

UNIVERSIDAD DE CÓRDOBA



Departamento de Informática y Análisis Numérico

*Minería de datos en series temporales:  
preprocesamiento, análisis, segmentación y predicción.  
Aplicaciones*

**Doctorado con Mención Internacional**

Programa de doctorado: Computación Avanzada, Energía y Plasmas

**Antonio Manuel Durán Rosal**

Directores

César Hervás Martínez

Pedro Antonio Gutiérrez Peña

Departamento de Informática y Análisis Numérico

Córdoba, marzo de 2019

TITULO: *Time series data mining: preprocessing, analysis, segmentation and prediction. Applications*

AUTOR: *Antonio Manuel Durán Rosal*

---

© Edita: UCOPress. 2019  
Campus de Rabanales  
Ctra. Nacional IV, Km. 396 A  
14071 Córdoba

<https://www.uco.es/ucopress/index.php/es/>  
[ucopress@uco.es](mailto:ucopress@uco.es)

---

UNIVERSITY OF CÓRDOBA



Department of Computer Science and Numerical Analysis

*Time series data mining:  
preprocessing, analysis, segmentation and prediction.  
Applications*

**International Doctorate**

Program: Advanced Computing, Energy and Plasmas

**Antonio Manuel Durán Rosal**

Supervisors

César Hervás Martínez

Pedro Antonio Gutiérrez Peña

Department of Computer Science and Numerical Analysis

Córdoba, March 2019



La memoria titulada "*Time series data mining: preprocessing, analysis, segmentation and prediction. Applications*", que presenta D. Antonio Manuel Durán Rosal para optar al Título de Doctor, ha sido realizada dentro del programa de doctorado "Computación Avanzada, Energía y Plasmas" del Departamento de Informática y Análisis Numérico de la Universidad de Córdoba bajo la dirección del Doctor D. César Hervás Martínez y del Doctor D. Pedro Antonio Gutiérrez Peña.

El doctorando D. Antonio Manuel Durán Rosal y los directores de la Tesis D. César Hervás Martínez y D. Pedro Antonio Gutiérrez Peña garantizamos, al firmar esta Tesis Doctoral, que el trabajo ha sido realizado por el doctorando, bajo la dirección de los directores de la Tesis y que, hasta donde nuestro conocimiento alcanza, en la realización del trabajo, se han respetado los derechos de otros autores a ser citados, cuando se han utilizado sus resultados o publicaciones.

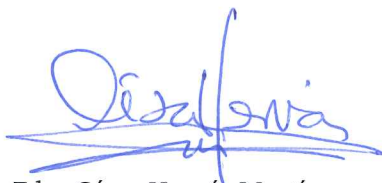
Córdoba, marzo de 2019

El doctorando

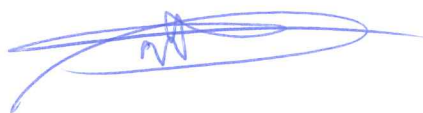


Fdo: Antonio Manuel Durán Rosal

Los directores



Fdo: César Hervás Martínez



Fdo: Pedro Antonio Gutiérrez Peña



**TÍTULO DE LA TESIS:**

Minería de datos en series temporales: preprocesamiento, análisis, segmentación y predicción. Aplicaciones.

Time series data mining: preprocessing, analysis, segmentation and prediction. Applications.

**DOCTORANDO:**

Antonio Manuel Durán Rosal

**INFORME RAZONADO DEL/DE LOS DIRECTOR/ES DE LA TESIS**

(se hará mención a la evolución y desarrollo de la tesis, así como a trabajos y publicaciones derivados de la misma).

En su Tesis, D. Antonio Manuel Durán Rosal ha abordado diferentes tareas de minería de datos en series temporales. La tesis se divide en cuatro partes correspondientes a la propuesta de nuevos métodos y a la optimización de los existentes para el procedimiento automático de análisis y preprocesamiento, la segmentación, la predicción, y el ajuste de distribuciones teóricas en series temporales. Estas metodologías están basadas en la creación de algoritmos bioinspirados, híbridos y de aprendizaje automático. Las diferentes metodologías propuestas han sido aplicadas a distintos problemas como la reconstrucción de valores perdidos en series de altura de ola, la detección de puntos de inflexión en series de paleoclimatología, la determinación de períodos comunes en series de índices bursátiles, la simplificación de series, la predicción de series de niebla y de altura de olas, y por último, el ajuste de distribuciones de probabilidad de series de altura de ola para la determinación de los umbrales para la metodología *Peak-Over-Threshold* asociada a distribuciones de valores extremos.

Estos resultados se han visto avalados por la publicación de más de una decena de artículos en revistas internacionales de impacto, así como otros tantos en conferencias de carácter nacional e internacional, mostrando la calidad científica de las contribuciones. Es por ello, que esta Tesis se ha presentado por compendio de artículos.

Por todo ello, se autoriza la presentación de la tesis doctoral.

Córdoba, 01 de Marzo de 2019

Firma del/de los director/es

Fdo.: César Hervás Martínez

Fdo.: Pedro Antonio Gutiérrez Peña





Esta Tesis Doctoral ha sido financiada en parte con cargo a los Proyectos **TIN2014-54583-C2-1-R** y **TIN2017-85887-C2-1-P** del Ministerio de Ciencia, Innovación y Universidades (MICINN), con fondos FEDER, y con el programa de Formación del Profesorado Universitario FPU, referencia **FPU14/03039**, del Ministerio de Educación y Formación Profesional / Ministerio de Cultura y Deporte (MECD). La estancia de investigación para la obtención de la Mención Internacional ha sido financiada por el anterior programa, bajo ayuda con referencia **EST17/00297**.

This Doctoral Thesis has been partially subsidised by the **TIN2014-54583-C2-1-R** and **TIN2017-85887-C2-1-P** projects of the Spanish Ministry of Science, Innovation and Universities (MICINN), FEDER funds, and the FPU Predoctoral Program of the Spanish Ministry of Education and Vocational Training / Ministry of Culture and Sport (MECD), grant reference **FPU14/03039**. The research stay has also been subsidised by the last program, grant reference **EST17/00297**.





## Mención de Doctorado Internacional

Esta Tesis cumple los criterios establecidos por la Universidad de Córdoba para la obtención del Título de Doctor con Mención Internacional. Para ello se presentan los siguientes requisitos:

1. Estancia predoctoral realizada en otros países europeos:
  - **School of Computer Science, University of Birmingham, Birmingham, Reino Unido.** Duración de tres meses desde el 1 de marzo hasta el 1 de junio de 2018. Tutor de la estancia: **Dr. Peter Tiño**, *Full professor of School of Computer Science (University of Birmingham)*.
2. Esta Tesis está avalada por los siguientes informes de idoneidad realizados por doctores de otros centros de investigación internacionales:
  - **Dr. Huanhuan Chen.** *Professor of School of Computer Science, University of Science and Technology of China (China)*.
  - **Dr. Mario Gongora.** *Associate Professor of School of Computer Science and Informatics, De Montfort University (Reino Unido)*.
3. La defensa de la Tesis y el texto se han realizado totalmente en inglés.
4. Entre los miembros del tribunal se encuentra un doctor procedente de un centro de educación superior europeo, tratándose del Dr. **David Elizondo**, *Professor of School of Computer Science and Informatics, De Montfort University (Reino Unido)*.

Córdoba, marzo 2019

El doctorando:



Fdo.: Antonio Manuel Durán Rosal



*La raíz de todo bien reposa en la tierra de la gratitud.*

Dalai Lama

## Agradecimientos

Me gustaría agradecer todo el apoyo mostrado por mis directores de Tesis, D. César Hervás y D. Pedro Antonio Gutiérrez. Gracias por su dedicación y por hacerme ver el incierto campo de la investigación como una oportunidad de futuro, una lucha constante consigo mismo y mejorar profesionalmente día a día. De corazón, gracias por ser personas antes que directores.

No puedo dejar de dar las gracias a mis amigos, y en especial a mis compañeros del Grupo AYRNA por hacer más fácil y llevaderos los problemas surgidos durante el desarrollo de este trabajo. Antonio, David, Javi, Juan Carlos, Julio, Manolo, María, Víctor, y un largo etcétera de personas que han pasado estos años por el laboratorio, gracias por colaborar en mi investigación ya sea trabajando cooperativamente, como competitivamente, ya que sin vosotros nada hubiera sido igual.

Por supuesto, reconocer la importancia de la familia en esta etapa de mi vida, este mérito es tanto vuestro como mío. A mis padres y mi hermano, que me han hecho mejor persona y han sabido implantar las bases de la educación en mí. A mis abuelas, por ser mi voz de la experiencia en todo momento. A mis tíos, que me han aconsejado como hermanos. A mis abuelos que sin estar presentes se que me animan a seguir en mi etapa formativa y profesional. A todos, os quiero.

Y cómo no, acordarme de la persona con la que llevo compartiendo más de los últimos nueve años de mi vida. Gracias Ana, gracias por aguantar mis cabreos, por apoyarme en cada una de las decisiones que he tomado personal y profesionalmente, por todos los momentos que han hecho convertir cada instante en inolvidable y desprender tanta felicidad. Pero sobre todo, gracias por darme el motivo de vivir y luchar cada día con una sonrisa en la cara, gracias por darme a nuestro pequeño, a nuestro Erik.

A todos, gracias de corazón.



# Index

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Machine learning . . . . .	1
1.2	Metaheuristics . . . . .	4
1.2.1	Genetic algorithms . . . . .	5
1.2.2	Coral reefs optimisation algorithms . . . . .	7
1.2.3	Particle swarm optimisation algorithms . . . . .	8
1.2.4	Hybrid algorithms . . . . .	10
1.2.5	Multiobjective algorithms . . . . .	11
1.3	Artificial neural networks . . . . .	13
1.3.1	Evolutionary artificial neural networks . . . . .	15
1.4	Time series . . . . .	17
1.4.1	Traditional prediction models . . . . .	17
1.4.2	Segmentation . . . . .	20
1.5	Extreme value theory . . . . .	22
1.6	Applications in real-world problems . . . . .	24
1.6.1	Tipping points . . . . .	24
1.6.2	Stock indexes . . . . .	26
1.6.3	Wave height time series . . . . .	27
1.6.4	Fog prediction . . . . .	28
<b>2</b>	<b>Motivation and objectives</b>	<b>31</b>
2.1	Motivation and challenges . . . . .	31
2.2	Objectives . . . . .	33
2.3	Summary of the Thesis . . . . .	34
2.4	Publications . . . . .	36
<b>3</b>	<b>Preprocessing: missing data reconstruction</b>	<b>41</b>
3.1	Massive missing data reconstruction in ocean buoys with evolutionary pro- duct unit neural networks . . . . .	41
<b>4</b>	<b>Time series segmentation</b>	<b>45</b>
4.1	Discovery of useful patterns . . . . .	45
4.1.1	Detection of early warning signals in paleoclimate data using a ge- netic time series segmentation algorithm . . . . .	47

4.1.2	Identification of extreme wave heights with an evolutionary algorithm in combination with a likelihood-based segmentation . . . . .	50
4.1.3	Identifying market behaviours using European stock index time series by a hybrid segmentation algorithm . . . . .	52
4.1.4	On the use of evolutionary time series analysis for segmenting paleoclimate data . . . . .	54
4.2	Time series size reduction . . . . .	56
4.2.1	A statistically-driven coral reef optimisation algorithm for optimal size reduction of time series . . . . .	57
4.2.2	A hybrid dynamic exploitation barebones particle swarm optimisation algorithm for time series segmentation . . . . .	60
4.3	Multiobjective time series segmentation . . . . .	74
4.3.1	Simultaneous optimisation of clustering quality and approximation error for time series segmentation . . . . .	74
<b>5</b>	<b>Prediction</b>	<b>77</b>
5.1	Detection and prediction of segments containing extreme significant wave heights . . . . .	78
5.2	Efficient fog prediction with multi-objective evolutionary neural networks .	81
<b>6</b>	<b>Statistical distribution-based learning</b>	<b>83</b>
6.1	On the use of a mixed distribution to fix the threshold for peak-over-threshold wave height estimation . . . . .	84
6.2	Distribution-based discretisation and ordinal classification applied to wave height prediction . . . . .	99
<b>7</b>	<b>Discussion and conclusions</b>	<b>101</b>
7.1	Conclusions . . . . .	101
7.1.1	Preprocessing . . . . .	101
7.1.2	Segmentation . . . . .	102
7.1.3	Prediction . . . . .	104
7.1.4	Statistical-distribution based learning . . . . .	105
7.2	Generic discussion and future work . . . . .	106



# Figure index

1.1.1	Paradigms of ML methods: a) Supervised learning with two labels b) Un-supervised learning c) Semisupervised Learning. . . . .	2
1.2.1	Flowchart of a standard GA. . . . .	5
1.2.2	Flowchart of the standard CRO algorithm. . . . .	7
1.2.3	Flowchart of the PSO algorithm. . . . .	8
1.2.4	Example of a Pareto front in a bidimensional minimisation MOP. Blue points represent the Pareto front, while the pink ones are the dominated solutions. . . . .	12
1.3.1	Example of a feed-forward neural network. . . . .	13
1.4.1	Example of time series segmentation with a time series of length $N = 35$ and 6 cut points ( $t_1 = 4, t_2 = 11, t_3 = 18, t_4 = 22, t_5 = 28, t_6 = 32$ ), represented by dashed lines. The resulting segments are: $s_1 = \{y_1, y_2, y_3, y_4\}$ , $s_2 = \{y_4, y_5, y_6, y_7, y_8, y_9, y_{10}, y_{11}\}, \dots, s_7 = \{y_{32}, y_{33}, y_{34}, y_{35}\}$ . . . . .	21
1.6.1	GISP2 time series. . . . .	25
1.6.2	IBEX35 time series. . . . .	27
1.6.3	SWH time series collected from buoy 46001 in the Gulf of Alaska. . . . .	28



# Acronyms

Acronym	Meaning
ANN	Artificial neural network
AM	Annual maximum
AR	Autoregressive
ARIMA	Autoregressive integrated moving average
ARMA	Mixed autoregressive moving average
BBePSO	Exploiting barebones particle swarm optimisation
BBPSO	Barebones particle swarm optimisation
BP	Backpropagation
COP	Combinatorial optimisation problem
CRO	Coral reef optimisation
DBBePSO	Dynamic exploitation barebones particle swarm optimisation
DO	Dansgaard-Oeschger
DT	Decision tree
EA	Evolutionary algorithm
EANN	Evolutionary artificial neural network
EP	Evolutionary programming
EPUNN	Evolutionary product unit neural network
EVT	Extreme value theory
FNN	Feed-forward neural network
GA	Genetic algorithm
GEV	Generalised extreme value
GISP2	Greenland ice sheet project two
GMOTSS	Evolutionary multiobjective time series segmentation algorithm
GPD	Generalised Pareto distribution
HA	Hybrid algorithm
HMM	Hidden Markov model
LR	Logistic regression
LS	Local search
MA	Moving average
MH	Metaheuristic
ML	Machine learning
MLE	Maximum likelihood estimation
MLP	Multilayer perceptron

Acronym	Meaning
MOEA	Multiobjective evolutionary algorithm
MOP	Multiobjective optimisation problem
NDBC	National Data Buoy Center
NGRIP	North Greenland ice core project
NNEP	Neural network evolutionary programming
NOAA	National Oceanic and Atmospheric Administration
NSGA-II	Nondominated sorting genetic algorithm
NWP	Numerical weather prediction
OCSVM	One class support vector machine
PLA	Piecewise linear approximation
POT	Peak-over-threshold
PSO	Particle swarm optimisation
PU	Product unit
PUNN	Product unit neural network
RBF	Radial basis function
RBFNN	Radial basis function neural network
REDSVM	Reduction applied to support vector machine
RVR	Runway visual range
SA	Simulated annealing
SCRO	Statistically-driven coral reefs optimisation
SSWH	Segments containing very high significant wave height
SU	Sigmoidal unit
SUNN	Sigmoidal unit neural network
SVDD	Support vector data descriptor

*Education is the passport to the future, for tomorrow  
belongs to those who prepare for it today.*

Malcolm X

# 1

## Introduction

This Thesis proposes novel methods for mining time series with the aim to automatically solve different real-world problems. In this way, this chapter introduces the context of this work. The main topics which have been considered in the present research are: machine learning (ML), metaheuristics (MHs), artificial neural networks (ANNs), time series, extreme value theory (EVT) for fitting the statistical distribution of a set of values, and several applications in real-world problems, such as, detection of tipping points (TP), analysis of stock market indexes, preprocessing, segmentation and prediction in wave height problems, and fog prediction in airports.

### 1.1. Machine learning

Nowadays, the big amount of data which is present in any field of science and in the daily life can have different ways of representations, such as databases or time series. The exponential growth of this data makes us using automatic techniques to extract knowledge from them, due to the impossibility to process and analyse these data conveniently in a manual way. In this work, the kind of data which is analysed, preprocessed, and, in general, mined, is in the form of time series. A time series can be defined as temporal data which are collected chronologically or as a function that varies during time.

The extraction of knowledge is commonly referred to by the term data science,

which involves a wide range of theories derived from mathematics, statistics or ML, among others. In this way, ML is one of the most important fields of the artificial intelligence, whose primary objective is to make possible the automatic learning of the computers (automatic extraction of knowledge) through examples of data.

ML methods can be divided depending on different criteria, among other things, considering the type of reasoning applied, regarding the manner in which the training data are presented to the learner or depending on the classification model itself. However, for this work, we consider more important to divide according to the learning task itself, that is, in supervised, unsupervised and semi-supervised learning [7]. Figure 1.1.1 shows a representation of these paradigms.

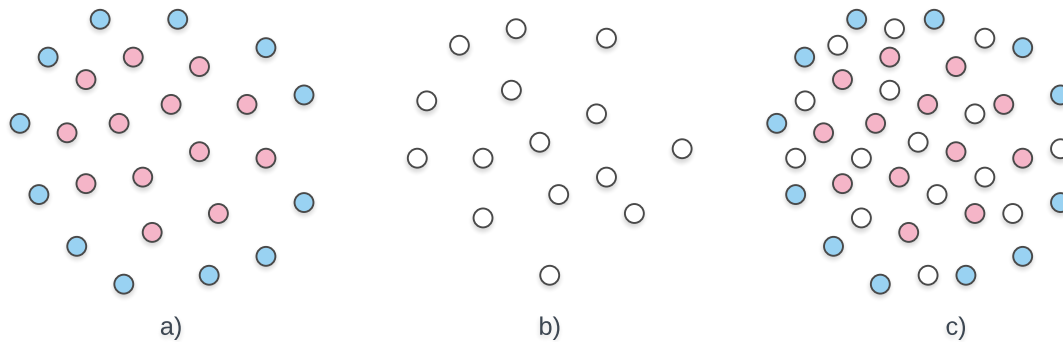


Figure 1.1.1: Paradigms of ML methods: a) Supervised learning with two labels b) Unsupervised learning c) Semisupervised Learning.

- Supervised learning: supervised learning is the most common learning problem in ML. It is said that a problem can be solved with a supervised learning algorithm when each example of data (normally presented as a vector of attributes  $X$ ) is labelled to a predefined membership class  $Y$ . For this reason, the main task is to design an automatic algorithm that learns a classification rule in a set of data called training data, which then produces the ideal output for each example of the test data. In other words, the algorithm learns decision rules from known data to predict the label of the unknown data. In this group of algorithms, we can distinguish a subdivision depending on the kind of label to predict.
  - Regression: the output variable  $Y$  is a real value. An example of an algorithm inside this group is linear regression [62].
  - Classification: the output variable  $Y$  is a discrete or nominal value. Classification methods are the most well-known ones in ML. There is a wide range of classification techniques, e.g. decision trees (DTs) [75], logistic regression (LR) [16], ANNs [79], or support vector machines (SVM) [31].

- Ordinal classification: when the classification of patterns is done into naturally ordered discrete labels, the paradigm is called ordinal classification or ordinal regression [83]. As can be seen, it is a mixture between the two previous kinds of algorithms given that the label to be predicted is discrete but there is also an order between the categories. However, this order is not quantified: for example, if we consider a problem where we have to classify a database of humans into *child*, *teenager*, *adult* or *ancient* labels, it is obvious that classifying a *child* as an *ancient* should be more penalised than classifying the same pattern as a *teenager*. Nevertheless, we cannot quantify the distance between classes, that is, we are sure that the distance between *child* and *teenager* is less than the distance between *child* and *adult*, but we do not know if the distance between *child* and *adult* is, for example, twice the distance between *child* and *teenager*. This a novel paradigm which is receiving a lot of attention, and in the Thesis is presented in one of the last publications.
- Unsupervised learning: in this case, the data is unlabelled [34], so the task differs from supervised learning. Unsupervised learning tries to discover several groups of data which present a similar structure, to determine the data distribution, or to project the data into a smaller dimensional space to visualise them. This paradigm includes the so-called clustering algorithms whose main objective is to make groups of patterns depending on the similarities of the input characteristics. Clustering methods are usually divided into partitional, hierarchical, and density-based algorithms. Partitional algorithms, such as K-means [53], need to know how many clusters are going to be discovered. Initially, the centre (centroid) of each cluster is chosen randomly, and during every iteration, it is moved in order to have more compact groups, well separated from other. Hierarchical algorithms can be agglomerative (each pattern is considered a cluster, and in each iteration, the nearest clusters are merged), or divisive (all patterns are considering into the same cluster, and in each iteration, the most different pattern is separated from its cluster). Finally, density based algorithms, such as DBSCAN [21], form their clusters using the density of the data in the space.
- Semi-supervised learning: when the problem presents a set of data which is labelled and a big set which is unlabelled, the paradigm is called semi-supervised learning. Sometimes labelled data is not accessible, or the labels depend on an expert in the area of the problem to solve. However, it can be affordable to label a subset of the dataset. Semi-supervised algorithms try to explore the unlabelled data structure with the aim to generate predictive models that work better than those which only use the labelled data.

ML includes a wide variety of algorithms with different goals depending on the problem to solve. However, all of them have a common characteristic which is to extract relevant information and knowledge from raw data. ML is used in many applications, e.g. in biomedicine, mathematics, weather forecasting, biometry, handwriting recognition, facial recognition, etc. In this Thesis, we made use of several ML algorithms, such as LR, SVM, or ANNs, among others. These algorithms have been used to improve them or merely to compare our results with those obtained in previous researches.

