35] Stathakis D. How many hidden layers and nodes? International Journal of Remote Sensing. 2009;**30**(8): 2133-2147

[36] Gao P, Chen C, Qin S. An optimization method of hidden nodes for neural network. In: 2010 Second International Workshop on Education Technology and Computer Science; 6–7 March 2010. Wuhan, China; 2010. pp. 53-56

[37] Bergmeir C, Benítez JM. On the use of cross-validation for time series predictor evaluation. Information Sciences. 2012;**191**:192-213

[38] Bergmeir C, Hyndman RJ, Koo B. A note on the validity of cross-validation for evaluating autoregressive time series prediction. Computational Statistics & Data Analysis. 2018;**120**:70-83

[39] Fukuchi JI. Subsampling and model selection in time series analysis. Biometrika. 1999;**86**(3):591-604

[40] Györfi L, Härdle W, Sarda P, Vieu P. Nonparametric Curve Estimation from Time Series. Vol. 60. Berlin: Springer; 2013

[41] Teräsvirta T, Lin CF, Granger CW. Power of the neural network linearity test. Journal of Time Series Analysis. 1993;**14**(2):209-220

[42] Lee TH, White H, Granger CW. Testing for neglected nonlinearity in time series models: A comparison of neural network methods and alternative tests. Journal of Econometrics. 1993; **56**(3):269-290

Section 4

Environmental Time Series Analysis

Chapter 9

Analysis and Modeling of Rainfed Crops Dynamics Based on NDVI Time Series in Central Spain

César Sáenz, Víctor Cicuéndez, Laura Recuero, Klaus Wiese, Alicia Palacios-Orueta and Javier Litago

Abstract

Rainfed crops occupy 76% of the cultivated area of Spain being distributed throughout the whole country. The yield of these crops depends on the great interannual variability of meteorological factors. The monitoring and prediction of crop dynamics is a key factor for their sustainable management from an environmental and socioeconomic point of view. Long time series of remote sensing data, such as spectral indices, allow monitoring vegetation dynamics at different spatial and temporal scales and provide valuable information to predict these dynamics through time series analysis. The objectives of this study are as follows: (1) To assess the dynamics of rainfed crops in a typical dryland area of Spain and (2) to build dynamic models to explain and predict the evolution of these crops. The NDVI time series of a rainfed cereal crop area of central Spain have been analyzed using statistical time series methods and their values were predicted using the Box-Jenkins approach. At the model identification stage, the evaluation of their autocorrelation functions, periodogram, and stationarity tests has revealed that most of these series are stationary and that their dynamics are dominated by annual seasonality. The selected preliminary dynamic model presents a good degree of adjustment for a 30% of the studied pixels.

Keywords: MODIS, remote sensing, Box-Jenkins, phenology, forecast, dynamic models

1. Introduction

Rainfed agriculture plays a crucial role directly or indirectly in global food supply, occupying 80% of the world's agricultural land and generating 60% of the total agricultural production [1, 2]. Rainfed crops, which depend mainly on precipitation for their growth, are vital to meet the food supply, especially in water-limited areas. The increasing global demand for food products driven by population growth, coupled with the climate change effects on food production, is increasing the concern about food security across countries [3, 4]. Climate change poses a significant threat to rainfed agricultural areas, as changing rainfall patterns, prolonged droughts, and more frequent extreme weather events may significantly reduce crop yields [5, 6]. In this context, the EU Green Deal recognizes the importance of preserving agricultural lands by implementing sustainable farming practices to deal with climate change and the loss of biodiversity, investing in research and innovation and promoting climate-resilient farming techniques [7, 8]. In this regard, monitoring rainfed agricultural areas and their productivity is essential to apply the most suitable management practices according to climate conditions.

In the last decades, satellite imagery has played a crucial role in monitoring rainfed crops, facilitating informed decision-making and contributing to the optimization of agricultural production. Particularly, remote sensing time series enable the assessment of crop dynamics, providing valuable information about past and current crop status and helping to monitor the growth and development of crops and to identify areas affected by water deficit, diseases, or pests [9–11]. This information is vital for implementing efficient management strategies, applying targeted phytosanitary treatments, and anticipating potential yield losses. In this regard, the United Nations is promoting the use of remote sensing technologies to monitor the implementation of the Sustainable Development Goals, such as "zero hunger," which is strongly related to the agricultural sector [12, 13].

The length and data frequency of remote sensing time series is fundamental to monitoring agricultural areas accurately. The Moderate Resolution Imaging Spectroradiometer (MODIS) images have been widely used to monitor large areas due to their high temporal resolution and the possibility to generate long time series of vegetation indices to assess crop dynamics [9, 14, 15].

The approach most widely extended is the use of phenometrics, which are the dates of specific phenological events derived from functions fitted to the time series of vegetation indices such as the Normalized Difference of Vegetation Index (NDVI) or Enhance Vegetation Index (EVI). The start, end, peak, and length of the growing season were used to classify crops [16], to evaluate crop calendars [17–19], and to map cropping intensity [20, 21] and temporal trends [22]. The main disadvantage of this approach lies in setting threshold values in time series with high variability, as in the case of drylands, which are highly dependent on climate variations. In addition, observations outside the specific phenological events are not considered, not exploiting the entire time series content [23].

Conversely, statistical time series analysis (TSA) in its temporal and frequency domains [24] enables the assessment of crop dynamics by considering all the observations and without setting threshold values. This approach provides several tools and methods to understand, model, and predict a variable by quantitatively identifying the temporal patterns based on its history. Its application to remote sensing data has constituted a significant progress when monitoring environmental variables allowing the analysis of extensive areas by considering the historical data of each pixel and the spatial coherence of the imagery [25, 26]. Nevertheless, implementing TSA methodologies at the pixel level poses a significant challenge due to the complexity of certain models and the high amount of data involved.

One of the most widely used techniques within the TSA approach is the autoregressive integrated moving average (ARIMA) models introduced in the seventies by Box and Jenkins [27] in which time series are modeled as a stationary stochastic process. These models have been applied to predict area and production of crops with different information sources [28, 29]. Although ARIMA models have been employed with remote sensing time series [30], the multiplicative seasonal autoregressive integrated moving average (SARIMA) models are commonly used due to the high seasonal component when using these data [31–33].

The conventional Box-Jenkins methodology is commonly employed through five stages: identification, estimation, validation, forecasting, and evaluation. Specifically, the identification step is typically time-consuming due to the need for visual inspection, posing challenges when applied to a significant number of pixels. Xiao et al. [31] obtained low accurate results when forecasting Leaf Area Index (LAI) using an estimated SARIMA model with the same coefficient for all MODIS time series (i.e., pixels). In contrast, Han et al. [34] obtained high accurate results when predicting Vegetation Temperature Condition Index (VTCI) at pixel level using an AR (1) model with specific coefficients for each time series. Other authors also found accurate predictions with the same approach using SARIMA models to forecast MODIS LAI time series [33] and NDVI time series from AVHRR sensor [32].

The use of Box-Jenkins approach with remote sensing data has already been applied over agricultural areas. Alhamad et al. [35] used ARIMA models to predict forage production with the aim of helping range managers when making stocking decisions operatively. Gonçalves et al. [36] used a SARIMA model applied to an agroclimatic index and NDVI time series to monitor sugarcane fields in Brazil. Tian et al. [37] constructed ARIMA models using VTCI time series from AVHRR sensor to forecast drought across croplands in the Guanzhong Plain of China. More recently, Bounouh et al. [38] applied ARIMA models to predict phenometrics from NDVI time series. Carreño-Conde et al. [39] proposed the combined use of Double Exponential Smoothing and ARIMA modeling approach with explanatory variables (ARIMAX) to better predict NDVI values for monitoring crop dynamics. Nevertheless, implementing TSA methodologies at the pixel level poses a significant challenge due to the complexity of certain models and the high amount of data involved [40].

More research is needed to characterize and model rainfed crop dynamics to predict their yields and ensure food supplies. Therefore, the aim of this research is to explain and predict the evolution of rainfed crops in central Spain, building a model using the Box-Jenkins approach applied to MODIS NDVI time series from 2002 to 2022.

2. Study area

The study area includes four representative regions of the drylands cropping systems in Spain in the province of Soria (**Figure 1**). Each area has a size of 314.16 km^2, and they were built around a meteorological station that belongs to the red SiAR [41]. The four stations are located in municipalities of Almazán, Hinojosa del Campo, Fuentecantos, and San Esteban de Gormaz. These regions are characterized by a Csb climate according to Köppen-Geiger classification with hot and dry summers and cold winters [42].

Table 1 shows the differences between sites in terms of precipitation and temperature annual values (study period 2002–2023) and the main soils of the four regions. The mean annual temperature ranges from 9.89 to 11.11°C and annual precipitation from 1078 to 1332 mm.

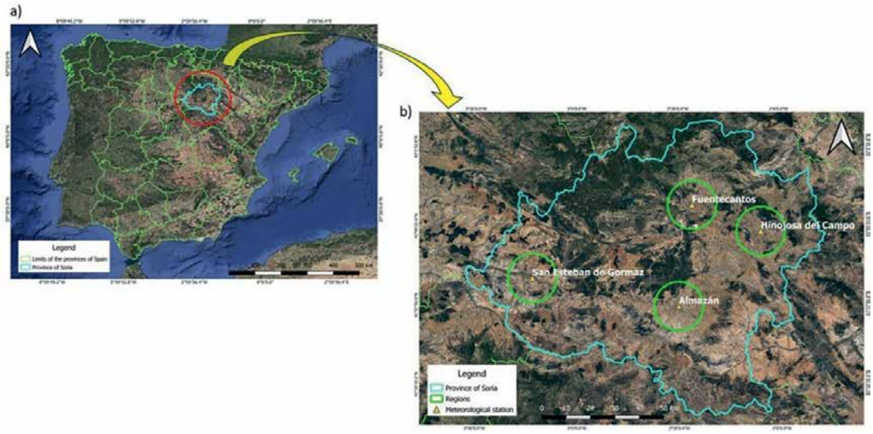


Figure 1.
Study area: Location of Soria in Spain (a) and a zoom into Soria (b) with the four studied regions: Region 1: San Esteban de Gormaz, Region 2: Fuentecantos, Region 3: Hinojosa del Campo, and Region 4: Almazán.

Region	Name	Mean annual temperature (°C)	Annual precipitation (mm)	Soil
Region 1	San Esteban de Gormaz	11.11	1212	Fluvisol/Cambisol
Region 2	Fuentecantos	9.89	1332	Cambisol
Region 3	Hinojosa del Campo	10.70	1189	Cambisol
Region 4	Almazán	10.68	1078	Cambisol

Table 1.
Meteorological and soil data for the four studied regions.

3. Materials

3.1 Remote sensing data

The remote sensing imagery used in the study consists of 1046 eight-day MODIS composites for the tile h17v04, spanning from 18th February 2000 to 9th November 2022. Satellite images were obtained from the MODIS MOD09A1 version 6.1 surface [43] reflectance product, available through the NASA Distributed Active Archive Centre, DAAC (https://lpdaac.usgs.gov/). This product consists of seven surface reflectance bands in the visible (VIS), near-infrared (NIR), and shortwave infrared (SWIR) regions at 500 m of spatial resolution. Specifically, the bands are centered around the following wavelengths: 470 nm (blue), 555 nm (green), 645 nm (red), 858 nm (NIR), 1240 nm (SWIR1), 1640 nm (SWIR2), and 2130 nm (SWIR3).

3.2 CORINE Land Cover (CLC) 2018

CLC2018 was used to delimit rainfed arable lands in Soria province using the class with the code 211 “non-irrigated arable lands” obtaining a total surface of 455.400 hectares (18,216 pixels). This dataset offers consistent and detailed information on land cover and land cover changes across Europe, being produced in the frame of the Copernicus Land Monitoring Service and coordinated by the European Environment Agency (EEA). The minimum mapping unit is 25 hectares for status layers and 5 hectares for land cover changes [44]. It was downloaded in the shapefile format from the Spanish platform for open datasets (https://datos.gob.es/es). Polygons were rasterized to 500 meters of spatial resolution to match with MODIS imagery and used to mask the other land cover classes and remove them from the analysis.

4. Methods

4.1 NDVI time series generation

MODIS imagery was downloaded from the NASA Distributed Active Archive Centre, DAAC, and re-projected to the UTM Zone 30 N WGS84 coordinate system. NDVI values for each composite were calculated [45] (Eq. (1)):

$$\mathrm{NDVI} = \frac{(NIR - \mathrm{R})}{(NIR + \mathrm{R})} \tag{1}$$

Where R and NIR represent pixel reflectance values in the red and near-infrared bands, respectively.

NDVI time series were generated and smoothed in order to avoid anomalous values. First, observations that fell outside the threshold of the mean plus/minus twice the standard deviation within a five-date period window were considered as outliers and replaced by the average of the previous and subsequent observations in the time series. Then, a Savitzky-Golay filter [46] was applied with a smoothing window width of 9 and a degree of the smoothing polynomial of 2 [47].

4.2 Subseting NDVI time series

The regions of interest (ROIs) were selected based on the location of four meteorological stations and delimited by intersecting circular surfaces with 10 km of radius around these stations and the non-irrigated arable lands identified with CLC 2018 dataset. As a result, a total of 2584 time series (i.e., pixels) were analyzed.

4.3 NDVI time series modeling

NDVI time series dynamics reflect vegetation conditions driven by climate and other factors resulting in a significant seasonality with an annual pattern repeated every 46 observations based on the frequency of MODIS data acquisition, which is 8 days. The presence of this seasonality is the main reason for non-stationarity in the NDVI time series.

Data-driven modeling based on Statistical Time Series Analysis in the frequential and temporal domains [24, 48] has been used to model and forecast NDVI time series. The procedure has been implemented using the original NDVI time series in the following stages [24] (**Figure 2**):

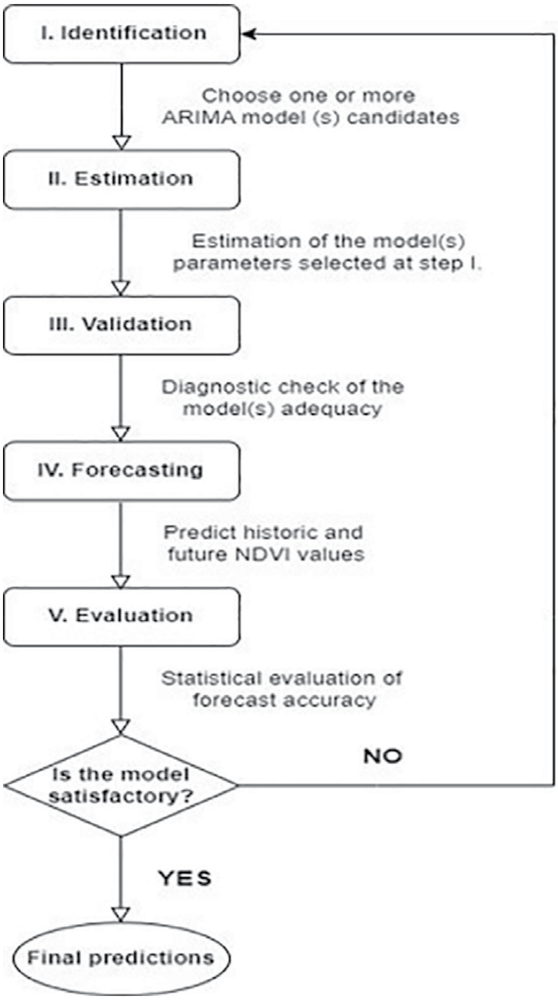


Figure 2.
Flow chart of the Box and Jenkins methodology.

4.3.1 Identification

In this step, NDVI time series dynamics were analyzed to identify their characteristic components such as seasonality, cycles, trends, or structural changes, among others. With this information, a specific model was proposed for each series. The identification stage was made at pixel level through the following stages:

I.1. Preliminary statistics analysis.
I.2. Estimation and analysis of the autocorrelation functions (regular and partial).
I.3. Estimation and analysis of the periodogram and use of white noise tests such as Fisher's Kappa (Fk) and Barlett's Kolgomoroff–Smirnoff (BKS) tests [49] to detect significant periodic components.

I.4. The regular and seasonal autoregressive and moving-average parameters were specified based on the preceding identification results to capture the dynamics of each series effectively.

4.3.2 Estimation

Nonlinear least squares methods were used to estimate the models specified in the previous stage, while the Akaike and Schwarz Information Criteria were employed to select the most suitable model [48]. The Student-t and F tests were used to evaluate the significance of the model parameters.

4.3.3 Validation

The NDVI variable was predicted using the selected models in the drylands of Soria. The adequacy of the estimated models was assessed using the autocorrelation in the model residuals through the Ljung-Box Q statistic [50]. If the test reveals that a substantial amount of residual autocorrelation persists in the estimated models, the model is considered invalid, and returning to the Identification step is necessary. Q Ljung-Box test [50] estimations were done for several lags within a logical temporal horizon (92 lags), given that the NDVI temporal frequency is 46 observations per year.

4.3.4 Forecasting

Historical and future NDVI values were predicted using the validated models.

4.3.5 Prediction evaluation

To assess the predictive capacity of the models, we used Theil's U inequality coefficient [51]. This coefficient is invariant to the measurement scale and ranges between 0 and 1. If U = 0, the prediction is perfect; if U = 1, the prediction is naïve (Eq. (2)). Theil's U inequality coefficient helps us also to detect the source of the prediction error. It is broken down into three proportions: (1) The bias proportion (U^B), which tells us how far the mean of the prediction is from the mean of the observed series; (2) the variance ratio (U^V), which tells us how far the predicted variance is from the variance of the observed series; and (3) the proportion of covariance (U^C), which measures the remaining and unsystematic errors of prediction. The sum of the 3 proportions is equal to 1. For a good prediction, the proportions of bias and variance should be close to zero, and most of the error should be concentrated in the proportion of covariance.

$$U = \frac{\left[\sum_{i=1}^{n} (F_i - O_i)^2\right]^{1/2}}{\left[\sum_{i=1}^{n} (O_i)^2\right]^{1/2}} \qquad (2)$$

Where:
U = Theil's U inequality coefficient.
F_i = Forecasted variable.

O_i = Observed variable.

n = number of observations.

Statistical analyses were conducted using SAS 9.2, while image processing was performed using ENVI 4.7.

5. Results

5.1 Identification

The identification of the NDVI series has been carried out using the Box-Jenkins methodology in the selected 2584 pixels of Soria in which rainfed agriculture is practiced.

Figures 3 and **4** show one selected pixel as an example of the identification stage with the original time series, the autocorrelation functions, and the periodograms.

A high temporal and spatial variability in the NDVI dynamics in these drylands has been identified, which makes it difficult to estimate models with sufficient statistical adequacy and high spatial representativeness. The analysis of the autocorrelation functions, the periodogram, and the augmented Dickey-Fuller test [52] rejected the presence of trends in most of these series.

The most common dynamic characteristic of all NDVI-studied time series was seasonality; however, this component presents a great spatial variability and a high temporal irregularity, which makes it especially difficult to incorporate it into a statistically significant and robust predictive model. In most of the time series, a dominant annual cycle was observed in period 46, with the highest ordinate, together with the presence of one or two secondary peaks of lower periodicity 23 or 14.33. Fisher's Kappa (Fk) and Barlett's Kolgomoroff-Smirnoff (BKS) tests [49] were used to detect significant periodic components.

It has also been detected in a high number of pixels that NDVI presents a significant conditional heteroscedasticity, which needs to be adequately treated, and, in many cases, 'ad hoc' in the specification and estimation of the models. To try to minimize the influence of this variance, the natural logarithm transformation has been used for all the NDVI time series to build the models.

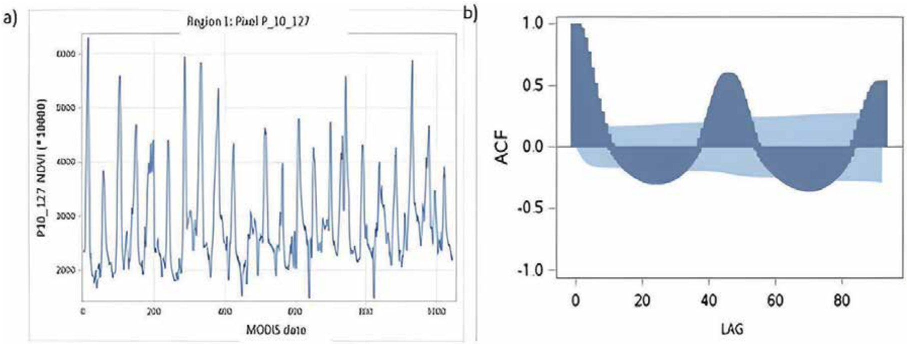


Figure 3.
Original time series (a) and autocorrelation function (ACF) (b) of pixel P_10_27 of Region 1: San Esteban de Gormaz.

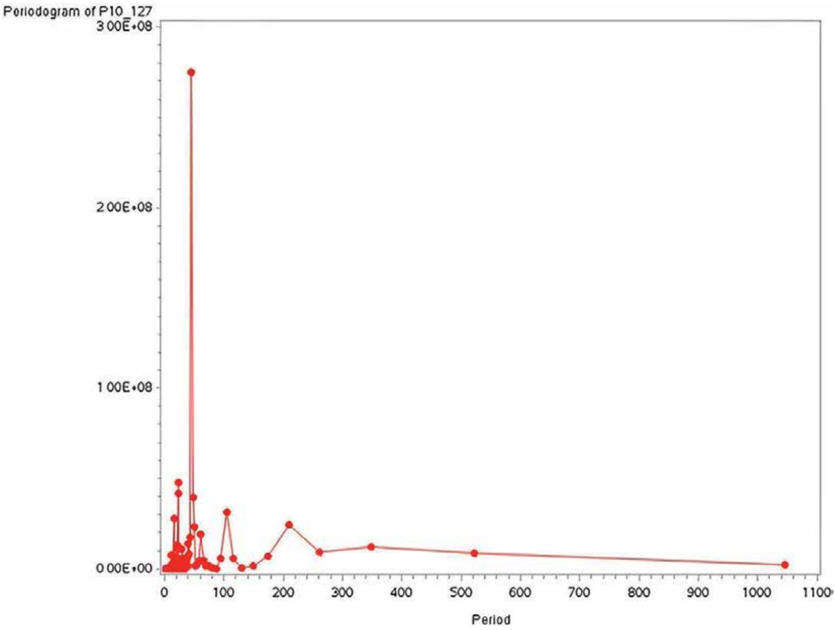


Figure 4.
Periodogram of pixel P_10_27 of Region 1: San Esteban de Gormaz.

5.2 Estimation

A great number of preliminary models were estimated, and between them, a small group of them were improved in terms of specification and estimation techniques. The final model with better statistical signification and quality and spatial representativeness (the one that adjusts correctly for a larger number of pixels) was considered to have a better performance in the study area.

A SARMA model was selected in which there were 15 autoregressive terms (AR) in the short, medium, and long term; another seasonal autoregressive term in lag 42 (SAR); 4 moving average terms in the medium term (MA); and an independent term (**Table 2**). All the estimated parameters show a very high statistical significance in the Student's t-test, except in some cases in which between two and five terms were not significative. It must be highlighted that the SAR term was not significative in a lot of cases, and it was the most difficult term to adjust to all the pixels. The model reached the minimization of the information criteria of Akaike and Schwarz (AIC and SBC). An example of the SARMA model applied to pixel P2_879 of Region 2 (Fuentecantos) is shown in **Table 2**.

5.3 Validation

This model was implemented to the 2584 selected pixels. According to this test, pixels have been divided into three levels (**Figure 5**). In level 1, there were the pixels that showed low Q Ljung-Box test values at short and especially at medium and at long terms. In level three, there were the pixels that showed high Q Ljung-Box test values at short, medium, and/or long term. Therefore, the models of these pixels could be considered invalid. In level 2, there were the pixels classified as an intermediate situation; thus,

Parameter	Coefficient	Std. Error	T-test	Prob > T	Lag
MU	7.961	0.079	100.4	<0.0001	0
MA 1,1	0.096	0.041	2.33	0.020	16
MA 1,2	−0.302	0.039	−7.8	<0.0001	17
MA 1,3	0.023	0.096	0.24	0.813	42
MA 1,4	0.010	0.03	0.31	0.754	43
AR 1,1	1.649	0.029	56.59	<0.0001	1
AR 1,2	−0.609	0.051	−12.04	<0.0001	2
AR 1,3	−0.246	0.049	−4.99	<0.0001	3
AR 1,4	0.117	0.042	2.76	0.006	4
AR 1,5	0.005	0.029	0.17	0.866	6
AR 1,6	0.498	0.045	11.15	<0.0001	8
AR 1,7	−0.995	0.061	−16.41	<0.0001	9
AR 1,8	0.612	0.05	12.24	<0.0001	10
AR 1,9	−0.007	0.025	−2.84	0.005	12
AR 1,10	−0.054	0.022	−2.49	0.013	15
AR 1,11	0.043	0.016	2.75	0.006	17
AR 1,12	−0.012	0.005	−2.31	0.021	25
AR 1,13	0.035	0.006	5.77	<0.0001	44
AR 1,14	−0.018	0.006	−2.86	0.004	51
AR 1,15	0.03	0.005	5.85	<0.0001	85
AR 2,1	0.034	0.101	0.34	0.735	42

The parameters which are not significant are shown in yellow.

Table 2.
Selected SARMA model applied to pixel P2_879 of Region 2 (Fuentecantos).

the models were not invalid, but they showed higher Q Ljung-Box test values than those of level 1. **Figure 5** shows the partial autocorrelation function (PACF) and white noise probability according to the Q Ljung-Box test of three selected pixels classified as level 1 (a), level 2 (b), and level 3 (c). In level 1, the PACF of the residuals was low and between the intervals of confidence. Likewise, the probability that the residuals of the models are white noise decreases as the lag order increases, being below 5% over the entire horizon. Therefore, the Q statistic is lower than the critical values of a Chi2 at 5% for all lags (except lag 20), and the null hypothesis that the residuals of the estimated model are white noise cannot be rejected. In level 3, the opposite situation could be observed.

Figure 6 showed the classification and distribution of the studied pixels into the three levels according to Q Ljung-Box test:

As **Table 3** shows, a total of 776 pixels were classified into level 1; thus, the model showed a good adequacy for 30% of the studied pixels. The spatial distribution of the three classes showed coherence with level 2 pixels between those with level 1 and 3. Specifically, the proportion of the pixels with level 1 class is higher in regions 1 and 3 than in regions 2 and 4, with most of the pixels being in the border areas of non-irrigated arable lands.

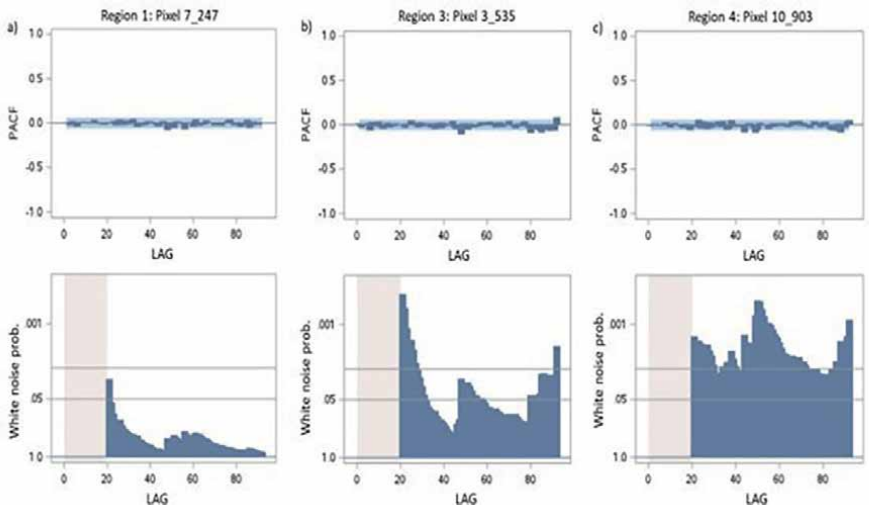


Figure 5.
Partial autocorrelation function (PACF) and white noise probability according to the Q Ljung-Box test of three selected pixels classified as level 1 (a), level 2 (b), and level 3 (c).

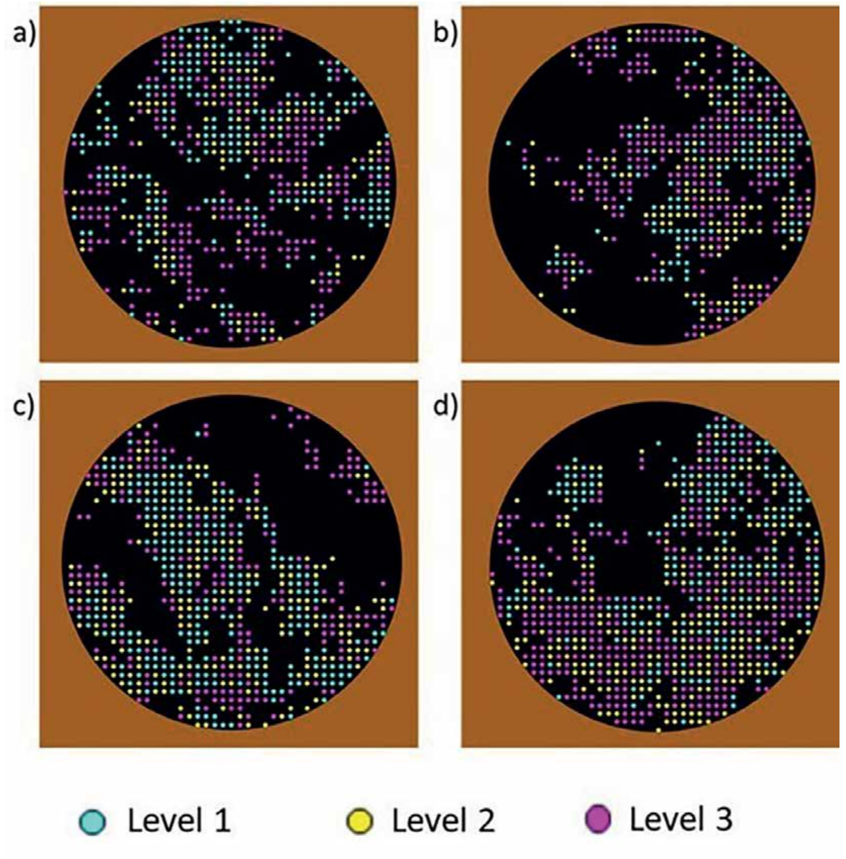


Figure 6.
Classification and distribution into three levels of the 2584 studied pixels: (a) Region 1: San Esteban de Gormaz, (b) Region 2: Fuentecantos, (c) Region 3: Hinojosa del Campo, and d) Region 4: Almazán.

Level	Region 1	Region 2	Region 3	Region 4	Sum	%
1	211	113	247	205	776	30.03
2	127	141	195	254	717	27.75
3	275	263	216	337	1091	42.22
Sum	613	517	658	796	2584	100

Table 3.
Number and percentage of the different pixel levels for the four selected regions.

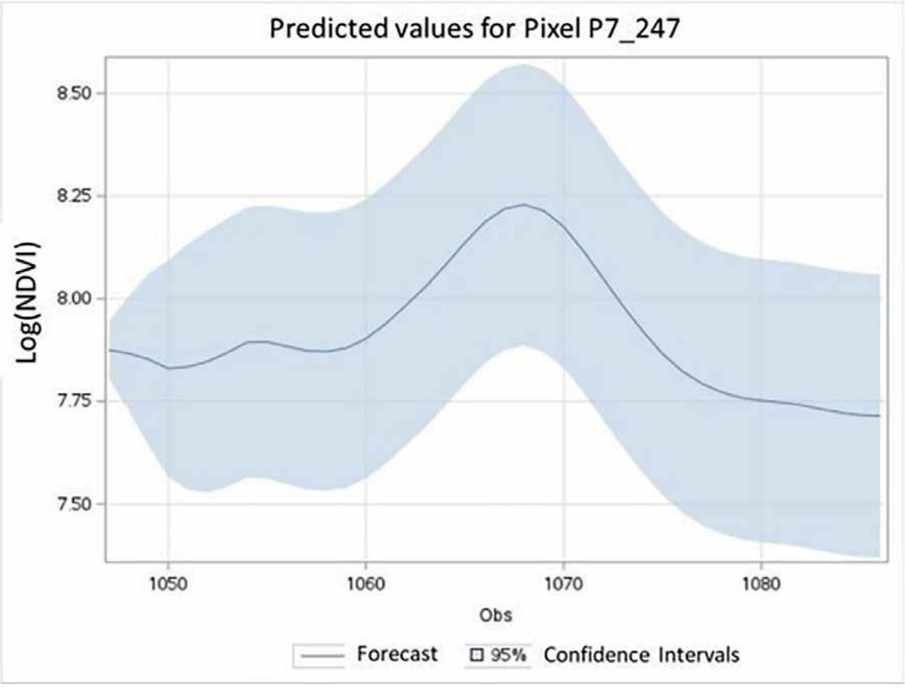


Figure 7.
Predicted 46 values for the Log(NDVI) for the pixel P7_247 of Region 1 (San Esteban de Gormaz) classified as level 1.

5.4 Forecasting

Historical values for the 1046 MODIS dates were predicted using the validated models. Future NDVI values were predicted for 46 MODIS dates (one year). **Figure 7** shows the predicted 46 values for the Log(NDVI) for the pixel P7_247 of Region 1 classified as level 1. As it can be observed, the predictions were accurate, and they were inside the intervals of confidence at 95%.

5.5 Prediction evaluation

Table 4 shows the accuracy of the 16 model forecasts using Theil's U inequality coefficient, which is broken down into three proportions: (1) The bias proportion

Region	Pixel	Level	U THEIL	U^B	U^V	U^C
1	P10_133	1	0.00516	1.49572E−05	0.00410	0.99589
	P7_247	1	0.00447	1.83618E−05	0.00427	0.99572
	P10_118	2	0.00571	2.77866E−05	0.00348	0.99649
	P11_894	2	0.00375	4.65361E−05	0.00503	0.99492
2	P2_880	1	0.00536	2.61424E-05	0.00359	0.99639
	P3_776	1	0.00529	2.23460E-05	0.00559	0.99438
	P2_936	2	0.00443	7.53839E-06	0.00278	0.99721
	P1_939	2	0.00544	1.51601E-05	0.00384	0.99614
3	P3_803	1	0.00637	2.36657E-05	0.00543	0.99454
	P3_996	1	0.01006	3.12941E-06	0.01324	0.98676
	P3_535	2	0.00590	5.99829E-05	0.00407	0.99587
	P3_536	2	0.00647	4.80825E-05	0.00569	0.99426
4	P11_826	1	0.00651	7.32773E-07	0.00474	0.99525
	P11_980	1	0.00789	3.28687E-05	0.00573	0.99423
	P11_362	2	0.00432	1.89298E-05	0.00452	0.99546
	P11_457	2	0.00393	4.48277E-05	0.01533	0.98462

Table 4.
Accuracy of the model forecasts for 16 pixels of level 1 and level 2 using the Theil's U inequality coefficient which is broken down into three proportions: (1) The bias proportion (U^B), (2) The variance ratio (U^V), (3) The proportion of covariance (U^C).

(U^B), (2) the variance ratio (U^V), and (3) the proportion of covariance (U^C). As can be observed, in all cases, Theil's U inequality coefficient was near to zero, showing a good predictive capacity of the models. In addition, most of the error was concentrated in the proportion of covariance, indicating good accuracy of the forecasts.

6. Conclusions

The analysis of the NDVI time series in the studied area has revealed a high temporal and spatial variability in its dynamics. However, it has made it possible to identify several common characteristics such as the absence of a long-term trend (autocorrelation, periodogram, and stationarity tests) and a very significant annual seasonality that dominates the dynamics of most of these series.

The selected model has shown high global adequacy in the construction stage of the dynamic model, and it has shown a satisfactory performance for 30% of the studied pixels.

This study is a first step in trying to model and forecast the NDVI of rainfed crops in mainland Spain. However, further research is needed to build an appropriate model valid for more pixels. In the next step, multivariate time series analysis must be applied to build dynamic NDVI models including variables such as precipitation and temperature, which are essential in the development of rainfed crops.