## 1.2. Metaheuristics

The analysis and design of algorithms are limited by the complexity of them. In the best case, an algorithm can be run in polynomial time (P-complexity). This kind of algorithms are used to solve easy problems and can be executed in a deterministic way on a computer. However, the most challenging issues are those involving NP-complexity, i.e. more complex problems with a much higher computational cost. For NP algorithms, it is not easy to find the best global solution, but it is possible to find good solutions close to the global one. In this context, we should define the meaning of heuristic and MH. On the one hand, in [72], the author affirmed that a heuristic is a technique that looks for good solutions (that is, almost optimal or effective) at a reasonable computational cost (efficient), although without guaranteeing feasibility or optimality of it. In some cases, you can not even determine how close to the optimum a particular feasible solution is. On the other hand, an MH is a method which includes heuristics and high-level procedures to solve a variety of general problems, able to escape for local optima, with incomplete information or limited computation capacity [6].

MHs are usually applied to solve combinatorial optimisation problems (COPs). A COP consists in optimising the values of some variables, i.e. maximising or minimising an objective function (sometimes with constraints). Formally, a maximisation problem could be defined as:

$$\max_{\mathbf{x} \in \mathcal{F} \subseteq \mathcal{S}} f(\mathbf{x}), \quad (1.1)$$

where  $\mathbf{x}$  is a vector of decision variables,  $f(\mathbf{x})$  is the objective function,  $\mathcal{S}$  is the search space, and  $\mathcal{F}$  is the subset of feasible solutions. The variables can be integer or real values.

MHs are classified depending on their inspiration (natural or not), the kind of objective functions (statics or dynamics), the number of neighbourhoods (one or more), the use of the memory, and more commonly, depending on the number of the solutions employed to guide the search. Attending to the last criterion, we can divide MHs in:

- Single solution MHs: the search is guided using one solution. Local search (LS) is



one of the main algorithms in this group. LS methods find the optimal solutions in a surrounding area. However, if the initialisation is not correct, the algorithm can reach a locally optimal solution. Simulated annealing (SA) [43] and tabú search (TS) [27] are examples of algorithms which try to avoid this problem.

- Population-based MHs: the search is guided by a set of solutions. They are considered robust because they perform a global search which quickly converges to high-quality areas. They are prepared to solve complex problems, and they can avoid local optima. There are many options for this kind of MHs. In this Thesis, we have focused on genetic algorithms (GA) [35], coral reef optimisation algorithms (CRO) [77], and particle swarm optimisation algorithms (PSO) [41], considering their improved variations.

### 1.2.1. Genetic algorithms

GAs are bioinspired MHs which simulate the evolution of the species [35]. They are one of the most extended types of MHs due to their properties and their capability of adaptation in new problems. The flowchart of a standard GA is summarised in Figure 1.2.1.

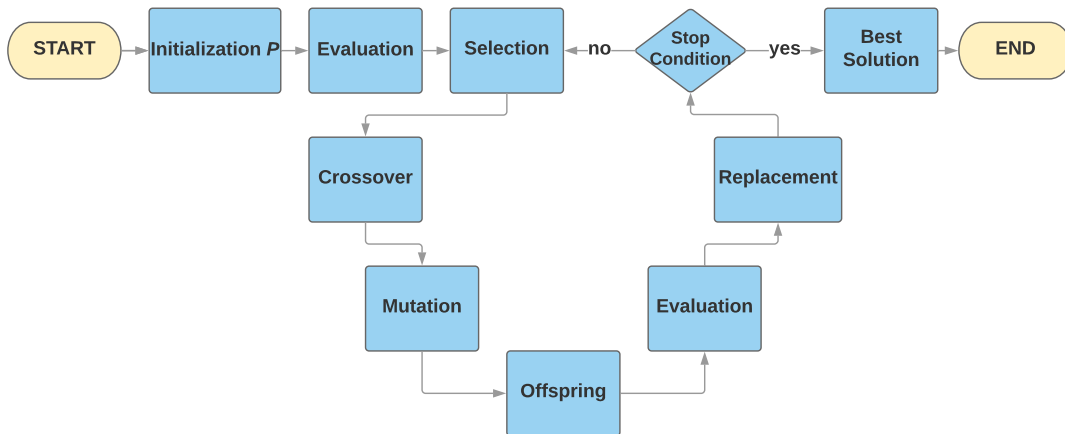


Figure 1.2.1: Flowchart of a standard GA.

In GAs, each solution is codified using a chromosome which is formed by binary, integer or real values, depending on the problem to solve. The algorithm starts initialising a population of solutions, which can be done by using different strategies, e.g. random or predefined ones.

Then, GA simulates the evolution for some generations. A selection process is applied to the whole population. This is done to choose some individuals as possible parents

to generate the offspring. There are many options in the literature, e.g. the parents can be randomly chosen, they can be selected as pairs of parents without repetition, or their objective function value can be used to select the more suitable solutions.

Once the parents are selected, for each parent, a given probability  $P_c$  (crossover probability) is used to decide whether the parent will be involved in a crossover process. Usually, when a parent  $c_1$  is selected to be crossed, the other parent  $c_2$  is randomly chosen from the set of parents, although this procedure could change depending on the algorithm. The crossover is usually applied to two parents, and it consists of an operation to create offspring solutions which preserve characteristics of both parents. This operator is essential for the convergence of GAs, and its main objective is to exploit the set of current solutions.

GAs are also endowed with mutation operations whose main objective is to guarantee the exploration of the search space during the evolution. After applying the crossover, each intermediate offspring solution  $c_i$  is selected to be mutated under a given probability  $P_m$  (mutation probability). This mutation usually consists in changing one or more elements of the chromosome by using different operations (e.g. adding white noise for real coding, performing a permutation for integer coding or flipping a bit for binary arrays). As can be seen, with these two operators, the algorithm is able to control the exploration (diversity) and the exploitation, with the purpose of maintaining an adequate convergence. Some versions of GAs adapt the probability during the evolution in order to have a greater exploration at the beginning and more exploitation at the end.

When the offspring, given by the mutation operator, is obtained, the algorithm creates the final offspring population by processing infeasible solutions, i.e. those which do not satisfy the constraints of the problem. Then, the GA evaluates the new solutions using the objective or fitness function, in order to give a score to each solution. The objective function is associated with the problem to solve. It is normalised in the interval  $[0, 1]$ , resulting in what is usually known as the fitness function of the GA.

GA simulates the natural selection principle by the selection operator and by the specific strategy used for the replacement of the new population in the next generation. During replacement, parent and offspring populations are merged into a new population. From it, a set of  $P$  individuals are considered to survive for the next generation. There are different strategies, but all of them need to reach a good compromise between fitted and diverse solutions. With this philosophy, in each iteration, the GA simulates the survival of the individuals in the current environment.

Finally, once the GA is finished, the best solution is returned.

### 1.2.2. Coral reefs optimisation algorithms

A novel evolutionary bioinspired strategy for search and optimisation is called CRO algorithm [77, 76]. It simulates the behaviour of the processes occurring in a real coral reef, which are summarised in Figure 1.2.2.

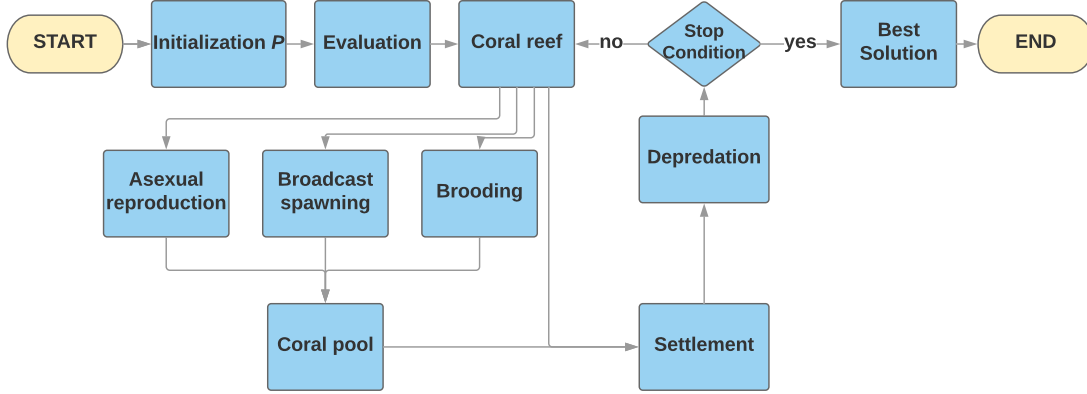


Figure 1.2.2: Flowchart of the standard CRO algorithm.

Given a coral reef with a size of  $P$  possible candidates, organized in a  $P_1 \times P_2$  grid, each position  $(i, j)$  is able to allocate a coral  $c_{i,j}$ , that is, a feasible solution for the optimisation problem to solve.

Firstly, CRO initialises the coral reef with random corals (solutions), maintaining some unfilled positions. These unoccupied positions represent holes to settle new corals in later phases of the evolution, allowing their growth. Typically, the percentage of free positions is predefined by a parameter  $\rho \in [0, 1]$ .

Once the initialisation is performed, the main block of the algorithm simulates the processes of reproduction and reef formation, which are recreated using different operators. There are two types of reproduction in the natural processes of coral reefs. The first one is called asexual reproduction, in which corals reproduce asexually by budding or fragmentation. In CRO, all corals are sorted according to their health (i.e. the fitness value in the optimisation problem). A fraction of them,  $F_a$ , are duplicated and mutated under a  $P_a$  probability to promote diversity. The new corals are settled in a pool of candidates solutions.

Secondly, the algorithm mimics the sexual reproduction, which includes the external sexual reproduction or broadcast spawning, and the internal sexual reproduction or brooding. On the one hand, broadcast spawning consists in selecting a uniform random fraction  $F_b$  of existing corals to be broadcast spawners. The main objective here is to generate new larvae. This procedure is carried out selecting two broadcast spawners and

applying any exploitation strategy, e.g. a crossover operator. The selection of the corals can be done uniformly, randomly or using any selection approach, but it is important to mention that any coral is selected only once. On the other hand, brooding represents the reproduction in hermaphrodite corals, which has been represented by considering the remaining  $1 - F_b$  percentage of individuals. This kind of reproduction is modelled using any type of mutation mechanism depending on the problem to solve. New larvae become part of the pool of candidates solutions.

When the reproduction is finished, the new larvae in the pool try to settle and grow in the reef. For each larva, a random position is  $(i, j)$  of the reef is randomly chosen. If this location is free, the new larva will settle. However, if this position is occupied, the new larva survives if it is healthier than the existing coral. In each iteration, each larva tries to look for a position for a maximum of  $\eta$  attempts.

Finally, CRO introduces a depredation procedure that simulates the death of the corals during the reef's formation. The depredation operator is applied to a  $F_d$  percentage of the worst corals under a given probability  $P_d$ . The operation consists in liberating the position for the next coral generation.

### 1.2.3. Particle swarm optimisation algorithms

The last important optimisation algorithm that have been taken into account in this Thesis is the PSO algorithms [41]. This kind of algorithm imitates the behaviour of a swarm of particles when they are looking for food (e.g. a bird flocking or a fish school). Figure 1.2.3 shows the main steps of PSO.

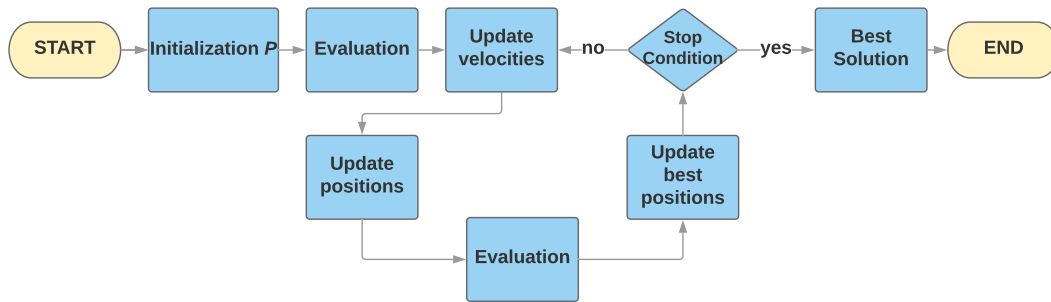


Figure 1.2.3: Flowchart of the PSO algorithm.

In PSO, a swarm corresponds with a set of  $P$  particles moving in a dimensional space of length  $D$ . Each particle  $i$  is a candidate solution of the studied problem, represented by an array  $\mathbf{x}_i$ . Also, there are other characteristics of the particles that the algorithm need to save to guarantee its correct performance. The direction and the rate of change in the movement are represented by the velocity of the particle  $\mathbf{v}_i$ , while the best position

found by the particle during the evolution is  $\mathbf{p}_i$ , that is, that position which has reached the higher value in the fitness function. The fitness function is represented by  $f(\mathbf{x}_i)$ , and it evaluates the quality of the solutions (as in other evolutionary algorithms, EAs). Moreover, an array with the best global solution ( $\mathbf{p}_g$ ) is also stored. It represents the best solution found by the whole swarm. In a maximisation problem, at iteration  $t$  of the algorithm, the solution  $\mathbf{p}_g^t$  is defined as:

$$\mathbf{p}_g^t = \arg \max_{\mathbf{p}} \{f(\mathbf{p}_g^{t-1}), f(\mathbf{p}_1^t), f(\mathbf{p}_2^t), \dots, f(\mathbf{p}_P^t)\}. \quad (1.2)$$

The evolution is done using the cooperation of the particles, considering what is called the cognitive component, i.e. the information of the given particle ( $\mathbf{p}_i$ ), and the social component, which includes the knowledge of the whole swarm ( $\mathbf{p}_g$ ). In this way, in each iteration  $t$ , PSO updates the velocities  $\mathbf{v}_i$  using the expression:

$$\mathbf{v}_i^t = w \cdot \mathbf{v}_i^{t-1} + \rho_1^t \cdot C_1 \cdot (\mathbf{p}_i^{t-1} - \mathbf{x}_i^{t-1}) + \rho_2^t \cdot C_2 \cdot (\mathbf{p}_g^{t-1} - \mathbf{x}_i^{t-1}), \quad (1.3)$$

where  $w$  is the inertia weight for controlling the effect of the previous velocities (a parameter used for velocity reduction, i.e. particles roaming),  $\rho_1, \rho_2 \sim U(0, 1)$  are uniform random values obtained at iteration  $t$ , and  $C_1, C_2$  are the acceleration constants.  $(\mathbf{p}_i - \mathbf{x}_i)$  and  $(\mathbf{p}_g - \mathbf{x}_i)$  represent the experience of the particle with respect to its best local solution and best global solution, respectively. Then, with the update velocity,  $\mathbf{v}_i^t$ , the new position of the particle is calculated as:

$$\mathbf{x}_i^t = \mathbf{x}_i^{t-1} + \mathbf{v}_i^t. \quad (1.4)$$

At the end, the best local position is  $\mathbf{p}_i^t = \arg \max \{f(\mathbf{p}_i^{t-1}), f(\mathbf{x}_i^t)\}$ , and the best global position is updated using Equation 1.2.

As can be seen, the current positions are calculated updating previous velocities. Two improved versions do not take into account these velocities. The first one is the barebones PSO (BBPSO) [40] where Equations 1.3 and 1.4 are replaced with the following expression:

$$x_{i,j}^t = N \left( \frac{p_{i,j}^{t-1} + p_{g,j}^{t-1}}{2}, |p_{i,j}^{t-1} - p_{g,j}^{t-1}| \right), \quad (1.5)$$

where  $j$  is the dimension to update, and  $N(\mu, \sigma)$  is a normal random distribution with  $\mu$  mean and  $\sigma$  standard deviation. Each position is generated with a Gaussian distribution, where  $\mu$  is the mean value between the best global and best local position, while  $\sigma$  is the absolute difference between them. This expression is based on the theoretical studies of Clerc and Kennedy [11], who confirmed that particles converge to a weighted average of the global and personal best positions.

The second one is called exploiting barebones PSO (BBPSO), where the position updates are reformulated as:

$$x_{i,j}^t = \begin{cases} N\left(\frac{p_{i,j}^{t-1} + p_{g,j}^{t-1}}{2}, |p_{i,j}^{t-1} - p_{g,j}^{t-1}|\right) & \text{if } a < 0.5, \\ p_{i,j}^{t-1} & \text{otherwise,} \end{cases} \quad (1.6)$$

where  $a$  is a random number generated from a uniform distribution  $U(0, 1)$ . The main difference concerning the BBPSO is that BBPSO searches solutions with a higher exploitation, that is, with a 0.5 probability, the  $j$ -th dimension of the particle  $i$  takes a value corresponding to the best local position. It is shown that this exploiting version outperforms other variants of PSO in many applications [65].

#### 1.2.4. Hybrid algorithms

In previous sections, we have defined some EAs. In general, they are able to find high-quality areas (those which contain solutions with a high value of fitness) in a lot of problems. For this reason, they are considered robust MHs. However, their principal disadvantage is that they are not good at finding the precise optimum in that area [36]. On the contrary case, LSs are stronger when they are looking for optima in one area, but a bad initialisation makes them very poor searching good solutions. Hybrid algorithms (HAs) were born for these reasons. The application of LSs in different parts of the evolutionary process is a way to prevent this problem. The idea is to combine the advantages of both kind of algorithms, that is, to use EAs (as global explorers) in order to reach high-quality areas, and then to apply LSs (local exploiters) to improve the solutions in that area.

There are different decisions that we have to take into account when the application of the LS in an EA is needed. The first one is referred to the kind of LS to apply, which usually depends on the studied problem and the ability of the LS to correctly solve it. Another important aspect is to decide when we want to apply the LS, i.e. in the beginning, in the end, or during the evolution. And finally, which individuals are going to be optimised, e.g. the best ones, the most representative ones, etc. These aspects affect the computational cost and the quality of the solutions. For example, the application of an LS in different parts of the evolution over a large set of solutions normally produces high-quality results, but the computational cost is huge. By contrast, the application of the LS to the best solution obtained by the EA will be faster but usually less robust. In this way, we have to establish a compromise between both extremes.

There exist different strategies previously used as the multistart approach, the Lamarckian learning, the Baldwinian learning, the partial Lamarckianism or the process of random linkage [44, 38, 87]. A special kind of hybridisation is produced when the hybrid

solutions obtained during the evolution are also included in the process of evolution for the following generations. In those cases, the combination is commonly known as memetic hybridisation. This procedure usually gets good results but is slower than standard hybridisation.

### 1.2.5. Multiobjective algorithms

A special kind of problems are those which need to be optimised considering more than one objective. These problems are called multiobjective optimisation problems (MOPs) [12], and they are formally defined using the following definitions (when considering minimisation problems):

1. MOP: the problem consists in find the vector  $\mathbf{x}^* = [x_1^*, x_2^*, \dots, x_n^*]^T$ , which satisfies the  $m$  inequality restrictions  $g_i(\mathbf{x}) \geq$  for  $i = 1, 2, \dots, m$ , the  $p$  equality restrictions  $h_i(\mathbf{x}) = 0$  for  $i = 1, 2, \dots, p$ , and optimises the function vector  $\mathbf{f}(\mathbf{x}) = [f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_k(\mathbf{x})]^T$ . The problem has  $k$  objectives and the functions  $\mathbf{f}(\cdot) : \Omega \rightarrow A$  represent the relation between the search space  $\Omega$  and the objective function space  $A$ . Given that there are multiple objectives in MOPs, the notion of optimum changes, and it is necessary to find a good compromise between them rather than obtaining a single solution (as in global optimization).
2. Pareto optimality: a solution  $\mathbf{x}^* \in \Omega$  is said to be a Pareto optimal if there is no  $\mathbf{x} \in \Omega$  whose function  $\mathbf{f}(\mathbf{x})$  dominates  $\mathbf{f}(\mathbf{x}^*)$ .
3. Pareto dominance: a vector  $\mathbf{u} = (u_1, u_2, \dots, u_k) \in A$  dominates another vector  $\mathbf{v} = (v_1, v_2, \dots, v_k) \in A$  (denoted by  $\mathbf{u} \succeq \mathbf{v}$ ), if and only if  $\forall i \in \{1, 2, \dots, k\}, u_i \leq v_i \wedge \exists i \in \{1, 2, \dots, k\} : u_i < v_i$ .
4. Pareto optimal set: for a given MOP to optimise  $\mathbf{f}(\mathbf{x})$ , the Pareto optimal set ( $\mathcal{P}^*$ ) is defined as:

$$\mathcal{P}^* := \{\mathbf{x}^* \in \Omega \mid \nexists \mathbf{x} \in \Omega, \mathbf{f}(\mathbf{x}) \succeq \mathbf{f}(\mathbf{x}^*)\}. \quad (1.7)$$

5. Pareto front: for a given MOP  $\mathbf{f}(\mathbf{x})$ , and a Pareto optimal set  $\mathcal{P}^*$ , the Pareto front ( $\mathcal{PF}^*$ ) is defined as:

$$\mathcal{PF}^* := \{\mathbf{u} = \mathbf{f}(\mathbf{x}) \mid \mathbf{x} \in \mathcal{P}^*\}. \quad (1.8)$$

6. Global minimum of a MOP: given a vector of functions  $\mathbf{f}(\cdot) : \Omega \subseteq \mathbb{R}^k \rightarrow \mathbb{R}^n$ ,  $\Omega \neq \emptyset$ , and  $k \geq 2$ , the set  $\mathcal{PF}^* : \mathbf{f}(\mathbf{x}^*)$  is called global minimum, if and only if  $\forall \mathbf{x} \in \Omega : \mathbf{f}(\mathbf{x}^*) \preceq \mathbf{f}(\mathbf{x})$ .
7. Weak Pareto optimality: a solution  $\mathbf{x}^* \in \Omega$  is a weakly nondominated solution if there is no  $\mathbf{x} \in \Omega$  such that  $f_i(\mathbf{x}) < f_i(\mathbf{x}^*)$ , for  $i = 1, 2, \dots, k$ .

8. Strict Pareto optimality: a point  $\mathbf{x}^* \in \Omega$  is strictly Pareto optimal if there is no  $\mathbf{x} \in \Omega$ ,  $\mathbf{x} \neq \mathbf{x}^*$ , such that  $f_i(\mathbf{x}) \leq f_i(\mathbf{x}^*)$ , for  $i = 1, 2, \dots, k$ .

From these definitions, for example, if we consider a bidimensional problem  $\mathbf{f}(\mathbf{x}) \in A \subseteq \mathbb{R}^2$ , the Figure 1.2.4 shows a Pareto front with seven points. In this way, we cannot conclude which is the best solution, but the seven solutions are better than those that are not in this Pareto front.

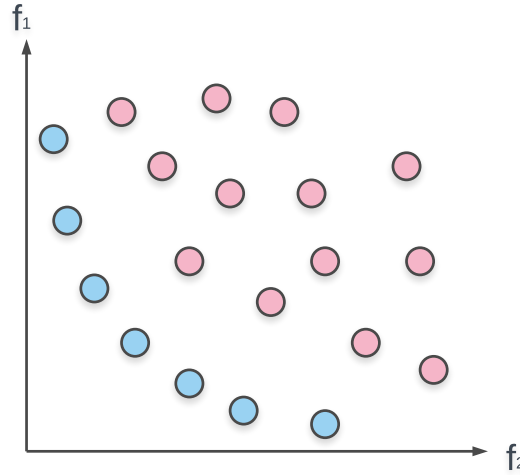


Figure 1.2.4: Example of a Pareto front in a bidimensional minimisation MOP. Blue points represent the Pareto front, while the pink ones are the dominated solutions.

An essential evolutionary algorithm for solving MOPs is the nondominated sorting GA (NSGA-II) [17], which is able to preserve the good solutions during the generations to guarantee the elitism. NSGA-II is one of the most extended algorithms in this field due to the application of two procedures.

On the one hand, NSGA-II sorts the population of solutions in different levels of fronts based on the dominance concept previously defined. The first front is formed by all nondominated solutions, while those solutions only dominated by solutions in the first front are in the second front. This procedure is extended to all solutions, and it is called fast nondominated sorting.

On the other hand, the crowding-distance is applied to sort the solutions inside each front. This procedure consists of sorting the population based on ascending order of magnitude of all objective functions. For each solution, the crowding distance is calculated, that is the sum of distances between its adjacent solutions in every objective function, taking into account that the best and the worst solution in each objective have a value equal to infinite. The solutions are ordered in descending order according to this distance.



### 1.3. Artificial neural networks

An ANN [7] is a modelling technique which simulates the biological nervous systems. Due to its powerful properties and characteristics, ANNs are commonly used in several complex real-world problems. Feed-forward neural networks (FNNs), with a hidden layer, are the most basic type of ANN, but also the most generally used. A representation of a FNN is shown in Figure 1.3.1.

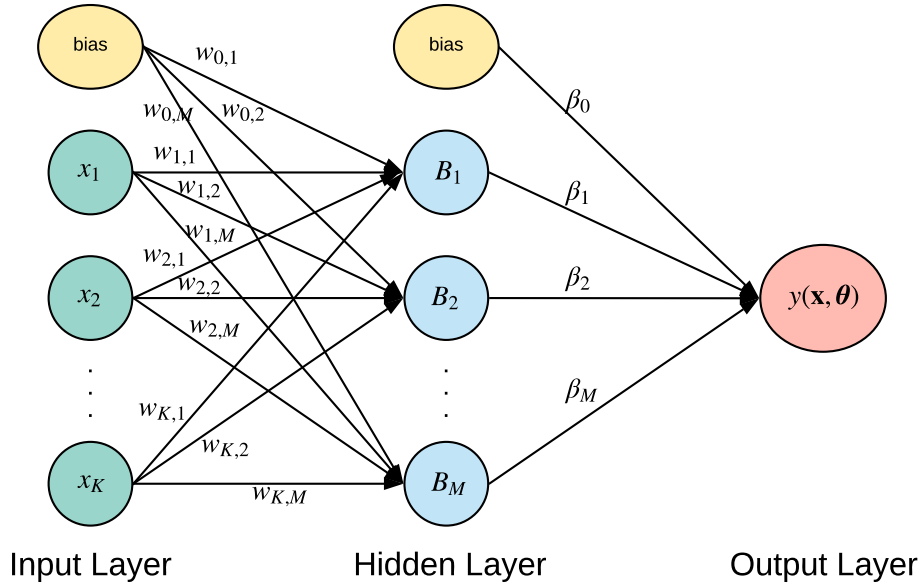


Figure 1.3.1: Example of a feed-forward neural network.

More formally, FNNs are a generalisation of a standard regression model with the following expression:

$$y(\mathbf{x}, \theta) = \beta_0 + \sum_{j=1}^M \beta_j B_j(\mathbf{x}, \mathbf{w}_j), \quad (1.9)$$

where  $M$  is the number of neurons in the hidden layer,  $B_j(\mathbf{x}, \mathbf{w}_j)$  is the basis function used for the  $j$ -th hidden neuron,  $\mathbf{x}$  represents the input independent variables,  $\mathbf{w}_j$  connects the input layer with the  $j$ -th neuron, while  $\beta_j$  represents the connection between neuron  $j$  of hidden layer and the output layer, and  $y(\mathbf{x}, \theta)$  is the function to optimise.  $B_j(\mathbf{x}, \mathbf{w}_j)$  is formed by an activation function which calculates the base value or total input arriving at the node and a transfer function which is the output of the neuron activation. The learning procedure in ANNs consists in estimating the values of  $\theta$  and the architecture of the neural network, that is, the number of nodes in hidden layer  $M$  and the number of connections between nodes (the network in Figure 1.3.1 is completely connected). Assuming an invariable predefined architecture, the learning is commonly performed by a

gradient descent algorithm, such as the backpropagation algorithm (BP) [74].

Generally, we can consider two types of activation functions for neurons:

- The *additive model* is the most common, and its output function is:

$$B_j(\mathbf{x}, \mathbf{x}_j) = h(w_{0,j} + w_{1,j}x_1 + w_{2,j}x_2 + \cdots + w_{K,j}x_K), \quad (1.10)$$

being  $\mathbf{w}_j = \{w_{0,j}, w_{1,j}, \dots, w_{K,j}\}$  the input weights for the connections to the node  $j$ ,  $h(\cdot)$  the transfer function, and  $w_{0,j}$  the node bias. There are several kinds of additive nodes: for instance, the perceptron [59] uses a step function, the sigmoidal units (SUs) can be based on logistic sigmoid, hyperbolic tangent or arctangent functions, and finally, the identity function is employed for linear nodes.

- The *multiplicative model* is a more recent strategy used for those cases in which there is a strong interaction between the input variables, and decision regions are not separable in hyperplanes [80]. The most general expression corresponds with the product unit (PU):

$$B_j(\mathbf{x}, \mathbf{w}_j) = x_1^{w_{1,j}} \cdot x_2^{w_{2,j}} \cdot \dots \cdot x_K^{w_{K,j}}, \quad (1.11)$$

where the bias  $w_{0,j}$  term does not make sense. PUs generalise other kinds of multiplicative units, due that the weights are real numbers.

Furthermore, considering the input characteristic space of the basis functions, we can classify them into:

- *Local or kernel functions*, such as radial basis functions (RBFs), present a higher value over a specific region of the input space. They are good at approximating isolated data but poorer in global environments and when the number of inputs is high.
- *Global or projection functions*, such as SUs or PUs, are better on the opposite case, that is, they behave better for global environments, but worse in the approximation of isolated data.

There are three well-known FNN types depending on the basis functions used in the hidden layer:

- *SU neural networks (SUNNs)* [49]: neural networks of SUs are also known as multilayer perceptrons (MLPs) and their nodes present an additive projection model. This family of units can approximate any given function with enough precision provided

that the number of hidden neurons is appropriately selected. An SU is defined as:

$$B_j(\mathbf{x}, \mathbf{w}_j) = \frac{1}{1 + e^{-(w_{0,j} + w_{1,j}x_1 + w_{2,j}x_2 + \dots + w_{K,j}x_K)}} = \frac{1}{1 + e^{-(w_{0,j} + \sum_{i=1}^K w_{i,j}x_i)}}. \quad (1.12)$$

- *PU neural networks (PUNNs)* [18]: they were introduced in an effort to learn appropriate high-order statistics for a given task. PUs enable a neural network to form high order combinations of inputs, with the advantages of increased information capacity and smaller network architectures when these interactions are present in the input variables. Also, they are considered universal approximators and they have been used in classification [55] and regression problems [54]. PUNNs use PU nodes in the hidden layer, following a multiplicative projection model with the output function:

$$B_j(\mathbf{x}, \mathbf{w}_j) = x_1^{w_{1,j}} \cdot x_2^{w_{2,j}} \cdot \dots \cdot x_K^{w_{K,j}} = \prod_{i=1}^K x_i^{w_{i,j}}. \quad (1.13)$$

- *RBF neural networks (RBFNNs)*: they consider kernel transfer functions, i.e. RBFNNs are those presenting RBFs in the hidden layer [7]. Each RBF makes an independent local approximation of the input space, normally using a Gaussian function. Then, the linear output layer combines the effect of the hidden nodes. The idea is that each node is placed in a region of the input space (centre of the region) with a specific radius, and the learning is made by moving the nodes through this space, varying the centre and this radius, in order to adjust the training data.

The activation function is similar to the Euclidean distance between the input pattern  $\mathbf{x}$  and the centre of the RBF ( $\mathbf{w}_j$ ), while the transfer function is generally the Gaussian function. In this way, the RBF function is:

$$B_j(\mathbf{x}, \mathbf{w}_j) = e^{-\frac{1}{2} \left( \frac{d(\mathbf{x}, \mathbf{w}_j)}{r_j} \right)^2}, \quad (1.14)$$

where  $d(\mathbf{x}, \mathbf{w}_j)$  is the Euclidean distance defined as:

$$d(\mathbf{x}, \mathbf{w}_j) = \|\mathbf{x} - \mathbf{w}_j\| = \sqrt{\sum_{i=1}^K (x_i - w_{i,j})^2}. \quad (1.15)$$

### 1.3.1. Evolutionary artificial neural networks

As we stated before, the most common algorithm used for training ANNs is BP, which only optimises the values of the connections. However, there are other ways to train ANNs that also take the architecture into account. In this work, we make use of evo-

lutionary programming (EPs) which are similar to the GAs previously defined. The main difference with respect to GAs is that EPs do not use crossover operators. The considered evolutionary ANN (EANN) was proposed in [33] under the name of neural network evolutionary programming (NNEP) algorithm. The general structure of NNEP algorithm is:

**EANN:**

**Input:** Database.

**Output:** ANN model.

- 1: Population initialisation.
- 2: **while** stopping criterion is not fulfilled **do**
- 3:   Calculate the fitness value for every individual (neural network).
- 4:   Rank the individuals according to their fitness.
- 5:   The best individual survive for the next generation.
- 6:   The best 10 % of individuals are replicated and substitute the worst 10 % of individuals.
- 7:   The best 10 % of individuals are parametrically mutated.
- 8:   The rest of the individuals (90 %) are structurally mutated.
- 9: **end while**
- 10: **return** The best individual at the end of the evolution.

The individuals represent ANN models. As can be seen, there are two mutation operators, but the crossover does not exist. The first one is called parametric mutator which can optimise the values of the network weights. If an ANN model is selected to be parametrically mutated, a Gaussian noise is added to its connections, that is, each weight is modified adding a value of a normal distribution  $N(0, \alpha \cdot T)$ , where  $\alpha$  is a dynamic parameter differently specified for every kind of connection:  $\alpha_1$  for the connections between the input layer and the hidden layer, and  $\alpha_2$  for the connections between the hidden layer and output layer.  $T = 1 - f(\mathbf{x})$ , where  $f(\mathbf{x})$  is the fitness function for network  $\mathbf{x}$ . The change is always accepted if it results in an improvement of the model. However, when the change causes a decrease of the fitness value, it is only allowed under a probability which is given by a simulated annealing procedure. On the other hand, the structural mutator is a more complicated procedure which implies a modification of the structure of the ANN model. When an ANN model is selected to be structurally mutated, a mutation is selected from the following five ones: node addition (add one or more neurons to the ANN model), node deletion (delete existing neurons of the ANN), connection addition (insert one or more connections between input and hidden layer, or between hidden layer and output layer), connection deletion, and node fusion. The node fusion mutation is performed over two neurons taking into account the following rules:

1. If a connection is present in both neurons, it will be present in the node resulting

from the fusion with a value equal to the average of the two original values.

2. If a connection is present in only one neuron, it will have the same value in the resulting neuron with a 0.5 probability.
3. If a connection is not present in any neuron, it will not be considered for the resulting neuron.

## 1.4. Time series

As stated in previous sections, a time series could be defined as temporal data which is collected chronologically, or simplifying, a function that varies across time. Time series data mining (TSDM) consists of several tasks, such as indexing of contents [23], anomaly detection [90], classification [91], analysis and preprocessing [29], segmentation [42], clustering [48], and prediction [89], among others.

In this work, we focus on time series analysis and preprocessing, which is used as a preceding step for multiple tasks, on time series segmentation (where clustering algorithms are also used) and, finally, on the time series prediction. The following subsection summarises the state-of-the-art of the main TSDM tasks considered in this Thesis.

### 1.4.1. Traditional prediction models

In general, given a time series  $Y = \{y_n\}_{n=1}^N$ , the prediction consists in the estimation of the value  $y_{N+l}$ . Traditional prediction models depend of the concept of stationarity. A time series  $Y$  is stationary if the distribution of  $\{y_1, y_2, \dots, y_k\}$  is the same as the distribution of  $\{y_{1+h}, y_{2+h}, \dots, y_{k+h}\}$ , that is, there are no systematic changes in the mean and variance. Furthermore, a time series is called weakly stationary if it satisfies two conditions:  $E[y_n] = \mu$  and  $Corr(y_n, y_{n+h}) = \sigma(h)$ . A white noise time series ( $U = \{u_n\}_{n=1}^N$ ), which is a set of identically distributed and independent random variables with common zero mean and constant variance,  $\sigma^2$ , is a simple example of stationary time series.

From these definitions, we present the following time series models: moving average (MA) model, autoregressive (AR) model, mixed autoregressive moving average (ARMA) model and autoregressive integrated moving average (ARIMA) model.

### Moving Average models

Given a  $U$  white noise time series with mean zero and  $\sigma^2$  variance,  $Y$  is an  $MA(q)$  (MA process of order  $q$ ), if:

$$y_n = \mu + \alpha_0 u_n + \alpha_1 u_{n-1} + \alpha_2 u_{n-2} + \cdots + \alpha_q u_{n-q}, \quad (1.16)$$

where  $\mu$  is the mean of the time series,  $\alpha = (\alpha_0, \alpha_1, \dots, \alpha_q)$  are the model parameters, and the  $u$ 's are scaled in order to have  $\alpha_0 = 1$ .

It is shown that  $E[y_n] = \mu$ , and the autocorrelation function does not depend on the time  $n$ , so the MA process is weakly stationary. Also, if  $U$  is a white noise, the process will be stationary.  $MA(\infty)$  is a special case:

$$y_n = \mu + \alpha_0 u_n + \alpha_1 u_{n-1} + \alpha_2 u_{n-2} + \dots \quad (1.17)$$

In order to simplify the formulation of these processes, the backshift operator ( $B$ ), which is defined by  $B^k y_n = y_{n-k}$ , provides another way to represent  $MA(q)$  models. In this way, Equations 1.16 and 1.17 can be reformulated as:

$$y_n = \mu + \alpha(B)u_n, \quad (1.18)$$

$$y_n = \mu + \theta(B)u_n, \quad (1.19)$$

respectively.  $\alpha(B) = I + \alpha_1 B + \cdots + \alpha_q B^q$  and  $\theta(B) = I + \theta_1 B + \theta_2 B^2 + \dots$

### Autoregressive models

Given a  $U$  white noise time series with zero mean and  $\sigma^2$  variance,  $Y$  is an  $AR(p)$  (AR process of order  $p$ ), if:

$$y_n = \delta + \beta_1 y_{n-1} + \beta_2 y_{n-2} + \cdots + \beta_p y_{n-p} + u_n, \quad (1.20)$$

where  $\delta$  is a constant, and  $\beta = (\beta_0, \beta_1, \dots, \beta_p)$  are the model parameters. Using the backshift operator, this expression can be reformulated as:

$$\beta(B)y_n = \delta + u_n, \quad (1.21)$$

where  $\beta(B) = I - \beta_1 B - \cdots - \beta_p B^p$ .

### Autoregressive Moving Average models

When a time series  $Y$  is formed by a combination of an  $AR(p)$  term and an  $MA(q)$  term, it is called ARMA process of order  $(p, q)$ , and it is given by:

$$y_n = \delta + \beta_1 y_{n-1} + \beta_2 y_{n-2} + \cdots + \beta_p y_{n-p} + u_n + \alpha_1 u_{n-1} + \alpha_2 u_{n-2} + \cdots + \alpha_q u_{n-q}, \quad (1.22)$$

which can be easily rewritten using the backshift operator  $B$  as:

$$\beta(B)y_n = \delta + \alpha(B)u_n. \quad (1.23)$$

### Autoregressive Integrated Moving Average models

Despite the big amount of time series that can be modelled with ARMA processes, there are a lot of non-stationary time series. This kind of time series is usually transformed into stationary time series by differencing the values. Taking into account the  $d$ -th difference, denoted by  $\nabla^d y_n = y_n - y_{n-d}$ , a  $Y$  time series is modelled with an  $ARIMA(p, d, q)$  model if:

$$\nabla^d y_n = \delta + \beta_1 y_{n-1} + \beta_2 y_{n-2} + \cdots + \beta_p y_{n-p} + u_n + \alpha_1 u_{n-1} + \alpha_2 u_{n-2} + \cdots + \alpha_q u_{n-q}, \quad (1.24)$$

which can be rewritten in the same way that Equation 1.23:

$$\beta(B)\nabla^d y_n = \delta + \alpha(B)u_n. \quad (1.25)$$

### Estimation and forecasting for ARMA models

Because all the models previously defined can be represented as an ARMA model (AR and MA models are incomplete ARMA models, and ARIMA models can be transformed into an ARMA model using differentiation), it is enough to describe the parameter estimation and the forecasting processes for ARMA models.

From the notation of the ARMA model (see Equation 1.22), there are  $p + q + 1$  parameters to be estimated. The most common method to estimate these parameters is the maximum likelihood estimation (MLE), which is based on the likelihood function  $L$ . It

can be defined by:

$$L(\beta, \alpha, \delta, \sigma^2|Y) = f(y_1, y_2, \dots, y_N; \beta, \alpha, \delta, \sigma^2) \quad (1.26)$$

$$= f(u_1, u_2, \dots, u_N; \beta, \alpha, \delta, \sigma^2) \quad (1.27)$$

$$= 2\pi^{-\frac{N}{2}} \sigma^{-N} \exp \left\{ -\frac{1}{2\sigma^2} \sum_{n=1}^N u_n^2(\beta, \alpha, \delta, \sigma^2) \right\}. \quad (1.28)$$

From it, we can derive the log-likelihood function:

$$Ln(L)(\beta, \alpha, \delta, \sigma^2) = - \left( \frac{N}{2} \ln(2\pi) + \frac{N}{2} \ln(\sigma^2) + \frac{1}{2\sigma^2} \sum_{n=1}^N u_n^2(\beta, \alpha, \delta, \sigma^2) \right), \quad (1.29)$$

whose maximisation is performed by iterative numerical procedures.

Once all the parameters are obtained for a set of  $N$  observations of  $Y$ , the forecasting can be immediately performed. To predict  $\hat{y}_N(l) = y_{N+l}$  which is a conditional expectation  $E(y_{N+l}|y_N, y_{N-1}, \dots, y_{N-p})$ , we have to consider that  $\hat{y}_N(l) = y_{N+l}$  and  $\hat{u}_n(l) = u_{n+l}$  if  $l \leq 0$ ; and  $\hat{u}_n(l) = 0$  if  $l > 0$ .

Taking into account these considerations the prediction could be made using the difference equation:

$$\hat{y}_N(l) = (1 + \beta_1)\hat{y}_N(l-1) - \beta_2\hat{y}_N(l-2) + \hat{u}_N(l) + \alpha_1\hat{u}_N(l-1) + \alpha_2\hat{u}_N(l-2), \quad (1.30)$$

where the computation of  $u_n$  is based on  $u_n = y_n - \hat{y}_{n-1}(1)$  (given  $y_0(1) = \mu$ ).

#### 1.4.2. Segmentation

The main contributions of this Thesis are in the field of time series segmentation. Time series segmentation consists in dividing the time series into a set of consecutive segments, in order to satisfy some objectives. Formally, given a time series  $Y = \{y_n\}_{n=1}^N$ , the procedure is to divide the values of  $y_n$  into  $m$  consecutive segments. For that, the time indexes, which are represented by  $n = 1, \dots, N$  are separated into subsequences, that is,  $s_1 = \{y_1, \dots, y_{t_1}\}$ ,  $s_2 = \{y_{t_1}, \dots, y_{t_2}\}$ ,  $\dots$ ,  $s_m = \{y_{t_{m-1}}, \dots, y_N\}$ , where the cut points (denoted by  $t_1 < t_2 < \dots < t_{m-1}$ ) are arranged in ascending order. Normally, each cut point belongs to the previous and the next segment, which allows analysing consistently the transition from one segment to the next. The Figure 1.4.1 graphically represents a example of segmentation.