Acknowledgements

We would like to thank the participation of Quasar s.l. y Prosecar s.l. in this research project. This research was conducted in the framework of the European project Forward. VC was supported by a postdoctoral Juan de la Cierva fellowship (FJC2021-046735-I) funded by the Spanish Ministerio de Ciencia e Innovación MCIN/AEI/10.13039/501100011033 and by the European Union « NextGenerationEU »/« PRTR ». LR was supported by a UPM grant as part of the national research program Recualificación del Sistema Universitario Español, financed by the Recovery and Resilience Package—NextGenerationEU (European Commission). CS was also supported by a predoctoral scholarship awarded by the Community of Madrid (No. IND2020/AMB-17747).

Conflict of interest

The authors declare no conflict of interest.

Author details

César Sáenz[1,2,3], Víctor Cicuéndez[4], Laura Recuero[1,3], Klaus Wiese[5],
Alicia Palacios-Orueta[3,6] and Javier Litago[1*]

1 Departamento de Economía Agraria, Estadística y Gestión de Empresas, ETSIAAB, Universidad Politécnica de Madrid (UPM), Ciudad Universitaria, Madrid, Spain

2 Quasar Science Resources S.L, Las Rozas de Madrid, Madrid, Spain

3 Centro de Estudios e Investigación para la Gestión de Riesgos Agrarios y Medioambientales (CEIGRAM), Universidad Politécnica de Madrid, Madrid, Spain

4 Departamento de Física de la Tierra y Astrofísica, Universidad Complutense de Madrid (UCM), Madrid, Spain

5 Escuela de Biología, Universidad Nacional Autónoma de Honduras, Honduras

6 Departamento de Ingeniería Agroforestal, ETSIAAB, Universidad Politécnica de Madrid (UPM), Madrid, Spain

*Address all correspondence to: javier.litago@upm.es

References

[1] Wani SP, Rockström J, Oweis T, editors. Rainfed Agriculture: Unlocking the Potential. Wallingford: CAB International; 2009

[2] Molden D, Vithanage M, Faures JM, Gordon L, Molle F, Peden D. 4.21-water availability and its use in agriculture. In: Treatise on Water Science. Oxford: Elsevier; 2011. pp. 707-732

[3] Rezapour S, Jooyandeh E, Ramezanzade M, Mostafaeipour A, Jahangiri M, Issakhov A, et al. Forecasting Rainfed agricultural production in arid and semi-arid lands using learning machine methods: A case study. Sustainability. 2021;**13**:4607. DOI: 10.3390/su13094607

[4] Turral H, Burke J, Faurès JM. Climate Change, Water and Food Security. Vol. 36. Rome, Italy: Food and Agriculture Organization of the United Nations (FAO); 2011

[5] Murray-Tortarolo GN, Jaramillo VJ, Larsen J. Food security and climate change: The case of rainfed maize production in Mexico. Agricultural and Forest Meteorology. 2018;**253-254**:124-131. DOI: 10.1016/j.agrformet.2018.02.011

[6] Kogo BK, Kumar L, Koech R. Climate change and variability in Kenya: A review of impacts on agriculture and food security. Environment, Development and Sustainability. 2021;**23**:23-43. DOI: 10.1007/s10668-020-00589-1

[7] Fetting C. The European Green Deal. Vienna, Austria; 2020

[8] Tataridas A, Kanatas P, Chatzigeorgiou A, Zannopoulos S, Travlos I. Sustainable crop and weed management in the era of the EU Green Deal: A survival guide. Agronomy. 2022;**12**:589. DOI: 10.3390/agronomy12030589

[9] Kibret KS, Marohn C, Cadisch G. Use of MODIS EVI to map crop phenology, identify cropping systems, detect land use change and drought risk in Ethiopia – An application of Google Earth Engine. European Journal of Remote Sensing. 2020;**53**:176-191. DOI: 10.1080/22797254.2020.1786466

[10] Angearu C-V, Ontel I, Boldeanu G, Mihailescu D, Nertan A, Craciunescu V, et al. Multi-temporal analysis and trends of the drought based on MODIS data in agricultural areas. Romania. Remote Sensing (Basel). 2020;**12**:3940. DOI: 10.3390/rs12233940

[11] Dong Y, Xu F, Liu L, Du X, Ye H, Huang W, et al. Monitoring and forecasting for disease and pest in crop based on WebGIS system. In: 2019 8th International Conference on Agro-Geoinformatics (Agro-Geoinformatics). IEEE; 2019. pp. 1-5. DOI: 10.1109/Agro-Geoinformatics.2019.8820620

[12] Holloway J, Mengersen K. Statistical machine learning methods and remote sensing for sustainable development goals: A review. Remote Sensing. 2018;**10**:1365. DOI: 10.3390/rs10091365

[13] Karthikeyan L, Chawla I, Mishra AK. A review of remote sensing applications in agriculture for food security: Crop growth and yield, irrigation, and crop losses. Journal of Hydrology (Amst). 2020;**586**:124905. DOI: 10.1016/j.jhydrol.2020.124905

[14] Ha TV, Uereyen S, Kuenzer C. Agricultural drought conditions over mainland Southeast Asia: Spatiotemporal

characteristics revealed from MODIS-based vegetation time-series. International Journal of Applied Earth Observation and Geoinformation. 2023;**121**:103378. DOI: 10.1016/j.jag.2023.103378

[15] Hentze K, Thonfeld F, Menz G. Evaluating Crop Area Mapping from MODIS Time-Series as an Assessment Tool for Zimbabwe's "Fast Track Land Reform Programme". PLoS One. 2016;**11**:e0156630. DOI: 10.1371/journal.pone.0156630

[16] Wardlow BD, Egbert SL, Kastens JH. Analysis of time-series MODIS 250 m vegetation index data for crop classification in the U.S. central Great Plains. Remote Sensing of Environment. 2007;**108**:290-310. DOI: 10.1016/j.rse.2006.11.021

[17] Whitcraft AK, Becker-Reshef I, Justice CO. Agricultural growing season calendars derived from MODIS surface reflectance. International Journal of Digital Earth. 2015;**8**:173-197. DOI: 10.1080/17538947.2014.894147

[18] de Castro A, Six J, Plant R, Peña J. Mapping crop calendar events and phenology-related metrics at the parcel level by object-based image analysis (OBIA) of MODIS-NDVI time-series: A case study in Central California. Remote Sensing. 2018;**10**:1745. DOI: 10.3390/rs10111745

[19] Mishra B, Busetto L, Boschetti M, Laborte A, Nelson A. RICA: A rice crop calendar for Asia based on MODIS multi year data. International Journal of Applied Earth Observation and Geoinformation. 2021;**103**:102471. DOI: 10.1016/j.jag.2021.102471

[20] Kuchler PC, Bégué A, Simões M, Gaetano R, Arvor D, Ferraz RPD. Assessing the optimal preprocessing steps of MODIS time series to map cropping systems in Mato Grosso, Brazil. International Journal of Applied Earth Observation and Geoinformation 2020;92:102150. https://doi.org/10.1016/j.jag.2020.102150.

[21] Guo Y, Xia H, Pan L, Zhao X, Li R. Mapping the northern limit of double cropping using a phenology-based algorithm and Google earth engine. Remote Sensing. 2022;**14**:1004. DOI: 10.3390/rs14041004

[22] Sisheber B, Marshall M, Mengistu D, Nelson A. Detecting the long-term spatiotemporal crop phenology changes in a highly fragmented agricultural landscape. Agricultural and Forest Meteorology. 2023;**340**:109601. DOI: 10.1016/j.agrformet.2023.109601

[23] Huesca M, Merino-de-Miguel S, Eklundh L, Litago J, Cicuéndez V, Rodríguez-Rastrero M, et al. Ecosystem functional assessment based on the "optical type" concept and self-similarity patterns: An application using MODIS-NDVI time series autocorrelation. International Journal of Applied Earth Observation and Geoinformation. 2015;**43**:132-148

[24] Box GE, Jenkins GM, Reinsel GC, Ljung GM. Time Series Analysis: Forecasting and Control. Fifth Edit. New Jersey: John Wiley & Sons, Inc; 2015

[25] Recuero L, Wiese K, Huesca M, Cicuéndez V, Litago J, Tarquis AM, et al. Fallowing temporal patterns assessment in rainfed agricultural areas based on NDVI time series autocorrelation values. International Journal of Applied Earth Observation and Geoinformation. 2019;**82**:101890. DOI: 10.1016/j.jag.2019.05.023

[26] Huesca M, Litago J, Palacios-Orueta A, Montes F, Sebastián-López A,

Escribano P. Assessment of forest fire seasonality using MODIS fire potential: A time series approach. Agricultural and Forest Meteorology. 2009;**149**:1946-1955. DOI: 10.1016/j.agrformet.2009.06.022

[27] Box GEP, Jenkins GM. Time Series Analysis: Forecasting and Control. San Francisco: Holden-Day; 1970

[28] Suresh KK, Krishna Priya SR. Forecasting sugarcane yield of Tamil Nadu using ARIMA models. Sugar Tech. 2011;**13**:23-26. DOI: 10.1007/s12355-011-0071-7

[29] Badmus MA, Ariyo OS. Forecasting cultivated areas and production of maize in Nigerian using ARIMA model. Asian Journal of Agricultural Sciences. 2011;**3**:171-176

[30] Piwowar JM, Ledrew EF. ARMA time series modelling of remote sensing imagery: A new approach for climate change studies. International Journal of Remote Sensing. 2002;**23**:5225-5248. DOI: 10.1080/01431160110109552

[31] Xiao Z, Liang S, Wang J, Jiang B, Li X. Real-time retrieval of leaf area index from MODIS time series data. Remote Sensing of Environment. 2011;**115**:97-106. DOI: 10.1016/j.rse.2010.08.009

[32] Fernández-Manso A, Quintano C, Fernández-Manso O. Forecast of NDVI in coniferous areas using temporal ARIMA analysis and climatic data at a regional scale. International Journal of Remote Sensing. 2011;**32**:1595-1617. DOI: 10.1080/01431160903586765

[33] Jiang B, Liang S, Wang J, Xiao Z. Modeling MODIS LAI time series using three statistical methods. Remote Sensing of Environment. 2010;**114**:1432-1444. DOI: 10.1016/j.rse.2010.01.026

[34] Han P, Wang PX, Zhang SY, Zhu DH. Drought forecasting based on the remote sensing data using ARIMA models. Mathematical and Computer Modelling. 2010;**51**:1398-1403. DOI: 10.1016/j.mcm.2009.10.031

[35] Alhamad MN, Stuth† J, Vannucci M. Biophysical modelling and NDVI time series to project near-term forage supply: Spectral analysis aided by wavelet denoising and ARIMA modelling. International Journal of Remote Sensing. 2007;**28**:2513-2548. DOI: 10.1080/01431160600954670

[36] Gonçalves RRV, Zullo J, Romani LAS, Nascimento CR, Traina AJM. Analysis of NDVI time series using cross-correlation and forecasting methods for monitoring sugarcane fields in Brazil. International Journal of Remote Sensing. 2012;**33**:4653-4672. DOI: 10.1080/01431161.2011.638334

[37] Tian M, Wang P, Khan J. Drought forecasting with vegetation temperature condition index using ARIMA models in the Guanzhong Plain. Remote Sensing. 2016;**8**:690. DOI: 10.3390/rs8090690

[38] Bounouh O, Essid H, Tarquis AM, Farah IR. Phenology as accuracy metrics for vegetation index forecasting over Tunisian forest and cereal cover types. International Journal of Remote Sensing. 2021;**42**:4644-4671. DOI: 10.1080/01431161.2021.1899331

[39] Carreño-Conde F, Sipols AE, de Blas CS, Mostaza-Colado D. A forecast model applied to monitor crops dynamics using vegetation indices (NDVI). Applied Sciences. 2021;**11**:1859. DOI: 10.3390/app11041859

[40] Huesca M, Litago J, Merino-de-Miguel S, Cicuendez-López-Ocaña V, Palacios-Orueta A. Modeling and forecasting MODIS-based fire potential

index on a pixel basis using time series models. International Journal of Applied Earth Observation and Geoinformation. 2014;**26**:363-376. DOI: .1016/j.jag.2013.09.003

[41] MAPA. 2023. Available from: https://servicio.mapa.gob.es/websiar/ [Accessed: June 17, 2023]

[42] AEMET/IMP. Atlas climático ibérico. Temperatura del aire y precipitación (1971-2000). Madrid: Closas-Orcoyen S. L.; 2011

[43] Vermote E. MODIS/Terra Surface Reflectance 8-Day L3 Global 500m SIN Grid V061. Missoula, MT, USA: NASA EOSDIS Land Processes DAAC; 2021

[44] Copernicus Land Monitoring Service. 2023. Available from: https://land.copernicus.eu/pan-european/corine-land-cover/clc2018 [Accessed: June 20, 2023]

[45] Tucker CJ. Red and photographic infrared linear combinations for monitoring vegetation. Remote Sensing of Environment. 1979;**8**:127-150

[46] Abraham S, Golay MJE. Smoothing and differentiation of data by simplified least squares procedures. Analytical Chemistry. 1964;**36**:1627-1639. DOI: 10.1021/ac60214a047

[47] Zhang TT, Qi JG, Gao Y, Ouyang ZT, Zeng SL, Zhao B. Detecting soil salinity with MODIS time series VI data. Ecological Indicators. 2015;**52**:480-489. DOI: 10.1016/j.ecolind.2015.01.004

[48] Hamilton JD. Time Series Analysis. Princeton: Princeton University Press; 1994

[49] Fuller WA. Introduction to Statistical Time Series. New York: John Wiley & Sons; 1976

[50] Ljung GM, Box G. On a measure of lack of fit in time series models. Biometrika. 1978;**65**:297-303

[51] Theil H. Principles of Econometrics. New York: John Wiley & Sons, Inc; 1971

[52] Dickey DA, Fuller WA. Distribution of the estimators for autoregressive time series with a unit root. Journal of the American Statistical Association. 1979;**74**:427-431. DOI: 10.1080/01621459.1979.10482531

Section 5

Financial and Economic Time Series Analysis

Chapter 10

Momentum Periods of Feedback Trading toward Exchange Rate Volatility in ASEAN Countries

Ivan Sudibyo, Zaafri Ananto Husodo and
Aulia Keiko Hubbansyah

Abstract

In this paper, the authors apply empirical evidence to demonstrate how important positive feedback trading factors are in understanding exchange rate behavior. Utilizing the GARCH augmented feedback model, or the exchange rate model set out by Laopodis, the author analyzes autocorrelation in exchange rate parameters and volatility in key ASEAN markets to yield deeper understanding of inducted exchange rates induced by the actions of positive feedback traders. The authors contend that positive feedback traders affect exchange rate volatility in ASEAN countries and induce the autocorrelation of negative returns within high exchange rate volatility. This study found that Singapore demonstrated a significant positive feedback trading during the period 1995–2014, while the authors further contend that Thailand, Indonesia, Malaysia, the Philippines, Brunei Darussalam, and Singapore also demonstrated positive feedback trading during the 1997–1998 Asian financial crisis. In addition to analyzing the positive feedback trading on exchange rate volatility, we also identify the exchange rate volatility spillover across the ASEAN countries. Related to this context, we found that Indonesia and Thailand play a dominant role as a dominant exchange rate volatility transmitter in the ASEAN region.

Keywords: feedback trading, volatility, momentum periods, high volatility, positive feedback trading

1. Introduction

In this paper, we provide empirical evidence that confirms how important positive feedback trading is as a factor in exchange rate behavior in key ASEAN markets. Utilizing Laopodis' [1] exchange rate model, we analyze autocorrelation of exchange rates and volatility for deeper understanding of inducted exchange rates caused by the positive feedback trading behavior of foreign exchange traders. Our thesis is that positive feedback traders affect exchange rate volatility in ASEAN countries and induce autocorrelation of negative returns within high exchange rate volatility.

Related to this concern, Keynes [2] asserted that while investor sentiment and market psychology play important roles in financial markets, it is also almost impossible for individual economic agents to impound all market information so as to make consistently rational investment decisions, as the following renowned quote elaborates: "Investment based on genuine long-term expectation is so difficult as to be scarcely practicable. He who attempts it must surely lead much more laborious days and run greater risks than he who tries to guess better than the crowd how the crowd will behave; and, given equal intelligence, he may make more disastrous mistakes" ([2] quoted by [3]). According to Keynes, it's hard to measure objective value from market fundamentals, and even were it possible, it will cost more to gather all relevant information in order to make such an assessment.

Indeed, exchange rate markets do not reflect fundamental value for most of the time [4]. Messe and Rogoff [5] provide the most distinguished empirical support of this perspective, one of a number of different exchange rate modeling approaches that emerged following the breakdown of the Bretton Woods exchange rate environment that prevailed up until the 1970s. Sub-sequent studies introduced non-linear features in exchange rate dynamics, such as initiated by Frankel and Froot [6] and further developed in the context of stock prices [4, 7–9] and Laopodis [1]. This paper is based on this approach.

Some country's exchange rates can be affected by monetary policy and market forces, the latter reflecting the actions of traders. In this study, we will focus on analyzing the traders' influence toward exchange rate through volatility that is formed by their trading pattern. One indication of traders' force in determining exchange rates is through estimation of feedback trading.

1.1 Research contributions

It is the authors' contention that feedback trading in key ASEAN countries' exchange rate markets causes increasing exchange rate volatility and causes such currencies to deviate from fundamental value, such as happened in the 1997–1998 Asian financial crisis. Hence, we make research of sub periods analysis due to change in exchange rate policy and the effect toward exchange rate volatility.

The exchange rate feedback trading research scope of this paper covers key ASEAN countries including Thailand, Indonesia, Malaysia, the Philippines, Brunei Darussalam, and Singapore. Precedent research examined pre and post crisis without analyzing sub-periods for exchange rate policy and its effect on exchange rate volatility.

The researcher motivation used the Laopodis model [1] is to identified the positive feedback trading effect on exchange rate volatilities in ASEAN countries more detailed with sub period analysis whereas the positive feedback trading was rarely discussed in precedent research about ASEAN countries' exchange rate volatility.

The paper's research focus is on sub-periods crisis during the 1997–1998 period because several subject countries of this study changed their exchange rate system during this time in attempts to mitigate the monetary and real economy impacts of the 1997–1998 Asian Financial Crisis. Such sub-period study of the exchange rate volatility experienced by the indicated ASEAN countries is also examined in 2007–2008 Global Financial Crisis period and thereafter.

2. Literature review

2.1 The trader

As with other prices, buyer and seller interactions, such as between economic agents like a household, a firm, and any other international trading currency company determine the exchange rate of two floating currencies, ceteris paribus. A market where international currency is traded is called foreign exchange market [10].

Traders are all economic agents who make sales or purchase transactions. Meanwhile, the main actors in exchange rate market are a commercial bank, an international trading firm, non-bank financial institutions such as management asset and insurance institutions, and central bank [10].

This paper refers to the studies on trader behavior conducted by Laopodis [1], De Long et al. [11], Sentana and Wadhwani [12]. From feedback trading behavior patterns, they divide the traders into two distinct groups:

- Positive feedback traders; that is, those traders who buy when prices are high and sell when prices are low ([1]; Sentana and Wadhwani [12]).

- Negative feedback trader; those traders who sell when prices are high and buy when prices are low ([1]; Sentana and Wadhwani [12]).

In Section 2.2 there is explanation regarding trader's influence on exchange rate volatility or, in other words, feedback trading as set out in Laopodis' exchange rate model [1].

2.2 Exchange rate model

Feedback Trading in Exchange Rate Market—Laopodis Theoritical Model [1]. Despite the existence of economic agents assumed to behave in a rational manner that maximizes the fundamental of smart money, other investors (noise-trader) use random price or trend trading. Both investors' interactions create instability in asset prices that push those asset prices away from their fundamental value.

2.2.1 Rational traders

The first group of traders (smart money) are trying to maximize their portfolio's (N) function exclusively in return and risk factor [12]. As result, this kind of investors will hold an optimal share fraction ($\mathrm{F}_{1,\ \mathrm{t}}$) from the market portfolio as below:

$$F_{1,t} = \left[(E_{t-1}(R_t) - \alpha)/k{\sigma_t}^2\right] \tag{1}$$

Where R_t is ex post return in t time, E_{t-1} is expectation operator in $t - 1$ time, α is return level at risk free asset, ${\sigma_\mathrm{t}}^2$ is the conditional variance at t time and k is the coefficient representing risk aversion level from the investor. With assumption k is positive, $k{\sigma_\mathrm{t}}^2$ is premium risk that needed in t time. Equation (1) is a mean–variance model from every asset requests which implies the risk asset demand increases with excess return as expected ($R_t - \alpha$) and reversing with ${\sigma_t}^2$ risk level.

2.2.2 Noise traders

The second type of investor, the noise traders are assumed to follow a feedback strategy where they tend to buy while prices are increasing and tend to sell when prices are falling. Thus, their functional demand where $F_{2,t}$ is share proportion owned by this group and p assumed positive. This strategy indicates noisy information trading, which is irrelevant in regards to the fundamental economy. Also, this will pull the asset price out like exchange rate from their essential value. Therefore, negative serial return correlation is seen as price increases followed by high demand that affects higher prices in the future. If $p < 0$, negative feedback trading is seen as a trader who buys in low price and sells in high price. Market equilibrium requests all demand should be hold by these two types of investor:

$$F_{1,t} + F_{2,t} = 1 \tag{2}$$

With the equality of (1) and (2), we can substitute the following

$$R_t = a + k\sigma_t^2 - k\sigma_t^2 \rho R_{t-1} \tag{3}$$

And we can assume the rational expectation until the Eq. (3) becomes stochastic as follows

$$R_t = a + k\sigma_t^2 - k\sigma_t^2 \rho R_{t-1} + \varepsilon_t \tag{4}$$

It indicates dependency in noise trader type, positive feedback trader will deliver negative serial correlation in return. It's common when the volatility level is higher, the negativity autocorrelation will increase. This point simply illustrates with Eq. (5)

$$R_t = a + k\sigma_t^2 + \left(\phi_0 + \phi_1\sigma_t^2\right)R_{t-1} + \varepsilon_t \tag{5}$$

Where the direct influence of noise trader (in constant level risk) comes from positive sign in $\varnothing_0$.

Therefore, to calculate the effect of intense trading during exchange rate appreciation, it needs the augmentation of Eq. (5) as follows:

$$R_t = a + k\sigma_t^2 + \left(\phi_0 + \phi_1\sigma_t^2\right)R_{t-1} + \phi_2|R_{t-1}| + \varepsilon_t \tag{6}$$

We conclude a simple Eq. (6) into:

$$R_t = a + k\sigma_t^2 + \phi_0 R_{t-1} + \phi_1\sigma_t^2 R_{t-1} + \phi_2|R_{t-1}| + \varepsilon_t \tag{7}$$

A positive feedback reflects in $\varnothing_1$ negative sign. The last part indicates the possibility of asymmetric trading behavior, where negative return is followed by higher feedback trading volume, if $\varnothing_2 > 0$. Positive and significant sign in asymmetric coefficient $\varnothing_2$ refers to the higher intention of positive feedback trading during appreciation rather than during depreciation. As cited in the Laopodis' previous research [1], this paper will utilize the Generalized Error Distribution [13] as an assumption of error distribution, with the GED's parameter $r > 0$. GED is a normal distribution if $r = 2$ and a leptokurtic if $r < 2$. If $r = 1$, the distribution is double exponential [13].

Related to the concern of this study, Laopodis' research [1] indicated that there are $\varnothing_0$ negative autoregressive and significant parameter in Franc, Singaporean Dollar, Peseta, Greek Drachma, Rupee, and Ringgit for all periods. It means that there is contrarian effect on those currencies. A $\varnothing_0$ positive autoregressive and significant parameter in Pound, Canadian Dollar, Mark, and Won indicated a bandwagon effect on those currencies.

An autoregressive $\varnothing_1$ parameter—that shows positive feedback trading and has similarity with herding (an agent who buy after prices increase)—will produce negative autocorrelation ($\varnothing_1$ negative parameter value). On the contrary, negative feedback trading that has similarity with profit taking or a trader who sell after the price has increased, will result in positive autocorrelation ($\varnothing_1$ positive parameter value). A $\varnothing_1$ positive and significant parameter value shows in British Pound, denotes negative feedback trading exists in the British Pound. A $\varnothing_1$ negative and significant parameter value can be found in Canadian Dollar, Lira, Mark, Peseta, Rupee, Ringgit, Mexican Peso, Singapore Dollar (SGD), Won and Baht. It indicates positive feedback trading on those currencies.

Asymmetric feedback trading signed by $\varnothing_2$ positive and significant parameter value. This leads into positive feedback trading that has intensively increased during appreciation rather than depreciation. It shows in Franc, Mark, Lira, Peseta, Mexican Peso, Rupee, and Won. A disparity value for an autocorrelation within pre- and post-crisis scenarios indicates a fluctuation in currency autocorrelation return. An autocorrelation will have positive value during quiet periods and will have negative value during volatile periods ([1]; Sentana and Wadhwani [12]).

By nature, traders' asymmetric behavior indicates that they depend on central banks to gain short term profit. This action can trigger instability in foreign exchange market either by smart money or noise trader. Besides, this asymmetric action can lead into less credible account and shows that traders depend on central bank's reserve to gain short term profit. At the end, ϕ_0 positive and significant autoregressive parameter can be termed as the bandwagon effect, which occurs when past currency movement followed by expected currency movement are on the same route.

3. Method, data and analysis

3.1 Data description

This research used daily basis data of ASEAN currency exchange rate start from 1995 to 2014. We used daily basis data in accordance to Laopodis [1]. To be informed, all exchange rates data toward US Dollar of the countries analyzed in this study were obtained from Thomson Reuters Datastream.

3.2 Model specification

3.2.1 Conditional variance

The conditional variance of the returns (σ_t^2) of various asset prices such as the exchange rate is found to be plagued by conditional heteroscedasticity as thus it can be modeled as a GARCH (1,1) process as follows:

$$\sigma_t^2 = \alpha_0 + \alpha_1\left(e_{t-1}^2\right) + \beta_1\left(\sigma_{t-1}^2\right)$$

And the half-life (HL) of a shock must be computed, which is defined as HL = ln (0.5)/ln(α1 + β1).

3.2.2 Estimation by Laopodis model

We adapt the econometric model proposed by Laopodis [1] which is called GARCH augmented feedback model. We apply maximum likelihood estimation [14] (BHHH algorithm) to estimate this model, with equation as follows:

$$R_t = a + k\sigma_t^2 + \phi_0 R_{t-1} + \phi_1\sigma_t^2 R_{t-1} + \phi_2|R_{t-1}| + \varepsilon_t \tag{8}$$

Where,

R_t: return in t time

a: constantan

k: coefficient investor's degree of risk aversion

σ_t^2: conditional variants return

ϕ_0: parameter that represents autoregressive process in return, if negative indicates contrarian effect (previous deviation from mean value that affects currency value moves in reverse). Whereas positive indicates bandwagon effect (previous currency observation is expected to move forward)

ϕ_1: autocorrelation feedback trading parameter: if negative, it presents a positive feedback trader; if positive, it presents negative feedback trader.

R_{t-1}: return in $t-1$

ϕ_2: asymmmetric trading behavior parameter: if positive, it dentoes a negative return (exchange rate value appreciation). It's followed by a higher volume of feedback trading. In other words, positive feedback trading is more intese during apreciation rather than during depreciation.

$|R_{t-1}|$: absolute value from $t-1$ return

ε_t: error term

4. Results

After examining the periods 1995–2014, the ASEAN countries do not exhibit significant positive feedback trading. Only Singapore that shows significant positive feedback trading (at α = 5%). The $\varnothing_1$ parameter value, in reaching thousands of units, indicates that the GARCH value in SGDs is very small, meaning that even if Singapore's Dollar has positive feedback trading, the Singapore currency is stable, fluctuating only in a small range. **Table 1** shows estimate result during 1995–2014 using maximum likelihood estimation GARCH augmented feedback model.

As can be shown from the **Table 1**, the autoregressive parameter is significant in the Philippines and Vietnam. The Philippines has negative autoregressive parameter yet significant (at α = 5%). This indicates a presence of contrarian effect in the Philippine currency during 1995–2014. This means previous currency movements resulted in the present currency moving in reverse. Vietnam has negative autoregressive parameter and significant (at α = 1%). This also shows a presence of contrarian effect in Vietnam's currency during 1995–2014. Vietnam has a significant

No	Countries	Periods	k	Φ_0	Φ_1	Φ_2	$\alpha 0$	$\alpha 1$	$\beta 1$	r	HL
1	Indonesia	1/1/1995–31/12/2014	0.108179	−0.000352	−1.798838	−0.0031	0.00000023**	0.401017** r	0.731644**	0.761625**	5.564
	prob		0.6925	0.9734	0.3125	0.8254	**0.0000**	**0.0000**	**0.0000**	**0.0000**	
2	Malaysia	1/1/1995–31/12/2014	0.613392	0.007723	−19.95998	−0.007387	0.00000000025*	0.493388**	0.738432**	0.711784**	3.3246
	prob		0.4435	0.4157	0.3814	0.4939	**0.0000**	**0.0000**	**0.0000**	**0.0000**	
3	Thailand	1/1/1995–31/12/2014	−0.022884	0.000157	−47.73276	−0.000444	0.0000000548**	0.148054**	0.871393**	1.026591**	35.988
	Prob		0.9882	0.99	0.2131	0.9794	**0.0000**	**0.0000**	**0.0000**	**0.0000**	
4	Philippines	1/1/1995–31/12/2014	0.560224	−0.023936*	−77.48001	−0.01876	0.00000000342*	0.124357**	0.893084**	1.0414**	40.088
	Prob		0.7767	**0.0485**	0.3074	0.2451	**0.0000**	**0.0000**	**0.0000**	**0.0000**	
5	Singapore	1/1/1995–31/12/2014	2.635466	−0.015648	−1316.924*	−0.010148	0.0000000579**	0.066105**	0.931445**	1.281344**	282.57
	Prob		0.607	0.3334	**0.0229**	0.6313	**0.0002**	**0.0000**	**0.0000**	**0.0000**	
6	Brunei D.	1/1/1995–31/12/2014	−0.037597	0.000082	−0.772018	−0.000284	0.000000156**	0.092572**	0.909251**	0.772919**	380.57
	Prob		0.9701	0.9932	0.9207	0.9853	**0.0000**	**0.0000**	**0.0000**	**0.0000**	
7	Cambodia	1/1/1995–31/12/2014	0.04838	0.000562	−0.606319	−0.001498	0.0000000158**	0.707803**	0.530158**	0.546711**	3.2471
	Prob		0.8506	0.6866	0.5891	0.8977	**0.0000**	**0.0000**	**0.0000**	**0.0000**	
8	Vietnam	1/1/1995–31/12/2014	−1.152384	−0.088325**	−12.24466	0.087888**	0.0000000183**	0.167304**	0.705865**	1.009344**	5.111
	Prob		0.8414	**0 0.0000**	0.9527	**0.0000**	**0.0000**	**0.0000**	**0.0000**	**0.0000**	
9	Myanmar	1/1/1995–31/12/2014	No arch effect								
	Prob										
10	Laos	01/01/1995–31/12/2014	No arch effect								
	Prob										

**sig at 0.1; **sig at 0.01.*

Table 1.
The estimation results.

asymmetric trading behavior too ($\varnothing_2$ positive and significant parameter). This indicates positive feedback trading in Vietnam's currency is more intense during appreciation rather than depreciation. Other ASEAN countries, during the period 1995–2014, are not indicated having any bandwagon effect or contrarian effect. Additionally, in the periods of 1995–2014, the other ASEAN countries are not indicated to have any asymmetric trading behavior. The "HL" value shows how many days of volatility persist in each currency.

4.1 Sub-periods exchange rate policy analysis

We made the sub-periods analysis arbitrary based on the date of authority change exchange rate policy, the period of US Financial crisis 2007–2008, and the period post US Financial crisis (2010–2014) which was begin with QE phase 2 period until 2014.

a. Indonesia

In the **Table 2**, we see Indonesia's currency displays no ARCH in 1995–1996 (before the Asian financial crisis). This shows Indonesia's currency was stable at that time. During early 1997 until before 14 August 1997, the Rupiah had begun to show an ARCH effect but no $\varnothing$ significant parameter. This is consistent with the fact that the Rupiah was, at this time, still managed under a floating rate within a crawling band policy.

On 14 August 1997, the Rupiah 's band intervention was released into a market mechanism called free floating policy that has been applied until now. During this period, the research resulted $\varnothing_0$ positive and significant parameter. This shows the presence of a bandwagon effect that is previous currency movement affected currency's expectation to move in the same way. Positive feedback trading appeared during this period by a presence of $\varnothing_1$ negative and significant parameter. Many factors impacted the Rupiah exchange rate's attenuation during this period. Trust and panic crisis led the society to buy US$ to stabilize their wealth, even to gain profit from that. A snowball effect undeniable, US$ became treasure. Many Indonesian wealthy had been prepared since previous years saving their assets abroad to anticipate domestic political instability. In other hand, private foreign debt either short-term or medium-term placed huge pressure on the Rupiah until the domestic foreign exchange reserves were left insufficient to repay the debt and the interest in the day of maturity [15]. Private foreign debt was accumulated since early 1990s and had reached an enormous amount even surpassing official government debt that had been decreasing in recent years. Mostly these private foreign loans were not hedged [15].

Indonesia's GDP growth in 1997 was 4.69% and in 1998 it plunged to −13.12%. The crisis impacted Indonesia's GDP growth drastically. While Indonesia's annual inflation rate (calculated from consumer prices) leapt higher from 6.22% in 1997 to 58.38% in 1998. From 14 August 1997 until 2014, volatility appeared in the Rupiah exchange rate, but showed no significant $\varnothing$ parameter that indicates the volatility affected by positive feedback trading, asymmetric trading, bandwagon nor contrarian effect.

Looking further at **Table 2**, in the period 2007–2008, the Rupiah did not exhibit ARCH effect. This was indicated that the Rupiah did not volatize during this period. The 2007–2008 global financial crisis affected Indonesia's economy, this was evidenced by decreasing annual GDP growth rates from 6.34% in 2007 to 6.01% in 2008. In 2007, annual inflation based on consumer prices in Indonesia was 6.4% and increased in 2008 to 9.77%. Indonesia's current account balance, as a percentage of GDP, also decreased from 2.42% in 2007 to 0.0024% in 2008.

No	Countries	Periods	Model	a	k	Φ_0	Φ_1	Φ_2	$\alpha 0$	$\alpha 1$	$\beta 1$	r
1	Indonesia	02/01/1995–31/12/1996	No arch effect									
	Prob											
		02/01/1997–13/08/1997	GARCH (1,1)	0.00000154	−0.681506	0.004938	−173.7801	−0.004701	0.000000282*	0.544318*	0.605745**	0.791666**
	Prob			0.9871	0.9477	0.9353	0.8273	0.9677	**0.0826**	**0.0311**	**0.0000**	**0.0000**
		14/08/1997–31/12/1998	GARCH (1,1)	0.002043	−0.133022	0.172647**	−6.713448**	−0.05331	0.0000635*	0.162186*	0.818367**	0.869269**
	Prob			0.1324	0.8039	**0.0001**	**0.0043**	0.4853	**0.0589**	**0.0157**	**0.0000**	**0.0000**
		14/08/1997–31/12/2014	GARCH (1,1)	0.00000238	−0.00447	−0.000503	−1.377,055	−0.000469	0.000000878**	0.404217**	0.69695**	0.781052**
	Prob			0.9496	0.9878	0.9507	0.4184	0.9727	**0.0000**	**0.0000**	**0.0000**	**0.0000**
		2007–2008	No arch effect									
	Prob											
		After global crisis	GARCH (1,1)	$6.83E^{-06}$	−0.194753	0.052516*	−482.0333*	0.044698	0.000000129*	0.244688**	0.80867**	0.781586**
	Prob	December 2010–2014		0.8773	0.9481	**0.0259**	**0.017**	0.2154	**0.0225**	**0.0000**	**0.0000**	**0.0000**

**sig at 0.1; **sig at 0.01.*

Table 2.
Estimation result for Indonesia.