There exist two main objectives which time series segmentation tries to satisfy. On the one hand, this procedure is applied to discover similarities between segments. For that, the methodology tries to group the segments into  $k$  clusters. Each segment will be associa-



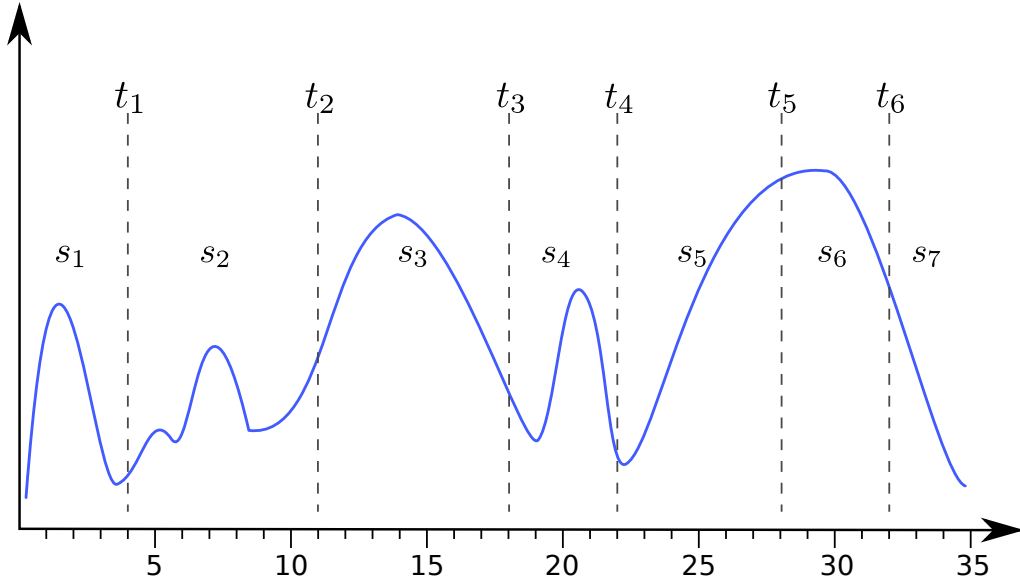


Figure 1.4.1: Example of time series segmentation with a time series of length  $N = 35$  and 6 cut points ( $t_1 = 4, t_2 = 11, t_3 = 18, t_4 = 22, t_5 = 28, t_6 = 32$ ), represented by dashed lines. The resulting segments are:  $s_1 = \{y_1, y_2, y_3, y_4\}$ ,  $s_2 = \{y_4, y_5, y_6, y_7, y_8, y_9, y_{10}, y_{11}\}$ ,  $\dots$ ,  $s_7 = \{y_{32}, y_{33}, y_{34}, y_{35}\}$ .

ted to a class label, with  $k$  possible values,  $\{C_1, \dots, C_k\}$ . Initially, in [2], authors affirmed that all points of a time series which belongs to the same cluster come from contiguous time instants. Then, methods to group segments instead simply points are proposed by many authors. In this way, Tseng et al. [86] created an algorithm where the segmentation is performed regarding the similarities between segments in a wavelet space. Segments of different length are compared and grouped by using a wavelet representation. Also, a meaningful clustering of subsequences from time series was tackled by using two efficient methods in [70]. All these methods are based on the idea of clustering the segments, so the use of clustering methods is required. In the last years, several algorithms have been proposed for time series clustering [3], with the aim of obtaining groups of time series with similar features. The most homogeneous clusters should be found while being as different as possible from the other clusters. In other words, the clustering process should look for the maximisation of the intercluster variance and the minimisation of the intracluster variance [48]. In fact, time series clustering is a very important task within TSDM, although, in this work, we only consider clustering for segmentation of a single time series.

On the other hand, time series segmentation is also applied with the objective of simplifying the time series, that is, replacing segments by simple model descriptions. This group of algorithms aims to reduce the number of points of the time series with minimum information loss, with the goal of alleviating the difficulty of processing and memory requirements. In this context, the authors in [63, 64] proposed the replacement of some of

the segments located by suitable approximations using bayesian approaches to financial time series data. Other well-known approaches follow the piecewise linear approximation (PLA), where linear regressions or interpolations are used for modelling each segment. The main contributions in this context have been made by Keogh [42]. The Top-Down procedure starts with one segment and recursively divides this segment into subsequences, based on the approximation error. On the contrary case, Bottom-Up starts considering each point as a segment and iteratively merges the two segments resulting in the minimum error of approximation. Sliding Window algorithm considers a fixed-length window, which is moved across the time series, and, when a new point is included in the segment and increases the cost over a threshold, the algorithm considers it as a cut point. The last algorithm, called SWAB, is a combination of the latter two, where a Bottom-up is run inside each Sliding Window. More recently, Fu [25] published a review where segmentation is presented as an optimisation problem, suitable to be solved by different techniques, such as EAs. In this context, new techniques related to the approximation of time series were proposed in [92], where authors developed a novel approach based on segmenting time series with connected lines under a predefined maximum error bound.

It is important to mention that both objectives are conflicting, that is, segmentations with low high approximation quality are usually related to a high number of segments, which are more difficult to be grouped. In this Thesis, we will also approach this multiobjective problem.

## 1.5. Extreme value theory

The last contributions in this Thesis are related to the determination of the distribution of a given time series, to use it for other posterior tasks, such as prediction or classification. This is motivated by a specific problem treated in this work, the determination of the distribution of values in Wave height time series (see section 1.6.3). This distribution is related to the EVT, being interesting to introduce this theory. EVT is associated to the maximum sample  $M_n = \max(X_1, \dots, X_n)$ , where  $(X_1, \dots, X_n)$  is a set of independent random variables with common distribution function  $F$ . In this way,  $Pr(M_n < x) = F^n(x)$  is the distribution of the maximum observations. The assumption of independence when  $X$  represent the wave height over a threshold is a hypothesis very acceptable because, for oceanographic data, a peak-over-threshold (POT) scheme is commonly used. POT selects extreme wave height events that are approximately independent [39]. Furthermore, MacKay and Johanning [52] stated that “The maximum wave heights in successive sea states can be considered independent, in the sense that the maximum height is dependent only on the sea state parameters and not on the maximum height in adjacent sea states”. This  $M_n$  variable is usually described with one of the three following distributions: Gumbel,

Frechet, or Weibull.

Annual maximum approach (AM) is a methodology within the EVT, where  $X$  represents the wave height collected in regular yearly periods, and  $M_n$  is formed by the maximum values per year. The statistical behaviour of AM can be described by the distribution of the maximum wave height regarding generalised extreme value (GEV) distribution [51]:

$$G(x) = \begin{cases} \exp \left\{ - \left[ 1 + \xi \left( \frac{x-\mu}{\sigma} \right) \right]^{\frac{1}{\xi}} \right\}, & \xi \neq 0, \\ \exp \left\{ - \exp \left( - \left( \frac{x-\mu}{\sigma} \right) \right) \right\}, & \xi = 0, \end{cases} \quad (1.31)$$

where  $0 < x < 1 + \xi \left( \frac{x-\mu}{\sigma} \right)$ ,  $-\infty < \mu < \infty$ ,  $\sigma > 0$  and  $-\infty < \xi < \infty$ . From these equations, it is easy to see that the model has three parameters: location ( $\mu$ ), scale ( $\sigma$ ), and shape ( $\xi$ ).

The estimation of the return values, corresponding to the return period  $T_p$ , is performed by inverting Eq. 1.31:

$$z_p = \begin{cases} \mu - \frac{\sigma}{\xi} \left[ 1 - \{-\log(1-p)\}^{-\xi} \right], & \xi \neq 0, \\ \mu - \sigma \log \{-\log(1-p)\}, & \xi = 0, \end{cases} \quad (1.32)$$

where  $G(z_p) = 1 - p$ . Then,  $z_p$  will be exceeded once per  $1/p$  years, which corresponds to  $T_p$ .

POT is another method contained in the EVT, where, instead of the maximum values per year, the statistical description is made with the values over a predefined threshold [15, 24]. Basic POT method has become a standard approach [57, 24, 9], which have been improved by several authors in recent years [68, 39].

The POT method considers that, if the AM approach uses a GEV distribution (Eq. 1.31), the peaks over a high threshold should result in the generalised Pareto distribution (GPD). The GPD fitted to the tail of the distribution gives the conditional non-exceedance probability  $P(H_{max} \leq x | H_{max} > u)$ , where  $u$  is the threshold level. The conditional distribution function can be calculated as:

$$P(X \leq x | X > u) = \begin{cases} 1 - \left( 1 + \xi^* \left( \frac{x-u}{\sigma^*} \right) \right)^{\frac{1}{\xi^*}}, & \xi^* \neq 0, \\ 1 - \exp \left( - \left( \frac{x-u}{\sigma^*} \right) \right), & \xi^* = 0. \end{cases} \quad (1.33)$$

There is consistency between the GEV and GPD models, meaning that the parameters can be related as  $\xi^* = \xi$  and  $\sigma^* = \sigma + \xi(u - \mu)$ . If  $\xi \geq 0$  the distribution is referred to as long-tailed, and the distribution is referred to as short-tailed on the contrary case.

The use of the GPD for modelling the tail of the distribution is also justified by asym-

ptotic arguments in [13]. In this paper, the author confirms that it is usually more convenient to interpret extreme value models in terms of return levels, rather than individual parameters. In order to obtain these return levels, the exceedance rates of thresholds have to be determined as  $P(X > u)$ . In this way, using Eq. 1.33 ( $P(X > x|X > u) = P(X > x)/P(X > u)$ ) and considering that  $z_N$  is exceeded on average every  $N$  observations, we have:

$$P(X > u) \left[1 + \xi^* \left(\frac{z_N - u}{\sigma^*}\right)\right]^{-\frac{1}{\xi^*}} = \frac{1}{N}. \quad (1.34)$$

Then, the  $N$ -year return level  $z_N$  is obtained as:

$$z_N = u + \frac{\sigma^*}{\xi^*} \left[(N * P(X > u))^{\xi^*} - 1\right]. \quad (1.35)$$

There are several methods for the estimation of the GEV and GPD parameters. In [13], the author describes the MLE methodology which was then used in [68] for the estimation of the parameters. However, it has an important drawback for two parameter distributions (for instance Weibull or Gamma): these distributions are very sensitive to the distance between the high threshold ( $u_2$ ) and the first peak [58]. For this reason, MLE could be used with two-parameter distribution when  $u_2$  reaches a peak. As this peak is excluded, the first value of the exceedance is as far from  $u_2$  as possible. A solution would be to use the three-parameter Weibull and Gamma distributions. However, MLE estimation of such distributions is complicated, and the algorithms usually fit two-parameter distributions inside a discrete range of location parameters [67].

## 1.6. Applications in real-world problems

Several real-world problems are considered in this Thesis to validate the methodologies proposed and solve some significant difficulties which have been found for these applications. Those which have more importance are the detection of TPs in paleoclimate time series, the analysis of stock indexes in financial problems, the detection, prediction and recovery of missing values in oceanographic data and, finally, the prediction of fog in the airport of Valladolid (Spain).

### 1.6.1. Tipping points

Palaeoclimatology studies the climate characteristics of the earth during its history, and it is a part of Palaeogeography science. Specifically, important climate variations and their causes are studied, at the same time that this science tries to give a description as accurate as possible of the characteristics of the climate for a specific moment in the his-

tory of the earth. The geological scale variation of the factors which determine the current climate, such as the energy of solar radiation, the astronomical and cosmic situations, the distribution of continents and oceans and the composition and dynamics of the atmosphere, constitute the most used factors in the deduction and explanation of paleoclimates.

Recently, some researches about dynamical systems, i.e. climate systems, affirm that they present critical transition points, called TPs. More formally, a climate TP consists of a small change which produces a strongly nonlinear response in the internal dynamics of the climate system, which changes its future state. In this way, the great climate changes in the history of the earth are characterised by turning points in the time series that represents the temperature (or one of its proxies, such as the concentration of the oxygen isotope in glaciers). These inflexion points cause the time series to pass from one stable state to another. An example of a climate time series with 18 TPs is shown in Figure 1.6.1, referred to as the Greenland ice sheet project two (GISP2)  $\delta^{18}\text{O}$  ice core data [5].

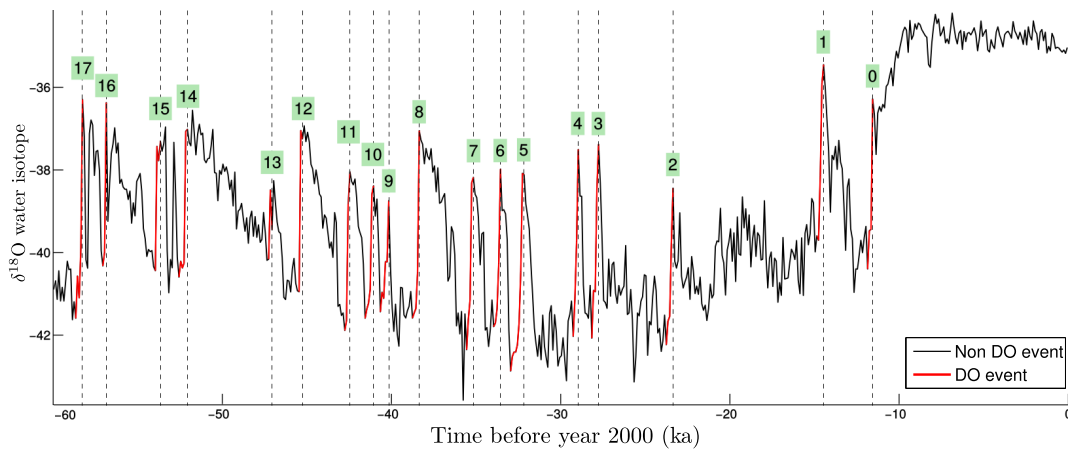


Figure 1.6.1: GISP2 time series.

The study of the causes of a TP and the characterisation of what happens before them is one of the most attractive areas in Paleoclimatology. An active scientific community is working on finding early warning signals of TP because climate TPs can affect millions of lives. Lenton [46] proposed a differentiation of many types of TPs and presented some indicators that can help to detect them, such as the increase of autocorrelation of the series values. Also, Dakos et al. [14] presented more particular techniques regarding data processing and indicators. They studied a set of methods using simulated data, concluding that there does not exist a unique best indicator for detecting a transition, and all the methods require a specific data-treatment.

Up to our knowledge, all previous works tackle the TP detection with statistical methods trying to select (by trial and error) the method more suitable to detect those transitions. They require an intensive data preprocessing that include, for instance, the use of Gaussian filters or rolling windows that introduces extra parameters (such as the width

of the Gaussian function or size of the window) which need to be optimised [46, 14]. The main limitation behind these methods is that different TPs and different statistical descriptors require different and specific treatments. For these reasons, in this Thesis, we solve this problem using new time series segmentation algorithms with a higher abstraction level.

### 1.6.2. Stock indexes

Regarding financial time series, the segmentation of stock market time series can be used for trend analysis. Recent studies have determined the movement of a stock based on a selection of some points of a given time series [69]. Gonzalez et al. [28] affirmed that the relevance of identifying phases in the stock market evolution consists in the fact that few economic phenomena attract more attention than the bullish and bearish markets (cycles) do. Bull markets are associated with persistently rising share prices, strong investor interest and expectation, and enhanced financial well-being.

Concerning the theoretical bases of the analysis of stock prices by experts found in the economic literature, there are three major theories for answering the questions of what and when to buy or sell. The first school believes that no investor can achieve above-average trading advantages based on historical and present information (the random walk hypothesis and the efficient market hypothesis). The second view is fundamental analysis in which analysts have undertaken in-depth studies into the various macroeconomic factors and have looked into the financial conditions and results of the industry concerned to discover the extent of correlation that might exist with the changes in the stock prices. The technical analysis presents the third view on market price prediction. They claim that there are recurring patterns in the market behaviour that can be identified and predicted. In such a process, they use some statistical parameters called technical indicators and charting patterns from historical data. Technical analysis is the science of recording, usually in graphic form, the actual history of trading (price changes, the volume of transactions, etc.) in a specific stock and then deducing from that pictured history the probable future trend [20]. Consequently, technical indicators are numerical values calculated by past prices, volumes, and other market statistics and are used to forecast future price movements. Recently, a fourth approach, known as cyclical, has made rapid progress and promises to contribute a great deal to our understanding of economic trends.

In this Thesis, we move between the cyclical and the technical analysis approaches, as our analysis is based on charts and figures (chartist analysis and financial patterns), but we search for characteristic phases or cycles in a long time series and identify the main financial patterns in each phase to analyse the behaviour of the European stock markets (see Figure 1.6.2 which shows the IBEX35 time series). Trend analysis studies also include

the well-known Elliott wave principle, Dow theory and related vocabulary, as primary or secondary trends. We investigate the characteristics of the resultant segments from a segmentation algorithm after a clustering phase following the analysis of the behaviour of *bearish*, *bullish* and *sluggish* markets [66], and *cycles*, *booms*, and *crashes* [47].

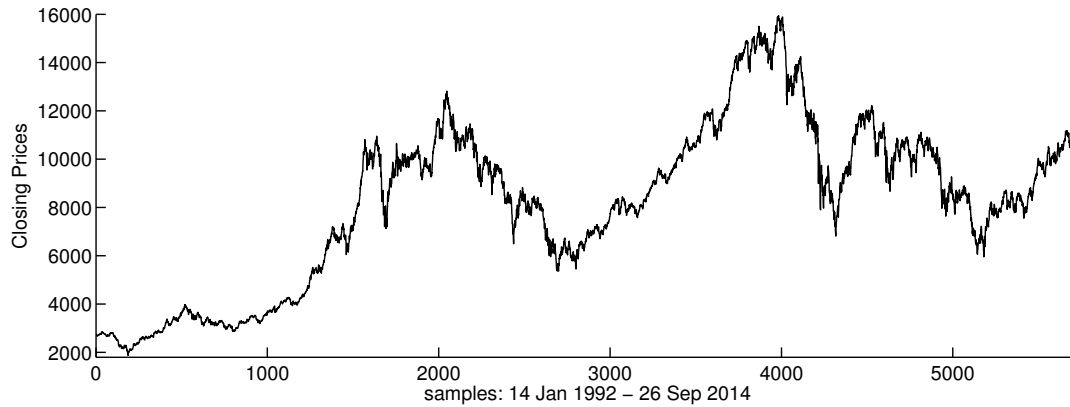


Figure 1.6.2: IBEX35 time series.

### 1.6.3. Wave height time series

Extreme ocean waves can cause significant risks in offshore structures. The development of these structures for oil extraction requires knowledge about the waves and their changes. Also, wave height time series are present in many real-world applications, such as the determination of energy power or for civil protection.

The oceanographic data that we use here is the significant wave height (SWH), and there are different statistical and mathematical methods to calculate it. We use the generic term  $H_s$  or simply SWH defined as the average trough to crest height in meters of the highest one-third of all the wave heights during a 20-minute sampling period [37]. This definition is given by the National Data Buoy Center (NDBC) and the National Oceanic and Atmospheric Administration (NOAA), which uses ocean buoys with special sensors to collect data. Figure 1.6.3 represents a time series of SWH collected in the Gulf of Alaska with identification number 46001.

One of the problems that has been tackled in this Thesis is the reconstruction of this kind of data. It is usually associated to a number of unexpected events, such as storms, which can make buoys break down [71], resulting in data gaps and, therefore, discontinuities in the buoys data time series, lasting from the causing event until the buoy is repaired/maintained. Some data analysis methods may allow data with gaps [50], but most statistical methods require all data recorded [85]. Due to this, the reconstruction of missing wave values has become an interesting topic in marine research.

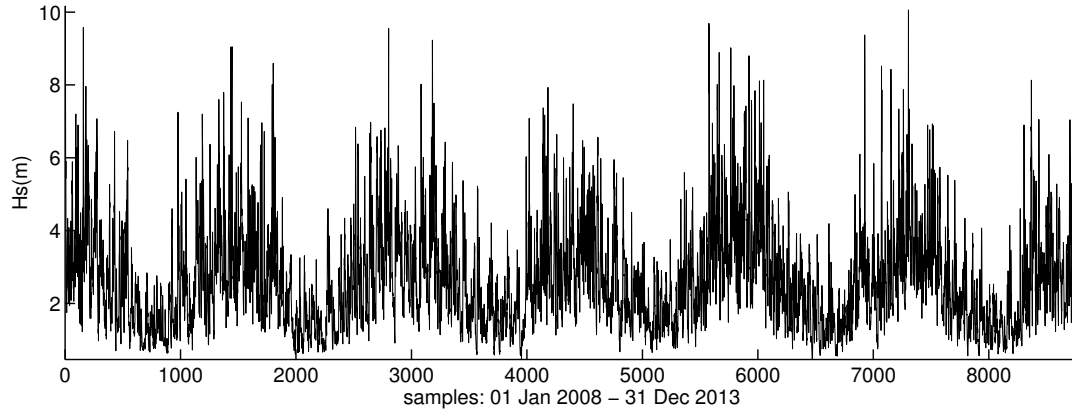


Figure 1.6.3: SWH time series collected from buoy 46001 in the Gulf of Alaska.

In the literature, there are different techniques to tackle this reconstruction. Firstly, authors used simple methods by generating random data [84] or used Monte Carlo methods to recover data in monthly mean sea level time series [82]. Other approaches are related with the use of ANN models, which is the case of the works of Haykin [30] and more recently, by including ANN in combination with the rough set theory [81]. Evolutionary computation is also applied in this context in the works of [61] and [4].

The remaining problem associated with this application is the detection and prediction of very large SWHs. For that, we consider extreme SWH as those with a substantial higher height in relation to other waves close in time. The goal of the detection is to determine intervals or segments in the time series which contains these waves and differentiate them from the rest. This step is completed with the prediction phase, that uses the segments to estimate a predictive model able to determine if a very large SWH event will be produced (or not) after a given subsequence of a time series. This methodology differs from the state-of-the-art methodologies which consider the prediction using the statistical distribution obtained with the EVT defined in Section 1.5. For example, Davison [15] introduced the POT methods that have been used then by other authors [9, 88].

Related with the last point, we propose a new way to determine the threshold in the POT approaches, in SWH time series.

#### 1.6.4. Fog prediction

The last real-world problem that has been considered in this Thesis is the prediction of fog presence for the airport of Valladolid. Aviation is one of the transport systems most affected by weather conditions. Considering airport operations, many factors reduce the visibility, and the most frequent one is fog formation. Fog is a collection of crystals suspended in the air, produced by low-level cloud over the ground. Foggy conditions have a



significant impact on the operation of taxiing, take-off and landing, and they have produced several accidents. Nowadays, the airport operation decisions are made using data collected from airports, but the runway visual range (RVR) is the most significant one. The RVR is a meteorological variable defined as the range over which the pilot of an aircraft on the central line of a runway can see the runway surface markings or the lights delineating the runway or identifying its centre line [1]. If RVR is low enough, the airport activates the low visibility procedures. In this way, it is interesting to predict this variable.

For this reason, to ensure safety conditions, all the employers need the most precise and reliable meteorological information to tackle the problems that involve low visibility conditions. So the continuous improvement of the prediction is a challenging problem. The most common approaches are associated with the use of numerical weather prediction (NWP) models, which have been considered by several authors. For instance, recent works evaluate the performance of high-resolution NWP models to forecast fog events [73]. Also, other research lines rely on the probabilistic forecasting given the uncertainty in weather [93]. The intrinsic limitations of NWP models paved the way for a new line of work focused on the post-processing of outputs from dynamical models [32], the application of the model output statistics [26] being, perhaps, the most widely used post-processing technique.

The main problem of these accurate NWP models is their computational cost, and that not all the meteorological service providers have the resources needed to access NWP models. Thus, other models can reach the same accuracy with economic solutions. In this sense, statistical approaches have been applied in fog formation prediction. Linear regression techniques have been used in [45]. Then, ANN models have acquired more importance, what can be seen in the works of Fabian et al. [22], which used an MLP, and Marzban et al. [56], which forecasted visibility at 39 airports with an MLP trained with information from different sources. Also, Dutta and Chaudhuri [19] obtained good results by using an MLP with back-propagation learning technique to forecast 3-hourly visibility during winter time at Kolkata airport (India).