We also examined the post-global financial crisis period from the end of 2010 until 2014. The Federal Reserve, The US Central Bank adopted an unprecedented quantitative easing policy. On November 2010, The Fed declared to begin QE phase 2 [16]. In 2013, The Fed began tapering to reduce bond purchasing since 18 December 2013 [17]. These Federal Reserve policy settings over the 2010–2014 period affected Indonesia's exchange rate. A $\varnothing_0$ positive autoregressive and significant parameter emerged in the Rupiah. Meanwhile $\varnothing_1$ negative and significant parameter indicated a bandwagon effect and positive feedback trading during that period.

The Rupiah's exchange rate had sloppy trend during this time. Autoregressive parameters showed bandwagon effect, a previous currency movement affected currency's expectation to move in the same way during this period. Hence, volatility in Rupiah during this time also was affected by positive feedback trading. In this period, annual GDP growth was decreasing from 6.2% in 2010 to 5.0% in 2014, while the current account surplus as a percentage of GDP also decreased from 0.7% in 2010 and moved to a deficit of −3.2% 2013.

b. Malaysia

Similar to the Indonesian Rupiah, Malaysia's Ringgit also did not demonstrate the ARCH effect before the Asian financial crisis period (based on data in the period 1995–30/06/1997). From July 1997 to August 1998, however, the Ringgit detached its value into floating. This was proved in the data by the presence of ARCH effect in Ringgit. Meanwhile, there was no $\varnothing$ significant parameter during this time. However, Malaysia obviously experienced the crisis' effect with GDP decreasing on an annual percentage basis from 7.3% in 1997 to −7.3% in 1998. At the same time, inflation on an annual percentage basis based on consumer price crawled higher from 2.66% in 1997 to 5.27% in 1998. The lowest Ringgit rate occurred on 8 January 1998, when it hits 4.6805 to the United States dollar (**Table 3**).

From September 1998 until 20 July 2005, Malaysia employed a fixed exchange rate system which was pegged toward US$ and floating toward other currencies. This is seen in $\varnothing_0$ and $\varnothing_2$ significant parameter derived in the author's modeling of the Ringgit's performance during this period. A $\varnothing_0$ negative and significant parameter indicated that the Ringgit had experienced a contrarian effect, and that a previous currency movement affected crosscurrent expectations during that period. Additionally, $\varnothing_2$ denoted negative and significant number. We can conclude that during the sub-period policy, an asymmetric trading behavior emerged in Ringgit and showed no significant positive feedback trading during this time. Therefrom, Ringgit was pegged to US$ movement at this period and just in fact, Ringgit moved around 3.8 US$.

After economic conditions had stable following the Asian financial crisis, Malaysia removed the Ringgit from its peg to the United States dollar on 21 July 2005 after China removed the Renminbi from its peg to the United States dollar. From 21 July 2005, Malaysia applied a managed floating exchange rate system once again.

We found a presence of $\varnothing_0$, $\varnothing_1$, and $\varnothing_2$ significant parameter from 21 July 2005 until the end of 2014. A $\varnothing_0$ positive and significant parameter denoted a bandwagon effect during this time. Positive feedback trading was indicated by a presence of $\varnothing_1$ parameter during this period. Meanwhile, a $\varnothing_2$ negative and significant parameter showed asymmetric trading behavior during that period. In other words, positive feedback trading was more intense during depreciation rather than appreciation.

No	Countries	Periods	Model	a	k	Φ_0	Φ_1	Φ_2	$\alpha 0$	$\alpha 1$	$\beta 1$	r
2	Malaysia	02/01/1995–30/06/1997	No arch effect									
	Prob											
		01/07/1997–31/08/1998	GARCH (1,1)	0.00126*	−0.884495	0.027907	−38.52741	0.05094	5.33E^{-06}	0.109995*	0.87854**	1.054552**
	Prob			**0.0821**	0.6404	0.6985	0.3941	0.5703	0.1486	0.0111	**0.0000**	**0.0000**
		09/1998–31/12/1998	GARCH (1,1)	0.0001	−62251E-01	0.878278**	−913.4873**	−0.937503**	0.000000233**	0.809532**	0.139062**	1.737544**
	Prob			0.5762	0.2638	**0.0000**	**0.0000**	**0.0000**	**0.0000**	0.0004	0.0095	**0.0000**
		09/1998–20/07/2005	GARCH (1,1)	3.81 E-09	−3.671559*	−0.129247**	−48.04654	−0.153701**	0.000000000117**	0.028744**	0.79958**	0.609419**
	Prob			0.9948	**0.0575**	**0.0000**	0.5547	**0.0000**	**0.0000**	**0.0000**	**0.0000**	**0.0000**
		2007–2008	GARCH (1,1)	−0.0000824	0.023844	0.161737**	−2498.654**	0.016359	7.83E^{-08}	0.265345**	0.79888**	0.982996**
	Prob			0.3767	0.9975	**0.0021**	**0.0097**	0.8067	0.2439	**0.0000**	**0.0000**	**0.0000**
		21/07/2005–31/12/2014	GARCH (1,1)	−0.00000291	7.400178*	0.047004*	−639.8292*	−0.069351*	4.06^{E-09}	0.155886**	0.874154**	1.050412**
	Prob			0.9283	**0.0269**	**0.0492**	**0.0925**	**0.0248**	0.1893	**0.0000**	**0.0000**	**0.0000**
		After global crisis	GARCH (1,1)	−9.03E^{-05}	13.13979*	0.017666	−190.2781	−0.070041	0.000000538**	0.080107**	0.889356**	1.154591**
	Prob	December 2010–2014		0.5272	**0.0146**	0.6952	0.7949	0.1672	**0.0429**	**0.0003**	**0.0000**	**0.0000**

**sig at 0.1; **sig at 0.01.*

Table 3.
Malaysia estimation results.

Ringgit had $\varnothing_0$ and $\varnothing_1$ significant parameter during 2007–2008. A bandwagon effect emerged during this period by a presence of $\varnothing_0$ parameter. A big number and $\varnothing_1$ negative parameter value showed a positive feedback trading during this period, albeit with a small range fluctuation. In the Malaysian economic system, 2007's annual GDP percentage growth was 6.29% and decreased to 4.83% in 2008, while the annual percentage change in inflation based on consumer prices grew from 2.02% to 5.44% in 2008. During this time period, Malaysia's current account balance itself increased from 15.3 in 2007 to 16.8 (% of GDP) in 2008.

There was a volatility in the Ringgit at the end of period 2010–2014, but it showed no $\varnothing$ significant parameter. This means that the volatility in the Ringgit wasn't affected by positive feedback trading, asymmetric trading, bandwagon nor contrarian effect. A variable macroeconomic performance was seen in Malaysia during the 2010–2014 period, but in 2013–2014, it climbed from 4.73 in 2013 to 6.03 (% of GDP). Malaysia's current account balance was lowering but still in positive condition. Malaysia's current account balance was recorded 10.906 (% of GDP) in 2010 and 3.74 in 2013 (% of GDP).

c. Thailand

The Baht demonstrated the ARCH effect just before crisis period (1995–1996), but had no $\varnothing$ significant parameter. In the crisis period (early 1997 until mid-1997) there was $\varnothing_0$ negative and significant parameter. This indicated a presence of contrarian effects in the trading of the Baht during that period. During 14–15 May 1997, the Baht was hit by enormous speculative attacks [18]. On 2 July 1997 after spending billions of dollars to defend Baht, the Thai Central Bank was forced to let its currency float [19]. The Thai Baht fell to its lowest point at 56 to the United States dollar on 12 January 1998, [19]. During the deepest of the crisis period from 2 July 1997 until 31 December 1998, the Baht exchange rate showed a bandwagon effect and positive feedback trading. Due this crisis Thailand's GDP growth was declining. In 1997 Thailand's annual GDP percentage growth was −1.4% and in 1998 plunged to −10.5%. Meanwhile, based on annual percentage changes in consumer prices, inflation in Thailand went higher from 5.6% in 1997 to 8.0% in 1998. Thailand's current account balance in 1997 was valued −2% of GDP (**Table 4**).

At the end of the period 2010–2014, the authors modeling highlights the volatility in the Thai Baht, but the research also pointed to an absence of $\varnothing$ significant parameter in the Baht, therefore indicating that the volatility wasn't affected by positive feedback trading, asymmetric trading, bandwagon nor contrarian effect.

During this period, Thailand experienced variable economic growth in terms of annual percentage changes in GDP growth, moving from 7.8% in 2010 to 0.7% in 2014. The current account balance in Thailand decreased during this period: In 2009, the current account balance was 8.3 (% of GDP) and then the negative current account balance happened in 2012–2013, hitting −0.3984 in 2012, easing further down to −0.97644 (% of GDP) in 2013, but recovering in 2014 to 3.51% of GDP.

d. Vietnam

Vietnam maintained the Vietnamese Dong (VND) in point around 10.000–11.000 for 3 years, 1993–1996 [20]. The research showed that Vietnam had the ARCH effect in the early 1995 until January 1997. Additionally, $\varnothing_0$ parameter denoted positive and significant value. This indicated that the bandwagon effect existed in the VND at that time. The $\varnothing_1$ parameter in the VND yielded negative yet significant value, on contrary the parameter had a big number. It exhibited a positive feedback trading on the

No	Negara	Periods	Model	a	k	Φ_0	Φ_1	Φ_2	$\alpha 0$	$\alpha 1$	$\beta 1$	r
3	Thailand	02/01/1995–31/12/1996	GARCH (1,1)	0.000101	−29.77067	−0.049986	19.78596	−0.013797	0.0000000334*	0.047783*	0.916995**	1.137272**
	Prob			0.1414	0.5775	0.5007	0.5196	0.8272	**0.0805**	**0.0481**	**0.0000**	**0.0000**
		01/01/1997–01/07/1997	GARCH (1,1)	0.00023	4.729407	−0.233357**	−285623	−0.1953	0.000000233	0.782481*	0.592685**	0.841169**
	Prob			0.0639	0.3937	**0.0098**	0.3151	0.1322	0.4241	**0.053**	**0.0004**	**0.0000**
		02/07/1997–31/12/1998	GARCH (1,1)	−0.001128*	20.166	0.116529*	−177.326**	0.0691	0.000025**	0.467093**	0.542638**	0.907214**
				0.0324	0.2222	**0.0265**	**0.0026**	0.2847	**0.0051**	**0.0014**	**0.0000**	**0.0000**
		2007–2008	GARCH (1,1)	0.000301	−1.474605	0.013949	−25.54712	−0.019957	0.00000706**	0.668198**	0.484566**	0.741343**
	Prob			0.0139	0.0897	0.6667	0.8312	0.567	**0.0061**	**0.0009**	**0.0000**	**0.0000**
		After global crisis	GARCH (1,1)	$-2.14E^{-05}$	−0.131412	−0.002294	−7919062	0.028256	0.000000259**	0.089813**	0.886711**	1.232899**
	Prob	December 2010–2014		0.8679	0.9928	0.9663	0.7932	0.5716	**0.0142**	**0.0001**	**0.0000**	**0.0000**

**sig at 0.1; **sig at 0.01.*

Table 4.
Thailand estimation result.

currency, even occurred the fluctuation in a very small range. As a result, the VND exchange rate pushed to its lowest point (it was during 1995 until January 1997). On 17 January 1997, the exchange rate was hitting 11,180. While $\varnothing_2$ negative and significant parameter value termed as a presence of the asymmetric trading behavior in the period. The positive feedback trading was more intense during depreciation rather than appreciation. During the Asian financial crisis (1997–1998), Vietnam expanded its band-intervention to ±5% in February 1997 and ±10% in October 1997 (**Table 5**).

We highlight the Vietnam's GDP growth (in 1997–1998) was decreasing but the value was still positive at 8.15 in 1997 and became 5.76 in 1998. The inflation based on consumer price in 1997, Vietnam had 3.2 inflation rate and increased to 7.26 in 1998. The VND exchange rate pushed to its lowest (13907) on 24 August 1998. It signified, even applied managed floating system, the VND still pushed to the lowest rate.

This research showed that there was no ARCH effect in the VND during 1 February 1997 until 31 January 1999. It indicated that SBV played significant role during this period in the VND movement. After financial crisis period in February 1999 until 2014, Vietnam changed its currency system into managed float regime. We can see $\varnothing_0$ negative and significant parameter. It indicated a presence of contrarian effect during this period. Also, this indicated that the VND had a significant asymmetric trading behavior ($\varnothing_2$ positive and significant parameter). This showed that a positive feedback trading in VND had more intense in appreciation rather than in depreciation.

The VND experienced an ARCH effect without $\varnothing$ significant parameter during 2007–2008. Vietnam's GDP growth decreased from 7.12 in 2007 to 5.66 in 2008. Based on consumer price, Vietnam's inflation went higher from 8.3 in 2007 to 23.11 in 2008. Vietnam's current account balance fell from −8.98 in 2007 to −10.91 (% of GDP) in 2008. In the end of period 2010 until 2014, the VND signed no ARCH effect. It concluded that there was no volatility in the VND during this period. Hence, SBV maintained the VND exchange rate during this period. Vietnam's GDP growth in this period was quite good, it was valued >5. In 2013, GDP growth was in 5.42 and moved to 5.98 in 2014. Vietnam's current account balance was moving from −3.68 in 2010 to 5.81 in 2012 and decreased slightly in 2013 to 5.53 (% of GDP). Though, SBV maintained the exchange rate with managed float system, the VND exchange rate still lowered since 2010 until 2014. The VND exchange rate was 18,469 in early January 2010 and hit its lowest point on 27 November 2014 to 21,390.

e. Philippine

The Philippine Peso (PHP) had been used floating system before crisis. Though, they still paid the effect of 1997–1998 the Asian financial crisis. This denoted $\varnothing_0$ positive and $\varnothing_1$ negative significant parameter. The author concludes that the Philippine Peso (PHP) experienced a bandwagon effect and positive feedback trading behavior during this period. The exchange rate pushed to its lowest point on 6 January 1998 to 45.2. This affected the Philippines GDP growth fell down. It fell from 5.18 in 1997 to −0.57 in 1998. It followed by the inflation increment based on consumer price, from 5.59 in 1997 to 9.23 in 1998. There signed no ARCH effect in 2007–2008. Meanwhile the financial crisis impacted the PHP that denoted in the decrement of GDP growth from 6.61 in 2007 to 4.15 in 2008. The inflation also increased in 2007. Based on consumer price, the inflation went almost tripled from 2.9 to 8.26 in 2008. The Philippine's current account balance during 2007–2008 fall from 5.40 in 2007 to 0.08 (% of GDP) in 2008 (**Table 6**).

No	Negara	Periods	Model	***a***	***k***	***Φ_0***	***Φ_1***	***Φ_2***	***α*0**	***α*1**	***β*1**	***r***
4	Vietnam	02/01/1995–31/01/1997	GARCH (1,1)	−0.0000419**	738.8006**	0.032935*	−806555.4**	−0.079674**	0.00000000227*	0.124039**	0.821613**	0.507143**
	Prob			**0.0000**	**0.0000**	**0.0772**	**0.0000**	**0.0000**	**0.0166**	**0.0086**	**0.0000**	**0.0000**
		01/02/1997–31/01/1999	No arch effect									
	Prob											
		01/02/1999–31/12/2014	GARCH (1,1)	0.000000368	−5.531236	−0.173693**	365.4061	0.163149**	0.00000000127**	0.124497**	0.871726**	1.084233**
	Prob			0.9425	0.5544	**0.0000**	0.3091	**0.0000**	**0.0000**	**0.0000**	**0.0000**	**0.0000**
		2007–2008	GARCH (1,1)	0.00000066	0.682373	−0.000872	−286.1015	−0.001408	0.0000000514**	0.279101**	0.716467**	0.696813**
	Prob			0.9746	0.9505	0.9789	0.9212	0.9709	**0.0001**	**0.0000**	**0.0000**	**0.0000**
	Prob	After global crisis December 2010–2014	No arch effect									

**sig at 0.1; **sig at 0.01.*

Table 5.
Vietnam result estimation.

No	Negara	Periods	Model	*a*	*k*	Φ_0	Φ_1	Φ_2	$\alpha0$	$\alpha1$	$\beta1$	*r*
5	Philippines	02/01/1995–31/12/1996	GARCH (1,1)	0.000000334	−0.4696	−0.067297*	−1209.4020	−0.0615	**0.0000**	0.1443	0.8763	0.7587
	prob			0.9905	0.9571	**0.0189**	0.3297	0.1671	**0.0402**	**0.0154**	**0.0000**	**0.0000**
		1/1/1997–31/12/1998	GARCH (1,1)	0.000000537	−2,.22548	0.249147**	−430.1327**	0.025304	0.0000000138*	0.3505**	0.710945**	0.713202**
	prob			0.9831	0.1933	**0.0000**	**0.0001**	0.5082	**0.0774**	**0.0000**	**0.0000**	**0.0000**
		2007–2008	No arch effect									
		After global crisis	No arch effect									
	prob	December 2010–2014										

*sig at 0.1; **sig at 0.01.*

Table 6.
Philippine estimation result.

After crisis in the end of 2010 until 2014, the PHP experienced no ARCH effect. This showed that the PHP had not volatility during this period. The GDP growth was in positive range from 7.63 in 2010 to 6.09 in 2014. Philippines current account balance was still positive. In 2010, Philippine's current account balance was 3.59, and increased in 2014 to 4.44 (% of GDP).

f. Singapore

The financial crisis also impacted the SGD. This was seen during 1997–1998. The $\varnothing_1$ parameter was valued negative and significant. This indicated a positive feedback trading during this period. The $\varnothing_1$ huge parameter denoted that fluctuation occurred in a small range. Therefore, Singaporean Dollar (SGD) sloped downward to its lowest on 12 January 1998 at point 1791.Meanwhile the SGD $\varnothing_2$ parameter termed as positive and significant. This exhibited that positive feedback trading was more intense during this appreciation rather than depreciation. Although the SGD was quite stable, GDP growth decreased from 8.29 in 1997 to −2.22 in 1998. The inflation based on consumer price also adjusted. It sank from 2.0 in 1997 to −0.26 in 1998. The SGD had a significant $\varnothing_0$ (positive) and $\varnothing_1$ (negative) parameter during 2007–2008. This indicated that there were a bandwagon effect and positive feedback trading behavior. This economic situation impacted the Singapore's GDP growth. It went down from 9.11 in 2007 to 1.7 in 2008 along with the increasing number of inflation from 2.09 in 2007 to 6.51 in 2008.Singapore's current account balance in this period was 25.97 in 2007 and sloped down to 14.4 (% of GDP) (**Table** 7).

There was a sign of volatility in the SGD during 2010–2014. But there were no $\varnothing$ significant parameter that denoted volatility affected by positive feedback trading, asymmetric trading, bandwagon or contrarian effect. Singapore's GDP growth decreased from 15.24 in 2010 to 2.91 in 2014. During this period, the current account balance was fluctuated but still at range between 17% and 23 (% of GDP). The current account balance in Singapore was 23.66 in 2010 and in 2015 it was valued 19.08 (% of GDP).

g. Brunei Darussalam

During 1997–1998, Brunei's currency had $\varnothing_1$ negative and significant parameter. It indicated a positive feedback trading behavior during this period. Therefore, the currency pushed to its lowest on 6 August 1998, at point 2.0112. On contrary, the GDP growth increased from −1.47 in 1997 to −0.55 in 1998. Meanwhile, the inflation based on consumer price decreased from 1.7 to −0.44 in 1998. The Brunei Dollar (BND) had a significant $\varnothing_0$ (positive) and $\varnothing_1$ (negative) parameter during 2007–2008. It indicated a presence of bandwagon effect and positive feedback trading behavior. Such as Singapore, the $\varnothing_1$ parameter was valued negative and had a large number. This showed a positive feedback trading in this period even affected it a small range of fluctuation. Brunei's GDP growth declined from 0.15 in 2007 to −1.9 in 2008. The inflation based on consumer price was increasing from 0.96 to 2.08 in 2008. Brunei's current account balance climbed up from 39.42 in 2007 to 48.2 (% of GDP) in 2008 (**Table 8**).

During 2010–2014, after financial crisis period, there was a volatility in the BND, but It had no significant $\varnothing$ parameter. It indicated that the volatility wasn't affected by positive feedback trading, asymmetric trading, bandwagon nor contrarian effect. The GDP growth was fluctuated during this period but inclined from −1.7 in 2013 to 5.3 in 2014. In 2012, the current account balance was 33.52 (% of GDP).

No	Countries	Periods	Model	a	k	Φ_0	Φ_1	Φ_2	$\alpha 0$	$\alpha 1$	$\beta 1$	r
6	Singapore	02/01/1995–31/12/1996	GARCH (1,1)	0.0000177	$-22472E^{+01}$	−0.121885*	−2318.9100	$2.11E^{-02}$	$4.98E^{-07}$	0.368049	0.579737	0.972402
				0.8266	0.2193	**0.0555**	0.7321	0.7441	**0.0068**	**0.0007**	**0.0000**	**0.0000**
		1/1/1997–31/12/1998	GARCH (1,1)	0.0000901	0.233784	0.051853	−1317.534*	0.133241*	$1.21\ E^{-07}$	0.125286**	0.890436**	0.968537**
	Prob			0.5011	0.9738	0.3028	**0.0414**	**0.0169**	0.2293	**0.0002**	**0.0000**	**0.0000**
		2007–2008	GARCH (1,1)	-0.0000558	4.489112	0.104117*	−4484.978**	0.004011	0.0000000269	0.000646**	0.997412**	1.765285**
	Prob			0.7256	0.548	**0.0366**	**0.0006**	0.9328		**0.0227**	**0.0000**	**0.0000**
		After global crisis	GARCH (1,1)	0.000101	−0.515869	−0.018236	−1114308	−0.062631	0.000000108**	0.075689**	0.916008**	1.393457**
	Prob	December 2010–2014		0.4174	0.9681	0.6638	0.521	0.225	**0.0509**	**0.0000**	**0.0000**	**0.0000**

**sig at 0.1; **sig at 0.01.*

Table 7.
Singapore estimation result.

No	Countries	Periods	Model	a	k	Φ_0	Φ_1	Φ_2	$\alpha 0$	$\alpha 1$	$\beta 1$	r
7	Brunei D.	02/01/1995–31/12/1996	No arch effect									
		1/1/1997–31/12/1998	GARCH (1,1)	0.00000535	−0.111147	0.004022	−40.99186*	−0.000969	0.000000583**	0.020133**	0.873668**	0.918024**
	Prob			0.9788	0.9523	0.8471	**0.0277**	0.9681	**0.0000**	**0.0000**	**0.0000**	**0.0000**
		2007–2008	GARCH (1,1)	−0.000252	12.86418	0.099232*	−5255.174**	−0.035262	0.000000000416**	0.000284**	1.004222**	1.356014**
	Prob			0.1222	0.446	**0.0631**	**0.0063**	0.5844	0.955	**0.9228**	**0.0000**	**0.0000**
		After global crisis	GARCH (1,1)	$3.77E^{-05}$	5.011454	−0.03235	−207.2297	−0.049652	0.000000105*	0.066026**	0.925894**	1.267054**
	Prob	December 2010–2014		0.7519	0.7158	0.4022	0.8814	0.3249	**0.0684**	**0.0000**	**0.0000**	**0.0000**

**sig at 0.1; **sig at 0.01.*

Table 8.
Brunei Darussalam estimation result.

h. Cambodia

Cambodian exchange rate during 1997–1998 had $\varnothing_0$ negative and significant parameter. During the Asian financial crisis period, Cambodian currency had a contrarian effect. The Cambodian Riel (KHR) had a $\varnothing_2$ positive and significant parameter. Hence, it indicated an asymmetric trading behavior during this period. In other words, a positive feedback trading was more intense during appreciation rather than depreciation. The lowest point of Cambodian exchange rate had been recorded at point 4015 on 24 July 1998. The GDP growth also declined from 5.61 in 1997 to 14.8 in 1998. Based on consumer price, the inflation climbed up from 7.96 in 1997 to 14.8 in 1998. The KHR had a $\varnothing_1$ (negative) and significant parameter during 2007–2008. During this period, the KHR termed as positive feedback trading behavior and had a big number in $\varnothing_1$ parameter. It was assessed as a positive feedback trading. Although it affected a small range of fluctuation. The Cambodia's economic system impacted the GDP growth. It sloped downward from 10.21 in 2007 to 6.69 in 2008. In 2007, the inflation based on consumer price was hitting 7.6 and increased to 24.9 in 2008 (**Table 9**).

Cambodia's current account balance went down from −4.9 (% of GDP) in 2007 to −7.9 (% of GDP) in 2008. In the end of 2010 until 2014, after financial crisis period, the KHR had a contrarian effect. It was demonstrated by a $\varnothing_0$ negative and significant parameter. The KHR was also valued $\varnothing_2$ negative and had a significant parameter. It showed an asymmetric trading behavior during that period and denoted that positive feedback trading was more intense during depreciation rather than appreciation. Cambodia's GDP growth indicated well because of the increase point from 5.9 in 2010 to 7.03 in 2014. But the Cambodia's current account balance sloped downward from −3.64 (% of GDP) in 2010 to −10.55 (% of GDP) in 2013.

i. Myanmar

Myanmar Kyat or MMK had no ARCH effect during Asian financial crisis (1997–1998). On April 2nd, 2012 until December 31st 2014, Myanmar already applied managed floating regime. But in fact, Myanmar still experienced no ARCH effect. Myanmar exchange rate was hitting its lowest point at 6.44002 on August 1st 1997. Myanmar GDP growth increased in very small range from 5.65 in 1997 to 5.86 in 1998. Meanwhile, the inflation based on consumer price increased from 29.69 in 1997 to 51.4 in 1998. After crisis period, during 2010–2014, the MMK indicated no ARCH effect. The economic growth in Myanmar was counted as stable. It had 8.24 point in 2013 and 8.49 point in 2014. But the current account balance sloped down from −1.68 in 2012 to −1.92 in 2013. After Myanmar exerted economic system from peg to SDR to managed floating system, the exchange rate moved from 6.41 to 818 Kyat on 2 April 2012.

j. Laos

Such as the Myanmar Kyat, the Laos Kip Rate (LAK) had not any ARCH effects during the Asian financial crisis 1997–1998. It also happened during 1995–2014. There was no ARCH effect in the LAK. Though the LAK sloped downward to its lowest with 4203.50 point on 11 December 1998.The GDP growth also decreased from 6.87 in 1997 to 3.96 in 1998. Meanwhile the inflation based on consumer price went higher from 27.5 in 1997 to 90.98 in 1998. There was ARCH effect in the LAK during 2007–2008 (**Table 10**).

The research found that the LAK was in $\varnothing_1$ (negative) and $\varnothing_2$ (negative) significant parameter. It exhibited a positive feedback trading during this period. A $\varnothing_2$ parameter showed that an asymmetric trading behavior and a positive feedback trading was more intense during depreciation rather than appreciation. It impacted to the GDP growth. The GDP growth was relatively increasing from 7.59 in 2007to 7.82 in 2008. The

No	Countries	Periods	Model	a	k	Φ_0	Φ_1	Φ_2	$\alpha 0$	$\alpha 1$	$\beta 1$	r
8	Cambodia	02/01/1995–31/12/1996	No arch effect									
		1/1/1997–31/12/1998	GARCH (1,1)	$9.E^{-15}$	$2.22E^{-10}$	−0.0000000184*	−0.00000346	0.0000000245*	$5.55E^{-10}$	0.011988**	0.988011**	0.092395**
	Prob			0.9882	0.5163	**0.085**	0.1641	**0.02**	—	**0.0000**	**0.0000**	**0.0000**
		2007–2008	GARCH (1,1)	−0.000000849	174.176	0.004716	−1389.412**	0.005051	0.000000255**	0.420766**	0.722673**	0.575157**
	Prob			0.9692	0.5025	0.6307	**0.0000**	0.5011	**0.0073**	**0.0006**	**0.0000**	**0.0000**
		After global crisis	GARCH (1,1)	$1.19E^{-07}$	−0.391192	−0.201847*	204.7175	−0.04108*	$1.01E^{-07}$	0.386119	0.721907	0.590749
	Prob	December 2010–2014		0.9966	0.8898	**0.0000**	0.4182	**0.0782**	**0.0000**	**0.0000**	**0.0000**	**0.0000**

**sig at 0.1; **sig at 0.01.*

Table 9.
Cambodia estimation result.

No	Countries	Periods	a	k	Φ_0	Φ_1	Φ_2	$\alpha 0$	$\alpha 1$	$\beta 1$	r
10	Laos	02/01/1995–31/12/1996									
		01/01/1995–31/12/2014									
	Prob										
		1/1/1997–31/12/1998									
	Prob										
		2007–2008	0.000152**	−194.9292**	−0.092729	−142721.6	−0.075832**	0.000000223*	0.22251**	0.770992**	0.562965**
	Prob		**0.0000**	**0.0000**	0.1978	0.0531	**0.0000**	**0.0366**	**0.0063**	**0.0000**	**0.0000**
	Prob	After global crisis									
		December 2010–2014									

**sig at 0.1; **sig at 0.01.*

Table 10.
Laos estimation result.

inflation rate (based on consumer price) climbed up from 4.52 in 2007 to 7.62 in 2008. During the period 2010–2014, ARCH effect did not show in the LAK. It indicated the currency had not volatility during this period. The Laos GDP growth obtained 8.52 point in 2010 but slowly downward to 7.45 in 2014. The Laos current account balance also sloped down from 0.4 in 2010 to −3.36 (% of GDP) in 2013. At the end, the LAK exchange rate fluctuated from 7500 until 8100 in this period.

4.2 Exchange rate volatility spillover in the ASEAN countries

In addition to analyzing the influence of positive and negative feedback trading on ASEAN exchange rate volatility, we are also very keen to analyze the exchange rate volatility spillover across the countries. For this purpose, we developed the six-variable VAR model consisting of countries' exchange rate volatility. In its operations, we exclude three ASEAN countries, namely Brunei Darussalam, Cambodia and Myanmar because of constraints on the data's availability. However, this exclusion does not significantly reduce the validity of the estimation, since the six countries used, namely Indonesia, Malaysia, the Philippines, Singapore, Thailand and Vietnam have greatly represented the ASEAN economy. To be noticed, the contribution of these six countries reaches 95% relative to the ASEAN economy [21].

Following Diebold and Yilmaz [22]; Hubbansyah and Husodo [23], we adapt the generalized VAR framework in identifying the spillover condition of the exchange rate in the ASEAN region. This method allows us to identify the relative contributions of own variables and other variable shocks to the forecast error variance decomposition (FEVD) of each variable in the VAR model which is calculated and displayed in tabular form [24].

In **Table 11**, we show spillover indices estimation for each variable based on 10-days ahead of the FEVD. For information, the ij^{th} entry is an estimated contribution to the fevd of variable i derived from innovation or shock to variable j. The diagonal element ($i = j$) represents the own-variable spillovers, while the off-diagonal element measures the cross-variable spillovers. The total spillover index (TSI) is equal to the sum of off-diagonal elements relative to the sum of total column (including the diagonal elements) that is expressed in percentage [22, 23, 25].

Based on the information presented in **Table 11**, it is known that the average value of exchange rate volatility spillover across countries—as indicated by the value of TSI—is 14.5%. It means that approximately 14.5% of the fevd of the variables comes from spillovers. The TSI, which receive a value of more than 10%, shows that exchange rate volatility spillover among the ASEAN countries to be relatively high.

Indonesia and Thailand have the largest directional spillover to others with value of 31.8% and 46.2% respectively. This means that the spillover of exchange rate volatility from both countries has a relatively greater contribution to the exchange rate volatility of other countries in ASEAN. In the case for Indonesia, part of its total directional spillover to others is contributed from the value of directional spillover to Thailand. Meanwhile, for Thailand, its total value of directional to others is largely contributed by the spillover which is transmitted to Indonesia and Singapore.

Directional including own, which shows a value greater than 100% in Indonesia and Thailand, is indicating that both countries are responsible for the magnification of exchange rate spillovers across the countries. It means that the exchange rate volatility spillovers originating in both countries have a substantial impact on the dynamics of exchange rate volatility in other country in ASEAN. Fundamentally, this could be happened due to the economic size of Indonesia and Thailand are bigger than the other four countries that are being analyzed [21].

	From (*j*)						
To (*i*)	**Vol_Ina**	**Vol_Thb**	**Vol_Php**	**Vol_Sgd**	**Vol_Myr**	**Vol_Vnd**	**From others**
Vol_Ina	77.74	19.55	0.67	0.61	0.69	0.73	22.3
Vol_Thb	19.05	79.15	0.45	0.19	0.77	0.39	20.9
Vol_Php	1.53	2.40	95.40	0.31	0.34	0.02	4.6
Vol_Sgd	7.41	17.79	1.34	72.55	0.91	0.01	27.4
Vol_Myr	2.36	3.73	0.16	1.11	92.59	0.04	7.4
Vol_Vnd	1.44	2.69	0.01	0.19	0.03	95.65	4.4
Directional to oOthers	31.8	46.2	2.6	2.4	2.7	1.2	86.9
Directional including Own	109.5	125.3	98.0	75.0	95.3	96.8	14.5% (TSI)

Source: data processed (1995–2014).

Table 11.
Exchange rate volatility spillover index.

Furthermore, although the average calculation of exchange rate volatility spillovers has provided good insight regarding the exchange rate volatility in the region, however this approach may mask interesting findings related to the pattern of the spillovers evolution under particular situation. Related to this concern, we are interested to analyze how total spillovers evolve over time and whether they are affected by major economic events, such as economic crisis. For this purpose, following Diebold and Yilmaz [22]; Antonakakis et al. [25] and Hubbansyah and Husodo [23], we do a rolling window estimation. In our case, we will estimate the six-variable VAR model using 200-days rolling window in order to obtain the variance decompositions, so that we have the total spillover indices in a time-varying fashion.

The estimation results are shown in the **Figure 1**.

The value of the spillover index of exchange rate volatility in ASEAN with 200-days of rolling window is found to vary over time. Although varied, the trend of time-varying of total spillovers is relatively high. An interesting fact that is known from the estimation results of time varying total spillovers is that there is a tendency of exchange rate volatility spillovers to increase in certain economic conditions, such as the Asian financial crisis and the US financial crisis. This indicates that there is a strong tendency of interaction among the countries' exchange rate volatility spillovers during the crisis period. This research finding that showed there was an increasing interconnectedness

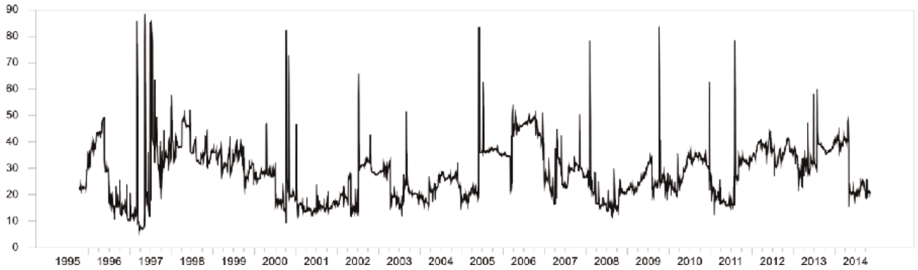


Figure 1.
Plots of the time-varying exchange rate volatility spillover indices based on 200-days rolling window estimation. [Source: Data processed (2018).]

between variables during the period of crises is consistent with the findings of Claessens et al. [26], Antonakakis et al. [25] and Hubbansyah and Husodo [23].

4.3 Structural breaks test

It's called a structural break when a time series abruptly changes at a point in time. This change could involve a change in mean or a change in the other parameters of the process that produce the series. Being able to detect when the structure of the time series changes can give us insights into the problem we are studying. Structural break tests help us to determine when and whether there is a significant change in our data. We made structural breaks test of the return and the result is given in **Figures 2–11**.