Finally, some researchers have used new artificial intelligence techniques. For example, Miao et al. [60] developed a fuzzy logic fog forecasting model for the cold season at Perth Airport, and Boneh et al. [8] proposed a Bayesian decision network based on data from the previous 34 years at Melbourne airport.

In this Thesis, we tackled this problem with a multiobjective evolutionary algorithm (MOEA) for training ANNs to improve the prediction of fog events without losing accuracy in the prediction of non-fog events.



*When you do what you fear most, then you can do anything.*

Stephen Richards

# 2

## Motivation and objectives

This chapter introduces the motivation and objectives of this work, and the main publications derived from the Thesis.

### 2.1. Motivation and challenges

From the previous chapter, four main concerns can be taken into account for TSDM. Firstly, the need to develop new preprocessing and analysis techniques to be used in order to alleviate the subsequent tasks. Secondly, the improvement of the state-of-the-art methods for time series segmentation when optimising both main objectives (clustering quality and approximation quality). Thirdly, the creation of time series prediction models which take into account not only the values of the time series but also higher levels of representation, such as the segments obtained in the segmentation. And finally, the use of statistical distributions to guide different time series methodologies.

Considering the ahead comments, we can synthesise the following open challenges:

- **Preprocessing techniques:** to improve the performance of subsequent tasks, we should generate methods for extracting the core information from time series. Given that time series are extracted from different sources of continuous data, they can lead to incorrect values or incomplete information. For example, when we focus on SWH

time series, the information is collected from buoys which are situated in different parts of the sea and oceans. As stated before, these buoys can be broken resulting in gaps across the time series. Consequently, we consider that an adequate method is needed to recover this kind of information. State-of-the-art approaches include ANNs models, but, up to author knowledge, there is no work which includes an EA to train ANNs of PUs. The resulting PUNN model could improve the estimation of the reconstructed values.

- Time series segmentation: as commented above, the main contributions of this Thesis will be made in the field of time series segmentation. Time series segmentation aims to find a set of ordered points with the objective of simplifying the time series or discovering useful patterns. State-of-the-art algorithms are mainly focused on the first objective using standard techniques. We consider that the application of different ML algorithms in combination with different MHs could improve the performance of the previous ones. In this way, a challenge is the improvement in terms of accuracy without resulting in too high computational cost. Regarding the second objective, the discovery of useful patterns, few works can be found in the literature, where the paper of Tseng et al. [86] is one of the most important. In this work, the authors proposed a GA where the representation of the chromosome is based on a wavelet transformation. We think that the representation could be improved using other information of the time series and, also, that the application and development of new MHs will lead to better patterns. Finally, a priori, both objectives (the optimisation of the clustering quality of the segments and the approximation of them) are in conflict, that is, the optimisation of one lead to a decrease of the quality of the other. This problem has not been previously solved in the literature, and we consider that the optimisation of both objectives in a multiobjective algorithm could result in an interesting challenge.
- Prediction: the prediction of future values of a time series implies the study of past values and, usually, in the literature, authors aim to predict the next value. We consider that transforming the prediction model into a classification model could be attractive, to determine not only the next value but also the next event in the time series. For that, a higher level of representation is needed. For example, the set of segments resulting in the segmentation procedure can be taken to construct a database (which maintains the temporal dimension) for predicting the next segment given a set of subsequences. In this way, the application and adaptation of different ML algorithms for classification in combination with MHs for the optimisation of the parameters will result in a better performance of traditional algorithms.
- Distribution-based learning: the POT approaches in the EVT need a threshold that is

usually fixed by trial and error procedures. Then, the distribution of the points over this threshold is fitted. However, in time series which contains extreme values (those which differs from a large set of points close to the average value), the analysis could be made taking into account the raw and complete time series. For that, we suppose that the determination of the statistical distribution of the whole time series would be able to determine the threshold used for this theoretical distribution. Also, using the statistical distribution of the whole time series, we could discretise its value. Then, this discretisation will be used to perform ordinal classification of the resulting segments, using this theoretical distribution instead of the histogram of time series, because the statistical distribution will be more stable than the observed values.

- Last, it should be highlighted that we consider necessary not the only the proposal of new methods but also the application of the developed models and algorithms to real-world problems. As discussed in the previous chapter, the problems considered include paleoclimate data, financial problems, SWH time series and fog formation.

## 2.2. Objectives

The present Thesis addresses the discussed open challenges in the previous section. All of them result in the following formal objectives:

1. To study and develop preprocessing and analysis techniques for time series with the aim of alleviating the difficulty of posterior tasks.
2. To develop and improve the state-of-the-art ANN models for missing values in SWH time series using EANNs based on PUs.
3. To study, adapt and create bioinspired algorithms for time series segmentation with the goal of discovering useful patterns in time series.
4. To adapt and develop bioinspired algorithms for time series segmentation with the aim of improving the accuracy of the solution obtained by the state-of-the-art algorithms.
5. To design a new multiobjective algorithm that takes into account both objectives during the optimisation, given that they are conflicting objectives, that is, an improvement of one means a deterioration in the quality of the other.
6. To use a high level of representation of subsequences of the time series for predicting future events: types of segments (using clustering) or intervals of values (using a standard discretisation).

7. To theoretically develop a method to determine the statistical distribution of SWH time series, for fitting the threshold needed for POT approaches.
8. To apply the proposed methods to the following real-world problems:
  - a) Detection of TPs in paleoclimate data.
  - b) Analysis of trends and phases in stock market indexes (financial data).
  - c) Reconstruction of massive missing data values in SWH time series.
  - d) Detection of largest wave height in oceanographic data.
  - e) Prediction of segments containing largest wave height in oceanographic data.
  - f) Predicting fog formation in airports.
  - g) Fitting statistical distribution for establishing the threshold of the POT method when applied to oceanographic data.

### 2.3. Summary of the Thesis

Currently, the amount of data which is produced for any information system is increasing exponentially. This motivates the development of automatic techniques to process and mine these data correctly. Specifically, in this Thesis, we tackled these problems for time series data, that is, temporal data which is collected chronologically. This kind of data can be found in many fields of science, such as palaeoclimatology, hydrology, financial problems, etc.

TSDM consists of several tasks which try to achieve different objectives, such as, classification, segmentation, clustering, prediction, analysis, etc. However, in this Thesis, we focus on time series preprocessing, segmentation and prediction.

Time series preprocessing is a prerequisite for other posterior tasks: for example, the reconstruction of missing values in incomplete parts of time series can be essential for clustering them. In this Thesis, we tackled the problem of massive missing data reconstruction in SWH time series from the Gulf of Alaska. It is very common that buoys stop working for different periods, what it is usually related to malfunctioning or bad weather conditions. The relation of the time series of each buoy is analysed and exploited to reconstruct the whole missing time series. In this context, EANNs with PUs are trained, showing that the resulting models are simple and able to recover these values with high precision.

In the case of time series segmentation, the procedure consists in dividing the time series into different subsequences to achieve different purposes. This segmentation can be done trying to find useful patterns in the time series. In this Thesis, we have developed novel bioinspired algorithms in this context. For instance, for paleoclimate data, an initial

genetic algorithm was proposed to discover early warning signals of TPs, whose detection was supported by expert opinions. However, given that the expert had to individually evaluate every solution given by the algorithm, the evaluation of the results was very tedious. This led to an improvement in the body of the GA to evaluate the procedure automatically. For significant wave height time series, the objective was the detection of groups which contains extreme waves, i.e. those which are relatively large with respect other waves close in time. The main motivation is to design alert systems. This was done using an HA, where an LS process was included by using a likelihood-based segmentation, assuming that the points follow a beta distribution. Finally, the analysis of similarities in different periods of European stock markets was also tackled with the aim of evaluating the influence of different markets in Europe.

When segmenting time series with the aim of reducing the number of points, different techniques have been proposed. However, it is an open challenge given the difficulty to operate with large amounts of data in different applications. In this work, we propose a novel statistically-driven CRO algorithm (SCRO), which automatically adapts its parameters during the evolution, taking into account the statistical distribution of the population fitness. This algorithm improves the state-of-the-art with respect to accuracy and robustness. Also, this problem has been tackled using an improvement of the BBPSO algorithm, which includes a dynamical update of the cognitive and social components in the evolution, combined with mathematical tricks to obtain the fitness of the solutions, which significantly reduces the computational cost of previously proposed coral reef methods.

Also, the optimisation of both objectives (clustering quality and approximation quality), which are in conflict, could be an interesting open challenge, which will be tackled in this Thesis. For that, an MOEA for time series segmentation is developed, improving the clustering quality of the solutions and their approximation.

The prediction in time series is the estimation of future values by observing and studying the previous ones. In this context, we solve this task by applying prediction over high-order representations of the elements of the time series, i.e. the segments obtained by time series segmentation. This is applied to two challenging problems, i.e. the prediction of extreme wave height and fog prediction. On the one hand, the number of extreme values in SWH time series is less with respect to the number of standard values. In this way, the prediction of these values cannot be done using standard algorithms without taking into account the imbalanced ratio of the dataset. For that, an algorithm that automatically finds the set of segments and then applies EANNs is developed, showing the high ability of the algorithm to detect and predict these special events. On the other hand, fog prediction is affected by the same problem, that is, the number of fog events is much lower than that of non-fog events, requiring a special treatment too. A preprocessing of different data coming from sensors situated in different parts of the Valladolid airport are used for

making a simple ANN model, which is physically corroborated and discussed.

The last challenge which opens new horizons is the estimation of the statistical distribution of time series to guide different methodologies. For this, the estimation of a mixed distribution for SWH time series is then used for fixing the threshold of POT approaches. Also, the determination of the fittest distribution for the time series is used for discretising it and making a prediction which treats the problem as ordinal classification.

The work developed in this Thesis is supported by twelve papers in international journals, seven papers in international conferences, and four papers in national conferences.

## 2.4. Publications

The following papers have been published in international journals:

- A. Nikolaou, P. A. Gutiérrez, **A. M. Durán-Rosal**, I. Dicaire, F. Fernandez-Navarro, and C. Hervás-Martínez. “Detection of early warning signals in paleoclimate data using a genetic time series segmentation algorithm”, *Climate Dynamics*, Vol. 44, April, 2015, pp. 1919-1933. JCR (2015): 4.708 Position: 8/84 (Q1). DOI: 10.1007/s00382-014-2405-0
- **A. M. Durán-Rosal**, C. Hervás-Martínez, A. J. Tallón-Ballesteros, A. C. Martínez-Estudillo, and S. Salcedo-Sanz. “Massive missing data reconstruction in ocean buoys with evolutionary product unit neural networks”, *Ocean Engineering*, Vol. 117, May, 2016, pp. 292 - 301. JCR(2016): 1.894 Position: 2/14 (Q1). DOI: 10.1016/j.oceaneng.2016.03.053
- **A. M. Durán-Rosal**, M. Dorado-Moreno, P. A. Gutiérrez, and C. Hervás-Martínez. “Identification of extreme wave heights with an evolutionary algorithm in combination with a likelihood-based segmentation”, *Progress in Artificial Intelligence*, Vol. 6, March, 2017, pp. 59-66. DOI: 10.1007/s13748-016-0105-1
- **A. M. Durán-Rosal**, J. C. Fernández, P. A. Gutiérrez, and C. Hervás-Martínez. “Detection and prediction of segments containing extreme significant wave heights”, *Ocean Engineering*, Vol. 142, September, 2017, pp. 268-279. JCR(2017): 2.214 Position: 2/14 (Q1). DOI: 10.1016/j.oceaneng.2017.07.009
- **A. M. Durán-Rosal**, M. de la Paz Marín, P. A. Gutiérrez, and C. Hervás-Martínez. “Identifying market behaviours using European Stock Index time series by a hybrid segmentation algorithm”, *Neural Processing Letters*, Vol. 46, December, 2017,



pp. 767–790. JCR(2017): 1.787 Position: 63/132 (Q2). DOI: 10.1007/s11063-017-9592-8

- **A. M. Durán-Rosal**, P. A. Gutiérrez, S. Salcedo-Sanz, and C. Hervás-Martínez. “A statistically-driven Coral Reef Optimization algorithm for optimal size reduction of time series”, *Applied Soft Computing*, Vol. 63. 2018, pp. 139-153. JCR(2017): 3.907 Position: 17/132 (Q1). DOI: 10.1016/j.asoc.2017.11.037
- **A. M. Durán-Rosal**, P. A. Gutiérrez, F. J. Martínez-Estudillo, and C. Hervás-Martínez. “Simultaneous optimisation of clustering quality and approximation error for time series segmentation”, *Information Sciences*, Vol. 442-443, May, 2018, pp. 186-201. JCR(2017): 4.305 Position: 12/148 (Q1). DOI: 10.1016/j.ins.2018.02.041
- **A. M. Durán-Rosal**, J. C. Fernandez, C. Casanova-Mateo, J. Sanz-Justo, S. Salcedo-Sanz, and C. Hervás-Martínez. “Efficient Fog Prediction with Multi-objective Evolutionary Neural Networks”, *Applied Soft Computing*, Vol. 70, September, 2018, pp. 347-358. JCR(2017): 3.907 Position: 17/132 (Q1). DOI: 10.1016/j.asoc.2018.05.035
- M. Pérez-Ortiz, **A. M. Durán-Rosal**, P. A. Gutiérrez, J. Sánchez-Monedero, A. Nikolaou, F. Fernández-Navarro, and C. Hervás-Martínez. “On the use of evolutionary time series analysis for segmenting paleoclimate data”, *Neurocomputing*, Vol. 326-327, January, 2019, pp. 3-14. JCR(2017): 3.241 Position: 27/132 (Q1). DOI: 10.1016/j.neucom.2016.11.101
- **A. M. Durán-Rosal**, P. A. Gutiérrez, Á. Carmona-Poyato, and C. Hervás-Martínez. “A hybrid dynamic exploitation barebones particle swarm optimisation algorithm for time series segmentation”, *Neurocomputing*, 2018. JCR(2017): 3.241 Position: 27/132 (Q1). Accepted.
- **A. M. Durán-Rosal**, P. A. Gutiérrez, S. Salcedo-Sanz, and C. Hervás-Martínez. “Dynamical Memetization in Coral Reef Optimization Algorithms for Optimal Time Series Approximation”, *Progress in Artificial Intelligence*. Accepted. DOI: 10.1007/s13748-019-00176-0
- **A. M. Durán-Rosal**, M. Carbonero, P. A. Gutiérrez, and C. Hervás-Martínez. “On the use of a mixed distribution to fix the threshold for Peak-Over-Threshold wave height estimation”, *Coastal Engineering*, 2019. JCR(2017): 2.674 Position: 21/128 (Q1). Under Review.

Also, some related works have also been published in the proceedings of international conferences:

- **A. M. Durán-Rosal**, M. de la Paz Marín, P. A. Gutiérrez, and C. Hervás-Martínez. “Applying a Hybrid Algorithm to the Segmentation of the Spanish Stock Market Index Time Series”. 13th International Work-Conference on Artificial Neural Networks (IWANN2015). 2015. pp. 69-79. DOI: 10.1007/978-3-319-19222-2\_6
- **A. M. Durán-Rosal**, P. A. Gutiérrez, F. J. Martínez-Estudillo, and C. Hervás-Martínez. “Time Series Representation by a Novel Hybrid Segmentation Algorithm”. 11th International Conference on Hybrid Artificial Intelligent Systems (HAIS2016). 2016. pp. 163-173. DOI: 10.1007/978-3-319-32034-2\_14
- **A. M. Durán-Rosal**, J. C. Fernández, P. A. Gutiérrez, and C. Hervás-Martínez. “Hybridization of neural network models for the prediction of extreme significant wave height segments”. 2016 IEEE Symposium Series on Computational Intelligence (IEEE SSCI2016). 2016. pp. 1-8. DOI: 10.1109/SSCI.2016.7850144
- **A. M. Durán-Rosal**, D. Guijo-Rubio, P. A. Gutiérrez, S. Salcedo-Sanz, and C. Hervás-Martínez. “A coral reef optimization algorithm for wave height time series segmentation problems”. International Work-Conference on Artificial and Natural Neural Networks (IWANN2017). 2017. pp. 673-684. DOI: 10.1007/978-3-319-59153-7\_58
- **A. M. Durán-Rosal**, D. Guijo-Rubio, P. A. Gutiérrez, and C. Hervás-Martínez. “Hybrid Weighted Barebones Exploiting Particle Swarm Optimization Algorithm for Time Series Representation”. Bioinspired Optimization Methods and their Applications (BIOMA2018). 2018. pp. 126-137. DOI: 10.1007/978-3-319-91641-5\_11
- A. Nikolaou, P. A. Gutiérrez, **A. M. Durán-Rosal**, F. Fernandez-Navarro, C. Hervás-Martínez, and M. Pérez-Ortiz. “Detection of early warning signals in paleoclimate data using a genetic time series segmentation algorithm”. European Planetary Science Congress (EPSC2018). 2018.  
URL: <https://meetingorganizer.copernicus.org/EPSC2018/EPSC2018-829-1.pdf>
- D. Guijo-Rubio, **A. M. Durán-Rosal**, A. M. Gómez-Orellana, P. A. Gutiérrez, and C. Hervás-Martínez. “Distribution-based discretisation and ordinal classification applied to wave height prediction”. 19th International Conference on Intelligence Data Engineering and Automated Learning (IDEAL2018). 2018. pp 171-179. DOI: 10.1007/978-3-030-03496-2\_20

And, finally, the following contributions have been made in national conferences:

- M. Dorado-Moreno, **A. M. Durán-Rosal**, D. Guijo-Rubio, P. A. Gutiérrez, L. Prieto, S. Salcedo-Sanz, and C. Hervás-Martínez. “Multiclass prediction of wind power ramp events combining reservoir computing and support vector machines”. Conferencia

de la Asociación Española para la Inteligencia Artificial (CAEPIA 2016). 2016. pp. 300-309. DOI: 10.1007/978-3-319-44636-3\_28

- **A. M. Durán-Rosal**, M. Dorado-Moreno, P. A. Gutiérrez, and C. Hervás-Martínez. “On the use of the beta distribution for a hybrid time series segmentation algorithm”. Conferencia de la Asociación Española para la Inteligencia Artificial (CAEPIA 2016). 2016. pp. 418-427. DOI: 10.1007/978-3-319-44636-3\_39
- **A. M. Durán-Rosal**, P. A. Gutiérrez, F. J. Martínez-Estudillo, and C. Hervás-Martínez. “Multiobjective time series segmentation by improving clustering quality and reducing approximation error”. XII Congreso Español de Metaheurísticas, Algoritmos Evolutivos y Bioinspirados (MAEB 2017). 2017. pp. 920-922.  
URL: <http://mic2017.upf.edu/proceedings/>
- **A. M. Durán-Rosal**, P. A. Gutiérrez, S. Salcedo-Sanz, and C. Hervás-Martínez. “An empirical validation of a new memetic CRO algorithm for the approximation of time series”. XIII Congreso Español de Metaheurísticas, Algoritmos Evolutivos y Bioinspirados (MAEB 2018), 2018. pp. 209-218. DOI: 10.1007/978-3-030-00374-6\_20



*“Data! Data! Data!... I can’t make bricks without clay.”*

Arthur Conan Doyle

# 3

## Preprocessing: missing data reconstruction

This chapter presents a new method to reconstruct massive missing data in SWH time series, as preprocessing for following tasks such as segmentation or prediction. As we stated in previous sections, this kind of time series are collected from buoys situated in different parts of the sea and oceans, so when a buoy is broken, the information that cannot be saved needs to be recovered.

### **Main publication associated to this chapter:**

- **A. M. Durán-Rosal**, C. Hervás-Martínez, A. J. Tallón-Ballesteros, A. C. Martínez-Estudillo, and S. Salcedo-Sanz. “Massive missing data reconstruction in ocean buoys with evolutionary product unit neural networks”, *Ocean Engineering*, Vol. 117, May, 2016, pp. 292 - 301. JCR(2016): 1.894 Position: 2/14 (Q1). DOI: 10.1016/j.oceaneng.2016.03.053

### **3.1. Massive missing data reconstruction in ocean buoys with evolutionary product unit neural networks**

The following paper presents the problem of massive missing data reconstruction in ocean buoys. These buoys are collecting data in very variable conditions, such as major storms, low temperatures, etc. Sometimes, it is difficult to obtain complete time series of

the measured variables due to malfunctioning. When considering a large number of buoys, it can be very hard to find a period of completeness (without missing data on it) in the data to form a proper training or test set. The reconstruction models in the state-of-the-art are complex to be interpreted in terms of the number of predictive variables.

In this paper, we tackle the problem of massive missing data reconstruction in ocean buoys, with an evolutionary PUNN (EPUNN). The method can reconstruct a massive amount of data. To do this, the method consists of two stages.

The first stage is associated with the reconstruction by the application of well-known linear models, i.e. transfer function and neighbour correlation. Firstly, the transfer function consists of analysing the correlation and estimate a linear function between an incomplete time series of wave height with other complete ones (without gaps). In this way, for each incomplete time series, the reconstruction is made using a complete one, when the correlation between both is higher than a given threshold. Secondly, the neighbour method is able to estimate missing values of the time series, adding the information of the correlation coefficient between this series and a complete one. This method is faster than the transfer function, given that it only needs one iteration.

Once the time series are reconstructed by linear models, the best series for each model is selected for the next stage. This second stage consists of an EPUNN algorithm that uses the outputs of the linear models as inputs for the net. For each time series, the reconstruction is made using the two more correlated inputs. The complete methodology is applied to six buoys located at the Gulf of Alaska (which forms a geographical grid) with identification numbers 46001, 46061, 46076, 46078, 46082 and 46085, with data from 2008 to 2013.

Some conclusions are extracted from the results obtained in this paper. Firstly, the EPUNN models get better results than sigmoid ones, and, also, they can be represented as linear models when a natural logarithm is applied to the input variables, resulting in very interpretable models. Secondly, this approach obtains lower approximation errors in coastal buoys, which is evident given that the range of values of offshore buoys is much higher. Finally, we notice the difficulty of the estimation of extreme values which is the main drawback of the proposed method. Consequently, we can conclude that the proposed approach is valid for a large number of applications for which an accurate estimation of extreme wave height is not an issue.