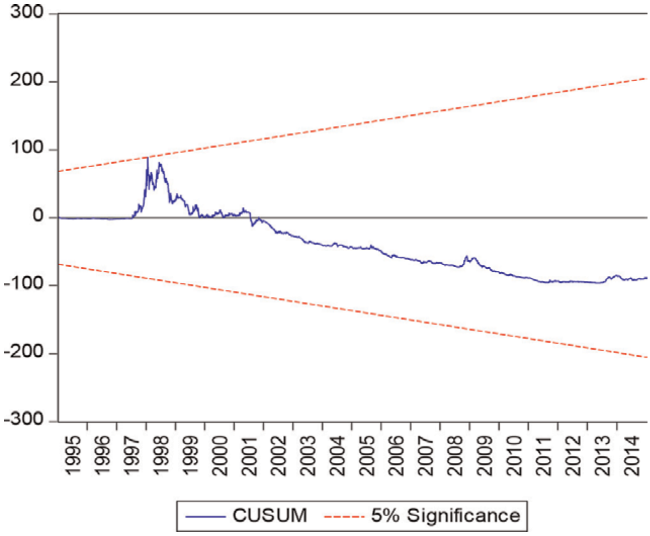


Figure 2.
USD-IDR

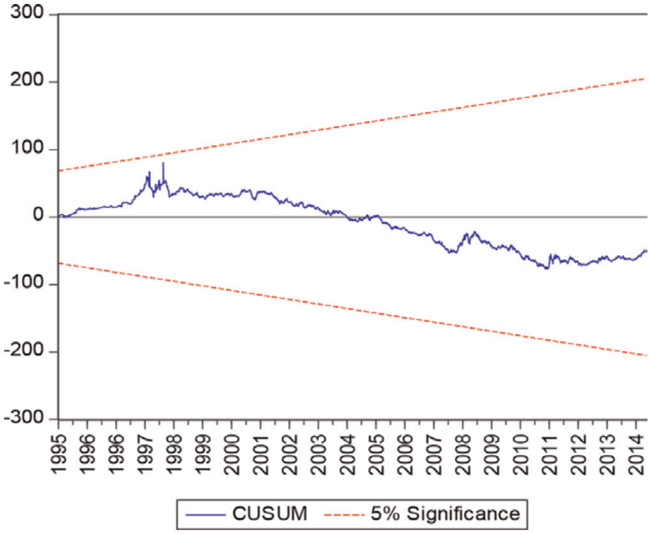


Figure 3.
USD-BND

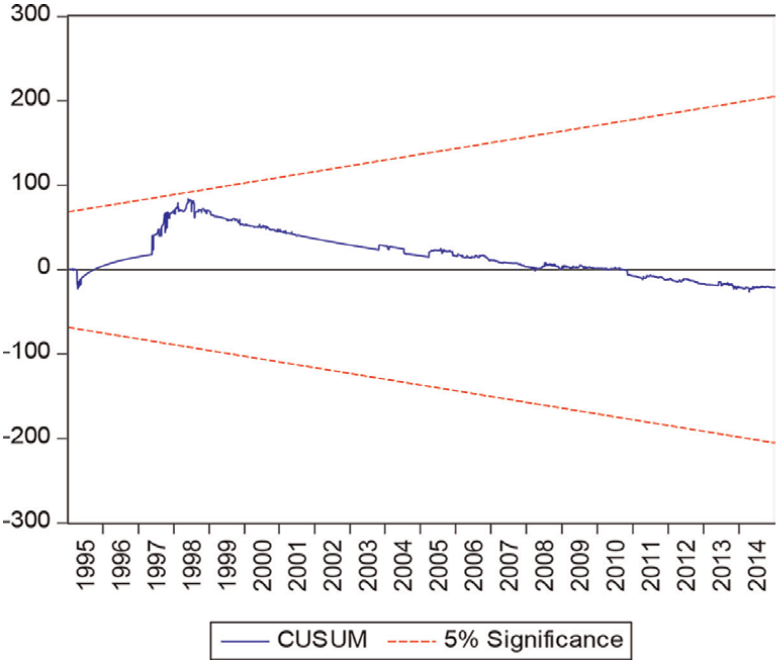


Figure 4.
USD KHR

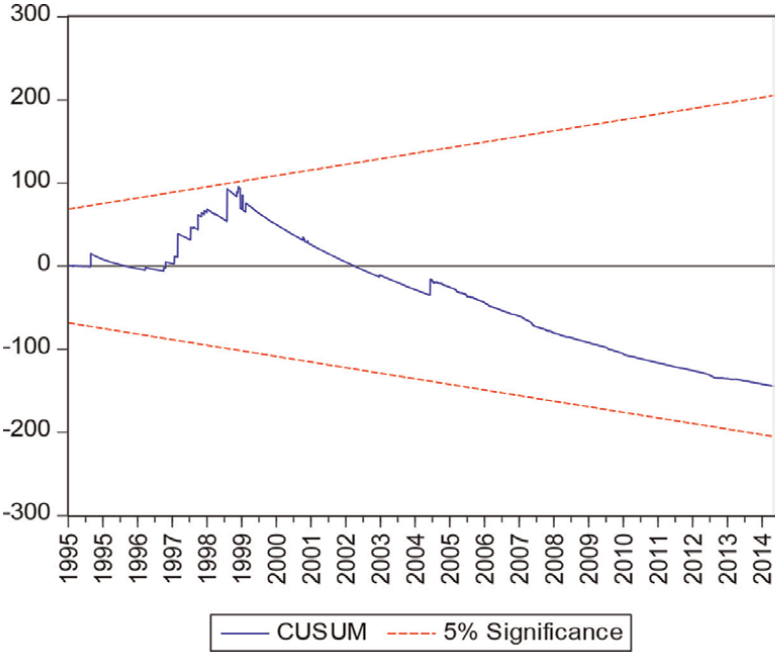


Figure 5.
USD LAK

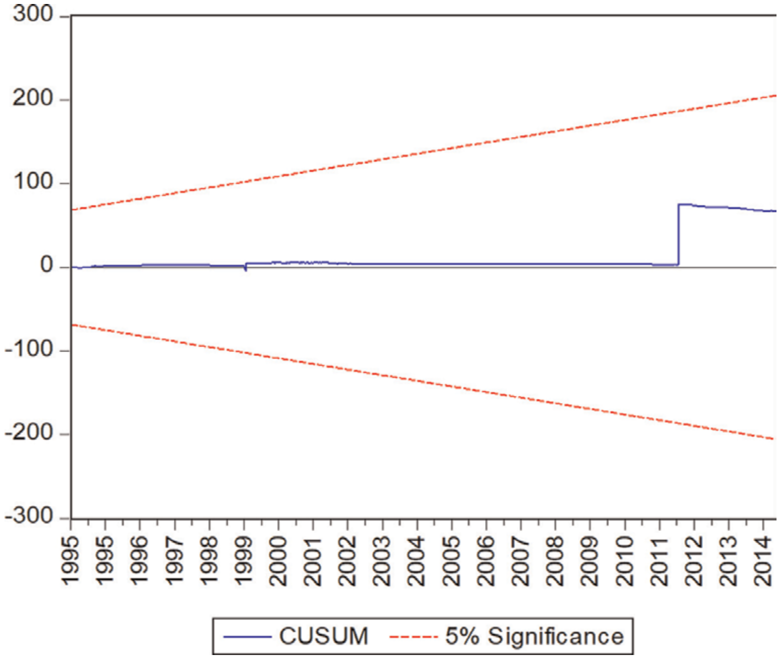


Figure 6.
USD MMK

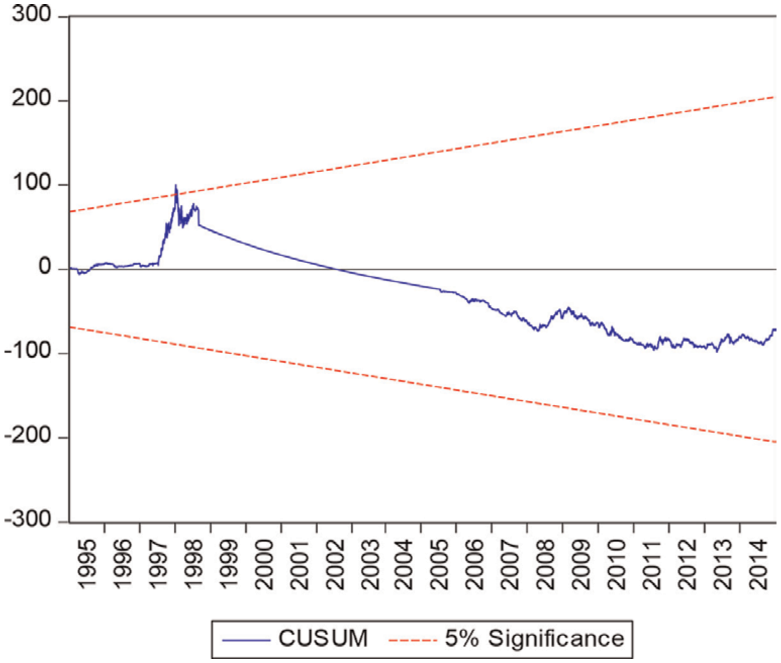


Figure 7.
USD MYR

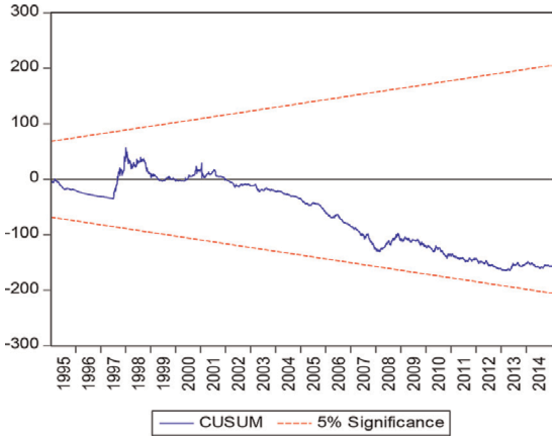


Figure 8.
USD PHP

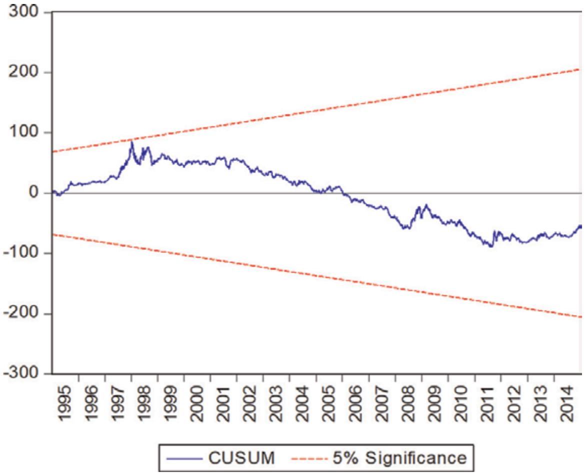


Figure 9.
USD SGD

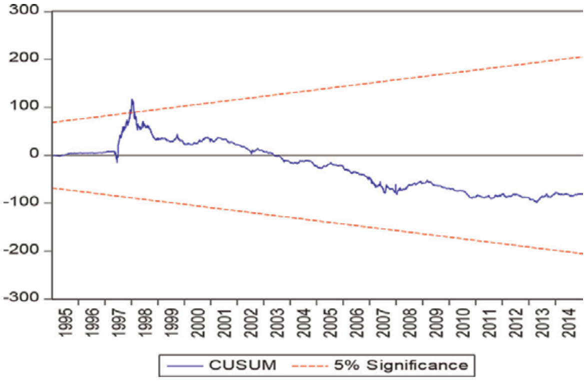


Figure 10.
USD THB

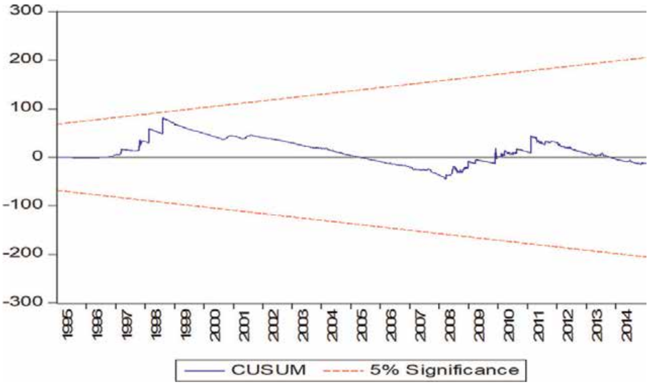


Figure 11.
USD VND

From **Figures 2–11**, we see that the IDR, MYR, SGD and THB have significance CUSUM test at 1998 which means the 1998 crisis have significantly changed the data.

5. Conclusion

The objective of this study is to analyze the autocorrelation in exchange rate parameters and volatility in key ASEAN markets to yield deeper understanding of inducted exchange rates induced by the actions of positive feedback traders. In addition to this concern, we also try to identify the exchange rate volatility spillover in the ASEAN. Related to the above, we found several important findings. First, the positive feedback trading is the main cause of the high volatility and weakening the exchange rate value. However, during 1995–2014, most of ASEAN currencies had not positive feedback trading except Singapore. But the condition is different when we analyze the data more detail by investigating the sub-periods where during the Asian financial crisis 1997/1998, most of all ASEAN currencies demonstrated a significant positive feedback trading. The crisis also affected the GDP growth and the inflation to almost all ASEAN countries. Second, the US financial crisis during 2007–2008 exhibited no significant impact through all ASEAN currencies. This impact can be seen from no ARCH sign such as in Indonesia, in Philippine, and in Myanmar. Meanwhile Malaysia, Singapore, Brunei, Cambodia, and Laos had a volatility in their exchange rate. Hence, their exchange rate indicated a positive feedback trading. Fluctuation also occurred in a small range. Thailand and Vietnam experienced a volatility without a sign of positive feedback trading. No ARCH effect in several periods affected by no volatility in currency movement of several countries. We do not exclude these countries because we want to show the characteristics of the exchange rates of each ASEAN country despite the different results of each country. For example, Myanmar and Laos applied a managed floating system. Third, in the context of exchange rate volatility spillover we found that Indonesia and Thailand play a dominant role as transmitter of exchange rate volatility in the ASEAN region. Fourth, in the context of time-varying fashion, the total spillovers tend to increase in a certain economic condition, such as economic crises. This research finding that showed there was an increasing interconnectedness between variables during period of crises is consistent with the findings of Claessens et al. [26], Antonakakis et al. [25] and Hubbansyah et al. [23].

Additional information

An earlier version of this chapter was previously published as a preprint in: Sudibyo I. Momentum Periods of Feedback Trading toward Exchange Rate Volatility [Internet]. Center for Open Science; 2021. Available from: http://dx.doi.org/10.31235/osf.io/yx8re. This initial draft of this chapter was first published as preprint: Sudibyo I. Momentum Periods of Feedback Trading toward Exchange Rate Volatility [Internet]. SocArXiv; 2021. Available from: osf.io/preprints/socarxiv/yx8re

Author details

Ivan Sudibyo[1]*, Zaafri Ananto Husodo[1] and Aulia Keiko Hubbansyah[2]

1 Faculty of Economics and Business, Universitas Indonesia, Indonesia

2 Faculty of Economics and Business, Universitas Pancasila, Indonesia

*Address all correspondence to: ivansudibyo14@gmail.com

References

[1] Laopodis NT. Feedback trading and autocorrelation interactions in the foreign exchange market: Further evidence. Economic Modelling. 2005;**22**: 811-827

[2] Keynes JM. The General Theory of Unemployment, Interest and Money. New York: Harcourt, Brace and World; 1936

[3] Hommes CH. Heterogeneous Agent Models in Economics and Finance, Tinbergen Institute Discussion Paper, No. 05-056/1. 2005

[4] De Grauwe P, Grimaldi M. Exchange rate puzzles: A tale of switching attractors. European Economic Review. 2006;**50**:1-33

[5] Messe R, Rogoff K. The Out-of-sample failure of empirical exchange rate models: Sampling error or misspecification? In: Exchange Rates and International Macroeconomics, National Bureau of Economic Research, Inc. Cambridge, MA, USA: National Bureau of Economic Research; 1983. pp. 67-112. Available from: https://EconPapers.repec.org/RePEc:nbr:nberch:11377

[6] Frankel JA, Froot K. Understanding the U.S. dollar in the eighties: The expectations of chartists and fundamentalists. NBER Working Paper no. R0957. Dec 1, 1987. Available from: https://ssrn.com/abstract=552102

[7] Bohl M, Reitz S. The influence of positive feedback trading on return autocorrelation: Evidence for the German stock market, Working Paper Series of Postgraduate Research Programme. 2002

[8] Brock W, Hommes C. Heterogeneous beliefs and routes to chaos in a simple asset pricing model. Journal of Economic Dynamics and Control. 1998; **22**:1235-1274

[9] Le Baron B. Some relations between volatility and serial correlations in stock market returns. Journal of Business. 1992;**65**:199-219

[10] Krugman PR, Obstfeld M. International Economics: Theory and Policy. 8th ed. Boston: Pearson Addison-Wesley; 2009

[11] De Long JB, Shleifer A, Summers LH, Waldmann RJ. Noise trader risk in financial markets. Journal of Political Economy. 1990;**98**(4):703-738

[12] Sentana E, Wadhwani S. Feedback traders and stock return autocorrelations: Evidence from a century of daily data. Economic Journal. 1992;**102**:415-425

[13] Nelson DB. Conditional heteroskedasticity in asset returns: A new approach. Econometrica. 1991;**59** (2):347-370

[14] Berndt EK, Hall HB, Hall RE, Hausman JA. Estimation and inference in nonlinear structural model. Annals of Economic and Social Measurement. 1974;**3**(4):1974

[15] Tarmidi LT. Krisis Moneter Indonesia: Sebab, Dampak, Peran IMF dan saran. Buletin Ekonomi Moneter dan Perbankan. 2003;**1**(4):1-25. DOI: 10.21098/bemp.v1i4.183

[16] Federal Reserve. Press Release. November, 3, 2010. Available from: http://www.federalreserve.gov/newsevents/press/monetary/2010 1103a.html

[17] CNBC News. Fed Begins Taper Program. CNBC. 18 Dec 2013. Available from: https://www.cnbc.com/2013/12/18/fed-begins-taper-program.html

[18] Hoontrakul P. Contagion in S. E. Asia Measuring Stock Market Co-Movement. 2001. Available at SSRN: http://ssrn.com/abstract=293319 or doi: 10.2139/ssrn.293319

[19] Yoon I-H. Financial crisis theories explaining the 1997 Thai financial crisis. Thammasat Economic Journal. 2006;**24** (1):127-157

[20] Hien MAMT. Solutions for exchange rate policy of transition economy of Vietnam. Dissertation zur Erlangung des Grades Doktor der Wirtschaftswissenschaft (Doctor rerumpoliticarum, Dr. rer. pol.) der Juristischen und WirtschaftswissenschaftlichenFakultät der Martin-Luther-Universität Halle-Wittenberg. 2007

[21] ASEAN Statistic. ASEAN Statistic Yearbook 2015. Jakarta: The ASEAN Secretariat; 2015

[22] Diebold FX, Yilmaz K. Better to give than to receive: Predictive measurement of volatility spillover. International Journal of Forecasting. 2012;**28**:57-66

[23] Hubbansyah AK, Husodo ZA. The interdependence between the financial sector and business sector in ASEAN 4 countries. Journal of Indonesian Economy and Business (JIEB). 2018;**33**(1):77-94

[24] Conefrey T, Cronin D. Spillover in Euro Area Sovereign Bond Markets. Central Bank of Ireland: Research Technical Paper, Vol. 5. 2013

[25] Antonakakis N, Breiteniechner M, Johann S. Business cycle and financial cycle spillover in the G7 countries. The Quarterly Review of Economics and Finance. 2015;**58**:154-162

[26] Claessens S, Kose MA, Marco ET. Financial Cycles: What? How? When? IMF Working Paper /11/76. 2011

Section 6

Engineering and Technological Applications of Time Series Analysis

Chapter 11

Electric Fault Diagnosis in Induction Machines Using Motor Current Signature Analysis (MCSA)

Francisco Javier Villalobos-Pina, Josue Augusto Reyes-Malanche, Eduardo Cabal-Yepez and Efrain Ramirez-Velasco

Abstract

Electric fault diagnosis is an important subject for ensuring the operational efficiency and reliability of induction machines, which are widely used in the industrial sector. Motor current signature analysis (MCSA) is an effective, non-invasive technique that has been useful for diagnosing faults in these machines. MCSA is applied on the acquired stator currents during the induction machine operation to detect and identify specific characteristics related to distinct faulty conditions. In this work, different methodologies for electric current analysis as instantaneous space phasor (ISP) module, spectral examination through Fourier transform, multiresolution inspection utilizing wavelet transform, and current phasor observation with fuzzy logic, are proposed for detecting and classifying short-circuit faults among coils of a stator winding in an induction motor, which has been modified to induce short-circuit faults with different severity degrees on its windings.

Keywords: instantaneous space phasor (ISP) module, Fourier transform, wavelet transform, current phasor, fuzzy logic

1. Introduction

Fault detection in induction machines is an important subject of electrical engineering and industry in general. These machines play a key role in widespread applications, from induction motors used in manufacturing processes to those used in propulsion systems for transportation and power generation. The reliable and efficient operation of induction machines is essential to guarantee continuous production and the security of the systems where they are employed. The Institute of Electrical and Electronics Engineers (IEEE) indicates that about 28 to 38% of induction motor faults occur in the stator [1], whereas the Electric Power Research Institute (EPRI) points out that 26% of the induction motor faults take place in the stator [2]. Different techniques for electric stator fault detection have been proposed in the literature. For instance, in Ref. [3], a two-stage methodology is proposed. In the first stage, mutual information is estimated from delayed stator current signals, which is used as input to C4.5 decision trees.

A multilayer perceptron neural network in the second stage performs the classification. Different on-line and off-line experimental tests are performed under unbalanced voltage, torque load variations, and short-circuit levels from 1 to 10%. The detection and location of inter-turn short-circuit (ITSC) faults in a three-phase induction motor are carried out in Ref. [4]. This technique employs the phase shifting between the stator currents and their corresponding voltages as input to a support vector machine (SVM), which is in charge of estimating the induction motor operational condition as healthy or with a short-circuit fault in one phase. In Ref. [5], a method for short-circuit fault diagnosis on the stator windings of a three-phase induction motor is presented. The method relies on the symmetrical-component concept. A mathematical model for an induction motor with short-circuit fault is introduced to analyze the rotating machine performance when a faulty condition occurs. A computational model of the motor is developed utilizing Simulink to extract its sequence components for the current and voltage signals. The negative sequence current can provide a fast and conclusive resolution about whether there is or there is not a short-circuit fault in an induction motor. The percentage variation of the negative-sequence current regarding its positive counterpart is the main fault indicator, which is utilized to categorize the short-circuit fault level in the stator windings through a neural network.

This work proposes the detection and classification of short-circuit faults on the stator windings through digital signal processing techniques such as: the Park instantaneous space phasor modulus, the fast Fourier transform (FFT), the multiresolution analysis through wavelet transform, and the phasor analysis of line currents from the stator utilizing fuzzy logic.

2. Electric stator fault diagnosis in induction machines

There are different fault conditions in three-phase induction motors, which can be classified as [6]:

- Rolling bearing faults.
- Stator faults.
- Broken rotor bars or short-circuit-ring breakage.
- Eccentricity faults.

2.1 Incipient stator short-circuit fault

A common fault in three-phase induction motors is the incipient short-circuit condition on the stator windings. In this work, this defect is produced artificially on the stator winding corresponding to the line current I_{ab} of a 3-Hp, 220 V, high-efficiency induction motor from Siemens, by modifying its windings to induce the short-circuit fault. This alteration is performed by rewinding the induction motor and bringing out several taps for generating different short-circuit stator fault scenarios, as portrayed in **Figure 1**.

The faulty condition in the induction motor is created by connecting terminal 7 with tap 5, suppressing 5 out of 45 coils in the corresponding winding, as depicted in the schematic diagram shown in **Figure 2**.

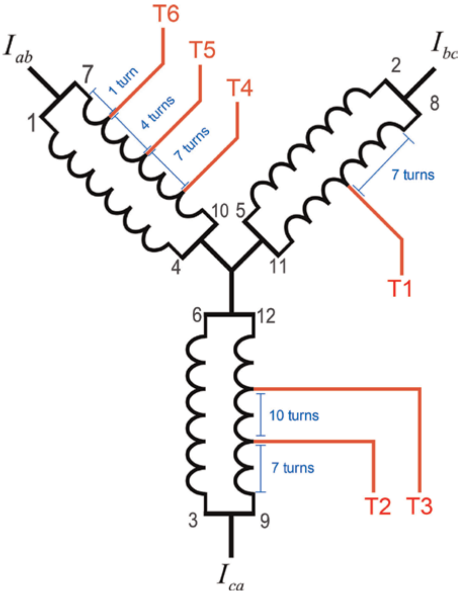


Figure 1.
Stator winding alteration taps.

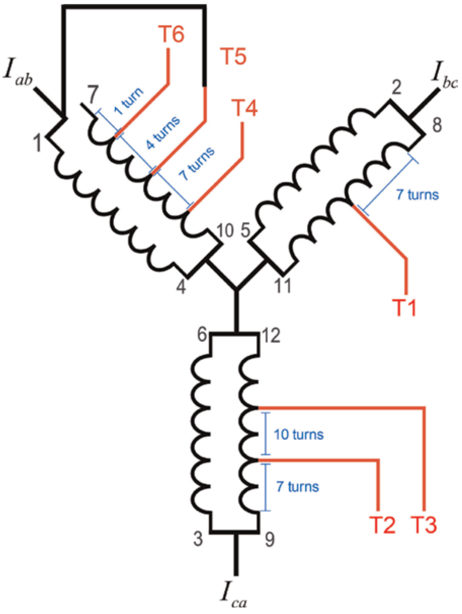


Figure 2.
Stator winding modification to produce a 5-turn short-circuit fault scenario.

Different techniques for induction motor fault diagnosis are assessed by performing 30 trials in healthy condition and 30 experiments in faulty condition, each one containing 128 discrete samples. The signal processing analysis is performed by a digital system based on the 32-bit, floating-point digital signal processor (DSP) *TMS*320*F*28335 from Texas Instruments, which is integrated in the sequential diagram displayed in **Figure 3**.

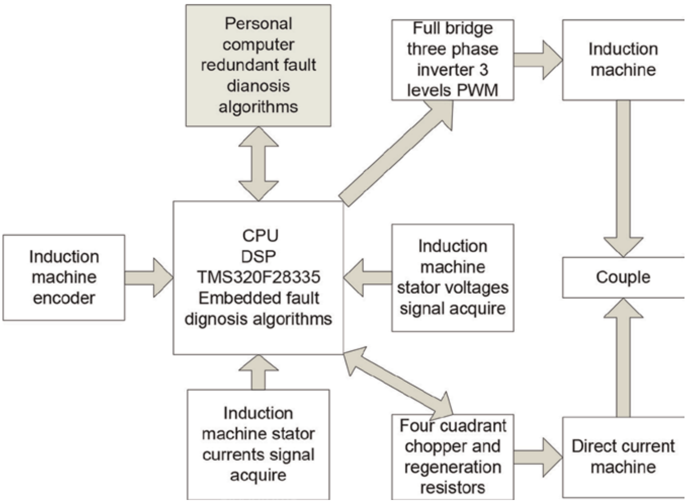


Figure 3.
Digital signal processing flowchart.

Figure 4 shows the experimentally acquired stator currents utilizing a sampling frequency $F_s = 3{,}840$ Hz. This sampling frequency is chosen to be a multiple of the induction machine nominal operation frequency 60 Hz, which produces a rotational speed of 1800 rpm.

The faulty scenario described above is created by reducing the number of coils in the phase-A winding of the induction motor. The coils 41 to 45 (i.e., five coils) are short circuited. The resulting stator currents are acquired experimentally and shown in **Figure 5**.

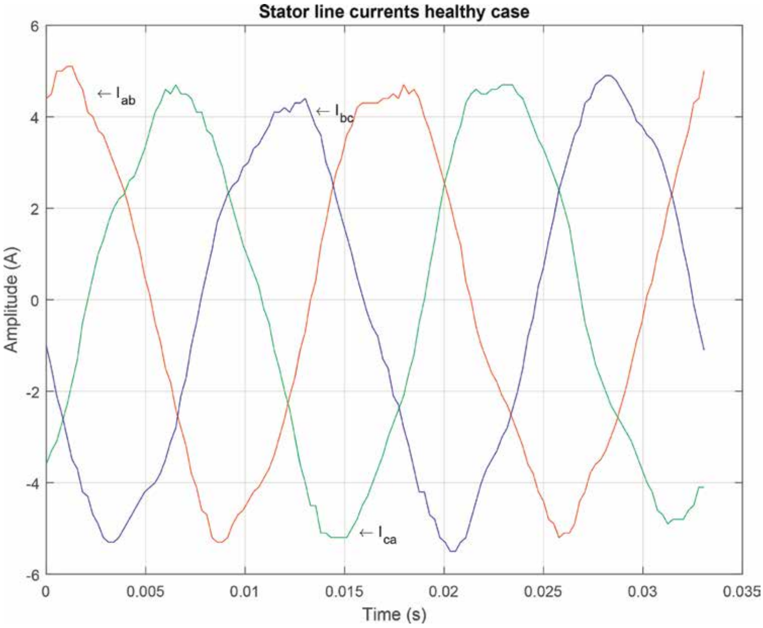


Figure 4.
Stator line currents for the healthy case.

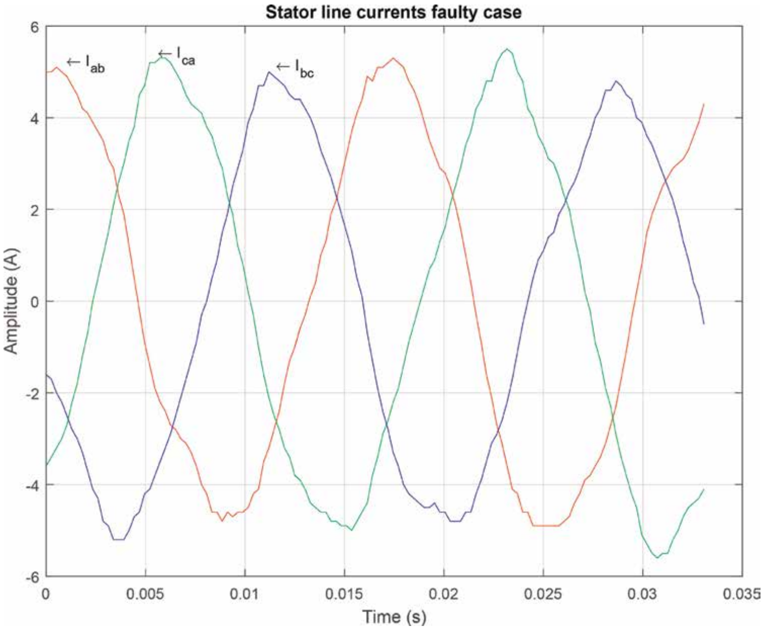


Figure 5.
Stator line currents for the faulty case.

3. Park instantaneous space phasor analysis

In a three-phase, unbalanced system with four wires, the line currents in phase a, phase b, phase c, and neutral are [7]:

$$\begin{aligned} i_a &= i_a^+ + i_a^- + i_a^0 \\ i_b &= i_b^+ + i_b^- + i_b^0 \\ i_c &= i_c^+ + i_c^- + i_c^0 \\ i_N &= i_N^+ + i_N^- + i_N^0 \end{aligned} \tag{1}$$

where i_a, i_b, i_c are the currents in phase a, phase b, phase c, and the neutral, respectively, with positive, negative, and zero sequence components. The instantaneous-space phasor currents for phase a, phase b, and phase c are defined as:

$$\begin{aligned} \tilde{I} &= \tilde{I}^+ + \tilde{I}^- + \tilde{I}^0 \\ \tilde{I}^+ &= \frac{2}{3}\left(i_a^+ + a i_b^+ + a^2 i_c^+\right) = \tilde{I}^+ e^{j(wt+\varphi^+)} \\ \tilde{I}^- &= \frac{2}{3}\left(i_a^- + a i_b^- + a^2 i_c^-\right) = \tilde{I}^- e^{j(-wt+\varphi^-)} \\ \tilde{I}^0 &= \frac{2}{3}\left(i_a^0 + a i_b^0 + a^2 i_c^0\right) = 0 \end{aligned} \tag{2}$$

The phasor squared magnitude is a meaningful quantity related to the instantaneous space phasor (ISP) that helps on separating the positive and negative sequence components as follows:

$$\left|\tilde{I}\right|^2 = \left|\tilde{I}^+ + \tilde{I}^-\right|^2 = \left(\tilde{I}^+\right)^2 + \left(\tilde{I}^-\right)^2 + 2\left(\tilde{I}^+\tilde{I}^-\right)\cos(2wt + \varphi^+ + \varphi^-) \tag{3}$$

The average squared ISP value is computed by:

$$\left\langle \left|\tilde{I}\right|^2 \right\rangle = \left(\tilde{I}^+\right)^2 + \left(\tilde{I}^-\right)^2 \tag{4}$$

From (4), $\left|\tilde{I}\right|^2$ has the distinctive feature of oscillating between a maximum value:

$$\left|\tilde{I}\right|_M^2 = \left(\tilde{I}^+ + \tilde{I}^-\right)^2 \tag{5}$$

and a minimum value:

$$\left|\tilde{I}\right|_m^2 = \left(\tilde{I}^+ - \tilde{I}^-\right)^2 \tag{6}$$

Solving Eqs. (5) and (6), the positive and negative sequence components are, respectively:

$$\begin{aligned} \hat{I}^+ &= \left(\left|\tilde{I}\right|_M + \left|\tilde{I}\right|_m\right)/2 \\ \hat{I}^- &= \left(\left|\tilde{I}\right|_M - \left|\tilde{I}\right|_m\right)/2 \end{aligned} \tag{7}$$

Hence, the ISP limit values can be used for obtaining the positive and negative sequence components. The same precept can be applied to the voltage signals. Therefore, $\tilde{I}^+; \tilde{I}^-; \tilde{V}^+$ y $\tilde{V}^-$ monitoring allows estimating the current and voltage unbalance; for instance, the current unbalance can be computed by:

$$\%IU = 100\frac{\hat{I}^-}{\hat{I}^+} = 100\frac{\left|\tilde{I}\right|_M - \left|\tilde{I}\right|_m}{\left|\tilde{I}\right|_M + \left|\tilde{I}\right|_m} \tag{8}$$

From the aforesaid, the ISP can be used for detecting and classifying electric faults in the induction-motor stator. The ISP or instantaneous space phasor is a widely used tool for analyzing three-phase systems, which allows determining the current signal unbalance in these systems by associating a three-phase reference framework into a two-phase scheme. There are two possible transformations to carry out this conversion: the variable framework scheme known as d-q transform, described in Eq. (9), and the fixed framework equivalence named Concordia or $\alpha - \beta$ transform, defined in Eq. (10). In this work, the Concordia transform is used; therefore, the stator currents are mapped into the two-phase stationary $\alpha - \beta$ reference frame.

$$\begin{bmatrix} i_d \\ i_q \\ i_o \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} \cos(\theta) & \cos\left(\theta - \frac{2\pi}{3}\right) & \cos\left(\theta + \frac{2\pi}{3}\right) \\ -\sin(\theta) & -\sin\left(\theta - \frac{2\pi}{3}\right) & -\sin\left(\theta + \frac{2\pi}{3}\right) \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} \tag{9}$$

where i_a, i_b, i_c are the instantaneous current values from the induction motor stator, and the instantaneous values from the Park vectors in the variable reference

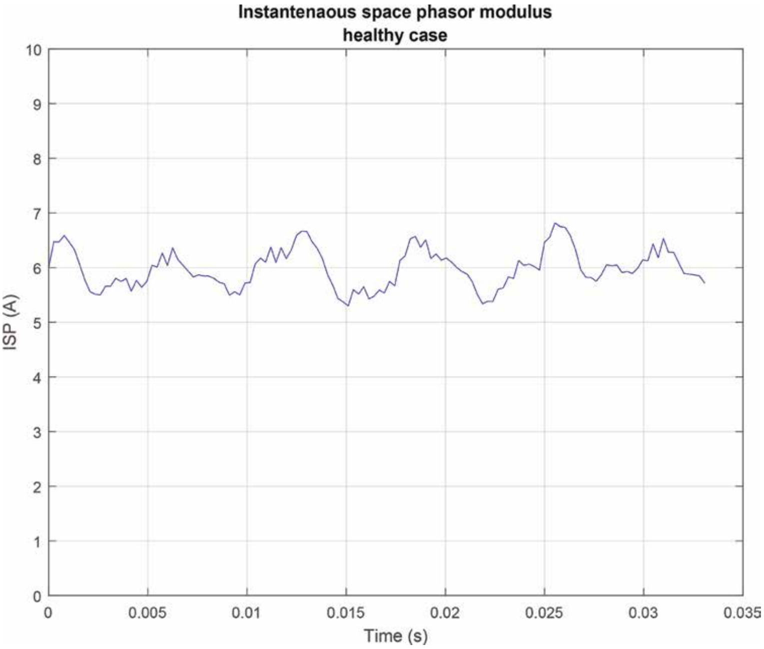


Figure 6.
ISP modulus healthy case.