# Massive missing data reconstruction in ocean buoys with evolutionary product unit neural networks



A.M. Durán-Rosal<sup>a,\*</sup>, C. Hervás-Martínez<sup>a</sup>, A.J. Tallón-Ballesteros<sup>b</sup>,  
A.C. Martínez-Estudillo<sup>c</sup>, S. Salcedo-Sanz<sup>d</sup>

<sup>a</sup> Department of Computer Science and Numerical Analysis, Universidad de Córdoba, Rabanales Campus, 14071 Córdoba, Spain

<sup>b</sup> Department of Languages and Computer Systems, Universidad de Sevilla, 41012 Seville, Spain

<sup>c</sup> Department of Management and Quantitative Methods, Universidad Loyola Andalucía, 41014 Seville, Spain

<sup>d</sup> Department of Signal Processing and Communications, Universidad de Alcalá, 28805 Alcalá de Henares, Madrid, Spain

## ARTICLE INFO

### Article history:

Received 15 May 2015

Received in revised form

23 February 2016

Accepted 22 March 2016

### Keywords:

Significant wave height

Missing values reconstruction

Product unit neural networks

Evolutionary algorithm

## ABSTRACT

In this paper we tackle the problem of massive missing data reconstruction in ocean buoys, with an evolutionary product unit neural network (EPUNN). When considering a large number of buoys to reconstruct missing data, it is sometimes difficult to find a common period of completeness (without missing data on it) in the data to form a proper training and test set. In this paper we solve this issue by using partial reconstruction, which are then used as inputs of the EPUNN, with linear models. Missing data reconstruction in several phases or steps is then proposed. In this work we also show the potential of EPUNN to obtain simple, interpretable models in spite of the non-linear characteristic of the neural network, much simpler than the commonly used sigmoid-based neural systems. In the experimental section of the paper we show the performance of the proposed approach in a real case of massive missing data reconstruction in 6 wave-rider buoys at the Gulf of Alaska.

© 2016 Elsevier Ltd. All rights reserved.

## 1. Introduction

Oceanographic buoys are measuring instruments used, among other purposes, to characterize wind-generated wave properties. The availability and accuracy of buoys data is crucial in very different problems and applications such as the design and maintenance of marine/coastal structures, wave height forecasting for safe ship navigation, or the design and operation of wave energy converters, etc. (López et al., 2013). There are different agencies such as the National Data Buoy Centre (NDBC) of the USA that maintain a large network of OBs to collect wave data on a regular basis (Londhe, 2008). A number of unexpected events can make buoys break down (such as storms, Rao and Mandal, 2005, navigation accidents, maintenance periods, etc.), causing data gaps, and therefore discontinuities in the buoys data time series, lasting from the causing event until the buoy is repaired/maintained. Some data analysis methods may allow gappy data (Liu and Wisberg, 2005), while most statistical methods require data to be gaps free (Thomson and Emery, 2014). Due to this point, the reconstruction of missing wave values has become a key topic in oceanic research.

\* Corresponding author.

E-mail address: [i92duroa@uco.es](mailto:i92duroa@uco.es) (A.M. Durán-Rosal).

Very different techniques for recovering of lost/missing OB data have been proposed in the last three decades, with prevalence of techniques focused on reconstruction of wave height data. The first approaches were quite naive, such as random sampling of data points suggested in Thompson (1971), or Monte Carlo methods applied to fill up gaps at random in a known time series of monthly mean sea level in Sturges (1983). In more recent works such as Soares and Cunha (2000) and Agrawal and Deo (2002), auto-regressive, auto-regressive moving average (ARMA) or auto-regressive integrated moving average (ARIMA) have been successfully applied to reconstruction of wave heights time series. Other constructive techniques such as cubic splines or fractal methods have been recently tested in wave height reconstruction problems (Liu et al., 2014).

In recent years, the number of works applying data machine learning (ML) techniques has been massive. Among ML techniques, neural networks (NNs) (Haykin, 1998) may have been the most used prediction methods. In Bhattacharya et al. (2003), NNs have been used to compute missing wave data in time series, measured at Europlatform station, in the North Sea. NNs have been found to be specially reliable to reach accurate estimations of missing wave data (Balas et al., 2004). In that work, feedforward multi-layer perceptrons (MLPs) and recurrent neural networks were trained by the steepest descent with momentum algorithm and the conjugate gradient algorithm, and their estimations were





*Divide the difficulties you examine in as many parts as possible for your best solution.*

René Descartes

# 4

## Time series segmentation

This chapter presents the main block of this Thesis, which is time series segmentation research works. The chapter is divided in the different objectives that can be tackled with this task, which are the discovery of useful patterns when optimising the clustering quality of the segments, the simplification of the time series by reducing their number of points and the optimisation of both objectives with a multiobjective algorithm.

### 4.1. Discovery of useful patterns

**Main publications associated to this section:**

- A. Nikolaou, P. A. Gutiérrez, **A. M. Durán-Rosal**, I. Dicaire, F. Fernandez-Navarro, and C. Hervás-Martínez. “Detection of early warning signals in paleoclimate data using a genetic time series segmentation algorithm”, *Climate Dynamics*, Vol. 44, April, 2015, pp. 1919-1933. JCR (2015): 4.708 Position: 8/84 (Q1). DOI: 10.1007/s00382-014-2405-0
- **A. M. Durán-Rosal**, M. Dorado-Moreno, P. A. Gutiérrez, and C. Hervás-Martínez. “Identification of extreme wave heights with an evolutionary algorithm in combination with a likelihood-based segmentation”, *Progress in Artificial Intelligence*, Vol. 6, March, 2017, pp. 59-66. DOI: 10.1007/s13748-016-0105-1

- **A. M. Durán-Rosal**, M. de la Paz Marín, P. A. Gutiérrez, and C. Hervás-Martínez. “Identifying market behaviours using European Stock Index time series by a hybrid segmentation algorithm”, *Neural Processing Letters*, Vol. 46, December, 2017, pp. 767–790. JCR(2017): 1.787 Position: 63/132 (Q2). DOI: 10.1007/s11063-017-9592-8
- M. Pérez-Ortiz, **A. M. Durán-Rosal**, P. A. Gutiérrez, J. Sánchez-Monedero, A. Nikolaou, F. Fernández-Navarro, and C. Hervás-Martínez. “On the use of evolutionary time series analysis for segmenting paleoclimate data”, *Neurocomputing*, Vol. 326-327, January, 2019, pp. 3-14. JCR(2017): 3.241 Position: 27/132 (Q1). DOI: 10.1016/j.neucom.2016.11.101

**Other publications associated to this section:**

- **A. M. Durán-Rosal**, M. de la Paz Marín, P. A. Gutiérrez, and C. Hervás-Martínez. “Applying a Hybrid Algorithm to the Segmentation of the Spanish Stock Market Index Time Series”. 13th International Work-Conference on Artificial Neural Networks (IWANN2015). 2015. pp. 69-79. DOI: 10.1007/978-3-319-19222-2\_6
- **A. M. Durán-Rosal**, M. Dorado-Moreno, P. A. Gutiérrez, and C. Hervás-Martínez. “On the use of the beta distribution for a hybrid time series segmentation algorithm”. *Conferencia de la Asociación Española para la Inteligencia Artificial (CAEPIA 2016)*. 2016. pp. 418-427. DOI: 10.1007/978-3-319-44636-3\_39
- **A. M. Durán-Rosal**, D. Guijo-Rubio, P. A. Gutiérrez, S. Salcedo-Sanz, and C. Hervás-Martínez. “A coral reef optimization algorithm for wave height time series segmentation problems”. *International Work-Conference on Artificial and Natural Neural Networks (IWANN2017)*. 2017. pp. 673-684. DOI: 10.1007/978-3-319-59153-7\_58
- A. Nikolaou, P. A. Gutiérrez, **A. M. Durán-Rosal**, F. Fernandez-Navarro, C. Hervás-Martínez, and M. Pérez-Ortiz. “Detection of early warning signals in paleoclimate data using a genetic time series segmentation algorithm”. *European Planetary Science Congress (EPSC2018)*. 2018.  
URL: <https://meetingorganizer.copernicus.org/EPSC2018/EPSC2018-829-1.pdf>

As can be seen, four main papers have been published in international journals. These papers are based on the idea of using GAs with a fitness function obtained by a clustering technique. In this way, the segmentation is done considering some statistical characteristics of the segments, such as the variance, the kurtosis, the skewness, the slope of a linear regression or the autocorrelation coefficient over the points belonging to a segment. This clustering is performed using a new deterministic  $k$ -means algorithm, where

the clustering results are always the same, that is, for a given segmentation, the clustering algorithm guarantees that the same fitness value will be obtained. The algorithm was then improved using a statistical likelihood-based segmentation over the segments obtained in the best segmentation of the GA. For detection in wave height time series, the hypothesis was that the points of time series are sampled from a beta distribution, which is specifically used in the presence of extreme values. In the case of financial time series, the assumption was made considering a normal distribution.

The conference papers show the initial validation of the algorithms and some experimental procedures using other alternatives of MHs, such as the CRO algorithm. The four main publications are now presented in the different subsections of this section.

#### **4.1.1. Detection of early warning signals in paleoclimate data using a genetic time series segmentation algorithm**

In this paper, we tackle the problem of detection of early warning signals before TPs in climate time series. As we stated, in previous chapters (see Section 1.6.1), a TP is a little change that produces a big consequence, that is, an inflexion point which makes that the dynamic of time series changes sharply.

For that, we propose a new segmentation algorithm consisting of a GA whose fitness function is provided by a clustering technique, i.e. it is a metric which measures the clustering quality of the time series segments. The algorithm starts with a random division of segments, and the evolutionary process modifies these cut points trying to optimise the clustering quality of them. The clustering is made according to the similarities in their statistical parameters: variance, skewness, kurtosis, the slope of a linear regression over the points of the segment, the mean squared error of the approximation of the segment, and lag-1 autocorrelation coefficient. All segments metrics are scaled to the range  $[0, 1]$  given that the differences from those metrics with larger ranges would disrupt others with smaller ranges.

The methodology is tested with two paleoclimate datasets: the GISP2 and the NGRIP (North Greenland ice core project)  $\delta^{18}\text{O}$  time series with a 20-year resolution. A 5-point average is made to reduce short-term fluctuations. In addition, synthetic datasets obtained from well-known dynamical systems are also studied. These time series are used to detect early warning signals of Dansgaard-Oeschger (DO) events, which are considered TPs in this kind of data.

Results agree that the GA can effectively analyse these DO events and discover similarities and differences in their statistical and dynamical characterisation. Specifically, warning signals, such as the increase in autocorrelation, variance and mean square error,

are robustly found in the GISP2  $\delta^{18}\text{O}$  dataset for DO 0, 1, 2, 4, 7, 8, 11, 12, and the NGRIP  $\delta^{18}\text{O}$  for DO 0, 1, 4, 8, 10, 11, 12. The increase in mean square error, suggesting nonlinear behaviour, has been found to correspond with an increase in variance prior to several DO events for  $\sim 90\%$  of the algorithm runs for the GISP2  $\delta^{18}\text{O}$  dataset and for  $\sim 100\%$  of the algorithm runs for the NGRIP  $\delta^{18}\text{O}$  dataset. The proposed approach applied to both synthetic data and paleoclimate datasets provides a novel visualisation tool of climate time series analysis.

# Detection of early warning signals in paleoclimate data using a genetic time series segmentation algorithm

Athanasia Nikolaou · Pedro Antonio Gutiérrez ·  
Antonio Durán · Isabelle Dicaire ·  
Francisco Fernández-Navarro · César Hervás-Martínez

Received: 29 November 2013 / Accepted: 6 November 2014 / Published online: 26 November 2014  
© Springer-Verlag Berlin Heidelberg 2014

**Abstract** This paper proposes a time series segmentation algorithm combining a clustering technique and a genetic algorithm to automatically find segments sharing common statistical characteristics in paleoclimate time series. The segments are transformed into a six-dimensional space composed of six statistical measures, most of which have been previously considered in the detection of warning signals of critical transitions. Experimental results show that the proposed approach applied to paleoclimate data could effectively analyse Dansgaard–Oeschger (DO) events and uncover commonalities and differences in their statistical and possibly their dynamical characterisation. In particular, warning signals were robustly detected in the GISP2 and NGRIP  $\delta^{18}\text{O}$  ice core data for several DO events (e.g.

DO 1, 4, 8 and 12) in the form of an order of magnitude increase in variance, autocorrelation and mean square distance from a linear approximation (i.e. the mean square error). The increase in mean square error, suggesting non-linear behaviour, has been found to correspond with an increase in variance prior to several DO events for  $\sim 90\%$  of the algorithm runs for the GISP2  $\delta^{18}\text{O}$  dataset and for  $\sim 100\%$  of the algorithm runs for the NGRIP  $\delta^{18}\text{O}$  dataset. The proposed approach applied to well-known dynamical systems and paleoclimate datasets provides a novel visualisation tool in the field of climate time series analysis.

**Keywords** Warning signals · Time series segmentation · Tipping points · Abrupt climate change · Genetic algorithms · Clustering

A. Nikolaou · I. Dicaire (✉) · F. Fernández-Navarro  
Advanced Concepts Team, European Space Research  
and Technology Centre (ESTEC), European Space Agency  
(ESA), Noordwijk, Netherlands  
e-mail: isabelle.dicaire@esa.int

A. Nikolaou  
e-mail: athanasia.nikolaou@esa.int

F. Fernández-Navarro  
e-mail: i22fenaf@uco.es; fafernandez@uloyola.es

P. A. Gutiérrez · A. Durán · C. Hervás-Martínez  
Department of Computer Science and Numerical Analysis,  
University of Córdoba, Córdoba, Spain  
e-mail: pagutierrez@uco.es

A. Durán  
e-mail: i92duroa@uco.es

C. Hervás-Martínez  
e-mail: chervas@uco.es

F. Fernández-Navarro  
Department of Mathematics and Engineering,  
Universidad Loyola Andalucía, Seville, Spain

## 1 Introduction

The statistical tools used to extract knowledge from time series analysis have undergone considerable development during the past decade (see Livina and Lenton 2007; Livina et al. 2011; Lenton et al. 2012; Scheffer et al. 2009; Dakos et al. 2008; Held and Kleinen 2004; Cimadoribus et al. 2013). Driven by the ultimate aim of understanding past climate variability, the above studies focused on statistical analysis of time series that demonstrate *threshold behaviour* as used in Alley et al. (2003). Candidate explanations for transitions of a system over thresholds link to dynamical systems analysis, which is used for gaining insight into internal variability modes and response to external forcing on both simple and complex systems (Saltzman 2001). Adopting the notation from Ashwin et al. (2012) the abrupt shift from a stable state to another stable state could be e.g. due to *B-tipping* or *N-tipping*. In *B-tipping* the system is driven past bifurcation points, where equilibrium solutions

#### 4.1.2. Identification of extreme wave heights with an evolutionary algorithm in combination with a likelihood-based segmentation

As we discussed in previous sections, the determination and clustering of events which contains larger wave heights in relation with other close in time is a challenge for the design of offshore structures. In this paper, we solve this problem by using a hybrid algorithm consisting of a GA in combination with a likelihood-based segmentation.

The GA is able to find the extreme events in a cluster, but the application of LS in different parts of the evolution improves the quality of the solutions obtained by this GA. The LS consists of a likelihood-based segmentation assuming that the points of the segments follow a beta distribution<sup>1</sup>. In this way, considering as main hypothesis that a segment  $s$  is a random sample from a  $X_t$  distribution, we have:

$$H_0 \equiv X_t \in B(\alpha, \beta), \quad (4.1)$$

and,

$$H_1 \equiv \begin{cases} X_{t_L} \in B(\alpha_L, \beta_L), \\ X_{t_R} \in B(\alpha_R, \beta_R). \end{cases} \quad (4.2)$$

where  $B(\alpha, \beta)$  is the beta distribution,  $\alpha$  and  $\beta$  are the parameters,  $X_{t_L}$  are the values at the left of the cut point  $t$ , and  $X_{t_R}$  are those which are at the right. The segmentation scheme is fundamentally based on the likelihood ratio test under these hypotheses.

Empirically, we validate the hybridisation following four strategies. The first one does not apply the LS. In the second one, the best solution of the last generation of the GA is improved by the application of the LS. The third one consists of the application of the LS to the best solution in 1/3, 2/3 and 3/3 of the total number of generations. And finally, in the last one, the algorithm applies the LS to the best 20 % individuals of the last generation.

The method is applied to three time series collected from buoys situated in Puerto Rico and the Gulf of Alaska. Results agree that the best methodology is the application of the LS to the best 20 % of the individuals in the last generation. Furthermore, the algorithm provides a well-defined cluster of segments which contains extreme values, allowing the study of the characteristics of this type of events.

---

<sup>1</sup>the beta distribution is a distribution specifically designed for correctly representing extreme values

# Identification of extreme wave heights with an evolutionary algorithm in combination with a likelihood-based segmentation

Antonio M. Durán-Rosal<sup>1</sup> · Manuel Dorado-Moreno<sup>1</sup> · Pedro A. Gutiérrez<sup>1</sup> · César Hervás-Martínez<sup>1</sup>

Received: 4 November 2016 / Accepted: 21 November 2016 / Published online: 19 December 2016  
© Springer-Verlag Berlin Heidelberg 2016

**Abstract** This paper presents four configurations of a genetic algorithm (GA) combined with a local search (LS) method for time series segmentation with the purpose of correctly recognising extreme values. The LS method is based on likelihood maximisation of a beta distribution. The proposal is tested on three real ocean wave height time series, where extreme values are frequently found. Concretely, the time series are taken from two oceanographic buoys in the Gulf of Alaska, and another one from Puerto Rico. The results show that the different combinations of LS improve the results of the GA. Furthermore, the algorithm provides segmentations where extreme values are grouped in a well-defined cluster, which allows the study of the characteristics of this type of events.

**Keywords** Time series segmentation · Evolutionary algorithms · Extreme value distributions · Clustering · Likelihood-based optimisation · Beta distribution

## 1 Introduction

Nowadays, the importance of time series data mining and machine learning has increased resulting in several works and studies. Time series can be defined as temporal data collected chronologically and they can be easily obtained from several applications. Their numerical and continuous nature and their difficulty to be processed, analysed and mined often leads to a discretisation of the continuous values into significant symbols [3,24]. This process, called “numeric-to-symbolic” (N/S) conversion, is considered as one of the best preprocessing techniques before mining time series. Other approaches [16,17] suggest dividing the time series using previously identified change points and substituting the segments with suitable functions.

Furthermore, it is well known that time series segmentation is one of the most important keys of time series representation and mining. The main objective of this process is to provide a more compact representation of the data by dividing the entire time series into a set of consecutive temporal periods, called segments. There are two main proposals to represent the time series into a high-level representation by segmenting them. On the one hand, this can be done using simple descriptions of the segments, i.e., using linear interpolations or linear regressions, with the objective of minimising their approximation error [4,12,23]. On the other hand, useful patterns can be found in the time series leading to segments, where two main tasks have to be considered [13]: matching of sequence patterns and recognition of periodical patterns.

Clustering methods, such as *k*-means, hierarchical clustering and expectation maximisation, has been used by many researchers for clustering time series [18,21]. Clustering can also be applied as part of the segmentation procedure to improve the conversion to symbols by finding similar-

✉ Antonio M. Durán-Rosal  
i92duroa@uco.es

Manuel Dorado-Moreno  
i92domom@uco.es

Pedro A. Gutiérrez  
pagutierrez@uco.es

César Hervás-Martínez  
chervas@uco.es

<sup>1</sup> Department of Computer Science and Numerical Analysis,  
University of Córdoba, Rabanales Campus, Albert Einstein  
building, 14071 Córdoba, Spain

#### 4.1.3. Identifying market behaviours using European stock index time series by a hybrid segmentation algorithm

The discovery of useful patterns embodied in a time series is of fundamental relevance in many real applications. Repetitive structures and common type of segments can also provide very useful information of patterns in financial time series. Time series segmentation is often used for trend analysis and the analysis of the movement of a stock market. Given that there are several important methodologies to answer the questions of what and when buying or selling, we focus on the cyclical and technical analysis approaches, due to our analysis is chartist, and it is based on the found financial patterns, i.e. we search for characteristic phases in a time series and identify the main financial patterns in each phase to analyse the behaviour of the stock market.

Specifically, we propose a new GA which is combined with an LS procedure based on the likelihood ratio optimisation, assuming that the values of the financial time series are normally distributed. In this sense, considering that a segment  $s$  is a random sample from a  $X_t$  distribution, the hypotheses are:

$$H_0 \equiv X_t \in N(\mu, \sigma), \quad (4.3)$$

and,

$$H_1 \equiv \begin{cases} X_{t_L} \in B(\alpha_L, \beta_L), \\ X_{t_R} \in B(\alpha_R, \beta_R). \end{cases} \quad (4.4)$$

where  $N(\mu, \sigma)$  is the normal distribution with mean  $\mu$  and standard deviation  $\sigma$ ,  $X_{t_L}$  are the values at the left of a cut point  $t$ , and  $X_{t_R}$  are those which are at the right. The characteristics of the segments are analysed following the behaviour of bearish, bullish and sluggish markets, and cycles, booms and crashes.

In our experiments, we apply the methodology to two stock market index time series: IBEX35 Spanish index (closing prices) and a weighted average (AVG) time series compound by the IBEX35 (Spanish), BEL20 (Belgian), CAC40 (French) and DAX (German) indexes. The algorithm maps the segments into a five-dimensional space including variance, skewness, kurtosis, slope of a linear regression over the points of the segment, and autocorrelation coefficient, with the aim of grouping them. Experimental results show that it is possible to discover similar patterns in both time series.



# Identifying Market Behaviours Using European Stock Index Time Series by a Hybrid Segmentation Algorithm

Antonio M. Durán-Rosal<sup>1</sup>  · Mónica de la Paz-Marín<sup>1</sup> ·  
Pedro A. Gutiérrez<sup>1</sup> · César Hervás-Martínez<sup>1</sup>

Published online: 25 January 2017  
© Springer Science+Business Media New York 2017

**Abstract** The discovery of useful patterns embodied in a time series is of fundamental relevance in many real applications. Repetitive structures and common type of segments can also provide very useful information of patterns in financial time series. In this paper, we introduce a time series segmentation and characterization methodology combining a hybrid genetic algorithm and a clustering technique to automatically group common patterns from this kind of financial time series and address the problem of identifying stock market prices trends. This hybrid genetic algorithm includes a local search method aimed to improve the quality of the final solution. The local search algorithm is based on maximizing a likelihood ratio, assuming normality for the series and the subseries in which the original one is segmented. To do so, we select two stock market index time series: IBEX35 Spanish index (closing prices) and a weighted average time series of the IBEX35 (Spanish), BEL20 (Belgian), CAC40 (French) and DAX (German) indexes. These are processed to obtain segments that are mapped into a five dimensional space composed of five statistical measures, with the purpose of grouping them according to their statistical properties. Experimental results show that it is possible to discover homogeneous patterns in both time series.