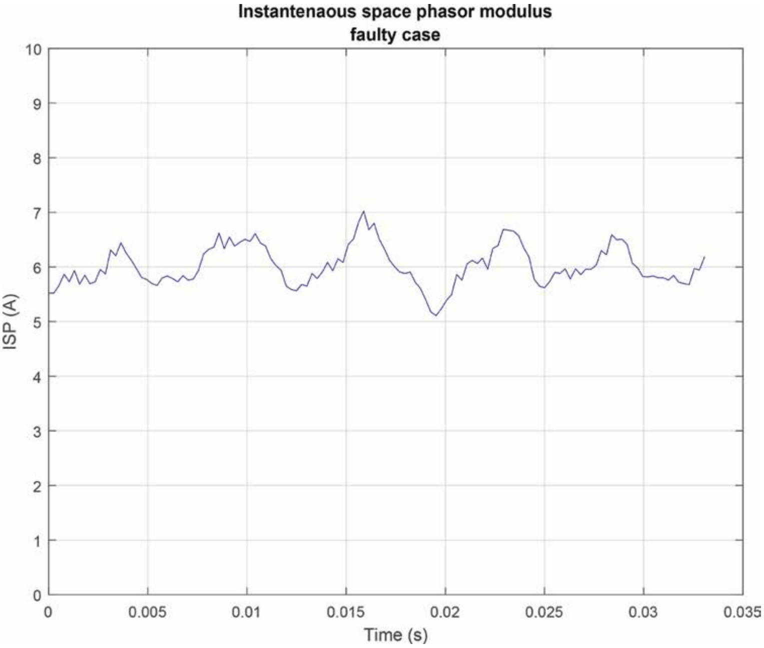


Figure 7.
ISP modulus faulty case.

framework d-q are i_d, i_q and i_0. For the stationary reference frame $\alpha - \beta$, or Concordia transform, the current instantaneous values are i_α, i_β and i_0.

$$\begin{bmatrix} i_\alpha \\ i_\beta \\ i_o \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & \cos\left(-\frac{2\pi}{3}\right) & \cos\left(\frac{2\pi}{3}\right) \\ 0 & -sen\left(-\frac{2\pi}{3}\right) & -sen\left(\frac{2\pi}{3}\right) \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} \tag{10}$$

The modulus of the instantaneous space phasor is obtained by:

$$\left\| i_{\alpha\beta} \right\|^2 = i_\alpha^2 + i_\beta^2 \tag{11}$$

Figure 6 depicts the experimentally obtained modulus of the Park phasor or ISP from the stator currents of a healthy motor utilizing the fixed framework or Concordia transform. **Figure 7** shows the ISP modulus of an incipient fault, where five coils of the phase a winding in the induction motor stator are short circuited.

4. Phasorial analysis

For analyzing short-circuit faults in the stator, the phasor relations in Eq. (13) and in Eq. (12) between line and phase currents of the induction-motor stator are considered in normal operation, utilizing the per-unit system, as it is illustrated in **Figure 8**.

Phasor representation of line currents:

$$\begin{aligned} I_{\text{ab}} &= \sqrt{3}\angle 90° \\ I_{\text{bc}} &= \sqrt{3}\angle -30° \\ I_{\text{ca}} &= \sqrt{3}\angle -150° \end{aligned} \tag{12}$$

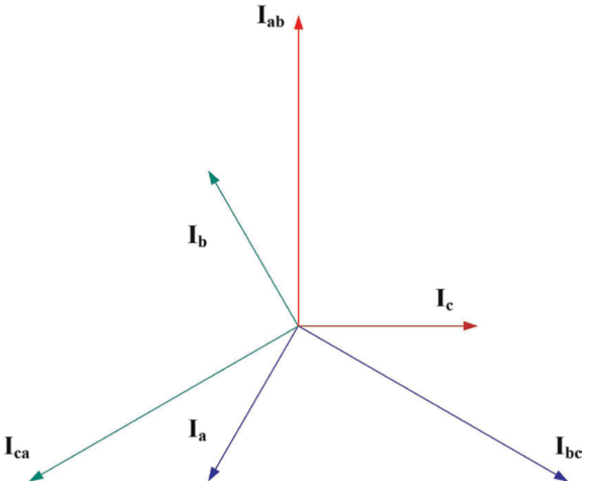


Figure 8.
Healthy current phasors of the line currents and phase currents.

Phasor representation of phase currents:

$$\begin{aligned} I_\mathrm{c} &= 1\angle 0^\circ \\ I_\mathrm{a} &= 1\angle -120^\circ \\ I_\mathrm{b} &= 1\angle 120^\circ \end{aligned} \tag{13}$$

For any balanced three-phase system connected to an induction motor, the supplied currents to the stator must comply with the following requirements because of the absence of a neutral connection:

$$\mathbf{I}_\mathrm{a} + \mathbf{I}_\mathrm{b} + \mathbf{I}_\mathrm{c} = 0 \tag{14}$$

Considering the stated before, the phasor components of each stator current are obtained utilizing both framework systems. It can be observed from **Figure 8**, Eq. (13), and Eq. (12) that a specific phase current is always perpendicular to a particular line current as follows:

$$\begin{aligned} &\mathbf{I}_\mathrm{bc} \perp \mathbf{I}_\mathrm{a} \\ &\mathbf{I}_\mathrm{ca} \perp \mathbf{I}_\mathrm{b} \\ &\mathbf{I}_\mathrm{ab} \perp \mathbf{I}_\mathrm{c} \end{aligned} \tag{15}$$

Figure 9 shows how each current phasor can be represented by components on the direction of corresponding components over the reference frame created by $\mathbf{I}_\mathrm{c}$ and $\mathbf{I}_\mathrm{ab}$ to obtain the expressions in Eq. (16).

$$\begin{aligned} \mathbf{I}_\mathrm{ca} &= \frac{1}{2}(-3\mathbf{I}_\mathrm{c} - \mathbf{I}_\mathrm{ab}) \\ \mathbf{I}_\mathrm{bc} &= \frac{1}{2}(3\mathbf{I}_\mathrm{c} - \mathbf{I}_\mathrm{ab}) \\ \mathbf{I}_\mathrm{ab} &= \mathbf{I}_\mathrm{ab} \\ \mathbf{I}_\mathrm{c} &= \mathbf{I}_\mathrm{c} \\ \mathbf{I}_\mathrm{a} &= \frac{1}{2}(-\mathbf{I}_\mathrm{c} - \mathbf{I}_\mathrm{ab}) \\ \mathbf{I}_\mathrm{b} &= \frac{1}{2}(-\mathbf{I}_\mathrm{a} + \mathbf{I}_\mathrm{ab}) \end{aligned} \tag{16}$$

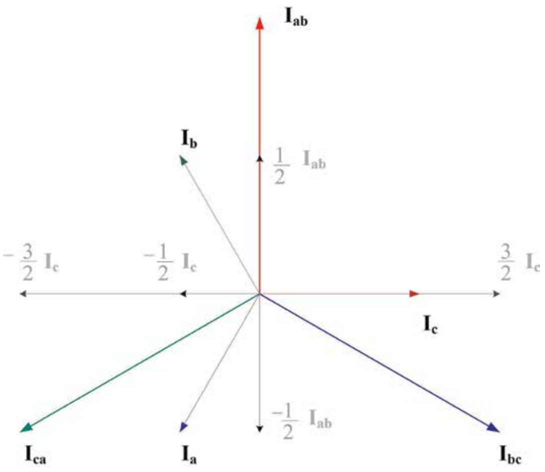


Figure 9.
Current phasor components on the direction of $\mathbf{I}_{ab} \perp \mathbf{I}_{c}$.

Following the same precept, it is possible to obtain equivalent expressions to those in Eq. (16) for the reference systems $\mathbf{I}_{ca}\perp\mathbf{I}_b$ and $\mathbf{I}_{bc}\perp\mathbf{I}_a$.

4.1 Phasorial short-circuit fault detection and isolation

A diagnosis system can be assumed from the corresponding phasor representations of phase and line currents to detect magnitude and angle variations over the current signals when a short-circuit fault takes place on the stator windings of an induction motor. **Figure 10** shows the trajectory variations on the phase and line current phasors when a short-circuit fault occurs on $\mathbf{I}_{ab}$.

If the phasor of line current $\mathbf{I}_{ab}$ increases because there is a short-circuit fault on some coils of the corresponding stator windings, then, the remainder phasor currents shift along parallel lines to the faulty line current. The variations on magnitude and direction of the current phasors are computed through Eq. (16). For instance, if a short-circuit fault takes place on the stator winding corresponding to the line current $\mathbf{I}_{ab}$, the reminder line currents shift along parallel lines in opposite direction to $\mathbf{I}_{ab}$, as depicted in **Figure 10**, where the faulty line current $\mathbf{I}_{ab}$ is represented in a vertical direction upwards; therefore, the line currents $\mathbf{I}_{ca}$ and $\mathbf{I}_{bc}$ shift vertically downward. On the other hand, it is worth it to notice that the magnitude and angle of the phase current $\mathbf{I}_c$ do not change, whereas the magnitude and angle of phase currents $\mathbf{I}_a$ and $\mathbf{I}_b$ do change. This behavior from line and phase current phasors, when a short-circuit fault occurs, emerges from choosing the $\mathbf{I}_{ab}\perp\mathbf{I}_c$ reference framework, which involves the phasor of the faulty line current $\mathbf{I}_{ab}$. Therefore, to diagnose a short-circuit fault it is necessary to take into account the phase and line current behavior. The detection and classification of the faulty condition can be achieved by observing the phase current that remains unchanging. In **Figure 11**, the magnitude and angle variations on the current signals are represented through the phasors in black color, when a faulty condition occurs on the line current $\mathbf{I}_{ab}$. It can be observed that just the phase current $\mathbf{I}_c$ remains constant, which signifies that the line current $\mathbf{I}_{ab}$ increases because there is a faulty condition.

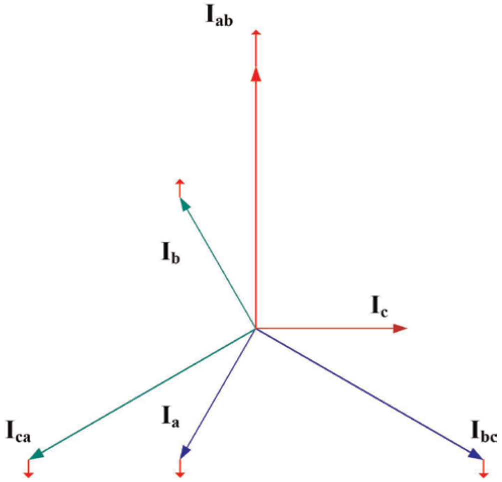


Figure 10.
Directions of current phasor variations in the event of a short-circuit fault in $\mathbf{I}_{ab}$.

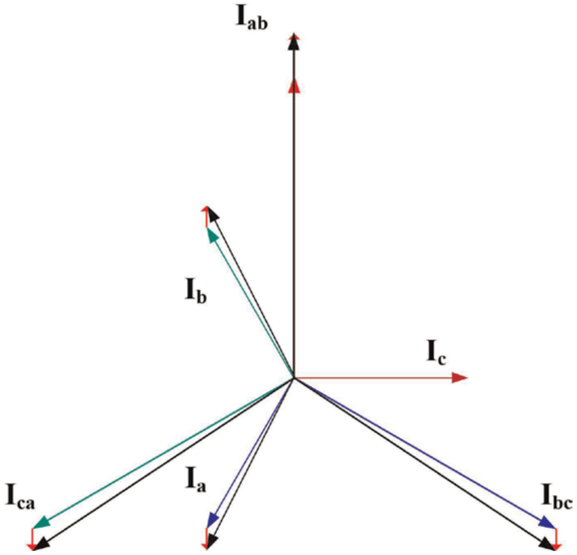


Figure 11.
Displacement of current phasors during a short-circuit fault in $\mathbf{I}_{ab}$.

4.2 Frequency analysis

The diagnosis of stator electric faults in an induction motor can be accomplished through the motor current signature analysis (MCSA), which is performed by examining in the frequency domain the ISP modulus defined in Eq. (11), taking into consideration the following spectral description [8]. Since the negative-sequence components of the current are directly related to the motor asymmetry, the presence of a faulty condition on the stator can certainly be indicated by a spectral analysis of the ISP [9]. For the stator electric fault, its frequency signature F_{stator} in the ISP spectrum takes place on integer multiples n $(n = 1,2,3, \dots, \infty)$ of twice the power-line frequency f_L.

$$F_{stator} = 2nf_L \tag{17}$$

The magnitude of the characteristic fault frequency decreases as the multiple n increases; therefore, the characteristic fault frequency for n = 1 is used in practice because it possesses the highest magnitude; hence:

$$F_{stator} = 2f_L \tag{18}$$

Figure 12 shows the spectra comparison between the healthy and faulty conditions, carried out experimentally, when five coils are short circuited in one stator winding of the induction motor.

Another tool, proposed in Ref. [10], for diagnosing electric faults consists in assessing the phasor representation of the line and phase currents. **Figure 13** shows the phasor frequency spectrum for the healthy and faulty conditions, which allows the diagnosis and classification of the faulty state, different from the ISP modulus that just identifies the fault presence.

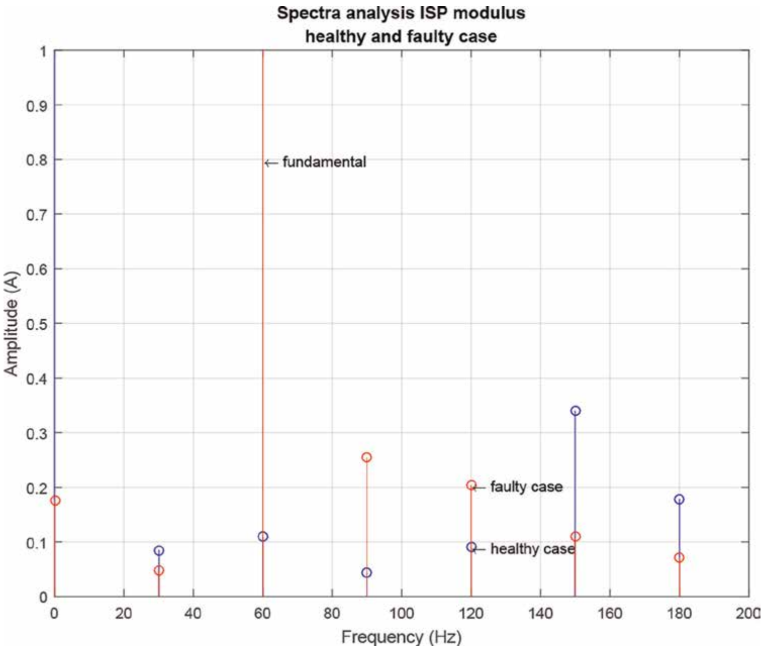


Figure 12.
Comparative analysis of ISP spectra from the healthy and faulty cases.

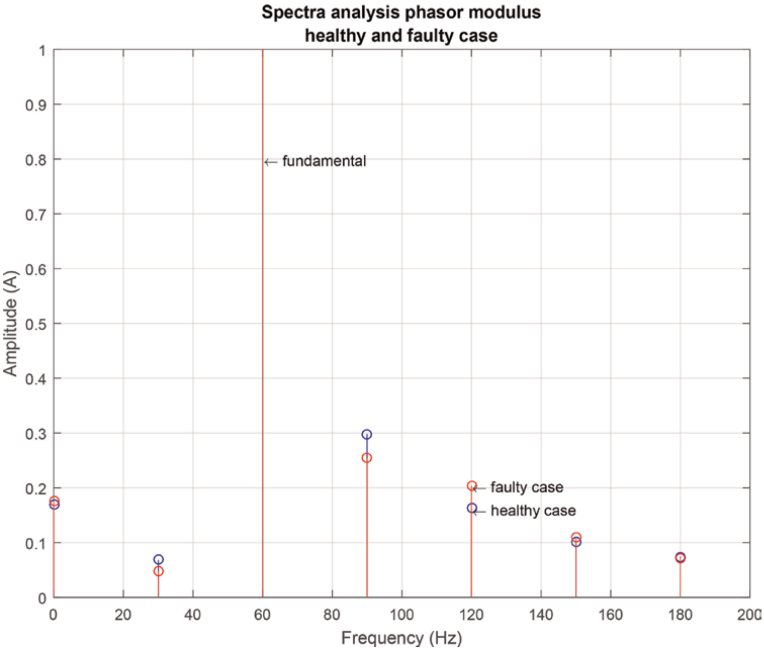


Figure 13.
Comparative analysis of phasor spectra from the healthy and faulty cases.

5. Wavelet analysis

The discrete wavelet transform (DWT) is a mathematical tool, with an effective structure, which allows breaking up a fault signature in the ISP modulus into distinct scales with different levels of resolution [11]. The DWT can be used as back-up tool, besides the frequency spectrum of the ISP modulus and the current phasor analyses, to perform a multiresolution analysis (MRA) for determining the root-mean-square (RMS) (i.e., effective) value of the wavelet coefficients on each frequency band. **Figure 14** portrays graphically the multilevel decomposition.

5.1 Haar mother wavelet function

The simplest mother wavelet function is the Haar function. In its discrete form, it relates to the Haar transformation, which decomposes one signal into two constituent elements, the approximation am and detail dm, with the same length.

The approximation coefficients for $m = 1,2,3, \dots, N/2$, where N is the original signal length, are computed by:

$$a_m = \frac{f_{2m-1} + f_{2m}}{\sqrt{2}} \tag{19}$$

The detail coefficients for $m = 1,2,3, \dots, N/2$ are computed by:

$$d_m = \frac{f_{2m-1} - f_{2m}}{\sqrt{2}} \tag{20}$$

The level-1 Haar transform is computed through:

$$f \overset{H_1}{\rightarrow} \left(a^1 | d^1\right) \tag{21}$$

Its corresponding inverse transformation is calculated by:

$$f = \left(\frac{a_1 + d_1}{\sqrt{2}}, \frac{a_2 - d_2}{\sqrt{2}}, \dots, \frac{a_{N/2} + d_{N/2}}{\sqrt{2}}, \frac{a_{N/2} - d_{N/2}}{\sqrt{2}}\right) \tag{22}$$

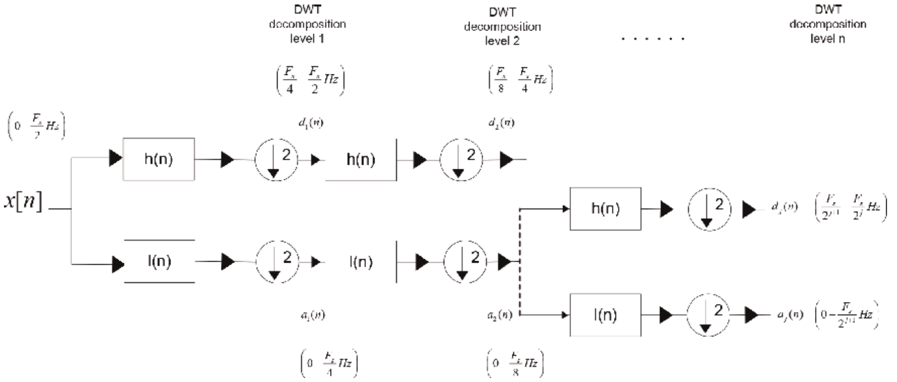


Figure 14.
Scheme for wavelet multiresolution analysis.

The second transformation level is obtained by decomposing the level-1 approximation coefficients a_1 to produce the level-2 approximation a_2 and detail d_2 coefficients.

$$f \xrightarrow{H_2} \left(a^2|d^2|d^1\right) \tag{23}$$

The Haar mother wavelet is used for obtaining multiple approximation levels to perform the MRA as follows:

$$\begin{aligned}
Wh1_1^1 &= \left(\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}, 0, 0, \dots, 0\right) \\
Wh1_2^1 &= \left(0, 0, \frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}, 0, 0, \dots, 0\right) \\
Wh1_2^1 &= \left(0,0,0,0, \frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}, 0, 0, \dots, 0\right) \\
&\vdots \\
Wh1_{N/2}^1 &= \left(0, 0, \dots, 0, \frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}\right)
\end{aligned} \tag{24}$$

The corresponding decomposition levels are given by:

$$\begin{aligned}
Vh1_1^1 &= \left(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}, 0, 0, \dots, 0\right) \\
Vh1_2^1 &= \left(0, 0, \frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}, 0, 0, \dots, 0\right) \\
Vh1_2^1 &= \left(0,0,0,0, \frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}, 0, 0, \dots, 0\right) \\
&\vdots \\
Vh1_{N/2}^1 &= \left(0, 0, \dots, 0, \frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}\right)
\end{aligned} \tag{25}$$

5.2 Experimental multiresolution analysis

The short-circuit fault in the induction-motor stator is inspected exhaustively through the MRA of the ISP utilizing the Haar mother wavelet. **Table 1** shows a

Level	Frequency band (Hz)	Healthy (RMS)	Faulty (RMS)
1	960.0–1920.0	0.119505	0.492522
2	480.0–960.0	0.223134	1.378206
3	240.0–480.0	0.461718	3.789614
4	120.0–240.0	0.989168	10.020093
5	60.0–120.0	0.533128	2.238111
6	30.0–60.0	0.597124	50.795688
7	15.0–30.0	0.619910	0.220971

Table 1.
RMS values from the MRA on the ISV utilizing the level-1, Haar-wavelet coefficients.

Level	Frequency band (Hz)	Healthy (RMS)	Faulty (RMS)
1	960.0–1920.0	0.113843	0.532535
2	480.0–960.0	0.243338	1.395585
3	240.0–480.0	0.501377	3.839840
4	120.0–240.0	0.852141	10.429046
5	60.0–120.0	0.731468	3.617190
6	30.0–60.0	0.984967	48.763207
7	15.0–30.0	0.492005	0.159099

Table 2.
RMS values from the MRA on the phasors utilizing the level-1, Haar-wavelet coefficients.

comparative analysis of RMS values for the wavelet coefficients on each frequency bad. The induction motor has an 1800 rpm rotational speed, with a 60-Hz fundamental frequency from the power supply. Since the induction motor has two poles, the fault characteristic frequency appears at twice its fundamental frequency $f_{fault} = 120$ Hz; hence, the RMS value increases from 0.989168 for the healthy case to 10.020093 for the faulty state. On the other hand, it is worth to notice that there is a significant difference between the wavelet-coefficients RMS values for a healthy and a faulty motor in the 30–60 Hz frequency band, which are 0.597124 and 50.795688, respectively, because of the fundamental-frequency variation in the induction motor.

The phasor analysis, which can detect and classify the faulty condition, is applied through a MRA for diagnosing short-circuit faults in the induction-motor stator, too. In **Table 2**, a significative change in the RMS value of the 120–240 Hz frequency band is observed, from 0.852141 for the healthy condition to 10.429046 for the faulty state. On the other hand, it is worth to notice the variation between the RMS values of the wavelet coefficients in the 30–60 Hz frequency band, from 0.597124 to 50.795688, for a healthy and a faulty condition, respectively, due to the fundamental-frequency variation in the induction motor.

6. Fault diagnosis and classification through fuzzy logic

Fuzzy logic offers the advantage of recognizing phase and line-current asymmetries. Since the line currents are obtained straight forward from the induction-motor stator, the phase currents must be estimated to compute the RMS values that are used as input to the fuzzy-logic system, which is based on the Mamdani model, to detect and classify the short-circuit fault, as shown in **Figure 15**.

In **Table 3**, the line-current RMS values do not show significant changes among them, neither the phase currents do. **Figure 16** shows the obtained results from introducing the RMS values to the fuzzy-logic system, which detects the induction motor condition as healthy by providing an outcome of 0.25 that is computed through the centroid method. The discourse universe in the fuzzy system takes the values from 0 to 1 for identifying the induction motor condition. For the healthy condition, the membership zone goes from 0 to 0.5, whereas the faulty condition region goes from 0.5 to 1.

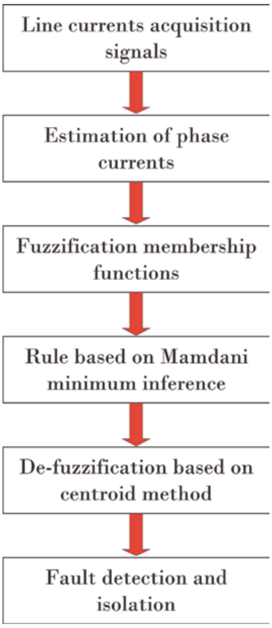


Figure 15.
Fuzzy logic algorithm based on Mamdani model.

Trial	I_a	I_b	I_c	I_{ab}	I_{ca}	I_{bc}
1	2.02	1.99	1.93	3.49	3.45	3.36
2	1.98	1.97	1.94	3.41	3.51	3.37
3	2.02	1.98	1.97	3.52	3.52	3.41
4	1.95	1.97	1.93	3.35	3.39	3.41
5	1.93	1.92	1.96	3.25	3.42	3.47

I_a, I_b, I_c *are phase currents estimated from the stator line currents.*
I_{ab}, I_{ca}, I_{bc} *are stator line currents.*

Table 3.
RMS values of line and phase currents in healthy condition.

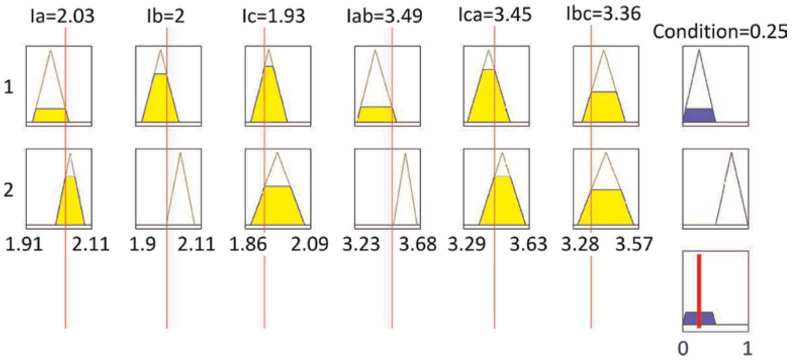


Figure 16.
Detection of the healthy operational condition of the induction-motor stator through fuzzy logic.

Trial	I_a	I_b	I_c	I_{ab}	I_{ca}	I_{bc}
1	2.04	2.01	1.98	3.55	3.49	3.42
2	2.09	2.03	1.99	3.62	3.47	3.43
3	2.07	2.02	1.88	3.58	3.38	3.29
4	1.99	2.02	1.99	3.51	3.49	3.49
5	2.02	2.06	1.99	3.57	3.50	3.46

$\mathbf{I}_a, \mathbf{I}_b, \mathbf{I}_c$ *are phase currents estimated from the stator line currents.*
$\mathbf{I}_{ab}, \mathbf{I}_{ca}, \mathbf{I}_{bc}$ *are stator line currents.*

Table 4.
RMS values of line and phase currents in faulty condition.

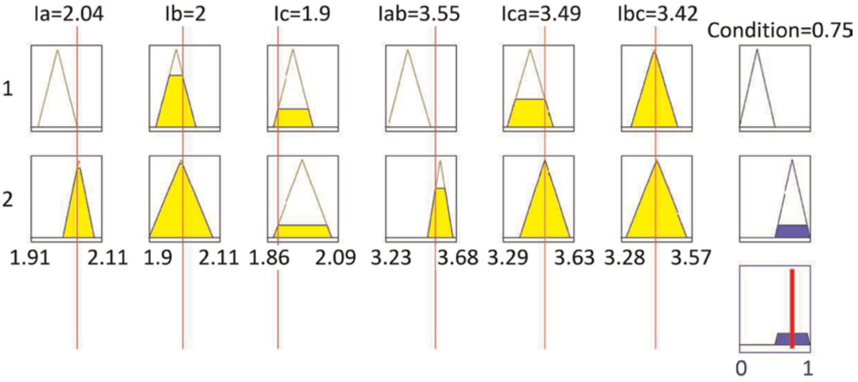


Figure 17.
Detection and classification of the short-circuit fault on the induction-motor stator utilizing fuzzy logic.

Connecting the Tap 5 as described in **Figure 2** generates a short-circuit fault on 5 out of 45 coils in the corresponding winding; therefore, the line current I_{ab} increases provoking the escalation of phase currents $\mathbf{I_a}$ and $\mathbf{I_b}$; however, $\mathbf{I_c}$ remains unaffected as described in **Table 4**. In this way, the fuzzy system is able to detect and classify the faulty condition by producing an outcome of 0.75, which points out that there is a short-circuit fault as depicted in **Figure 17**.

7. Conclusions

This work presents several techniques for detecting short-circuit faults in three-phase induction motors. Among these approaches are spectral-frequency tools as discrete Fourier transform and discrete wavelet transform, which employ the instantaneous space phasor (ISP) modulus without achieving the fault classification. Hence, a novel technique relaying on the phasor analysis of the three-phase currents is put forward, which in addition to detect the faulty condition, it is capable of classifying through a fuzzy system with a Mamdani model by examining the current RMS values as input data.

Acknowledgements

This work was partially supported by DEPI-ITA-TecNm (Departamento de Posgrado e Investigacion—Instituto Tecnologico de Aguascalientes/Tecnologico Nacional de Mexico), Departamento de Produccion y Seguridad Industrial, Universidad Tecnologica de Aguascalientes, Aguascalientes, Multidisciplinary Studies Department, Engineering Division, Campus Irapuato-Salamanca, University of Guanajuato.

Notes/thanks/other declarations

Thanks to the Consejo Nacional de Humanidades Ciencias y Tecnologias (CONAHCYT) for the Doctoral program scholarship.

Abbreviations

MCSA	Motor Current Signature Analysis
IEEE	The Institute of Electrical and Electronics Engineers
EPRI	Electric Power Research Institute
ITSC	Inter Turn Short Circuit
SVM	Support Vector Machine
IM	Induction Machine
ISP	Instantaneous Space Phasor
MRA	Multiresolution Analysis
DSP	Digital Signal Processor

Nomenclature

T1	Stator motor winding artificial fault tap 1
T2	Stator motor winding artificial fault tap 2
T3	Stator motor winding artificial fault tap 3
T4	Stator motor winding artificial fault tap 4
T5	Stator motor winding artificial fault tap 5
T6	Stator motor winding artificial fault tap 6
T7	Stator motor winding artificial fault tap 7
I_a	Phase A current
I_b	Phase B current
I_c	Phase C current
I_{ab}	Phase A to B line current
I_{bc}	Phase B to C line current
I_{ca}	Phase C to A line current
F_{stator}	Stator Fault
f_L	Line frequency
I_d	d current in the dq reference frame
I_q	q current in the dq reference frame
I_α	α current in the α-β reference frame
I_β	β current in the α-β reference frame
I_0	0 current in the dq reference frame and α-β reference frame

Author details

Francisco Javier Villalobos-Pina[1*†], Josue Augusto Reyes-Malanche[2†],
Eduardo Cabal-Yepez[3†] and Efrain Ramirez-Velasco[1†]

1 Departamento de Ingeniería Electrica Electronica, TecNm/Instituto Tecnologico de Aguascalientes, Aguascalientes, Mexico

2 Departamento de Produccion y Seguridad Industrial, Universidad Tecnologica de Aguascalientes, Aguascalientes, Mexico

3 Multidisciplinary Studies Department, Engineering Division, Campus Irapuato-Salamanca, University of Guanajuato, Guanajuato, Mexico

*Address all correspondence to: francisco.vp@aguascalientes.tecnm.mx

† These authors contributed equally.

IntechOpen

References

[1] Garcia-Calva T, Morinigo-Sotelo D, Fernandez-Cavero V, Romero-Troncoso R. Early detection of fault in induction motors - a review. MDPI Energies. 2022; **15**(1):1713-1721. DOI: 10.3390/en15217855

[2] Jigyasu R, Mathew L, Sharma A, Chatterji S. A review of condition monitoring and fault diagnosis methods for induction motor. In: Second International Conference on Intelligent Computing and Control Systems (ICICCS 2018). IEEE Xplore; 2018. pp. 1713-1721. DOI: 10.1109/ICCONS.2018.8662833. ISBN:978-1-5386-2841-6

[3] Bazan G, Scalassara P, Endo W, Goedtel A, Palácios R, Godoy W. Stator short-circuit diagnosis in induction motors using mutual information and intelligent systems. 2019;**66**(4): 3237-3246. DOI: 10.1109/TIE.2018.2840983

[4] Bensaoucha S, Brik Y, Moreau S, Bessedik S, Ameur A. Induction machine stator short-circuit fault detection using support vector machine. The International Journal for Computation and Mathematics in Electrical and Electronic Engineering. 2021;**40**(3): 373-389. DOI: 10.1108/COMPEL-06-2020-0208

[5] Rajamany G, Srinivasan S, Rajamny K, Natarajan RK. Induction motor stator inter turn short circuit fault detection in accordance with line current sequence components using artificial neural network. Journal of Electrical and Computer Engineering. Hindawi. 2019;**2019**:1-11. DOI: 10.1155/2019/4825787

[6] Nandi S, Toliyat HA, Li X. Condition monitoring and fault diagnosis of electrical motors a review. IEEE Transactions on Energy Conversion. 2007;**20**(4):719-729

[7] Milanez DL, Emanuel AE. The instantaneous-space-phasor a powerful diagnosis tool. IEEE Transactions on Instrumentation and Measurement. 2003;**51**(1):143-148

[8] Cruz S, Cardoso AJM. Multiple reference frames theory: A new method for the diagnosis of stator faults in three-phase induction motors. IEEE Transactions on Energy Conversion. 2005;**20**(1):289-299

[9] Cruz SMA, Cardoso AJM. Stator winding fault diagnosis in three-phase synchronous and asynchronous motors, by the extended Park's vector approach. IEEE Transcations on Industry Applications. 2001;**37**(5):1227-1233

[10] Reyes-Malanche JA, Villalobos-Pina FJ, Ramırez-Velasco E, Cabal-Yepez E, Hernandez-Gomez G, Lopez-RamirezCruz M. Short-circuit fault diagnosis on induction motors through electric current phasor analysis and fuzzy logic. MDPI Energies. 2023; **16**(516):1-15. DOI: 10.3390/en16010516 [Accessed: January 03, 2023]

[11] Walker JS. A Primer on Wavelets and their Scientific Applications. 2nd ed. Chapman & Hall/CRC Taylor and Francis Group; 2008

Chapter 12

A Comparative Analysis of Image Encoding of Time Series for Anomaly Detection

Chris Aldrich

Abstract

A novel approach to anomaly detection in time series data is based on the use of multivariate image analysis techniques. With this approach, time series are encoded as images that make them amenable to analysis by pretrained deep neural networks. Few studies have evaluated the merits of the different image encoding algorithms, and in this investigation, encoding of time series data with Euclidean distance plots or unthresholded recurrence plots, Gramian angular fields, Morlet wavelet scalograms, and an ad hoc approach based on the presentation of the raw time series data in a stacked format are compared. This is done based on three case studies where features are extracted from the images with gray level co-occurrence matrices, local binary patterns and the use of a pretrained convolutional neural network, GoogleNet. Although no method consistently outperformed all the other methods, the Euclidean distance plots and GoogleNet features yielded the best results.

Keywords: image encoded time series, wavelet spectrograms, Gramian angular fields, Euclidean distance plots, recurrence plots, gray level co-occurrence matrices, local binary patterns, GoogleNet

1. Introduction

Anomaly detection in nonlinear time series is a field that focuses on identifying unusual or abnormal patterns in time series data that exhibit nonlinear behavior. Time series data refers to a sequence of observations collected over time, and nonlinear time series data refers to data that does not follow a linear trend or relationship.

Anomalies, also known as outliers or aberrations, are data points or patterns that deviate significantly from the expected or normal behavior of the time series. These anomalies may represent critical events, irregularities, faults, or rare occurrences that are of interest for further investigation.

Detecting anomalies in nonlinear time series is a difficult task, since traditional methods based on linear assumptions or statistical measures may not effectively capture the nonlinear relationships and patterns present in the data.

These challenges can be addressed by use of nonlinear models and time series analysis methods specifically designed to handle nonlinear data. They aim to capture the intricate structures, dependencies, and irregularities present in the time series and identify deviations from the expected behavior. This can be accomplished by learning a representation of the original or normal time series behavior that can be used as a reference for the identification of anomalous patterns in the time series.

The quality of these models depends on the quality of the features that invariably have to be extracted from the time series to enable analysis. In this chapter, imaging of time series is empirically investigated by comparing different methods of image generation, as well as the extraction of features from these images. One of the major drivers in time series imaging is that this allows the use of pretrained convolutional neural networks and currently emerging vision transformers [1] that have recently redefined the state of the art in image analysis.

This is a major advantage in time series classification, where end-to-end learning direct from the images can be used. In what is referred to as transfer learning, this often entails further training of some of the feature layers of the networks, which could markedly improve the performance of the model, even if relatively few data are available for this purpose [2].

In the context of anomaly or change point detection in time series data, where labeled data are not necessarily available, this is not a direct option. Under these circumstances, the problem can be treated as a multivariate statistical process monitoring problem, where images are used as a basis for comparison. This would require a reference time series or sets of time series from which image features could be extracted. New time series data would subsequently be encoded as images, and the features extracted from these time series could then be compared with those from the reference time series in some formal monitoring scheme.

In this investigation, such formal monitoring approaches are not considered. Instead, the focus is on the comparative merits of the imaging and feature extraction methods, as these would be critical to the performance of any monitoring scheme.

The rest of the chapter is organized as follows. In the next section, a brief overview of image encoding of time series data is given. This is followed by a summary of the analytical methodology used in the investigation. The next three sections deal with three different case studies. In the final section, the results are discussed and the conclusions of the study are summarized.

2. Imaging of time series data

The rationale for using image-encoding techniques of time series data is that it allows for the use of well-established image processing and computer vision techniques for analysis. It also provides a visual representation that can be easily interpreted by humans, aiding in the understanding of complex temporal patterns.

Over the last decade, different approaches have been investigated. Broadly, these can be categorized as dealing with univariate or multivariate data, as outlined in **Figure 1**. Methods can be grouped into imaging of univariate and multivariate time series. Some methods, such as Euclidean distance plots and their thresholded version, recurrence plots, can deal naturally with time series in more than one dimension, while others, like Gramian angular field plots, cannot. Regardless, multivariate time series can be handled by stacking or multibanding of images obtained from individual time series, for example [3, 4].

Some feature extraction methods, such as pretrained convolutional neural networks that are increasingly used with imaged time series, can also naturally processes

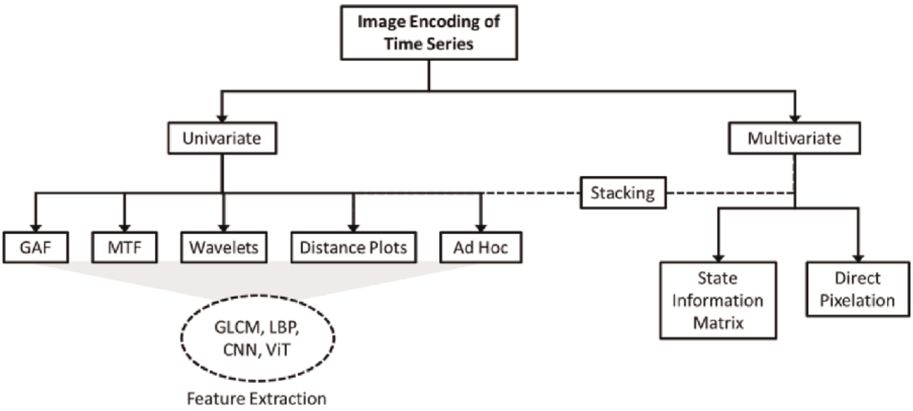


Figure 1.
Image-encoding methods for time series.

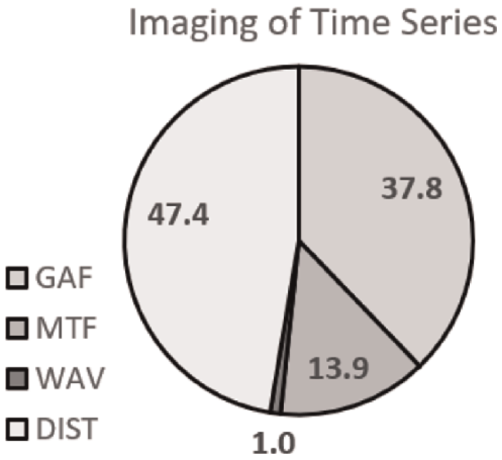


Figure 2.
Comparative analysis of the use of the most popular approaches for image encoding of time series data.

image triplets, so in principle, they could handle up to three different types of (grayscale) images (**Figure 2**).

Gramian angular fields: GAFs have been one of the first approaches used to encode time series for multivariate image analysis [5] and remains the most popular approach for this purpose. Generally, GAF has been used in diverse fields ranging from tool wear classification with CNNs [6] to identification of nontechnical losses in power systems [7], recognition of wearable sensor-based human activity [8], fault detection in transmission lines [9], time series classification with vision transformers [10], and so on.

Distance and recurrence plots [11, 12]: Recurrence plots are graphical representations of the recurrence of a signal's pattern. They are created by comparing each point in a signal to all other points and determining if they are close enough to be considered recurrent. Recurrence plots display the time points at which recurrences occur as black pixels or dots on a two-dimensional grid, providing insights into the signal's temporal structure and periodicity.

They are essentially thresholded distance plots, with the Euclidean distance as the most popular measure of proximity between points. Recurrence plots underpin recurrence quantification analysis, a well-established approach to time series analysis with applications in a wide range of disciplines. Although recurrence plots per se could also be used directly as a basis for feature extraction by means other than RQA, this has approach has not been widely established. On the other hand, unthresholded distance plots are seeing growing use in time series analysis [13–15].

Wavelet scalograms [16–19]: Wavelet analysis involves decomposing a signal into different frequency components using wavelet transforms. By representing the resulting coefficients as an image, known as a wavelet spectrogram, both time and frequency information can be captured simultaneously. The intensity or color of each pixel represents the magnitude of the corresponding frequency component. Image encoding of time series data with wavelet scalograms is relatively new and not as well-established as encoding with Gramian angular fields or recurrence plots, for example.

Ad hoc methods: These could include methods specific to the domain, such as heat maps for furnaces, day-hour power consumption heat maps [20], or climatic events with a time axis and a spatial dimension and time delay embeddings [21, 22] that are not captured in recurrence plots.

3. Analytical methodology

The analytical methodology is shown schematically in **Figure 3**. It consists of several steps. The first is segmentation of the time series by a moving window of a user-specified length, b, moving with a step size, s. If $b = s$, the time series is segmented into a number of contiguous segments. Otherwise, if $b > s$, the segments are overlapping. In some predictive monitoring schemes, it may also be possible to use $b < s$ parameter configurations.

3.1 Image encoding

Four methods were used to encode the time series into images. These were based on Euclidean distance matrices of the time series, Gramian angular field matrices, imaging derived from Toeplitz matrices of time series, as well as wavelet scalograms, as briefly summarized below.

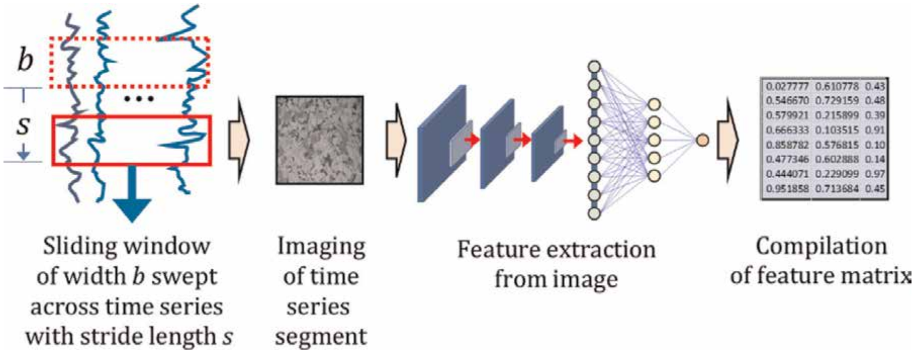


Figure 3.
General approach to image encoding and feature extraction from time series.

3.1.1 Euclidean distance plots

Euclidean distance plots are obtained from the distances between pairs of points in the time series segment, as shown in **Figure 4**. Imaging of the pairwise distance values (d_{ij}) yields a Euclidean distance plot. Thresholded Euclidean distance plots, also referred to as recurrence plots, are widely used in recurrence quantification analysis of time series.

3.1.2 Gramian angular fields

Gramian angular fields can serve as a basis for imaging signals, particularly time series data. It represents the dynamics of a signal by encoding pairwise angles between its data points in a visual form. GAF provides a novel way to capture temporal relationships and patterns within a signal.

To obtain the Gramian angular field of a time series, $y_i \in [-1, 1]$, for all $i = 1, 2, \ldots N$ and time stamp $t_i = 0, 1, \ldots N$, the time series in the Cartesian coordinate system is converted to a polar coordinate system with the following equations. In these equations, N serves as a constant to regularize the span of the polar coordinates.

$$\theta_i = \arccos(y_i) \tag{1}$$

$$r_i = \frac{t_i}{N} \tag{2}$$

$$GAF = \begin{bmatrix} \cos(\theta_1 + \theta_1) & \cos(\theta_1 + \theta_2) & \ldots & \cos(\theta_1 + \theta_N) \\ \cos(\theta_2 + \theta_1) & \cos(\theta_2 + \theta_2) & \cdots & \cos(\theta_2 + \theta_N) \\ \vdots & \vdots & & \vdots \\ \cos(\theta_N + \theta_1) & \cos(\theta_N + \theta_2) & \cdots & \cos(\theta_N + \theta_N) \end{bmatrix} \tag{3}$$

Finally, the Gramian angular field (GAF) matrix is imaged, with the elements of the field serving as pixel values.

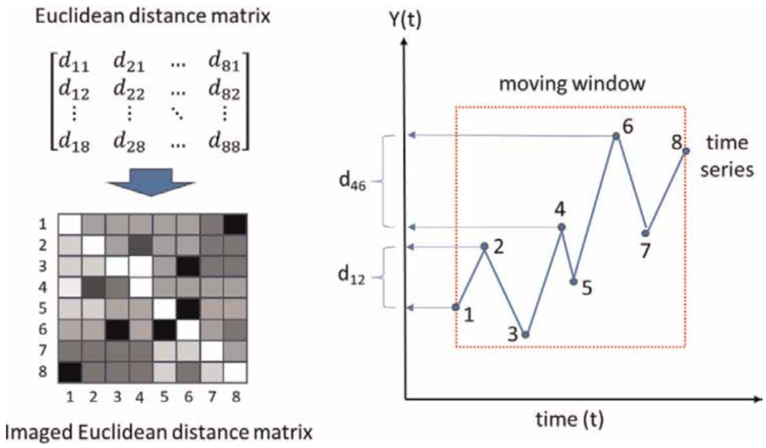


Figure 4.
Derivation of Euclidean distance plots from time series data.

3.1.3 Toeplitz stacking of time series segments

As an ad hoc approach to imaging of the time series, it is stacked by generating a Toeplitz or diagonal-constant matrix from the time series, $\mathbf{Y} = \{y_i\} \in \mathbb{R}^N, i = 1, 2, \ldots N$ in a circulant form as indicated by Eq. (4)

$$T = \begin{bmatrix} y_1 & y_2 & \cdots & y_N \\ y_2 & y_1 & \cdots & y_{N-1} \\ \vdots & \vdots & \ddots & \vdots \\ y_N & y_{N-1} & \cdots & y_1 \end{bmatrix} \tag{4}$$

3.1.4 Wavelet scalograms

Morlet wavelets were used to generate wavelet spectrograms of the time series segments. Morlet wavelets are a type of complex-valued wavelet that combines the benefits of both wavelets and Fourier analysis and are commonly used in time-frequency analysis and signal-processing tasks to extract information about the time-varying frequency content of a signal.

Mathematically, the Morlet wavelet can be defined as:

$$\psi(t) = Ae^{i\omega t}e^{-\frac{t^2}{2\sigma^2}} \tag{5}$$

where $\psi(t)$ represents the complex-valued Morlet wavelet at time t, A is a normalization constant, ω is the angular frequency, and σ is the standard deviation of the Gaussian envelope.

In practice, Morlet wavelets with basis functions as shown in **Figure 5** are often used in wavelet transforms or continuous wavelet transforms (CWT). The wavelet

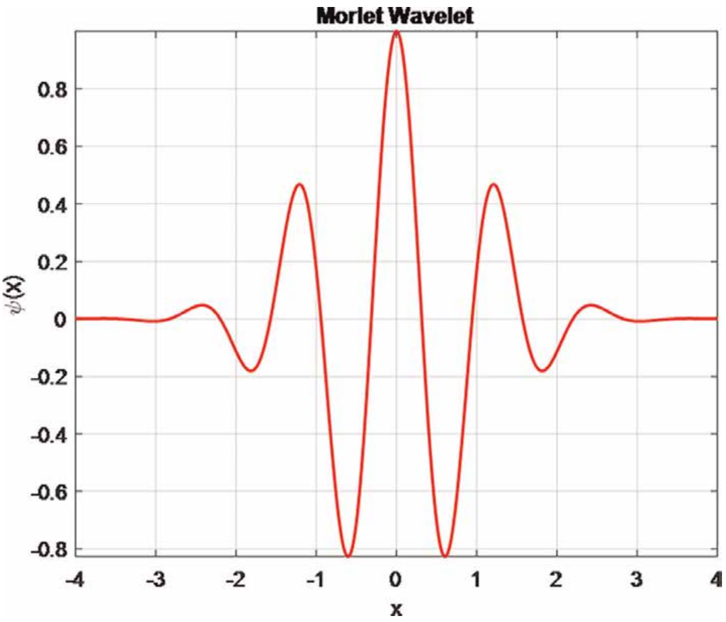


Figure 5.
Morlet waveform. (Source: Author).

transform convolves the Morlet wavelet with the signal of interest at different scales or frequencies, generating a time-frequency representation known as a scalogram. This scalogram provides insights into the signal's frequency content and how it evolves over time, and it is these images that were used to encode the time series segments.

3.2 Feature extraction

3.2.1 Gray-level co-occurrence matrices

Gray-level co-occurrence matrices (GLCMs) are statistical tools used to describe the spatial relationships between pixel intensities in an image. GLCMs capture the frequency of occurrence of pairs of pixel values at specific spatial displacements or directions within an image, as indicated in **Figure 6**. As such, they represent the joint probability distributions of pixel intensity values for a given set of pixels and their respective neighboring pixels within a defined distance and direction.

More formal descriptions can be found elsewhere, for instance, Löfstedt et al. [23], but basically, GLCMs and features derived from them are generated by:

a. Choosing a distance and direction (e.g., horizontal, vertical, or diagonal) for the pixel pairs.

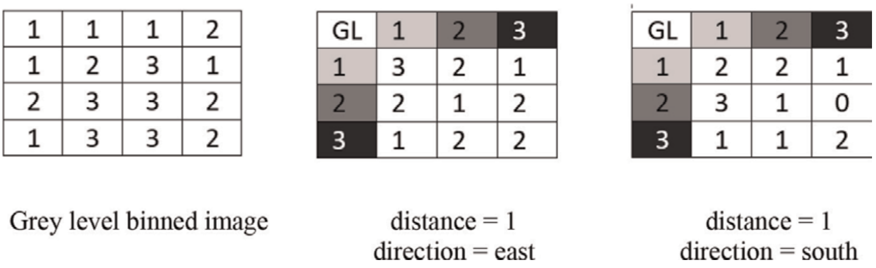
b. Creating a matrix with the number of rows and columns equal to the number of gray levels in the image (usually 256 for 8-bit images).

c. Checking the designated neighboring pixel for each pixel in the image according to a chosen distance and direction.

d. Incrementing the corresponding element in the GLCM matrix based on the pixel pair values.

e. Normalization of the GLCM, so the elements of the matrix can be interpreted as probability estimates.

The resulting GLCM matrix can then be used to derive various texture features, of which the Haralick features or texture descriptors are most commonly used [24]. Four of these features were used in this study, namely, contrast, correlation, energy, and homogeneity, as defined by Eqs. (1)–(4). In all cases, the number of gray levels that were used in the gray level co-occurrence matrices was 8 and the offset between pixels was a unit distance at an angle of zero degrees.



1	1	1	2
1	2	3	1
2	3	3	2
1	3	3	2

Grey level binned image

GL	1	2	3
1	3	2	1
2	2	1	2
3	1	2	2

distance = 1
direction = east

GL	1	2	3
1	2	2	1
2	3	1	0
3	1	1	2

distance = 1
direction = south

Figure 6.
Gray-level image (left) from which two gray-level co-occurrence matrices (middle and right) are derived.

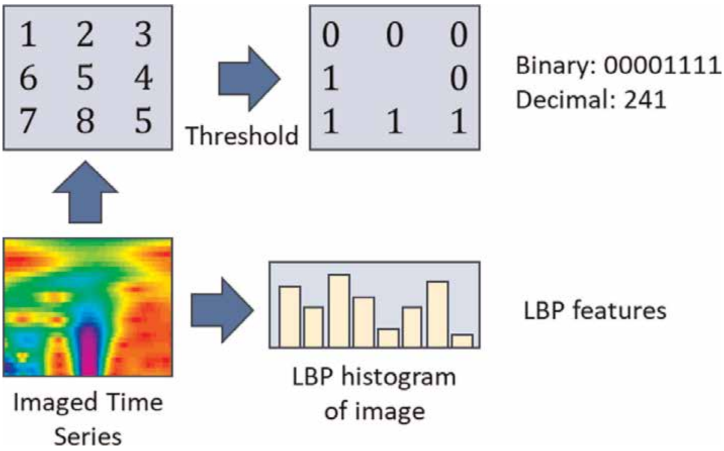


Figure 7.
Feature extraction from images with local binary patterns.

Contrast: Captures the intensity contrast between neighboring pixels.

$$CON = \sum_i \sum_j (i - j)^2 G_{ij} \tag{6}$$

Correlation: Measures the linear dependency between pixel pairs with m_i and s_i (m_j and s_j), indicating the mean and standard deviation of the elements in the i'th row (j'th column) of the co-occurrence matrix.

$$COR = \sum_i \sum_j \frac{(i - m_i)(j - m_j)G_{ij}}{s_i s_j} \tag{7}$$

Energy: Quantifies the homogeneity of the image texture.

$$E = \sqrt{\sum_i \sum_j \mathrm{G}_{ij}^2} \tag{8}$$

Homogeneity: Reflects the closeness of the GLCM values to the diagonal, indicating homogeneous textures.

$$H = \sum_i \sum_j \frac{G_{ij}}{1 + (i - j)^2} \tag{9}$$

3.2.2 Local binary patterns

Local binary patterns (LBPs) are designed to capture the texture information of an image by comparing the intensity of a central pixel with its surrounding neighbors [25, 26]. The basic idea is to convert the local image patch around each pixel into a binary pattern. The binary pattern is then used to represent the texture characteristics of that particular region. Local binary pattern methods have among other been used by Mitiche et al. [27] for extraction of features from encoded time series data.

The basic process of computing LBP involves the following steps, as indicated in **Figure 7**.

a. Selection of a central pixel: For each pixel in the image, a local neighborhood around that pixel is considered.

b. Comparison of central pixel with neighboring pixels: The intensity value of the central pixel is compared with the intensities of its neighbors. The comparisons are performed in a circular or square region around the central pixel.

c. Binary pattern formation: For each comparison, if the intensity of the neighboring pixel is smaller than the intensity of the central pixel, the result is set to 0; otherwise, it is set to 1. These binary values are then concatenated to form a binary pattern for that specific region.

d. Histogram creation: The binary patterns are collected for all the pixels in the image, and a histogram is created, showing the frequency of occurrence of different binary patterns.

With the parameter settings in **Table 1**, 59 LBP features were generated for each image.

3.2.3 GoogleNet

GoogleNet [28], also known as Inception-v1, is a deep convolutional neural network (CNN) architecture that was developed by researchers at Google for image classification tasks. It was the winner of the ImageNet Large Scale Visual Recognition Competition (ILSVRC) in 2014.

To use GoogleNet for feature extraction from images, the network is typically truncated or frozen after the desired layer. The earlier layers capture low-level features such as edges, corners, and basic textures, while deeper layers capture more complex and abstract features specific to the task on which the network was trained (e.g., ImageNet classification). These higher-level features can then be used as input to other machine learning models or for further analysis.

One of the most common uses of GoogleNet is in transfer learning. Transfer learning involves leveraging the pretrained weights of GoogleNet on a large-scale dataset (e.g., ImageNet) and fine-tuning it on a smaller, task-specific dataset. By doing so, one can benefit from the powerful feature extraction capabilities of the network without needing to train it from scratch, even if the target dataset is relatively

Hyperparameter	Comment	Setting
P	Number of pixel neighbors	8
r	Central pixel neighborhood radius	1
Other	Encoding of rotation information	NO
	Interpolation method used to compute pixel neighbors	Linear
	Normalization of LBP histograms	L2

Table 1.
Local binary pattern hyperparameter settings.

small. In all cases, features were extracted from the images by simply passing the images through GoogleNet; that is, no further training was done, and the features were generated solely from the pretraining of GoogleNet on the ImageNet database.

For each image, 1024 features were extracted from the layer labeled “in GoogleNet.”

3.3 Evaluation of features

The quality of the features derived by means of different algorithms from images generated by different encoding schemes was evaluated through their use as predictors in machine learning models, specifically random forests [29]. In all cases, the hyperparameter settings of the random forests that were used are summarized in **Table 2**. This provided for as consistent an evaluation scheme as possible for the features.

4. Case study 1: Bivariate time series

In the first case study, a simulated bivariate Gaussian time series is considered, as shown in **Figure 8**. The 2000 sample series is generated with a zero mean vector and unit variances. The covariance matrix of the first 1000 samples was invariant, with $\Sigma_{1:1000} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$. However, from the 1001st sample index, the series started to change slowly, as the covariance was incrementally increased with each sample over the following 1000 samples, such that $\Sigma_{1001:2000} = \begin{bmatrix} 1 & s/1000 \\ s/1000 & 1 \end{bmatrix}$, with $s = 1, 2, \ldots 1000$.

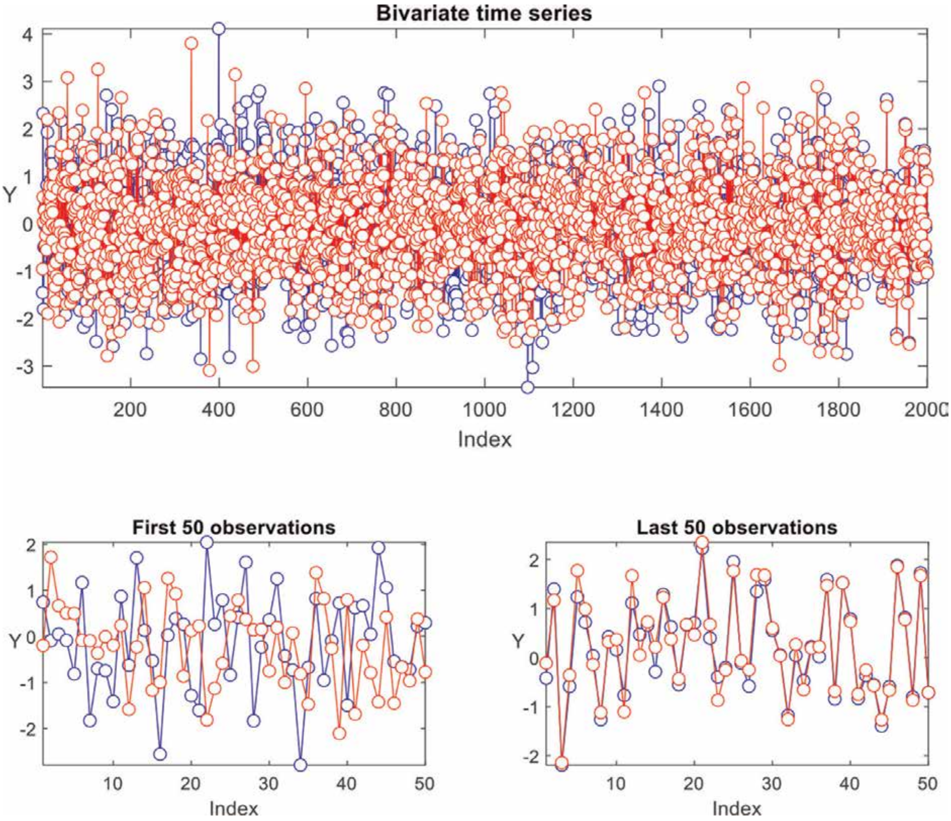
A close-up view of the time series is provided in the lower panel of **Figure 8**. As can be seen from this panel, the two time series move independently initially (left, lower panel), and toward the end, the two time series follow essentially the same trajectories.

A moving window with a size $b = s = 1000$ was used to segment the time series, and each segment was encoded as an image, examples of which are shown in **Figure 9**. A total of 100 images for each time series segment were generated. The Gramian angular field images were stacked horizontally for the two time series, of which the correlations varied from 0 to 1, as explained above.

4 GLCM, 59 LBP, and 1024 GoogleNet features were extracted from the images. These features could be visualized by projecting them to a two-dimensional space with a t-distributed stochastic neighbor embedding (t-SNE) algorithm [30] using a Euclidean distance metric with a perplexity parameter value of 30, as shown in **Figure 10**.

Hyperparameter	Description	Value
n_{trees}	Number of trees	200
n_{try}	Number of observations drawn at each split	$0.7N$
m_{try}	Number of variables drawn at each split	$\frac{M}{3}$
Replacement	TRUE/FALSE	TRUE
Node size	Minimum number of samples in a terminal node	5
Splitting rule	Criterion on which splitting of nodes was based	Gini

Table 2.
Hyperparameters of random forest constructed with a training data set consisting of N *samples and* M *variables.*



Bivariate time series

First 50 observations

Last 50 observations

Figure 8.
Bivariate time series considered in case study 1.

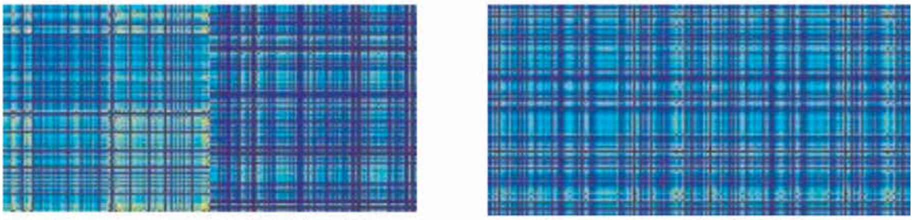
These projections tend to preserve the topological structure of the original data in the high-dimensional spaces; in other words, features that are similar (representing images that are similar) would tend to be located in the same area on the t-SNE map, while features that are different would tend to be segregated in the *t*-SNE map. As indicated by the bottom panel of these score plots, wavelet imaging facilitated the best segregation between the two time series.

To further quantify the performance of the features, they were used as predictors in random forest models, as discussed in Section 3.3. In each case, the features were used in a binary classification problem to discriminate as best as possible between the two time series.

For evaluation purposes, the out-of-bag (OOB) errors of the random forest models are shown as a function of the number of trees in the forest (each had 200 trees) in **Figure 11**. In this figure, each panel represents a different imaging method. GLCM features are represented by blue curves, LBP features by red curves, and the GoogleNet features by black curves.

The best results were obtained with the wavelet scalograms, as indicated by the lower right panel in the graph and more specifically by the features that were generated by GoogleNet. Conversely, features derived from the Toeplitz matrix and Gramian angular field images were not as predictive, noting that a classification error

Euclidean distance plot (r = 0) Euclidean distance plot (r = 1)



Gramian angular field (r = 0) Gramian angular field (r = 1)

Toeplitz matrix (r = 0) Toeplitz matrix (r = 1)

Morlet wavelet scalogram (r = 0) Morlet wavelet scalogram (r = 1)

Figure 9.
Imaging of the time series data in case study 1 (b = 100, s = 100). Images in the left column are from the invariant series (r = 0), while those on the right are from the variant section of the time series (r = 1).

of 0.5 would be equivalent to random features with no predictive value, given that there were only two equisized classes to predict.

5. Case study 2: Effect of imaging and feature extraction on anomaly detection in a nonlinear dynamic system

In the second case study, the effect of time series preprocessing, imaging method, and feature extraction on the ability of a principal component model to detect changes in the dynamics of a nonlinear time series are considered. For this purpose, a simulation of the Thomas attractor is investigated. Imaging is done by use of Euclidean biplots, Gramian angular fields, and wavelet scalograms, as discussed in more detail below.

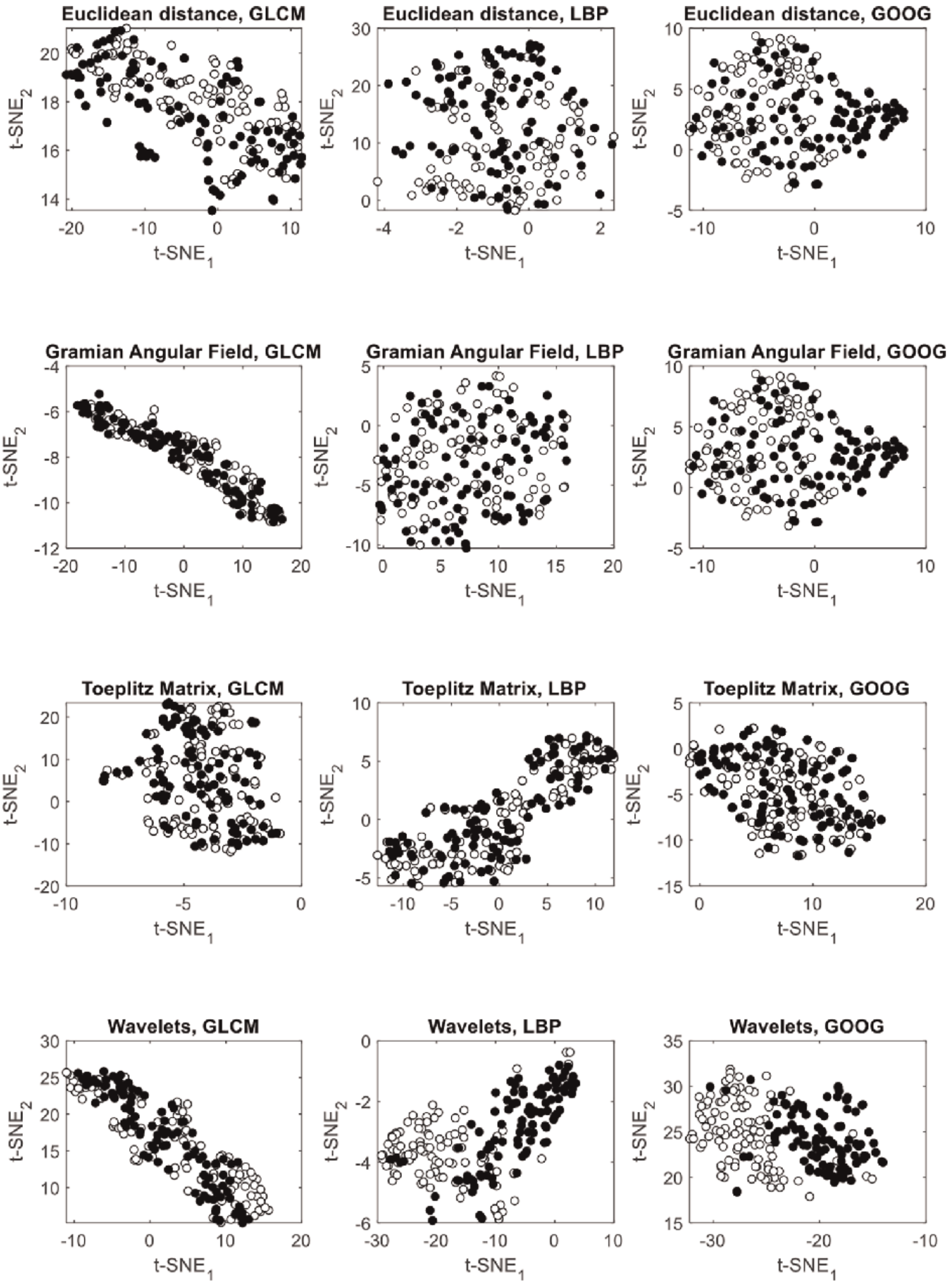


Figure 10.
Visualization of image features in case study 1 with t-SNE score plots. White and black markers show the first and second parts of the time series, respectively.

5.1 Thomas attractor

In the second case study, a univariate nonlinear time series is considered. This time series was obtained by simulating the Thomas attractor [31–33]. The Thomas attractor is a type of strange attractor that arises in a three-dimensional dynamical

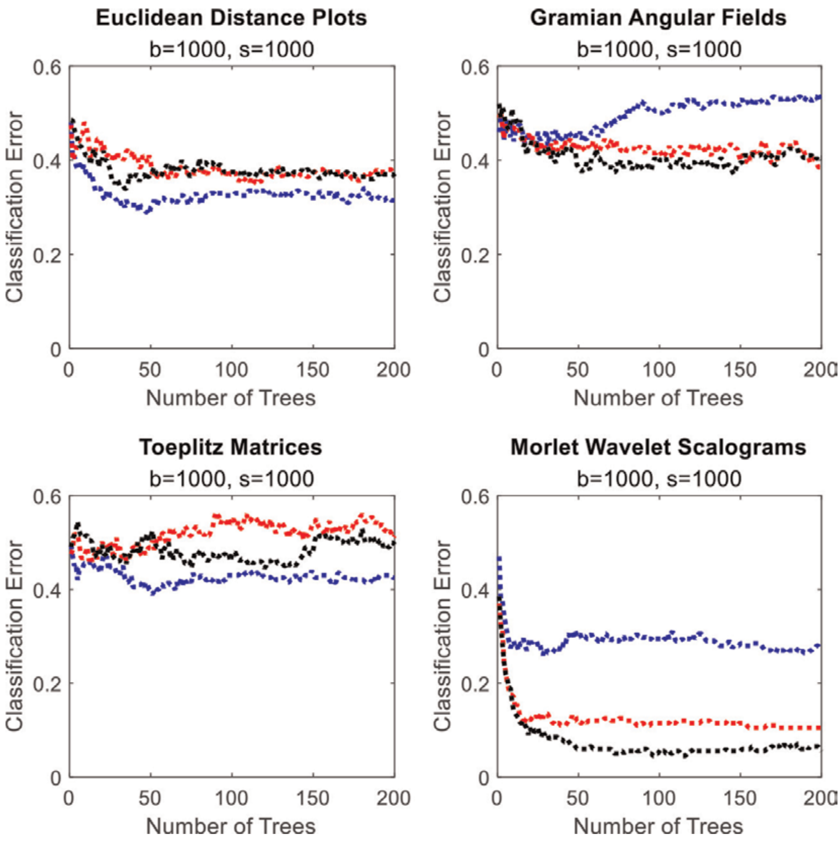


Figure 11.
Out-of-bag (OOB) errors of random forest models discriminating between the different realizations of the bivariate system in case study 1, using GLCM (blue), LBP (red), and GoogleNet (black) features derived from different images.

system and is named after its discoverer, mathematician and physicist Michael Thomas. The attractor is defined by the following system of three nonlinear differential equations:

$$\frac{dx}{dt} = \sin(y) - b_1 x \tag{10}$$

$$\frac{dy}{dt} = \sin(z) - b_1 y \tag{11}$$

$$\frac{dz}{dt} = \sin(x) - b_1 z \tag{12}$$

where b_1 is a positive constant.

This system of equations exhibits chaotic behavior, meaning that the solutions of the equations are highly sensitive to initial conditions. The behavior of the system is characterized by a complex, non-repeating pattern of trajectories in three-dimensional space, which forms the Thomas attractor. b_1 corresponds to how dissipative the system is and acts as a bifurcation parameter.