**Keywords** Time series segmentation · Hybrid algorithms · Clustering · European stock market indexes

---

Antonio M. Durán-Rosal  
i92duroa@uco.es

Mónica de la Paz-Marín  
mpaz@uco.es

Pedro A. Gutiérrez  
pagutierrez@uco.es

César Hervás-Martínez  
chervas@uco.es

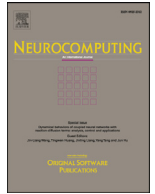
<sup>1</sup> Department of Computer Science and Numerical Analysis, University of Córdoba, Rabanales Campus, Albert Einstein Building, 14071 Córdoba, Spain

#### **4.1.4. On the use of evolutionary time series analysis for segmenting paleoclimate data**

The following paper presents an extension of those presented in previous sections for detecting early warning signals in paleoclimate data. The paper involves the following contributions:

- The evaluation of the segmentation is automated using two criteria. The first one is related to the comparison of the segmentation to an ideal segmentation given by experts in the area. The second one measures the stability of the algorithm.
- The fitness function of the algorithm is selected from a set of 10 clustering validity indexes, considering the best one from a battery of experiments.
- Some improvements in the algorithm are also included, such as binary coding, mutation and a constraint for the minimum segment size.
- Finally, a simple model of prediction is derived from the obtained segments.

These contributions improve the results of the paper presented in Section 4.1.1. The experiments performed and the results obtained agree that Calisíky and Harabasz index, which has been found to be one of the best-performing ones for adjusting the number of clusters, is the best cluster validity index for the evolutionary algorithm. With this index, the algorithm is capable of detecting all the DO events except number 9 and 13 in the GISP2 dataset, while in the NGRIP time series, four events are not detected: 2, 9, 13 and 16.



# On the use of evolutionary time series analysis for segmenting paleoclimate data



M. Pérez-Ortiz<sup>a,1,\*</sup>, A.M. Durán-Rosal<sup>b</sup>, P.A. Gutiérrez<sup>b</sup>, J. Sánchez-Monedero<sup>a</sup>, A. Nikolaou<sup>c</sup>, F. Fernández-Navarro<sup>a</sup>, C. Hervás-Martínez<sup>b</sup>

<sup>a</sup> Department of Quantitative Methods, Universidad Loyola Andalucía, Third Building, Córdoba 14004, Spain

<sup>b</sup> Department of Computer Science and Numerical Analysis, University of Córdoba, Rabanales Campus, Albert Einstein building, Córdoba 14071, Spain

<sup>c</sup> German Aerospace Center Institute of Planetary Research Planetary Physics Rutherfordstraße 2 12489 Berlin

## ARTICLE INFO

### Article history:

Received 20 November 2015

Revised 9 June 2016

Accepted 1 November 2016

Available online 18 September 2017

### Keywords:

Time series segmentation

Genetic algorithms

Clustering

Paleoclimate data

Tipping points

Abrupt climate change

## ABSTRACT

Recent studies propose that different dynamical systems, such as climate, ecological and financial systems, among others, present critical transition points named to as *tipping points* (TPs). Climate TPs can severely affect millions of lives on Earth so that an active scientific community is working on finding early warning signals. This paper deals with the development of a time series segmentation algorithm for paleoclimate data in order to find segments sharing common statistical patterns. The proposed algorithm uses a clustering-based approach for evaluating the solutions and six statistical features, most of which have been previously considered in the detection of early warning signals in paleoclimate TPs. Due to the limitations of classical statistical methods, we propose the use of a genetic algorithm to automatically segment the series, together with a method to compare the segmentations. The final segments provided by the algorithm are used to construct a prediction model, whose promising results show the importance of segmentation for improving the understanding of a time series.

© 2017 Published by Elsevier B.V.

## 1. Introduction

In contrast to the famous statement of Linnaeus “*natura non facit saltus*” (or nature makes no leaps), it has been proven that some points of no return, thresholds and phase changes are widespread in nature and these are often non linear [1]. Such events can be rarely anticipated and some of them can have detrimental consequences on Earth’s climate and large-scale impacts on human and ecological systems. This increases the imperious necessity of studying, analysing and developing techniques for characterising them in order to construct reliable early warning systems. Although the human being have influenced their local environment for millennia, e.g. reducing biodiversity, it is now, since the industrial revolution, that truly global changes are being noticed [2,3]. Examples that are currently receiving attention include the potential collapse of the Atlantic thermohaline circulation, the dieback of the Amazon rainforest or the decay of the Greenland ice sheet [1]. Formally, a climate “tipping point” (TP, also known as “little things can make a big difference”) occurs when a small change in forcing

triggers a strongly nonlinear response in the internal dynamics of part of the climate system, qualitatively changing its future state.

The critical relevance of early TPs detection has produced a growing attention of the scientific community. Lenton differentiates between several types of TPs, and presents some indicators that can help to detect them, such as the increase of autocorrelation of the series values [4]. In [5], more concrete techniques regarding data processing and indicators are presented. They study a bank of methods using only simulated ecological data, concluding in concordance with the literature that there is no unique best indicator for identifying an upcoming transition. They also conclude that all the methods require specific data-treatment. Up to our knowledge, all previous works tackle the TP detection with statistical methods trying to select (by trial and error) the method (and the time-window) most suitable to detect those transitions. They require an intensive data preprocessing that includes, for instance, the use of Gaussian filters or rolling windows that introduce extra parameters (such as the width of the Gaussian function or size of the window) that need to be optimised [4,5]. The main limitation behind these methods is that different TPs require specific treatment, which is the specific objective that this paper tries to tackle.

Although one of the main areas of research for time series is their modelling [6], time series segmentation is emerging as a very interesting field, aiming to provide a compact representation of the

\* Corresponding author.

E-mail address: [i82perom@uco.es](mailto:i82perom@uco.es) (M. Pérez-Ortiz).

<sup>1</sup> This paper has been invited to be included in the “Special Issue Neurocomputing-HAIS2014”.

## 4.2. Time series size reduction

### Main publications associated to this section:

- **A. M. Durán-Rosal**, P. A. Gutiérrez, S. Salcedo-Sanz, and C. Hervás-Martínez. “A statistically-driven Coral Reef Optimization algorithm for optimal size reduction of time series”, *Applied Soft Computing*, Vol. 63. 2018, pp. 139-153. JCR(2017): 3.907 Position: 17/132 (Q1). DOI: 10.1016/j.asoc.2017.11.037
- **A. M. Durán-Rosal**, P. A. Gutiérrez, Á. Carmona-Poyato, and C. Hervás-Martínez. “A hybrid dynamic exploitation barebones particle swarm optimisation algorithm for time series segmentation”, *Neurocomputing*, 2018. JCR(2017): 3.241 Position: 27/132 (Q1). Accepted.

### Other publications associated to this section:

- **A. M. Durán-Rosal**, P. A. Gutiérrez, F. J. Martínez-Estudillo, and C. Hervás-Martínez. “Time Series Representation by a Novel Hybrid Segmentation Algorithm”. 11th International Conference on Hybrid Artificial Intelligent Systems (HAIS2016). 2016. pp. 163-173. DOI: 10.1007/978-3-319-32034-2\_14
- **A. M. Durán-Rosal**, D. Guijo-Rubio, P. A. Gutiérrez, and C. Hervás-Martínez. “Hybrid Weighted Barebones Exploiting Particle Swarm Optimization Algorithm for Time Series Representation”. *Bioinspired Optimization Methods and their Applications (BIOMA2018)*. 2018. pp. 126-137. DOI: 10.1007/978-3-319-91641-5\_11
- **A. M. Durán-Rosal**, P. A. Gutiérrez, S. Salcedo-Sanz, and C. Hervás-Martínez. “An empirical validation of a new memetic CRO algorithm for the approximation of time series”. *XIII Congreso Español de Metaheurísticas, Algoritmos Evolutivos y Bioinspirados (MAEB 2018)*, 2018. pp. 209-218. DOI: 10.1007/978-3-030-00374-6\_20
- **A. M. Durán-Rosal**, P. A. Gutiérrez, S. Salcedo-Sanz, and C. Hervás-Martínez. “Dynamical Memetization in Coral Reef Optimization Algorithms for Optimal Time Series Approximation”, *Progress in Artificial Intelligence*. Accepted. DOI: 10.1007/s13748-019-00176-0

Generally, the contributions in this part are focused on developing new MHs for the problem of time series size reduction. Firstly, to alleviate a problem of the CRO algorithm, which is the large amount of parameters that need to be defined, a new SCRO algorithm is proposed, where the parameters of the algorithm are dynamically updated during the evolution. In this way, the operators are applied based on the statistics of centralisation

and dispersion of the fitness function of the population in each iteration. Secondly, the idea is to improve the statistical BBPSO, considering a modification of the Gaussian function for having a better exploration at the beginning of the evolution and better exploitation at the end. These two main publications are now presented in the different subsections of this section.

#### 4.2.1. A statistically-driven coral reef optimisation algorithm for optimal size reduction of time series

Segmenting time series with the aim of reducing the number of points is an interesting challenge for simplifying the data without losing important information. In this work, we proposed a new variant of the standard CRO algorithm for this purpose. The main problem of the CRO is its configuration, due to the large amount of parameters which need to be defined. We propose a new version, where the parameters are updated dynamically throughout the iterations of the algorithm, depending on the fitness distribution of the population. The best solution obtained by this algorithm, which is called SCRO, is then applied a local optimisation using two well-known LS methods, Bottom-Up and Top-Down, which are able to improve the quality of the solution in terms of the approximation error.

Let  $f_{ij}$  be the fitness function of the solution  $i$  in iteration  $j$ , and let  $\bar{f}_j$  and  $S_{f_j}^2$  be the mean value and the variance of the corals at iteration  $j$ . The SCRO algorithm modifies the parameters of CRO in the following way:

1. Free positions: at the beginning, those corals whose fitness verifies  $f_{i1} \notin (\bar{f}_1 - S_{f_1}, 1]$  are deleted.
2. Asexual reproduction: a random coral from the set of corals whose fitness verifies  $f_{ij} \in (\bar{f}_j + S_{f_j}, 1]$  is mutated and considered candidate solution.
3. External sexual reproduction: those corals with a fitness function verifying  $f_{ij} \in (\bar{f}_j - S_{f_j}, 1]$  are externally sexually reproduced.
4. Internal sexual reproduction: the remaining corals ( $f_{ij} \in [0, \bar{f}_j - S_{f_j}]$ ) are mutated in each generation with internal sexual mutation.
5. Depredation: at the end of each iteration, the algorithm eliminates those corals whose fitness verifies  $f_{ij} \in [0, \bar{f}_j - 2S_{f_j}]$ .

As can be seen, the parameters are updated in each iteration, and the user does not need to specify them in the configuration. The methodology is tested in 16 time series collected from different sources, including financial time series, SWH time series and

benchmark time series. SCRO is compared against other state-of-the-art methods, which are Bottom-Up, Top-Down, GA, PSO and standard CRO, showing that our algorithm outperforms the rest of methods and that it has the same performance than the standard CRO, but without the need of specifying the parameters.



# A statistically-driven Coral Reef Optimization algorithm for optimal size reduction of time series

Antonio M. Durán-Rosal<sup>a,\*</sup>, Pedro A. Gutiérrez<sup>a</sup>, Sancho Salcedo-Sanz<sup>b</sup>, César Hervás-Martínez<sup>a</sup>

<sup>a</sup> Department of Computer Science and Numerical Analysis, University of Córdoba, Córdoba, Spain

<sup>b</sup> Department of Signal Processing and Communications, Universidad de Alcalá, 28805 Alcalá de Henares, Madrid, Spain

## ARTICLE INFO

### Article history:

Received 22 June 2017

Received in revised form 6 October 2017

Accepted 21 November 2017

Available online 26 November 2017

### Keywords:

Data mining

Time series segmentation

Coral reef optimization

Hybrid algorithm

## ABSTRACT

This paper is focused on reducing the number of elements in time series with minimum information loss, with specific applications on time series segmentation. A modification of the coral reefs optimization metaheuristic (CRO) is proposed for this purpose, which is called statistical CRO (SCRO), where the main parameters of the algorithm are adjusted based on the mean and standard deviation associated with the fitness distribution. Moreover, the algorithm is combined with the Bottom-Up and Top-Down methodologies (traditional local search methods for time series segmentation), resulting in a hybrid methodology (HSCRO). We evaluate the performance of these algorithms using 16 time series from different application areas. The statistically-driven version of CRO is shown to improve the results of the standard CRO, eliminating the necessity of manually adjusting the main parameters of the algorithm and dynamically adjusting these parameters throughout the evolution. Moreover, when compared with other local search methods and metaheuristics from the state of the art, HSCRO shows robust segmentation results, consistently obtaining lower approximation errors.

© 2017 Elsevier B.V. All rights reserved.

## 1. Introduction

Time series data mining is currently an important area of research in different fields of science and engineering. In this way, time series can be easily obtained from different source areas, including biology, financial problems, climate-related applications, renewable energy, hydrology, etc. They are used with the objective of solving different problems, such as clustering, indexing, classification, structure discovery or anomalies detection, among others.

A very important task within time series data mining is the problem of time series segmentation [1]. It consists of dividing the time series into different non-overlapping segments, based on series of cut points. Depending on the application tackled, the specific objective of time series segmentation can be different: A first group of methods tries to discover useful patterns of the time series, based

on the similarities between the segments. In [2], a genetic algorithm was proposed for this purpose, where the cut points are optimized in terms of the similarities between the different segments obtained. The method proposed in [3] achieves indirect sequence clustering by using an online recursive fuzzy clustering, which is found to be stable in the presence of outliers. Fuzzy segmentation of multivariate time-series has been also tackled in [4], using a modified Gath-Geva clustering. Another work in this direction is [5], where a fixed-length window was used for representing the time series using simple patterns obtained by a segmentation procedure. A closely related topic is the segmentation for anomaly detection. It has been intensively studied in signal processing to locate abrupt changes along the time series [6–9]. Finally, the characterization of Tipping Points can also be approached by using this type of time series segmentation [10], where common patterns which occur before these events are found and used as early warning signals in paleoclimate time series.

On the other hand, a second group of segmentation methods aim to reduce the number of points (amount of data) in the time series without losing the essential information. In other words, these algorithms try to simplify the time series, alleviating the difficulty of processing, analysing or mining complete time series databases.

\* Corresponding author at: Department of Computer Science and Numerical Analysis, University of Córdoba, Rabanales Campus, Albert Einstein Building 3rd Floor, 14071 Córdoba, Spain.

E-mail addresses: [i92duroa@uco.es](mailto:i92duroa@uco.es) (A.M. Durán-Rosal), [pagutierrez@uco.es](mailto:pagutierrez@uco.es) (P.A. Gutiérrez), [sancho.salcedo@uah.es](mailto:sancho.salcedo@uah.es) (S. Salcedo-Sanz), [chervas@uco.es](mailto:chervas@uco.es) (C. Hervás-Martínez).

#### 4.2.2. A hybrid dynamic exploitation barebones particle swarm optimisation algorithm for time series segmentation

Another work proposed in this Thesis for applying segmentation to reduce the number of points of time series is presented in this section. From the previous work, we observed that the computational cost of time series segmentation could be reduced using properly adapted PSO algorithms. The quality of the solutions seemed to be improved too.

Consequently, in this work, we propose a new variant of the BBPSO, which automatically adapts the parameters of the normal distribution to update the positions in each iteration (see Equation 1.6). In this way, the proposed algorithm, which is called DBBePSO, updates the importance of the social and cognitive components during the iterations, with the aim of having a better exploration at the beginning and better exploitation at the end of the evolution. For that, we propose a modified Gaussian distribution with a new parameter  $\lambda$ :

$$x_{i,j}^t = \begin{cases} N\left(\frac{p_{i,j}^{t-1} + p_{g,j}^{t-1}}{2}, \lambda |p_{i,j}^{t-1} - p_{g,j}^{t-1}|\right) & \text{if } U(0, 1) < 0.5, \\ p_{i,j}^{t-1} & \text{otherwise,} \end{cases} \quad (4.5)$$

where  $\lambda$  is updated dynamically over the generations from an initial value of 1 to a final value of 0.1:

$$\lambda = \frac{0.9(L - l)}{L} + 0.1, \quad (4.6)$$

where  $L$  is the maximum number of evaluations allowed to the algorithm (stop criterion), and  $l$  is the current number of evaluations.

The stop criterion is the number of evaluations which is established based on the length of each time series. Also, the hybridisation (which consists in modifying the cut points using Bottom-Up and Top-Down procedures in their iterative versions) is made using a different strategy, i.e. in the beginning, the 50 % of the total of particles are applied a local search, and, at the end, the best solution after the evolutionary process is also optimised.

All of these modifications and adaptations of PSO are tested in 15 datasets collected from different sources. The experimental validation confirms that the DBBePSO and its hybrid version, called HDBBePSO, lead to better results when compared to other state-of-the-art algorithms, such as traditional methods, GA, PSO, BBePSO, and an optimal method called Salotti. Moreover, the computational cost is drastically reduced.



# A hybrid dynamic exploitation barebones particle swarm optimisation algorithm for time series segmentation

Antonio M. Durán-Rosal\*, Pedro A. Gutiérrez, Ángel Carmona-Poyato, César Hervás-Martínez

*Department of Computer Science and Numerical Analysis, University of Córdoba, Córdoba, Spain*

---

## Abstract

Large time series are difficult to be mined and preprocessed, hence reducing their number of points with minimum information loss is an active field of study. This paper proposes new methods based on time series segmentation, including the adaptation of the particle swarm optimisation algorithm (PSO) to this problem, and more advanced PSO versions, such as barebones PSO (BBPSO) and its exploitation version (BBePSO). Moreover, a novel algorithm is derived, referred to as dynamic exploitation barebones PSO (DBBePSO), which updates the importance of the social and cognitive components throughout the generations. All these algorithms are further improved by considering a final local search step based on the combination of two well-known standard segmentation algorithms (Bottom-Up and Top-Down). The performance of the different methods is evaluated using 15 time series from various application fields, and the results show that the novel algorithm (DBBePSO) and its hybrid version (HDBBePSO) outperform the rest of segmentation techniques.

## Keywords:

Time series size reduction, time series segmentation, particle swarm optimisation, hybrid algorithm

---

## 1. Introduction

Recently, time series data mining (TSDM) has become an important field of research in science and engineering [1, 2]. Time series can be obtained from different areas, such as climate [3], hydrology [4], finances [5], satellite images [6], etc. They are used for different tasks depending on the objective of the researchers and the application areas, e.g. classification [7, 8], forecasting [9, 10], tipping point detection [11], clustering [12], similarity assessment [13, 14] or segmentation [15]. Specifically, time series segmentation is an important task, which consists of cutting the time series in some specific points trying to achieve different objectives, which are generally related to two points of view.

Firstly, time series segmentation can be used to discover useful patterns or segments in time series. Chung et al. [16] proposed a genetic algorithm for this purpose, using the similarities between the segments for optimising the cut points. Tseng et al. [17] combined a genetic algorithm with a clustering procedure and considered the discrete wavelet transformation (DWT) for the representation of the segments. The genetic algorithm proposed in [11] is aimed to characterise tipping points (TPs) and analyse the common patterns which occur before them, in order to create early warning signals in paleoclimate time series. Furthermore, a full analysis of different metrics

for clustering evaluation and a first approximation to forecast TPs using the patterns previously identified were made in [3]. Fuzzy segmentation of multivariate time series was approached by a modified Gath-Geva clustering algorithm in [18], and an online recursive fuzzy clustering for indirect sequence clustering was proposed in [19]. Anomaly detection has been widely analysed for signal processing with the aim of locating abrupt changes along the time series [20]. There are many more applications of this kind of time series segmentation, such as the detection of important events in stock price time series [9, 21] or the detection and prediction of wave height extreme events combining a genetic algorithm with artificial neural networks [22].

On the other hand, the second group of time series segmentation algorithms tries to tackle the difficulty of processing and mining large time series. Their large amount of data (i.e. their high dimensionality) makes them very difficult to analyse. Because of this reason, and considering the fact that data mining is constrained by three types of limited resources (time, memory and sample size), different algorithms have been proposed with the aim of reducing the dimensionality or the number of points of time series. In the literature, time series segmentation techniques are also called time series representation procedures. These methods reduce the dimension of a given time-series by transforming it into a new representation space [23]. In general, TSDM tasks can be classified as first-hand processing (i.e. dimensionality reduction) or second-hand processing (further analysis of time series). Time series representation methods are first-hand processing algorithms, being useful for reducing the number of points of the time series while keeping their fun-

---

\*Corresponding author at: Department of Computer Science and Numerical Analysis, University of Córdoba, Rabanales Campus, Albert Einstein Building 3rd Floor, 14071 Córdoba, Spain. Tel.: +34 957 218 349; Fax: +34 957 218 630. E-mail addresses: i92duroa@uco.es, pagutierrez@uco.es, chervas@uco.es

damental characteristics [24]. In this context, the authors in [25, 26] proposed a method based on dividing the time series using previously identified change points and represented the segments with suitable approximations. Piecewise linear approximation (PLA) is a global term referring to all the algorithms which reduce the number of points in the time series with a minimum information loss, based on linear interpolations or regressions [27]. Top-Down and Bottom-Up approaches proposed by Keogh et al. [27] are two simple PLA algorithms, based on iteratively reducing the approximation error. There are some other representations algorithms, such as adaptive piecewise constant approximation (APCA) [28] or symbolic aggregate approximation (SAX) [29].