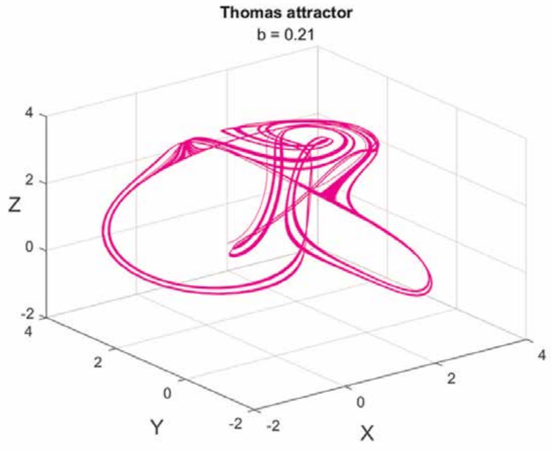


Figure 12.
The Thomas attractor in phase space.

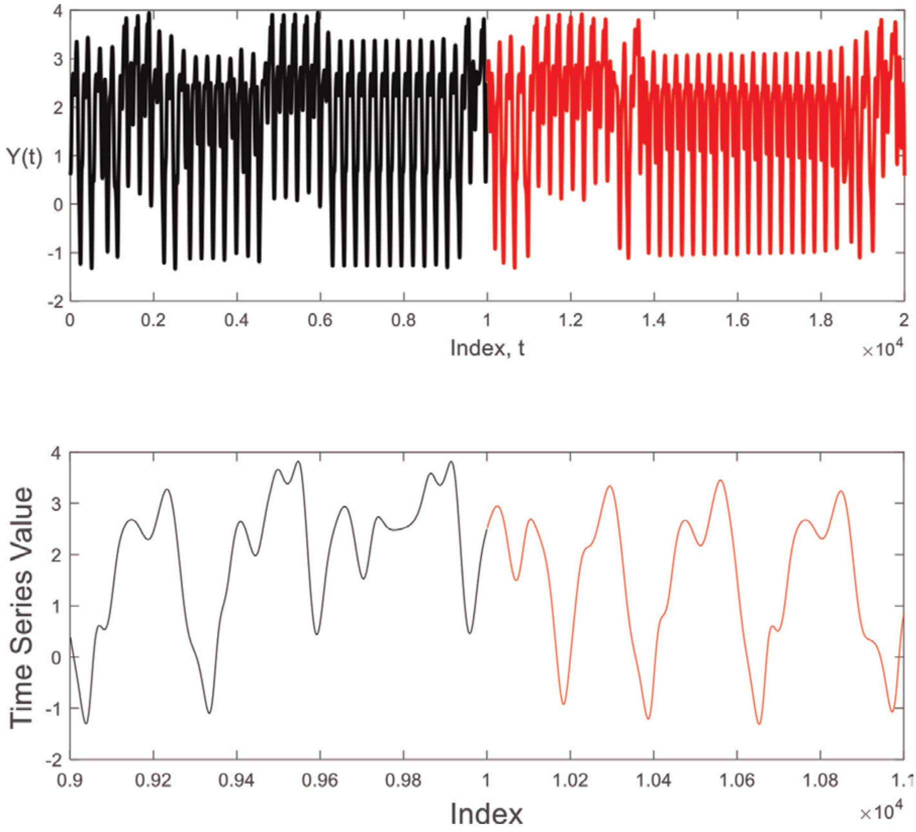


Figure 13.
Thomas attractor with persistent change in dynamics at index = 10,000 (top) and a close-up of the same data at the change point (bottom).

Changes in the behavior of the attractor can therefore be easily simulated by making small changes in the value of the parameter b. For this case study, the attractor was simulated by generating 20,000 samples of x, y, and z. For a parameter value of $0 < \mathrm{b} < 0.33$, the system exhibits chaotic behavior, with up to six separate coexisting attractors. An example of such as attractor is shown in **Figure 12**.

Only one of the variables (x) was retained for purpose of the analysis. 10,000 samples were generated based on a parameter value b = 0.2 and another 10,000 with $b_1 = 0.21$. These time series were concatenated to yield a single time series consisting of 20,000 samples, as shown in **Figure 13**.

5.2 Effect of imaging of time series

The time series was segmented with a window size $b = [100, 1000]$ and step size $s = [100, 1000]$. Each segment was imaged with Euclidean distance plots, Gramian angular field matrices, Toeplitz stacks, and Morlet wavelets. In addition, GLCM, LBP, and GoogleNet features were extracted from these images, as before. Examples of these images are shown in **Figure 14**.

The features extracted with gray level co-occurrence matrices, local binary patterns, as well as using GoogleNet can be visualized in **Figure 15**. Overall, there does not appear to be marked differences in the imaging methods, and this is also borne out by the data shown in **Figure 16**. On the whole, the LBP features combined with the Euclidean distance plots yielded the best results.

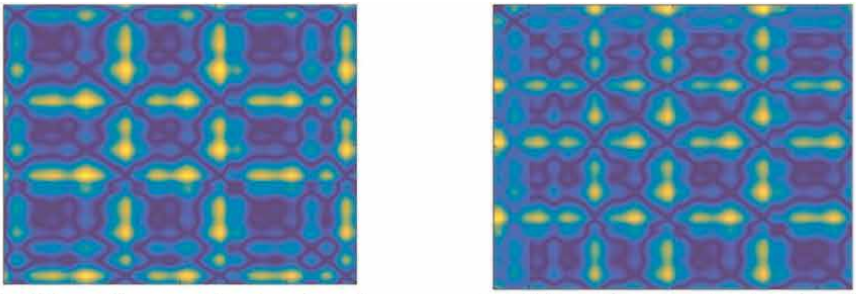
6. Case study 3: Power consumption of a SAG mill

In the final case study, measurements of the power consumption collected from an IsaMill on an industrial copper processing circuit in Western Australia, as described in more detail by Napier and Aldrich [34], are considered. The time series consisted of 80,000 measurements collected over a two-month period of operation.

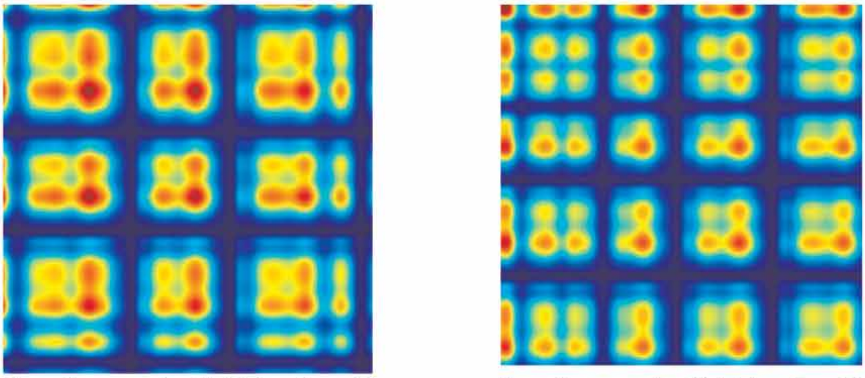
A surrogate time series with the same distribution of measurements, as well as having the same autocorrelation as the real data, was generated using an iterative amplitude adjusted Fourier transform algorithm. These algorithms are commonly used in nonlinear time series analysis to test hypotheses about the (non)linearity or deterministic nature of the time series based on pivotal test statistics, such as the correlation dimension of the time series.

In this case study, such formal tests are not conducted, but the extent to which the real data can be distinguished from the surrogate data could be seen as an indication of the nonlinear deterministic nature of the time series. The real data and the surrogate data are shown in **Figure 17**.

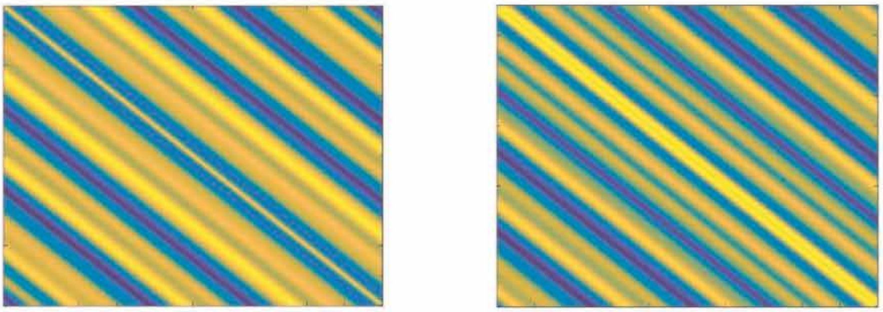
Both the real time series and its surrogate were encoded in images based on the use of Euclidean distance plots, Gramian angular fields, Toeplitz matrices of the time series segments, as well as Morlet wavelet scalograms, as shown in **Figure 18**. As before, features were extracted from these images by use of gray level co-occurrence matrices, local binary patterns, and GoogleNet, and these can be visualized in **Figure 19**. By and large, the two time series segments are



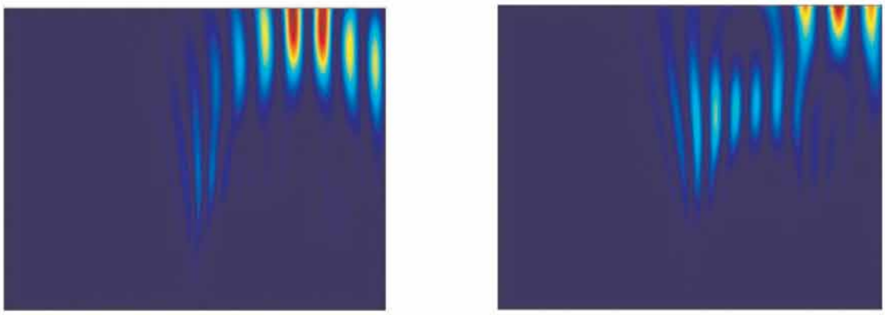
Euclidean distance plot (section A) Euclidean distance plot (section B)



Gramian angular field (section A) Gramian angular field (section B)



Toeplitz matrix (section A) Toeplitz matrix (section B)



Morlet wavelet scalogram (section A) Morlet wavelet scalogram (section B)

Figure 14.
*Examples of images of the Thomas attractor shown in **Figure 13**.*

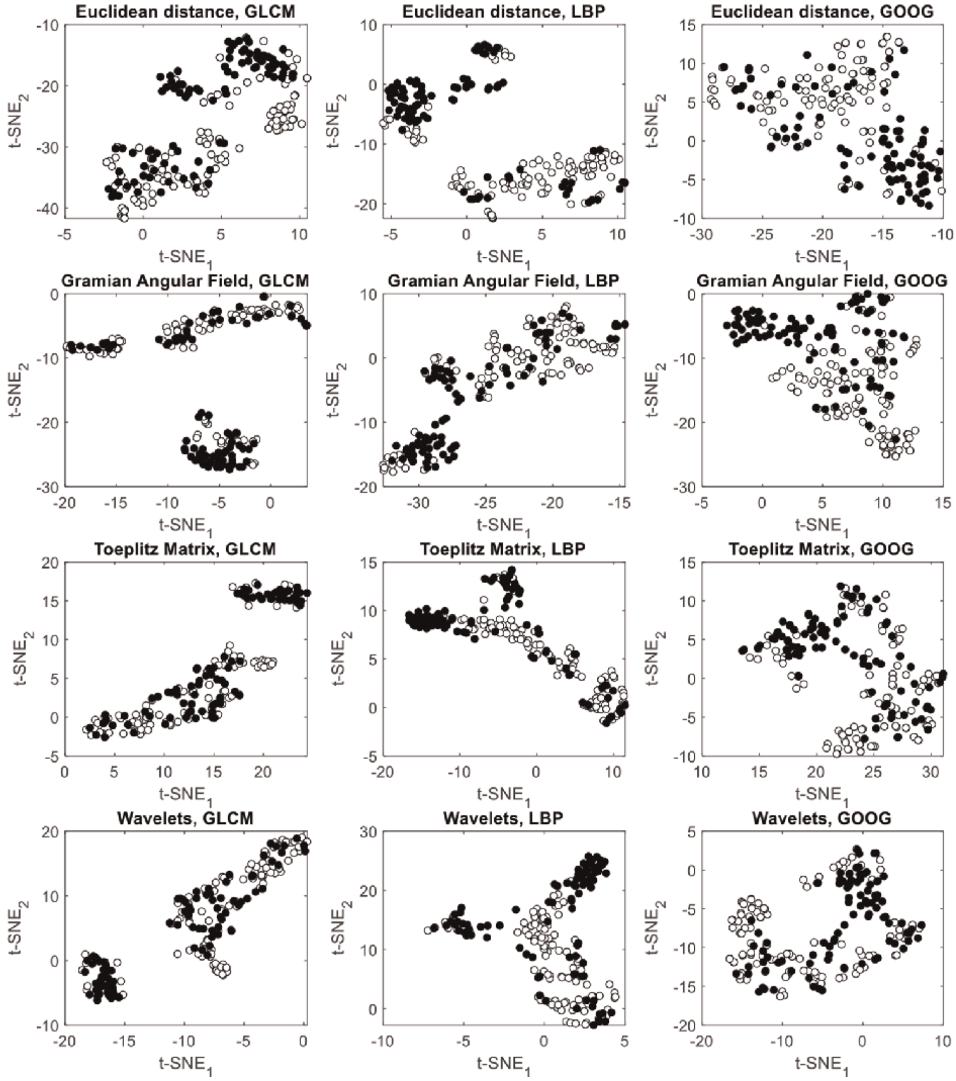


Figure 15.
Visualization of image features in case study 2 with t-SNE score plots. White and black markers show the first and second parts of the time series, respectively.

significantly separated in the t-SNE score spaces of all the images, with the exception of the wavelets.

These features were subsequently used as predictors in random forest models trained to discriminate between the real and surrogate time series. The out-of-bag (OOB) errors are graphically portrayed in **Figure 20**.

As can be seen from these results, the GoogleNet features extracted from the Euclidean distance plots and Gramian angular fields gave the best results, being able to discriminate between the two time series with an error of approximately 7%.

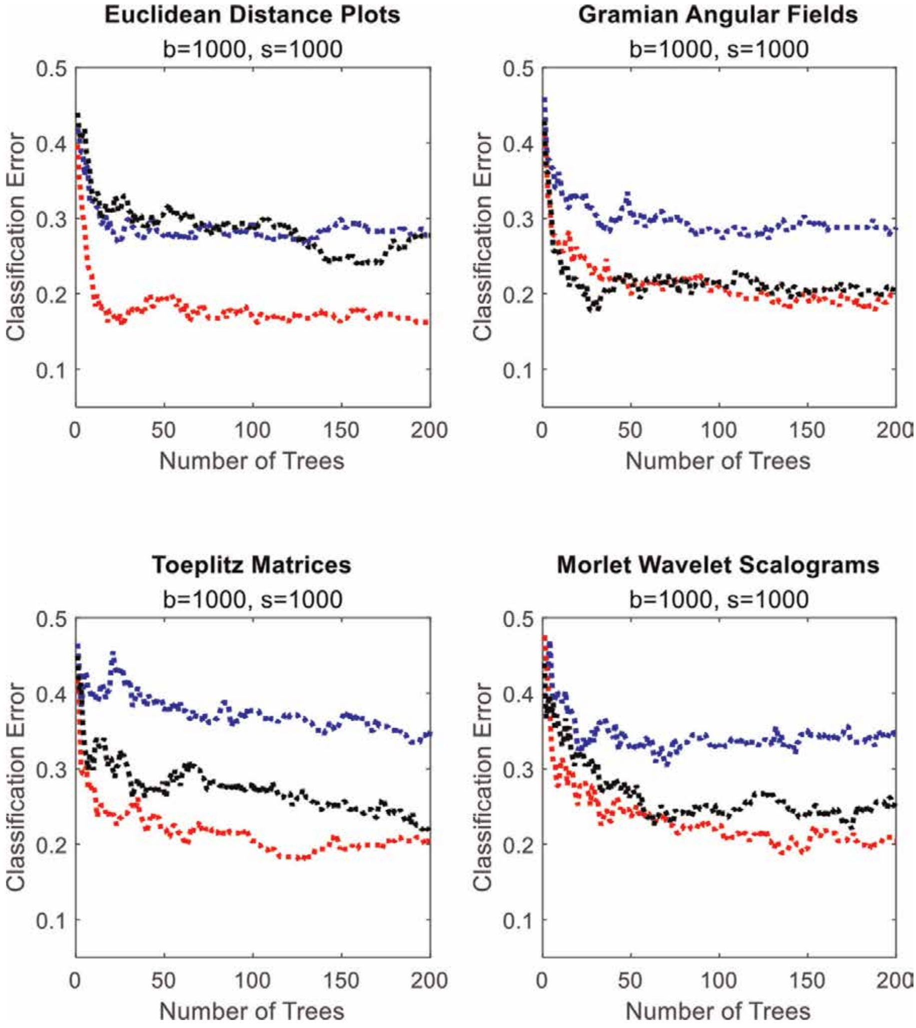


Figure 16.
Effect of different image encodings and features in discriminating between two different realizations of the nonlinear Thomas time series in case study 2 shown in ***Figure 13****. GLCM, LBP, and GoogleNet features are respectively represented by blue, red, and black curves.*

Features extracted from Euclidean distance plots consistently yielded the best performance. In addition, the GLCM features were consistently outperformed by the other feature sets.

7. Discussion and conclusion

When the time series is imaged, the capture of information contained in the time series depends on the characteristics of the image. Euclidean distance plots, for

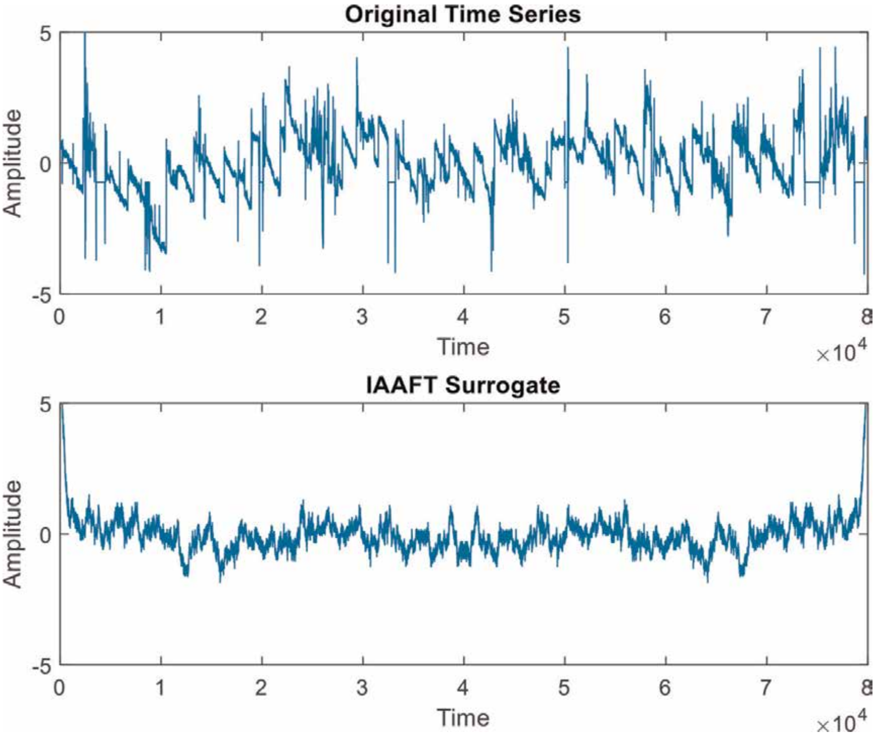


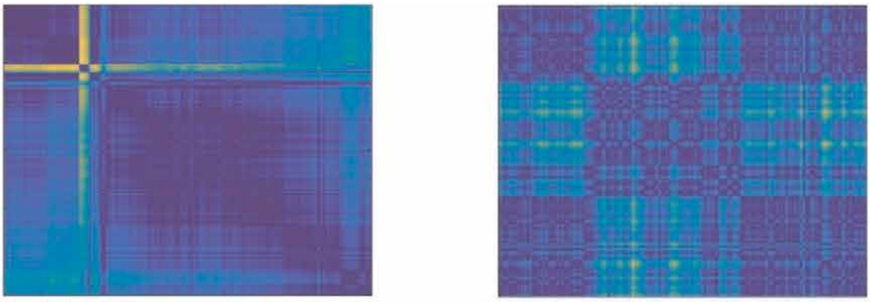
Figure 17.
Scaled power consumption on an industrial copper grinding circuit, as well as an iterative amplitude adjusted Fourier transform surrogate of the data.

example, tend to capture recurrent behavior in the time series well. Although some comparative analysis has been conducted [35], the effects of different imaging approaches have not been studied very widely as yet. In another study, Yuan et al. [36] have compared recurrence plots, wavelets, and Markov transition fields in seismographic data. Although the wavelet-based images gave the best results, the differences were marginal. Song et al. [37] have likewise compared recurrence plots, Gramian angular difference fields, and Markov transition fields and have concluded that the recurrence plots yielded the best results in their application related to fault detection in manufacturing processes.

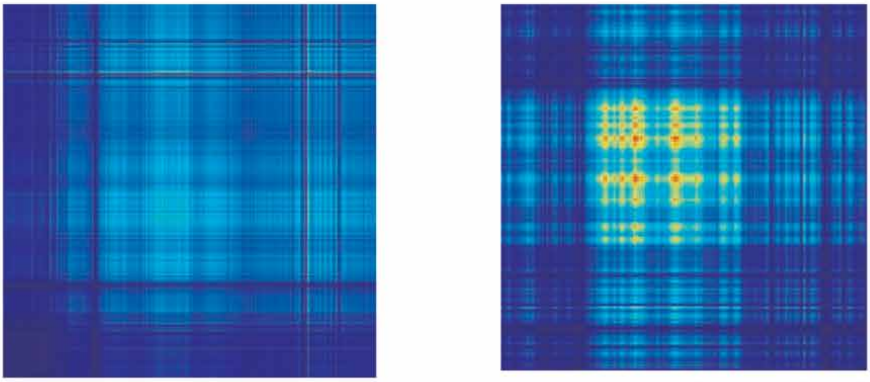
Overall, the feature extraction algorithms considered in this study can be compared by their rankings in each case study. That is, each algorithm can be awarded a score of 1 (best), 2, or 3 (worst) based on its performance in each trial. Where two algorithms perform equally, the ranks are shared. For example, if two algorithms perform equally well and best, each is awarded a score of 1½.

For the 12 trials, that is, four image types in three case studies, the results are summarized in **Table 3**, with a lower score better than a higher score. This shows that on average GoogleNet features performed somewhat better than LBP features and markedly better than GLCM features.

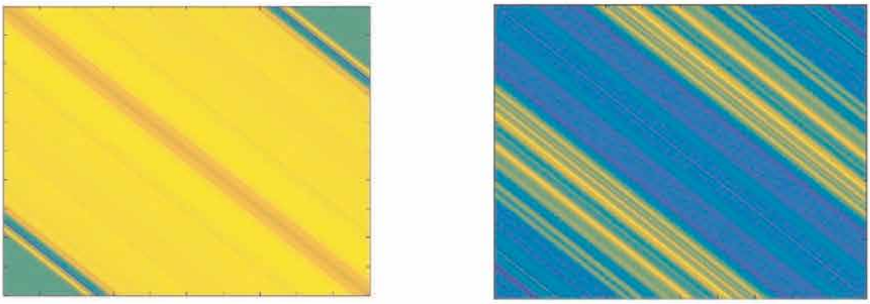
The different imaging methods can also be analyzed on a similar basis, which on average yield the result D > G > W > T; that is, the best models were associated with Euclidean distance plots, although it is clear that none of the imaging methods consistently outperformed the other.



Euclidean distance plot (section A) Euclidean distance plot (section B)



Gramian angular field (section A) Gramian angular field (section B)



Toeplitz matrix (section A) Toeplitz matrix (section B)

Morlet wavelet scalogram (section A) Morlet wavelet scalogram (section B)

Figure 18.
Examples of images of the power consumption on the industrial copper grinding circuit shown in ***Figure 17****.*

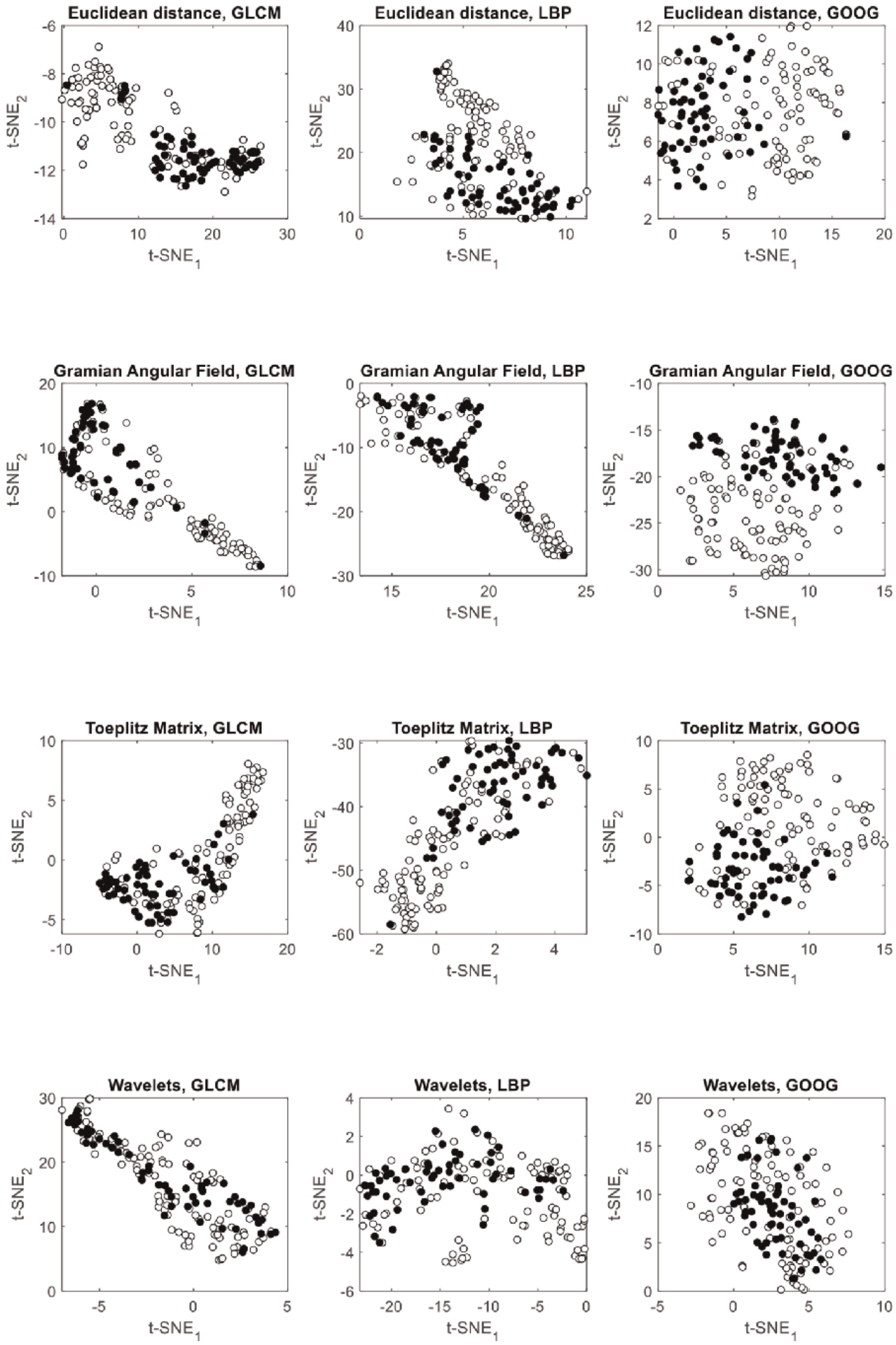


Figure 19.
Visualization of image features in case study 3 with t-SNE score plots. White and black markers show the first and second parts of the time series, respectively.

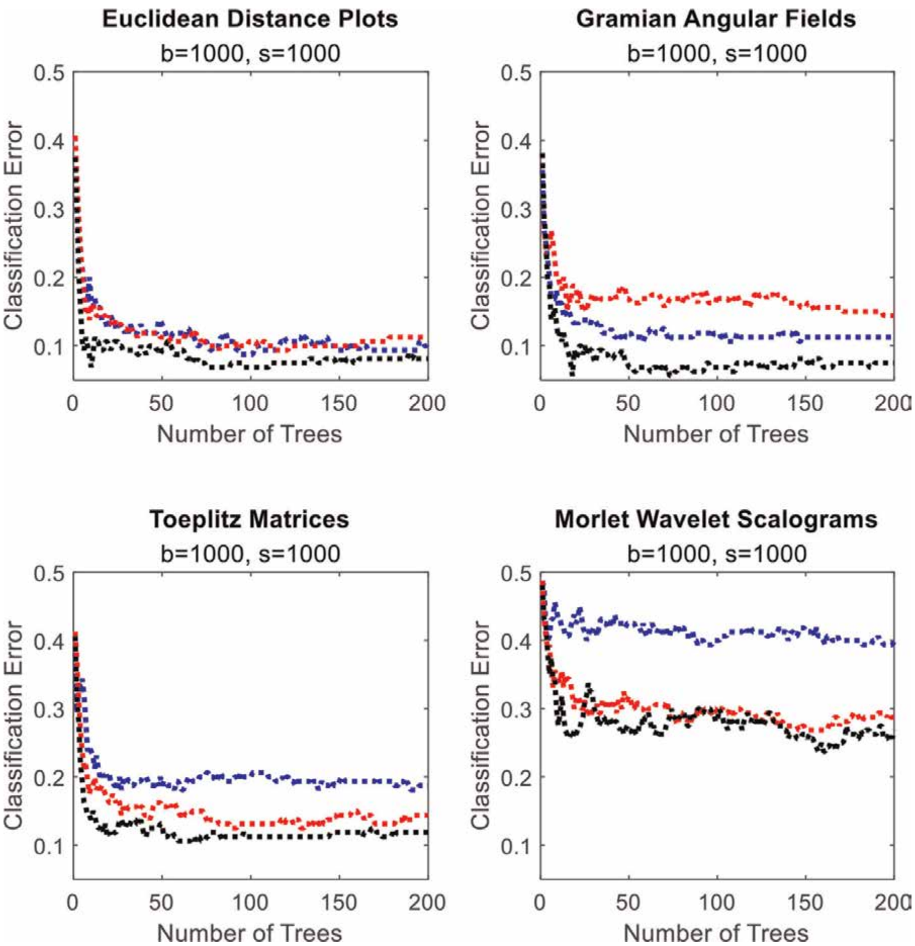


Figure 20.
Effect of imaging method and feature extraction on the characterization of the dynamics of power consumption of a mill on a copper circuit in case study 3. GLCM, LBP, and GoogleNet features are respectively represented by blue, red, and black curves.

Case study	Imaging*	GLCM	LBP	GoogleNet
1	D	1	2½	2½
	G	3	2	1
	T	1	2½	2½
	W	3	2	1
2	D	2½	1	2½
	G	3	1½	1½
	T	3	1	2
	W	3	1	2

Case study	Imaging*	GLCM	LBP	GoogleNet
3	D	2	2	2
	G	2	3	1
	T	3	2	1
	W	3	1½	1½
Total		**29½**	**22**	**20½**

**D = Euclidean distance plot, G = Gramian angular field, T = Toeplitz stacking, and W = Wavelet scalogram.*

Table 3.
Ranking of feature extraction methods across the three case studies. Smaller values are better.

These results should be considered as preliminary only, and further validation would be required with more diverse time series data. Other imaging methods or more advanced implementation of the basic approaches would also have to be considered.

Author details

Chris Aldrich
Western Australian School of Mines, Curtin University, Perth, WA, Australia

*Address all correspondence to: chris.aldrich@curtin.edu.au

References

[1] Li Z, Li S, Yan X. Time Series as Images: Vision Transformer for Irregularly Sampled Time Series. arXiv: 2303.12799v1 [cs.LG]. 1 Mar 2023

[2] Aldrich C, Liu X. Quantitative texture analysis with convolutional neural networks. In: IoT-Enabled Convolutional Neural Networks: Techniques and Applications. Denmark: River Publishers; 2023. Available from: https://ieeexplore.ieee.org/Xplorehelp/browsing-ieee-xplore/river-publishers#learn-more-about-river-publishers

[3] Abidi A, Ienco D, Abbes AB, Farah IR. Combining 2D encoding and convolutional neural network to enhance land cover mapping from satellite image time series. Engineering Applications of Artificial Intelligence. 2023;**122**:106152. DOI: 10.1016/j.engappai.2023.106152

[4] Wang C-C, Kuo C-H. Detecting dyeing machine entanglement anomalies by using time series image analysis and deep learning techniques for dyeing-finishing process. Advanced Engineering Informatics. 2023;**55**:101852. DOI: 10.1016/j.aei.2022.101852

[5] Wang Z, Oates T. Imaging time-series to improve classification and imputation. In: Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence (IJCAI 2015), Buenos Aires, Argentina, 25-31 July. Palo Alto, CA, USA: AAAI Press; 2015

[6] Zhou X, Yu T, Wang G, Guo R, Fu Y, Sun Y, et al. Tool wear classification based on convolutional neural network and time series images during high precision turning of copper. Wear. 2023;**522**: 204692. DOI: 10.1016/j.wear.2023.204692

[7] Chen Y, Li J, Huang Q, Li K, Zhao Z, Ren X. Non-technical losses detection with Gramian angular field and deep residual network. Energy Reports. 2023;**9**:1392-1401. DOI: 10.1016/j.egyr.2023.05.183

[8] Qin Z, Zhang Y, Meng S, Qin Z, Choo KKR. Imaging and fusing time series for wearable sensor-based human activity recognition. Information Fusion. 2020; **53**:80-87. DOI: 10.1016/j.inffus.2019.06.014

[9] Zhang Q, Qi Z, Cui P, Xie M, Din J. Detection of single-phase-to-ground faults in distribution networks based on Gramian angular field and improved convolutional neural networks. Electric Power Systems Research. 2023;**221**: 109501. DOI: 10.1016/j.epsr.2023.109501

[10] Jiang H, Liu L, Lian C. Multi-modal fusion transformer for multivariate time series classification. In: 14th International Conference on Advanced Computational Intelligence (ICACI); Wuhan, China, 15-17 July, 2022. IEEE. 2022. pp. 284-288. DOI: 10.1109/ICACI55529.2022.9837525

[11] Marwan N. A historical review of recurrence plots. European Physical Journal ST. 2008;**164**(1):3-12. DOI: 10.1140/epjst/e2008-00829-1.S2CID 119494395

[12] Zbilut JP, Webber CL Jr. Embeddings and delays as derived from quantification of recurrence plots. Physics Letters A. 1992;**171**(3–4):199-203. DOI: 10.1016/0375-9601(92)90426-M

[13] Debayle J, Hatami N, Gavet Y. Classification of time-series images using deep convolutional neural networks. In: Proceedings of the 10th International Conference on Machine Vision (ICMV 2017). Vienne: SPIE; 2018. DOI: 10.1117/12.2309486

[14] Hou Y, Aldrich C, Lepkova K, Machuca L, Kinsella B. Monitoring of carbon steel corrosion by use of electrochemical noise and recurrence quantification analysis. Corrosion Science. 2016;**112**:63-72

[15] Hou Y, Aldrich C, Lepkova K, Kinsella B. Identifying corrosion of carbon steel buried in iron ore and coal cargoes based on recurrence quantification analysis of electrochemical noise. Electrochimica Acta. 2018;**283**:212-220

[16] Abbasi H, Bennet L, Gunn AJ, Unsworth CP. 2D wavelet scalogram training of deep convolutional neural network for automatic identification of micro-scale sharp wave biomarkers in the hypoxic-ischemic EEG of preterm sheep. In: (2019) Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS; Berlin, Germany, 23-27 July 2019. IEEE. 2019. pp. 1825, 8857665-1828

[17] Ahmad S, Ahmad Z, Kim C-H, Kim J-M. A method for pipeline leak detection based on acoustic imaging and deep learning. Sensors. 2022;**22**(4):1562

[18] Roy AD, Islam MM. Detection of epileptic seizures from wavelet scalogram of EEG signal using transfer learning with AlexNet convolutional neural network. In: ICCIT 2020-23rd International Conference on Computer and Information Technology, Proceedings, Dhaka, Bangladesh, 19-21 December 2020. Art. no. 9392720. IEEE. 2020

[19] Sharan RV. Spoken digit recognition using wavelet scalogram and convolutional neural networks. In: 2020 IEEE Recent Advances in Intelligent Computational Systems, RAICS 2020. Thiruvananthapuram, India: (Virtual Conference), 3-5 December 2020. Art. no. 9332505. 2020. pp. 101-105

[20] Almaghrabi S, Rana M, Hamilton M, Rahaman MS. Solar power time series forecasting utilising wavelet coefficient. Neurocomputing. 2022;**508**:182-207

[21] Packard NH, Crutchfield JP, Farmer JD, Shaw RS. Geometry from a time series. Physical Review Letters. 1980; **45**:712

[22] Sauer TD. Attractor reconstruction. Scholarpedia. 2006;**1**(10):1727. DOI: 10.4249/scholarpedia.1727

[23] Löfstedt T, Brynolfsson P, Asklund T, Nyholm T, Garpebring A. Gray-level invariant Haralick texture features. PLoS One. 2019;**14**(2):e0212110. DOI: 10.1371/journal.pone.0212110

[24] Haralick RM, Shanmugam K, Dinstein I. Textural features for image classification. IEEE Transactions on Systems, Man, and Cybernetics. 1973;**3**: 610-621. DOI: 10.3390/s22041562

[25] Ojala T, Pietikäinen M, Harwood D. A comparative study of texture measures with classification based on featured distributions. Pattern Recognition. 1996; **29**:51-59

[26] Ojala T, Pietikäinen M, Mäenpää T. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2002;**24**(7):971-987. DOI: 10.1109/TPAMI.2002.1017623

[27] Mitiche I, Morison G, Nesbitt A, Hughes-Narborough M, Stewart BG, Boreha P. Imaging time series for the classification of EMI discharge sources. Sensors. 2018;**18**(9):3098. DOI: 10.3390/s18093098

[28] Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, et al. Going deeper with convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; Boston, MA, USA. 7-12 June 2015. IEEE. 2015. pp. 1-9

[29] Breiman L. Random forests. Machine Learning. 2001;**45**:5-32

[30] Van der Maaten LJP, Hinton GE. Visualizing high-dimensional data using t-SNE. Journal of Machine Learning Research. 2008;**9**:2579-2605

[31] Butusov DN, Ostrovskii VY, Tutueva AV, Savelev AO. Comparing the algorithms of multiparametric bifurcation analysis. In: XX IEEE International Conference on Soft Computing and Measurements (SCM); St. Petersburg, Russia, 24-26 May 2017. IEEE. 2017. DOI: 10.1109/SCM.2017.7970536

[32] McDonald B, Roy Choudhury S. The Thomas attractor with and without delay: Complex dynamics to amplitude death. Discontinuity, Nonlinearity, and Complexity. 2020;**9**(1):27-45. DOI: 10.5890/DNC.2020.03.003

[33] Sprott JC, Chlouverakis Konstantinos E. Labyrinth chaos. International Journal of Bifurcation and Chaos. 2007;**17**(6):2097-2108

[34] Napier LFA, Aldrich C. An IsaMill™ soft sensor based on random forests and principal component analysis. IFAC-PapersOnLine. 2017;**50**(1):1175-1180. DOI: 10.1016/j.ifacol.2017.08.270

[35] Lee G, Kwon D, Lee C. A convolutional neural network model for SOH estimation of Li-ion batteries with physical interpretability. Mechanical Systems and Signal Processing. 2023;**188**: 110004. DOI: 10.1016/j.ymssp.2022.110004

[36] Yuan X, Tanksley D, Jiao P, Li L, Chen G, Wunsch D. Encoding time-series ground motions as images for convolutional neural networks-based seismic damage evaluation. Frontiers in Built Environment. 2021;**729**:660103. DOI: 10.3389/fbuil.2021.660103

[37] Song J, Lee YC, Lee J. Deep generative model with time series-image encoding for manufacturing fault detection in die casting process. Journal of Intelligent Manufacturing. 2023;**34**: 3001-3014. DOI: 10.1007/s10845-022-01981-6

Section 7

Time Series Analysis in Cultural Contexts

Chapter 13

Fractal Dynamics and Fibonacci Sequences: A Time Series Analysis of Cultural Attractor Landscapes

Rob G. Sacco

Abstract

This study explores the intricate relationship between fractal structures and cultural evolution through time series analysis. Utilizing Fibonacci time series modeling, the author predicts the formation and stabilization of cultural attractors—emergent phenomena that arise from dynamic populations within cognitive landscapes. The model backtests significant milestones in human cultural history, ranging from the Copper Age to the Internet era, and correlates them with Fibonacci time series. The findings suggest that cognitive development, demographic structure, and cultural transmission are key factors that influence the self-organization and dynamic stabilization of cultural attractor landscapes. This study offers a novel perspective on the optimization of information flow in cultural evolution and has implications for interdisciplinary studies in computer science, mathematics, statistics, and geography.

Keywords: fractal structures, cultural evolution, time series analysis, Fibonacci time series, cultural attractors, cognitive landscapes, information flow, cumulative culture, demographic structure, self-organization

1. Introduction

The intersection of mathematics, computer science, and cultural studies has long been fertile ground for interdisciplinary research [1]. One of the most intriguing areas of study within this intersection is the application of time series analysis to understand the evolution of cultures [2]. While time series analysis has traditionally been used in fields such as economics, epidemiology, and geography, its application to cultural studies opens up new avenues for understanding the complex dynamics of cultural evolution.

This study aims to contribute to this growing body of knowledge by exploring the role of fractal structures and Fibonacci sequences in shaping cultural attractor landscapes.

The concept of cultural attractors has been a subject of interest in anthropology, sociology, and psychology for several decades [3]. Cultural attractors are emergent phenomena that arise from the collective cognitive landscapes of individuals within society. These attractors serve as focal points, around which cultural norms, beliefs, and practices coalesce. However, the mechanisms by which these attractors form and stabilize are not fully understood.

Recent advancements in time series analysis have offered a mathematical framework for studying these mechanisms based on the Fibonacci sequence [4]. The Fibonacci sequence, a series of numbers in which each number is the sum of the two preceding ones, has been found to have applications in various natural phenomena, including the growth patterns of plants, structure of galaxies, and even the stock market [5]. This study explores the potential of the Fibonacci sequence as a predictive tool for understanding the formation and stabilization of cultural attractors.

The primary objective of this research is to apply time series analysis, specifically Fibonacci time series modeling, to predict the formation and stabilization of cultural attractors. The paper aims to:

1. Develop a theoretical framework that integrates the Fibonacci sequence with the concept of cultural attractors.

2. Backtest this model against major milestones in human cultural history, ranging from ancient epochs to contemporary times [6].

3. The correlation between the Fibonacci time series and the formation of cultural attractors is used to understand the factors that contribute to their stability.

4. Explore the implications of these findings for optimizing information flow in cultural transmission [7].

The scope of this research is limited to the application of Fibonacci time series modeling for predicting cultural attractors. The primary focus is on the Fibonacci sequence and its applicability to cultural studies. One of the limitations of this study is the availability and quality of historical data for backtesting the model. Additionally, the study does not delve into the micro-level psychological factors that contribute to individual cognition and behavior, focusing instead on macro-level patterns and trends.

This research employs a mixed-methods approach that combines qualitative case studies with quantitative time series analysis [8]. The first step involved a comprehensive literature review to understand the existing theories and models related to cultural attractors and time series analysis. Following this, Fibonacci time series modeling is applied to predict the formation of cultural attractors. Historical data on major cultural milestones are collected from Wikipedia. The results are then validated through correlation analysis to determine the effectiveness of Fibonacci time series modeling in predicting cultural attractors. Finally, the findings are discussed in the context of their implications for interdisciplinary research in computer science, mathematics, statistics, and geography [9].

2. Literature review

The application of mathematical models and computational methods to the study of cultural evolution is a burgeoning field of research. This literature review aims to provide an overview of the key areas that intersect with the focus of this paper: time series analysis in cultural studies, the Fibonacci sequence and fractals, the concept of cultural attractors, and previous models and their limitations.

2.1 Time series analysis in cultural studies

Time series analysis has traditionally been employed in disciplines such as economics, epidemiology, and environmental science to forecast future events based on historical data. However, its application in cultural studies is relatively new, but growing. Researchers [10] have used time series analysis to study cultural drift and the diffusion of cultural traits. Similarly, studies have applied time series models to understand the dynamics of cultural change over time [11]. These studies have laid the groundwork for the application of time series analysis in predicting cultural phenomena; however, they have often focused on linear models that may not capture the complexity of cultural evolution.

2.2 The Fibonacci sequence and fractals

The Fibonacci sequence is a series of numbers where each number is the sum of the two preceding ones, starting with 0 and 1 (0, 1, 1, 2, 3, 5, 8, ...). This sequence describes various natural phenomena, from the arrangement of leaves on a stem to the spiral structure of galaxies. Mathematically, the sequence is closely related to fractals and complex structures that look similar at any level of magnification [5]. In 1983 [5], research explored the application of fractals in the natural sciences, but its application in the social sciences, particularly in cultural studies, is still an emerging field. The potential for using the Fibonacci sequence and fractals as tools for understanding the nonlinear, complex nature of cultural evolution is an area ripe for exploration.

2.3 Cultural attractors

The concept of cultural attractors is rooted in the broader theory of attractor landscapes in complex systems. In cultural studies, attractors represent the stable states in which cultures tend to gravitate over time. Researchers [12, 13] have explored the mechanisms through which cultural attractors form and stabilize. They argue that cultural attractors emerge from the collective cognitive landscapes of individuals within society and serve as focal points around which cultural norms and practices coalesce. However, these studies often lack a quantitative framework for predicting the formation and stabilization of cultural attractors, which this paper aims to provide.

2.4 Previous models and their limitations

Several models have been proposed for understanding the dynamics of cultural evolution. Agent-based models [1] have been used to simulate the spread of cultural traits among individuals. Similarly, mathematical models such as the Moran process have been applied to study cultural drift [3]. While these models offer valuable insights, they often suffer from limitations, such as oversimplification of complex cultural phenomena and lack of predictive power. Moreover, they usually do not account for the fractal nature of cultural evolution, which can be better captured through the application of the Fibonacci sequence.

3. Theoretical framework

The theoretical framework of this research is anchored in three main concepts: Fibonacci time series modeling, cultural attractor landscapes, and fractal structures in cultural evolution. These concepts are interwoven to create a comprehensive model for understanding the dynamics of cultural evolution through time series analysis.

3.1 Fibonacci time series modeling

Fibonacci time series modeling serves as the mathematical backbone of this research. Originating from the Fibonacci sequence, a series of numbers, where each number is the sum of the two preceding ones (0, 1, 1, 2, 3, 5, 8, ...), this approach aims to predict the formation and stabilization of cultural attractors. The sequence is applied to a time series, where each point represents a significant cultural milestone. The gaps between these milestones were measured in Fibonacci numbers, each representing a day, creating a predictive model for future cultural events.

The application of Fibonacci sequence in this context is not arbitrary. This sequence has been found to describe various natural phenomena, suggesting an underlying order in seemingly chaotic systems [5]. For example, it has been observed in random diffusion-limited aggregation processes [14]. By applying this sequence to the study of cultural evolution, it is hypothesized that cultural attractors can be predicted and understood in a more structured manner. This method offers a quantitative framework that complements existing qualitative theories, filling a gap in the current literature.

3.2 Cultural attractor landscapes

Cultural attractor landscapes serve as conceptual frameworks for understanding the dynamics of cultural evolution. These landscapes are multi-dimensional spaces, where each point represents a possible state of a culture, and the "height" of each point indicates its stability or attractiveness. Over time, cultures tend to gravitate towards the "valleys" or stable states in these landscapes, forming cultural attractors.

The concept of cultural attractors is not new, but its quantitative analysis has been limited. This study aims to fill this gap by applying the Fibonacci time series to these landscapes. By doing so, we can predict the states that are likely to become stable attractors and understand the factors that contribute to their formation and stabilization [12, 13]. This offers a more nuanced understanding of cultural evolution, moving beyond simple models that fail to capture its complexity.

3.3 Fractal structures in cultural evolution

Fractal structures offer a lens by which the complexity of cultural evolution can be understood. A fractal is a complex structure that appears similar at any level of magnification, suggesting a form of self-similarity across different scales. In the context of cultural evolution, this means that the mechanisms driving change at the micro level (individual or community) are similar to those at the macro level (society or civilization).

This fractal nature is not unique to cultural systems. The fractal structure observed in human and primate social networks [15] offers a compelling parallel to the fractal nature of cultural evolution, suggesting that such structures may inherently optimize information flow across different scales.

The inclusion of fractal structures in this theoretical framework is crucial for two reasons. First, it allows for a more accurate representation of the complex, nonlinear nature of cultural evolution. Traditional linear models often fail to capture this complexity, leading to inaccurate predictions and interpretation. Second, fractal structures offer a method to understand how cultural attractors form and stabilize at different scales, from individual cognitive landscapes to societal norms and practices.

This theoretical framework offers a comprehensive model for understanding the dynamics of cultural evolution by integrating Fibonacci time series, cultural attractor landscapes, and fractal structures. It provides both a quantitative and conceptual toolset for predicting the formation and stabilization of cultural attractors, filling gaps in the existing literature, and offering new avenues for interdisciplinary research.

4. Methodology

The methodology employed in this study was designed to test the theoretical framework outlined in the previous section. It encompasses three main components: data collection, analysis, and validation. Each of these components is essential for ensuring the reliability and validity of the research findings.

4.1 Data collection

The first step in the research process is data collection. Due to the interdisciplinary nature of this study, the data sources are diverse, encompassing a wide range of information from historical records to scholarly articles, primarily gathered from Wikipedia. The core data comprise significant cultural milestones, which are plotted as initial conditions in a Fibonacci time series. The selection of these milestones is based on their influence in shaping cultural attractors, with Wikipedia serving as the primary source of this historical and cultural information.

Secondary data includes scholarly articles and books that provide insights into the concepts of cultural attractors, fractal structures, and time series analysis. These data are essential for contextualizing primary data and developing a theoretical framework [11].

4.2 Data analysis

Once the data are collected, the next step is the data analysis. Rather than employing complex statistical methods, such as Pearson correlation coefficients, this study uses a simpler and more straightforward approach. The primary tool for this is a percentage deviation analysis, comparing the actual dates of cultural milestones with the predicted dates based on the Fibonacci time pattern.

Significance of the deviation was determined using a lower threshold of 2.5%. Any deviations below this threshold are considered significant and indicative of a strong alignment between the cultural milestones and the Fibonacci sequence.

4.3 Validation methods

Validation was an essential part of the methodology used in this study. To guarantee the reliability and applicability of the findings, several validation techniques were used. The primary method included cross-validating the correlation analysis results by employing multiple cultural milestones. This process tests the findings' generalizability, ensuring that the conclusions drawn are not unique to a specific dataset, but hold true across different sets of data.

In addition to cross-validation, this study emphasized meticulous data verification. The data were checked multiple times for accuracy and consistency, thus reinforcing the robustness of the research outcomes. This rigorous examination ensured that the data accurately reflected the intended cultural milestones and supported the validity of the Fibonacci time series modeling approach.

5. Case studies

To validate the theoretical framework and methodology, this study employs a diverse set of historical milestones that span significant periods in human cultural history.

- Copper Age (c. 4000 BCE): During the Copper Age, around 4000 BCE, human societies experienced a pivotal moment marked by the widespread use of copper for tools and weaponry, setting the stage for future metallurgical innovations.

- Iron Age (c. 1000 BCE): The Iron Age, beginning around 1000 BCE, saw the dominance of iron in tool and weapon production, leading to significant advancements in technology and warfare.

- Classical Age (c. 500 BCE): Around 500 BCE, the Classical Age witnessed the rise of influential civilizations, such as Rome and Greece, known for their contributions to philosophy, law, and architecture.

- Common Era Age (c. 1 CE): Starting with the year 1 CE, the Common Era Age signifies the modern calendar era.

- Medieval Age (c. 500 CE): After the fall of the Western Roman Empire in 476 CE, the Medieval Age emerged, marked by feudalism, the Byzantine Empire, and the Islamic Golden Age.

- Early Modern Age (c. 14th CE): The Early Modern Age, beginning in the fourteenth century, saw the Italian Renaissance and the Age of Exploration, fostering cultural and geographical transformation.

- Scientific Age (c. mid-seventeenth century CE): The Scientific Age, roughly starting in the mid-seventeenth century, witnessed a surge in scientific discoveries and methodologies, challenging traditional beliefs.

- Industrial Age (c. mid-eighteenth century CE): Starting in the mid-eighteenth century, the Industrial Age brought mechanization, urbanization, and the Industrial Revolution, reshaping societies and economies.

- Modern Age (c. twentieth century CE): The twentieth century Modern Age was defined by World Wars, technological leaps, and cultural shifts that shaped the contemporary world.

- Contemporary Age (c. twenty-first century CE): The twenty-first century Contemporary Age represents the current era characterized by rapid technological advancements, globalization, and complex geopolitical dynamics.

These case studies collectively serve as practical applications of the research methodology and offer valuable insights into the formation and stabilization of cultural attractors across different historical periods.

6. Results

The results of this study provide empirical evidence supporting the theoretical framework and methodology outlined in the previous sections. The findings are organized into three main categories: correlation analysis, information flow optimization, and implications for cumulative culture.

6.1 Correlation analysis

The application of percent deviation analysis in this study primarily focuses on forward-looking correlations. This approach calculates deviations between actual and predicted dates, using a 2.5% significance threshold, to determine how earlier cultural milestones predict later ones. This study specifically avoids retrospective correlations, such as how a later event like a Scientific (Sci) milestone correlates with an earlier event like the Copper Age. Instead, it emphasizes the predictive power of earlier milestones over later milestones.

To understand how percent deviation analysis applies, consider the reference date of 1967 from the "Cp" column. To identify dates within ±2.5% of 1967, the lower and upper bounds are calculated by multiplying 1967 by 0.975 and 1.025, resulting in approximately 1917.8 and 2016.2, respectively. Any date falling within this range is considered within the 2.5% threshold. In this example, the subsequent table columns are examined to identify any dates that fall within this range, thereby confirming if they deviate by less than 2.5% from 1967.

Table 1 presents the dataset, with each column representing a specific historical age and its corresponding predicted dates, based on the Fibonacci sequence. Here's how the data points align within a 2.5% deviation threshold, organized by column.

1. Cp (Copper Age)—projected year 1967

   - 1983 (Scientific, within 2.5%): NASA's STS-7 mission, the beginning of the Internet (ARPANET adopted TCP/IP, leading to a network that became the basis for the Internet).

   - 1956 (Industrial, within 2.5%): IBM invented the first hard disk. This year was significant in the field of technology and industry, marking a pivotal moment in the evolution of data storage and computing

   - 1918–1978 (Modern Age, within 2.5%): end of World War I, mid-century technological and geopolitical changes.

   - 2000–2017 (Contemporary, within 2.5%): millennium events, 9/11 attacks, 2007–2008 financial crisis.

Fib no.	Days	Years	Cp	Ir	Clas	CE	Med	EMod	Sci	Ind	Mod	Cont
0	0.000	0.000	−4000.000	−1000.000	−500.000	1.000	500.000	1300.000	1650.000	1750.000	1900.000	2000.000
1	1.000	0.003	−3999.997	−999.997	−499.997	1.003	500.003	1300.003	1650.003	1750.003	1900.003	2000.003
1	1.000	0.003	−3999.995	−999.995	−499.995	1.005	500.005	1300.005	1650.005	1750.005	1900.005	2000.005
2	2.000	0.005	−3999.989	−999.989	−499.989	1.011	500.011	1300.011	1650.011	1750.011	1900.011	2000.011
3	3.000	0.008	−3999.981	−999.981	−499.981	1.019	500.019	1300.019	1650.019	1750.019	1900.019	2000.019
5	5.000	0.014	−3999.967	−999.967	−499.967	1.033	500.033	1300.033	1650.033	1750.033	1900.033	2000.033
8	8.000	0.022	−3999.945	−999.945	−499.945	1.055	500.055	1300.055	1650.055	1750.055	1900.055	2000.055
13	13.000	0.036	−3999.910	−999.910	−499.910	1.090	500.090	1300.090	1650.090	1750.090	1900.090	2000.090
21	21.000	0.058	−3999.852	−999.852	−499.852	1.148	500.148	1300.148	1650.148	1750.148	1900.148	2000.148
34	34.000	0.093	−3999.759	−999.759	−499.759	1.241	500.241	1300.241	1650.241	1750.241	1900.241	2000.241
55	55.000	0.151	−3999.608	−999.608	−499.608	1.392	500.392	1300.392	1650.392	1750.392	1900.392	2000.392
89	89.000	0.244	−3999.364	−999.364	−499.364	1.636	500.636	1300.636	1650.636	1750.636	1900.636	2000.636
144	144.000	0.395	−3998.970	−998.970	−498.970	2.030	501.030	1301.030	1651.030	1751.030	1901.030	2001.030
233	233.000	0.638	−3998.331	−998.331	−498.331	2.669	501.669	1301.669	1651.669	1751.669	1901.669	2001.669
377	377.000	1.033	−3997.299	−997.299	−497.299	3.701	502.701	1302.701	1652.701	1752.701	1902.701	2002.701
610	610.000	1.671	−3995.627	−995.627	−495.627	5.373	504.373	1304.373	1654.373	1754.373	1904.373	2004.373
987	987.000	2.704	−3992.923	−992.923	−492.923	8.077	507.077	1307.077	1657.077	1757.077	1907.077	2007.077
1597	1597.000	4.375	−3988.548	−988.548	−488.548	12.452	511.452	1311.452	1661.452	1761.452	1911.452	2011.452
2584	2584.000	7.079	−3981.468	−981.468	−481.468	19.532	518.532	1318.532	1668.532	1768.532	1918.532	2018.532
4181	4181.000	11.455	−3970.014	−970.014	−470.014	30.986	529.986	1329.986	1679.986	1779.986	1929.986	2029.986
6765	6765.000	18.534	−3951.479	−951.479	−451.479	49.521	548.521	1348.521	1698.521	1798.521	1948.521	
10,946	10946.000	29.989	−3921.490	−921.490	−421.490	79.510	578.510	1378.510	1728.510	1828.510	1978.510	
17,711	17711.000	48.523	−3872.967	−872.967	−372.967	128.033	627.033	1427.033	1777.033	1877.033	2027.033	

Fib no.	Days	Years	Cp	Ir	Clas	CE	Med	EMod	Sci	Ind	Mod	Cont
28,657	28657.000	78.512	−3794.455	−794.455	−294.455	206.545	705.545	1505.545	1855.545	1955.545		
46,368	46368.000	127.036	−3667.419	−667.419	−167.419	333.581	832.581	1632.581	1982.581	2082.581		
75,025	75025.000	205.548	−3461.871	−461.871	38.129	539.129	1038.129	1838.129	2188.129			
121,393	121393.000	332.584	−3129.288	−129.288	370.712	871.712	1370.712	2170.712				
196,418	196418.000	538.132	−2591.156	408.844	908.844	1409.844	1908.844					
317,811	317811.000	870.715	−1720.441	1279.559	1779.559	2280.559	2779.559					
514,229	514229.000	1408.847	−311.594	2688.406	3188.406							
832,040	832040.000	2279.562	1967.967									
1,346,269	1346269.000	3688.408	5656.375									
2,178,309	2178309.000	5967.970										
3,524,578	3524578.000	9656.378										

Table 1.
Comparison of Fibonacci sequence (fib no.) with historical ages (years).

2. Ir (Iron Age)—projected year 1280
    - 1300 (Early Modern, within 2.5%): Renaissance beginnings, cultural, and scientific advancements.
3. Clas (Classical Age)—projected year 1779
    - 1777 (Scientific, within 2.5%): enlightenment period, scientific, and philosophical progress.
    - 1750—1798 (Industrial, within 2.5%): early Industrial Revolution developments.
4. CE (Common Era)—projected year 1409
    - 1378—1427 (Early Modern, within 2.5%): late Medieval period, Renaissance beginnings.
5. Med (Medieval)—projected year 1908
    - 1877 and 1955 (Industrial, within 2.5%): industrial era growth and transformation.
    - 1900—1948 (Modern, within 2.5%): World Wars and the interwar period, societal shifts.
6. EMod (Early Modern)—projected year 1838
    - 1855 (Scientific, within 2.5%): mid-nineteenth-century scientific and industrial progress.
    - 1798–1877 (Industrial, within 2.5%): core period of the Industrial Revolution.
7. Sci (Scientific)—projected year 1982
    - 1955 (Industrial, within 2.5%): post-WWII industrial development, Cold War beginnings.
    - 1948–2027 (Modern, within 2.5%): post-WWII and Cold War era, twenty-first-century advancements.
    - 2000–2029 (Contemporary, within 2.5%): Digital Revolution, globalization.
8. Ind (Industrial)—projected year 1955
    - 1907–1978 (Modern, within 2.5%): twentieth century, marked by global conflicts and changes.
    - 2000–2002 (Contemporary, within 2.5%): early twenty-first-century technological advancements.

9. Mod (Modern)—projected year 1978
    - 2000–2029 (Contemporary, within 2.5%): early twenty-first-century events, technological progress (**Table 1**).

These results revealed a consistent pattern of alignment between key historical periods and later cultural epochs, fitting within a 2.5% significance threshold. Specifically, the Copper Age, beginning around 4000 BCE and projected to align with 1967, shows significant correspondence with later milestones in the Scientific (1983), Industrial (1956), Modern (1918–1978), and Contemporary (2000–2017) Ages. In a similar vein, the Medieval Age, starting near 500 CE with pivotal year projected as 1908, demonstrates predictive alignment with the Industrial (1877 and 1955) and Modern (1900–1948) Ages. These observed alignments suggest that culturally significant periods, such as the Copper and Medieval Ages, exert a discernible influence on the timing of key events in subsequent eras, such as the Scientific, Modern, and Contemporary Ages. This pattern supports the hypothesis that the Fibonacci sequence exerts a widespread predictive effect across the different stages of cultural evolution.

6.2 Information flow optimization

One of the key objectives of this research was to explore the implications of the findings for optimizing the information flow in cultural transmission. The strong correlation between the Fibonacci sequence and the formation of cultural attractors suggests that there is an optimal way to structure the information flow to facilitate the emergence and stabilization of these attractors [9].

For instance, a case study of the Scientific Revolution revealed that the intervals between key milestones, such as the publication of Copernicus's heliocentric model and Newton's laws of motion, were consistent with Fibonacci numbers. This suggests that the rate at which groundbreaking scientific ideas were disseminated and accepted by the broader community followed an optimal path, thereby contributing to the formation of stable cultural attractors.

These findings are not only consistent with the theoretical framework but also resonate with external research. These findings on the optimization of information flow in cultural attractor landscapes resonate with recent work that demonstrated that the fractal structure of human and primate social networks is critical for dynamic self-organization and exhibits a form of collective intelligence [15].

6.3 Implications for cumulative culture

These findings have profound implications for the concept of cumulative culture, which refers to collective learning and knowledge accumulation over generations. The strong correlation between the Fibonacci sequence and cultural attractors suggests a mathematical basis for the emergence of cumulative culture [11]. This provides a framework for understanding how cultural knowledge can be effectively transmitted and accumulated over time.

For example, the analysis of the Copper Age revealed that key developments, such as the advent of metalworking, urbanization, and the establishment of trade networks, predicted milestones in later periods, including the Scientific, Industrial, and Modern Ages. This suggests that these critical milestones in human history were not mere coincidences, but were integral parts of a larger, patterned progression that underpinned the evolution of complex societies. This alignment with the Fibonacci sequence implies a fractal-like structure in the unfolding of human civilization, highlighting the systematic accumulation of cultural knowledge over time.

In summary, the results validate the theoretical framework and methodology employed in this research. They demonstrated that the Fibonacci time series is a robust tool for predicting the formation and stabilization of cultural attractors. Moreover, the findings offer new insights into the optimization of information flow in cultural transmission and have significant implications for the concept of cumulative culture.

7. Discussion

The discussion section aims to interpret the results of the research, identify its limitations, and explore its practical applications. The findings from the correlation analysis, information flow optimization, and implications for cumulative culture were examined in depth to provide a comprehensive understanding of the research outcomes.

7.1 Interpretation of results

The results obtained from the percentage deviation analysis support the research hypothesis that cultural attractors can be predicted and understood through mathematical sequences. Several cultural milestones in the case studies showed deviations below the 2.5% significance threshold, suggesting that the formation of cultural attractors aligns closely with the Fibonacci sequence. This supports the idea that cultural evolution is not a random process but is guided by underlying structures that can be quantified [1, 6].

In the context of understanding the emergence of order, it is argued that complex systems often exhibit self-organization and the generation of new structures in response to fluctuations and perturbations [16]. In 'A New Kind of Science,' similar concepts are explored in the context of complex systems [17], offering insights into how simple rules can lead to complex behaviors in various domains, including cultural evolution.

Findings related to information flow optimization indicate that there is an optimal path for the dissemination and acceptance of cultural elements, which facilitates the formation of stable cultural attractors [9]. In the context of the Scientific Age case study, it becomes apparent that the intervals between groundbreaking scientific discoveries closely align with the Fibonacci numbers. Notably, this alignment extended to the later Invention of the World Wide Web by Tim Berners-Lee in 1989, reinforcing the significance of mathematical patterns in shaping cultural evolution.

Finally, the implications for the cumulative culture are profound. This research provides a mathematical framework for understanding how cultural elements can be effectively transmitted across generations, thereby contributing to the collective knowledge and advancement of society [11]. This is exemplified by cases where foundational elements such as the Copper Age followed a Fibonacci sequence that later predicted the emergence of the Scientific Age, Industrial Age, and Modern Age.

7.2 Limitations and future research

While this research provides compelling evidence supporting the theoretical framework, it is not without limitations. One of the primary limitations is the availability and quality of historical data to backtest the model. The data points used in the case studies are significant cultural milestones; however, they are not exhaustive, and the research could benefit from a more comprehensive dataset.

Another limitation is the focus on macro-level cultural phenomena, without delving into the micro-level psychological and social factors that contribute to individual behavior and cognition. Understanding these micro-level factors could provide a more nuanced view of how cultural attractors form and stabilize.

One of the limitations of this study was the simplicity of the data analysis method. While percent deviation analysis is straightforward and accessible, it may not capture the full complexity of the relationship between cultural milestones and mathematical sequences. Future research should employ more complex statistical methods to provide a more nuanced understanding of this relationship.

Future research could address these limitations by incorporating more diverse datasets and by extending the model to include micro-level variables. Additionally, the application of Fibonacci time series modeling to other types of time series data, such as economic or environmental indicators, could provide valuable insights into the universality of the observed patterns.

7.3 Practical applications

This study has several practical applications in various disciplines. In cultural studies and anthropology, these findings offer a quantitative tool for predicting the formation and stabilization of cultural norms and practices [1, 6]. This could be particularly useful for policymakers and social scientists aiming to influence cultural change in the desired direction.

In the field of computer science and data analytics, Fibonacci time series modeling provides a novel approach for time series analysis that could be applied to various types of data, from stock market trends to social media analytics [9].

Moreover, this study has implications for education and knowledge dissemination. Understanding the optimal paths for information flow could help educators and content creators design more effective curricula and communication strategies [11].

In conclusion, this study provides a robust theoretical and methodological framework for understanding the dynamics of cultural evolution. While there are limitations that future research could address, the findings offer valuable insights into the optimization of information flow in cultural transmission and have significant implications for the concept of cumulative culture.

8. Conclusion

This research represents a groundbreaking interdisciplinary effort to blend mathematics and cultural studies. Applying Fibonacci time series modeling to diverse historical periods, from the Copper Age to the Contemporary Age, it explored various cultural phenomena. The findings revealed a strong correlation between the Fibonacci sequence and the development of cultural attractors, affirming the research hypothesis. Additionally, they suggested an optimal pathway for information flow in cultural transmission, impacting our understanding of cumulative culture.

Significantly, this study contributes a new theoretical framework that connects mathematical sequences with cultural attractors, thus addressing a gap in the existing literature. Fibonacci time series modeling, a novel tool for predicting the emergence and stability of cultural attractors, has applications in a range of disciplines, including cultural studies, anthropology, biology, psychology, computer science, and education. It provides a quantitative tool for policymakers, social scientists, and educators to understand and influence cultural evolutions.

Furthermore, this research encourages interdisciplinary collaboration by breaking traditional academic silos. It underscores the non-random nature of cultural evolution, guided by mathematical structures, and highlights the intricacies and patterns inherent in human historical narratives.

In summary, this study presents a novel approach to decoding cultural evolution, pointing towards a structured and patterned progression in the history of human civilization.

Author details

Rob G. Sacco
Fibonacci LifeChart, Toronto, Canada

*Address all correspondence to: robgsacco@gmail.com

References

[1] Axelrod R. The Complexity of Cooperation: Agent-Based Models of Competition and Collaboration. Princeton: Princeton University Press; 1997

[2] Barabási A-L. Network Science. Cambridge: Cambridge University Press; 2016. DOI: 10.1017/CBO9781316827452

[3] Cavalli-Sforza LL, Feldman MW. Cultural Transmission and Evolution: A Quantitative Approach. Princeton: Princeton University Press; 1981

[4] Sacco RG, Torday JS. Systems biology of human aging: A Fibonacci time series model. Progress in Biophysics and Molecular Biology. 2023;**177**:24-33. DOI: 10.1016/j.pbiomolbio.2022.10.005

[5] Mandelbrot BB. The Fractal Geometry of Nature. New York: W. H. Freeman and Company; 1983. 468 p

[6] Nowak MA, Highfield R. SuperCooperators: Altruism, Evolution, and Why we Need each Other to Succeed. New York: Free Press; 2011. 336 p. DOI: 10.1007/s13398-014-0173-7.2

[7] Strogatz SH. Sync: How Order Emerges from Chaos in the Universe, Nature, and Daily Life. New York: Hyperion; 2003. 352 p

[8] Mitchell M. Complexity: A Guided Tour. New York: Oxford University Press; 2009. 368 p

[9] Goldenfeld N, Kadanoff LP. Simple lessons from complexity. Science. 1999;**284**(5411):87-89. DOI: 10.1126/science.284.5411.87

[10] Bentley RA, Lipo CP, Herzog HA, Hahn MW. Regular rates of popular culture change reflect random copying. Evolution and Human Behavior. 2007;**28**(3):151-158. DOI: 10.1016/j.evolhumbehav.2006.10.002

[11] Mesoudi A. Cultural Evolution: How Darwinian Theory Can Explain Human Culture and Synthesize the Social Sciences. Chicago: University of Chicago Press; 2011. DOI: 10.7208/chicago/9780226520452.001.0001

[12] Sperber D. Explaining Culture: A Naturalistic Approach. Oxford: Blackwell; 1996

[13] Claidière N, Scott-Phillips TC, Sperber D. How Darwinian is cultural evolution? Philosophical Transactions of the Royal Society B: Biological Sciences. 2014;**369**(1642). DOI: 10.1098/rstb.2012.0404

[14] Arneodo A, Argoul F, Bacry E, Muzy JF, Tabard M. Fibonacci sequences in diffusion-limited aggregation. In: Growth Patterns in Physical Sciences and Biology. Vol. 304. 1993. pp. 191-202

[15] West BJ, Culbreth G, Dunbar RI, Grigolini P. Fractal structure of human and primate social networks optimizes information flow. Proceedings of the Royal Society A. 2023;**479**(2274):20230028

[16] Prigogine I, Stengers I. Order out of Chaos: Man's New Dialogue with Nature. New York: Bantam Books; 1984. 349 p

[17] Wolfram S. A New Kind of Science. Champaign, IL: Wolfram Media; 2002. 1197 p

Edited by Jorge Rocha,
Sandra Oliveira and Cláudia M. Viana

Time series analysis describes, explains, and predicts changes in a phenomenon through time. People have utilized techniques that add a distinctive spatial dimension to this type of analysis. Major applications of spatiotemporal analysis include forecasting epidemics, analyzing the development of traffic conditions in urban networks, and forecasting/backcasting economic risks such as those associated with changing house prices and the occurrence of hazardous events. This book includes contributions from researchers, scholars, and professionals about the most recent theory, models, and applications for interdisciplinary and multidisciplinary research encircling disciplines of computer science, mathematics, statistics, geography, and more in time series analysis and forecasting/backcasting.

Published in London, UK

IntechOpen

ISBN 978-0-85466-054-4
9 780854 660544