The work presented in this paper is focused on the second group, specifically on PLA representation methods, whose objective is to reduce the number of elements in time series with minimum information loss. For this purpose, we use a modification of a particle swarm optimisation algorithm (PSO) [30] for segmenting time series, considering also two different related versions, barebones PSO (BBPSO) and exploiting barebones PSO (BBePSO). PSO is an evolutionary algorithm which simulates the social and cognitive behaviour of a set of particles, such as birds or fish when looking for food. In this way, PSO optimises problems considering a set of candidate solutions, denoted as particles (in our case, segmented time series), which move along the search space. In PSO, the social component refers to the best global position found by the algorithm, while the cognitive one is the best solution found by the individual particle. In general, PSO can be more easily adapted to the specific problem being tackled, as fewer parameters have to be configured when compared to other metaheuristics, such as genetic algorithms or ant colony systems. On the other hand, BBPSO avoids the use of velocities and, instead, considers a normal distribution to decide whether the update should take the best global position into account or the best local one [31]. Finally, BBePSO adds an exploiting component to BBPSO, improving convergence [31]. PSO has been applied in many real problems, including hydrology prediction [32], video tracking [33], power system state estimation [34], etc.

In standard PSO, and also in BBPSO, and BBePSO, the importances of the social component (exploration) and the cognitive component (exploitation) are not updated during the generations. In this paper, we propose a new formulation, where the social component is more important at the beginning of the evolution, while the cognitive component is more important at the end, resulting in that we call dynamic BBePSO (DBBePSO).

On the other hand, evolutionary algorithms (EAs) are able to perform a global multi-point search, converging to high quality areas. In this sense, they are considered robust heuristics that can be applied in different problems. The main problem with EAs is that they are not good at finding the precise optimum in these high-quality areas [35]. To solve this issue, several authors combine EAs with a local search (LS) procedure to improve the best solutions. The idea is to combine the advantages of the EA (global explorer) and the advantages of the LS (local exploiter), resulting in hybrid algorithms. The hybridisation can be made in different ways, which are very important in terms of

accuracy and computational cost. Some of the strategies previously used include the multi-start approach, the Lamarckian learning, the Baldwinian learning, the partial Lamarckianism and the process of random linkage [36, 37, 38]. In this way, we combine the previously presented algorithms with a LS procedure, consisting in removing a number of cut points with a Bottom-Up methodology and, then, adding the same number of cut points using the Top-Down procedure [27]. All the algorithms are applied to the segmentation of several time series in the experimental section of the paper, and hybrid DBBePSO obtains very good results which outperform the state of the art algorithms considered.

The rest of the paper is organised as follows: Section 2 briefly presents the main parts of the PSO, BBPSO, and BBePSO algorithms. Section 3 describes the new PSO proposal, while Section 4 includes the different considerations needed for adapting all the algorithms for time series segmentation. Section 5 shows the considered time series, which are extracted from real-world applications and different public repositories, the experimental setting and the statistical analysis of the results obtained. Finally, the paper is concluded in Section 6.

## 2. Particle Swarm Optimisation algorithm and its advanced versions

The particle swarm optimisation (PSO) [30] is an evolutionary-type algorithm for search and optimisation, based on the simulation of a swarm of particles, i.e., birds or fish, looking for food. In PSO, a swarm is formed by a set of  $P$  particles in a  $D$ -dimensional space, being  $D$  the length of the particles. Each particle  $i$  is a candidate solution of the studied problem, and it is represented by the following characteristics at iteration  $t$ : the current position of the particle  $\mathbf{x}_i^t$ , the current velocity of the particle  $\mathbf{v}_i^t$  and the best position found by the particle  $\mathbf{p}_i^t$ . The fitness function evaluates the quality of a particle  $\mathbf{x}_i$  and is presented by  $f(\mathbf{x}_i)$ . The velocity of the particle represents the direction and the rate of change in the movement of the particle at iteration  $t$ , while the best position  $\mathbf{p}_i^t$  is the value of the  $\mathbf{x}_i$  visited by the particle resulting in the best fitness. Moreover, an array with the best global solution ( $\mathbf{p}_g$ ) is also stored, which is defined as  $\mathbf{p}_g^t = \arg \max_p \{f(\mathbf{p}_g^{t-1}), f(\mathbf{p}_1^t), f(\mathbf{p}_2^t), \dots, f(\mathbf{p}_p^t)\}$  (considering a maximisation problem). Thus, the evolution is possible due to the cooperation of the particles, considering the local best position  $\mathbf{p}_i$  (cognitive component) and the global best position  $\mathbf{p}_g$  (social component).

For each iteration of a PSO algorithm, the velocity  $\mathbf{v}_i$  is updated in the following way:

$$\mathbf{v}_i^t = w \cdot \mathbf{v}_i^{t-1} + \rho_1^t \cdot C_1 \cdot (\mathbf{p}_i^{t-1} - \mathbf{x}_i^{t-1}) + \rho_2^t \cdot C_2 \cdot (\mathbf{p}_g^{t-1} - \mathbf{x}_i^{t-1}), \quad (1)$$

where  $w$  is the inertia weight,  $\rho_1^t, \rho_2^t$  are uniform random values obtained at iteration  $t$ ,  $\rho_1, \rho_2 \sim U(0, 1)$ , and  $C_1, C_2$  are the acceleration constants. The  $w$  parameter controls the impact of the memory with respect previous velocities. The cognitive component  $(\mathbf{p}_i - \mathbf{x}_i)$  represents the experience of the particle with respect to its best-found solution, while the social component

$(\mathbf{p}_g - \mathbf{x}_i)$ , represents the experience with respect to the global best solution. The position of the particles is then updated using the expression:

$$\mathbf{x}_i^t = \mathbf{x}_i^{t-1} + \mathbf{v}_i^t. \quad (2)$$

Finally, the individual best position  $\mathbf{p}_i$  and the global best position  $\mathbf{p}_g$  are also updated in each iteration.  $\mathbf{p}_i$  is updated as:

$$\mathbf{p}_i^t = \begin{cases} \mathbf{p}_i^{t-1} & \text{if } f(\mathbf{x}_i^t) \leq f(\mathbf{p}_i^{t-1}), \\ \mathbf{x}_i^t & \text{if } f(\mathbf{x}_i^t) > f(\mathbf{p}_i^{t-1}), \end{cases} \quad (3)$$

while, for the global best position, we have:

$$\mathbf{p}_g^t = \arg \max_{\mathbf{p}} \{f(\mathbf{p}_g^{t-1}), f(\mathbf{p}_1^t), f(\mathbf{p}_2^t), \dots, f(\mathbf{p}_P^t)\}. \quad (4)$$

The PSO algorithm is repeated during a predefined number of iterations or until velocity updates are near zero. The quality of the solutions (particles) is measured by a fitness function (in this section we have considered maximisation problems). Algorithm 1 illustrates the flowchart diagram of the PSO algorithm, which summarises the previously defined steps.

---

**Algorithm 1** Pseudo-code for the PSO algorithm

---

**Input:** Valid values for the parameters controlling the PSO algorithm

**Output:** A solution with the best *fitness* value found by the algorithm

- 1: Initialise the swarm randomly
  - 2: Evaluate the initial swarm
  - 3: **while not** stop condition **do**
  - 4:   Update velocities (Eq. 1)
  - 5:   Update positions (Eq. 2)
  - 6:   Evaluate the new swarm
  - 7:   Update personal best positions (Eq. 3)
  - 8:   Update global best positions (Eq. 4)
  - 9: **end while**
  - 10: Return the best individual (final solution) from the swarm
- 

### 2.1. Barebones PSO

One of the improved versions of PSO is the barebones PSO (BBPSO) [31]. This algorithm does not take into account the velocities to update the current position of the particles in the swarm. Instead, BBPSO replaces Equations 1 and 2 with the following expression for the  $j$ -th dimension:

$$x_{i,j}^t = N\left(\frac{p_{i,j}^{t-1} + p_{g,j}^{t-1}}{2}, |p_{i,j}^{t-1} - p_{g,j}^{t-1}|\right), \quad (5)$$

where  $N(\mu, \sigma)$  is a normal random distribution with  $\mu$  mean and  $\sigma$  standard deviation, and  $i = 1, \dots, P$ ,  $j = 1, \dots, D$ . This equation is based on theoretical studies confirming that particles converge to a weighted average of the global and personal best positions [39]. In this way, each dimension of each particle is selected from a Gaussian distribution where the mean is the average value of the global and local best positions, and the difference between them is used as the standard deviation. This procedure allows taking large steps when the personal best positions are far away from the global best positions.

### 2.2. Exploiting barebones PSO

In [31], Kennedy also proposed an alternative version of the BBPSO, called exploiting barebones PSO (BBePSO), where the velocity and position updates are replaced with:

$$x_{i,j}^t = \begin{cases} N\left(\frac{p_{i,j}^{t-1} + p_{g,j}^{t-1}}{2}, |p_{i,j}^{t-1} - p_{g,j}^{t-1}|\right) & \text{if } U(0, 1) < 0.5, \\ p_{i,j}^{t-1} & \text{otherwise.} \end{cases} \quad (6)$$

This equation establishes a 0.5 probability that the  $j$ -th dimension of the particle  $i$  changes to the corresponding personal best position. In this way, the BBePSO searches with a higher degree of exploitation than BBPSO. In general, this exploiting version outperforms other variants of PSO [40]. Unlike standard PSO, the barebones variants (BBPSO and BBePSO) do not need a value for the weight and the acceleration coefficients, so they are more suitable for those application problems where the value of these parameters is difficult to be estimated.

## 3. Dynamic exploiting barebones PSO

In this work, a dynamic BBePSO (DBBePSO) algorithm is proposed, where the importance of the social and the cognitive components are updated along the generations.

As we mentioned before, DBBePSO updates the current positions of each particle ( $\mathbf{x}_i$ ) in a similar way that BBePSO. However, in our proposal, the importance of the exploration and the exploitation are dynamically updated over the generations using a modified Gaussian distribution:

$$x_{i,j}^t = \begin{cases} N\left(\frac{p_{i,j}^{t-1} + p_{g,j}^{t-1}}{2}, \lambda |p_{i,j}^{t-1} - p_{g,j}^{t-1}|\right) & \text{if } U(0, 1) < 0.5, \\ p_{i,j}^{t-1} & \text{otherwise.} \end{cases} \quad (7)$$

The novelty is the multiplicative parameter  $\lambda$  in the standard deviation of the distribution. It is known that evolutionary algorithms work better when the exploration is higher at the beginning but lower at the end [41]. To do so,  $\lambda$  is updated dynamically over the generations from an initial value of 1 to a final value of 0.1:

$$\lambda = \frac{0.9(L - l)}{L} + 0.1, \quad (8)$$

where  $L$  is the maximum number of evaluations allowed to the algorithm (stop criterion), and  $l$  is the current number of evaluations. As can be observed, when the number of evaluations is 0, then  $\lambda$  is 1.0; and it decreases to 0.1 when  $l$  is close to  $L$ . It is important to mention that the  $\lambda$  update is done at the beginning of each iteration  $t$  of the algorithm.

## 4. Adapting the algorithms for time series segmentation

### 4.1. Problem definition

Given a time series  $Y = \{y_n\}_{n=1}^N$ , the main goal is to split the time series by dividing the values into  $m$  consecutive segments, taking into account that the error approximation of these segments needs to be as lower as possible. In other words, from all the time indexes ( $n = 1, \dots, N$ ), a set of  $m - 1$  cut points are selected, being presented in ascending order ( $t_1 < t_2 < \dots <$

$t_{m-1}$ ). In this way, the set of the resulting segments is composed by  $s_1 = \{y_1, \dots, y_{t_1}\}$ ,  $s_2 = \{y_{t_1}, \dots, y_{t_2}\}$ ,  $\dots$ ,  $s_m = \{y_{t_{m-1}}, \dots, y_N\}$ , and the algorithm has to determine the values of the  $m - 1$  cut points. Note that the cut points are part of two segments, the precedent segment and the posterior one, while the rest of points belong to a single segment. In order to reduce the amount of information, each segment is approximated using linear interpolation between the initial and the final points (i.e. the cut points delimiting the segment).

It is important to mention that the search space is very large. Consequently, the use of evolutionary algorithms is proposed in this paper.

#### 4.2. Particle representation

The position of a particle is represented by an array (chromosome) of real values ( $\mathbf{x}_i$ ), where the length of the chromosome is the same that the number of segments minus one ( $m - 1$ ), i.e. the number of cut points. Each chromosome element  $x_{i,j}$  stores a real value, which is rounded to the closest integer in order to obtain the value of the  $j$ -th cut point ( $t_{i,j}$ ). For example, the chromosome of length 5,  $\mathbf{x}_i = \{1.68, 5.76, 12.12, 15.30, 20.10\}$  corresponds to the following cut points,  $\mathbf{t}_i = \{2, 6, 12, 15, 20\}$ . An example of a particle for this problem is shown in Figure 1. It is important to note that the values of the chromosome need to be presented in ascending order (see section 4.5).

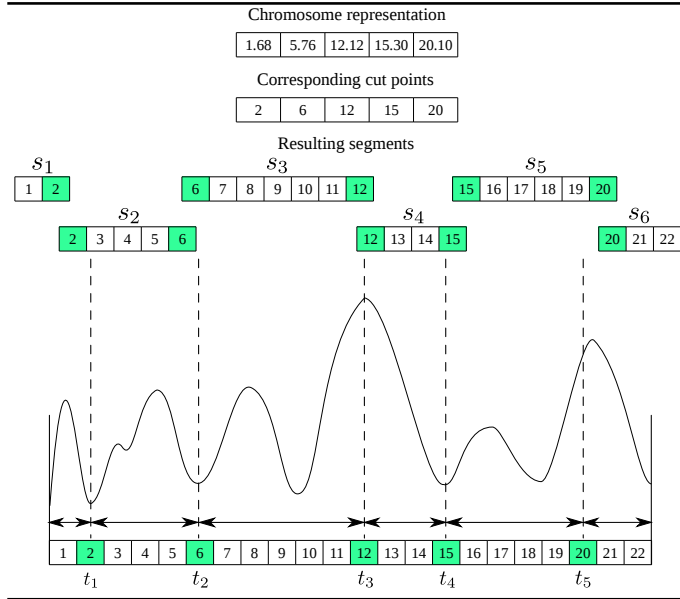


Figure 1: Chromosome representation:  $\mathbf{x}_i = \{1.68, 5.76, 12.12, 15.30, 20.10\}$ .

#### 4.3. Initialisation of the swarm

The population of the swarm is a set of  $P$  arrays of real values with a length of  $m - 1$ . In the initial population, the cut points are randomly selected taking into account that they must be subscripted in ascending order, and each cut point has to be unique (it is not possible to have two cut points with the same value). Note that the initial population is formed by integer values, but, during the generations, these positions are updated with real values.

#### 4.4. Fitness evaluation

As we stated before, the main goal of this type of time series segmentation is to reduce the number of points without losing important information. For that, we optimise the error produced by the approximation with respect to the original time series values. Thus, the fitness function is defined as minimising the difference between each real value of the time series and its corresponding approximation. The approximation error of the  $n$ -th point of the time series in the swarm is defined as:

$$e_n(\mathbf{x}_i) = (y_n - \hat{y}_n(\mathbf{x}_i)), \quad (9)$$

where  $y_n$  is the real value of the  $n$ -th point in the time series, and  $\hat{y}_n(\mathbf{x}_i)$  is the PLA approximation value obtained by a linear interpolation in the chromosome  $\mathbf{x}_i$ . The fitness function considered for the complete chromosome is the root mean square of the  $e_n(\mathbf{x}_i)$  (RMSE), which is formally defined as:

$$RMSE(\mathbf{x}_i) = \sqrt{\frac{1}{N} \sum_{n=1}^N e_n^2(\mathbf{x}_i)}. \quad (10)$$

Due to the fact that this metric needs to be minimised, the final fitness function is  $f = \frac{1}{1+RMSE(\mathbf{x}_i)}$ , which is bounded in the interval  $[0, 1]$ . Figure 2 shows the evaluation process of the chromosome used in Figure 1.

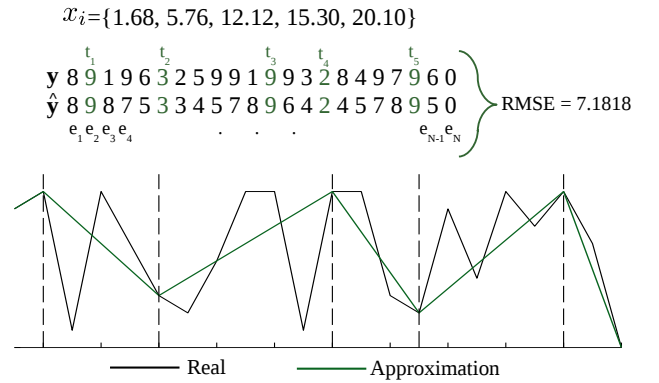


Figure 2: Example of evaluation.

#### 4.5. Repair solutions

In time series segmentation, there are certain constraints that have to be satisfied to ensure proper solutions:

1. The first one is that the time index  $n$  must be presented in ascending order, and therefore the values of the chromosome ( $x_{i,1} < x_{i,2} < \dots < x_{i,m-1}$ ). After applying the position updates, it can be possible that a too large step is taken for one of the dimensions, making the value of the cut point to be higher than the next in the chromosome ( $x_{i,k} > x_{i,k+1}$ ) or lower than the previous one ( $x_{i,k} < x_{i,k-1}$ ). In order to avoid this problem, after updating the positions, the algorithm sorts the cut points of the chromosomes (particles) in ascending order.

2. The second constraint is related to the values of the first and the last genes of the chromosome. The value of the first gene should not be smaller than 1.5 ( $x_{i,1} \geq 1.5$ ), and the value of the last one should not be higher than  $N - 0.5$  ( $x_{i,m-1} < N - 0.5$ ). This is because the first and the last point of the time series can not be cut points, and the nearest integers of a value lower than 1.5 or a value higher than  $N - 0.5$  are 1 and  $N$ , respectively. If this constraint is not satisfied, the chromosome is rescaled:

$$\mathbf{x}'_i = \frac{\mathbf{x}_i^t - \min\{\mathbf{x}_i^t\}}{\max\{\mathbf{x}_i^t\} - \min\{\mathbf{x}_i^t\}} (\max\{\mathbf{x}_i^{t-1}\} - \min\{\mathbf{x}_i^{t-1}\}) + \min\{\mathbf{x}_i^{t-1}\}, \quad (11)$$

where the  $\min(\mathbf{x})$  and  $\max(\mathbf{x})$  represent the minimum and the maximum value of the array  $\mathbf{x}$ , respectively.

#### 4.6. Hybridisation procedure

A local search strategy is used to further improve the quality of the solutions, based on the combination of Bottom-Up and Top-Down algorithms [27]. Bottom-Up considers each element of the time series as a possible cut point, and, during the iterations, the two adjacent segments incurring in a lowest cost are merged, that is, those adjacent segments whose merging results in the minimum increase of error. Top-Down is the complementary algorithm, which works with the opposite philosophy. At the beginning, the complete time series is considered as a segment, and Top-Down recursively splits the segment considering the point resulting in the maximum error decrease. Both algorithms are run until some stopping criteria are met (related with the approximation error). Our proposed local search methods consists in removing a percentage of the cut points of the best solution using the Bottom-Up strategy and then adding the same number of cut points using the Top-Down algorithm.

To use these algorithms, we have modified the implementations proposed in [27] in such a way that, for both, the stopping criteria is the number of segments to merge or cut, respectively. Note that the implementation of Top-Down presented in [27] is recursive, so we have transformed it into an iterative method.

We have considered the following strategy for combining this local search with the metaheuristics (GA, PSO and the different PSO variants): at the beginning of the evolution, a 50% of the population is randomly selected, and these individuals are improved by the local search. After that, the metaheuristic is applied to the complete population, including standard random individuals and the ones improved by the local search method. Finally, the best solution obtained by the metaheuristic is also applied a local search.

#### 4.7. DBBePSO algorithm for time series segmentation

This section summarises the work-flow of the DBBePSO presented in Section 3 for time series segmentation, including all the considerations previously exposed. The main steps of the algorithm are summarised in Algorithm 2. Very similar pseudocodes are used for adapting the rest of PSO variants.

---

#### Algorithm 2 Dynamic BBBePSO for time series segmentation

---

**Input:** Time series.

**Output:** Segmented time series.

- 1: Initialise a random initial particle swarm (population).
  - 2: Evaluate the initial population.
  - 3: **while not** stop condition **do**
  - 4:   Update the importance of the social and cognitive components.
  - 5:   Update the positions of the particles.
  - 6:   Repair solutions.
  - 7:   Evaluate the new population (particle swarm).
  - 8:   Update the best global and the best local positions.
  - 9: **end while**
  - 10: Apply the local search to the best solution obtained by the DBBePSO.
  - 11: **return** Best solution after the local search.
- 

## 5. Experimental results and discussion

This section analyses the time series considered for validating the different methods, the experimental setting and the results obtained.

### 5.1. Datasets used in our experiments

In this work, we evaluate the performance of the DBBePSO algorithm in several synthetic and real-world time series collected from public repositories, to test its robustness in different scopes of application. The time series used are the following:

- Synthetic time series
  - UCR time series: four datasets from the UCR Time Series Classification Archive [42] has been selected. Originally, these time series are divided into training and test, because it is a time series classification repository. As we are facing time series segmentation, we have joined some of the training patterns in order to have larger length time series. The time series selected are Hand Outlines, with a total of 8127 points, and Mallat, Phoneme and StarLightCurves, all of them with 8192 observations.
  - Donoho-Johnstone time series: this series is extracted from a benchmark repository [43, 44, 45], which is widely used in the neural net and machine learning community. The Donoho-Johnstone benchmarks are formed by four functions to which random noise can be added to produce an infinite number of datasets. In this work, we have considered the function Blocks with medium noise, producing a total of 2048 observations<sup>1</sup>.
- Real-world application time series

---

<sup>1</sup>All these time series can be downloaded from <https://sites.google.com/site/icdmmdl/>

- Stock prices time series from financial applications: five different indexes has been selected. The first one is IBEX35 time series (called IBEX since this moment for simplicity). It is one of the Spanish official indexes of the Madrid stock market. The time series consists of a total of 5730 observations considering daily values from 14 January 1992 to 26 September 2014. The rest of time series includes market rates collected from four banks (BBVA, Deutsche Bank, Intesa San Paolo, and Société Générale). These four series have a length of 4174 points, considering daily values from 1 January 1999 to 9 February 2015.
- Wave height time series (Hs): four time series of significant wave height collected from buoys of the National Data Buoy Center of the USA [46] have been used. Two buoys are collecting data in the Gulf of Alaska (with registration number 46001 and 46075), and the rest are from Puerto Rico (41043 and 41044). One value every six hours from 1st January 2008 to 31st December 2013 is considered for buoy 46001 (8767 observations), while data from 1st January 2011 to 31st December 2015 are considered for the rest of buoys (7303 observations for each one).
- Arrhythmia dataset contains cardiology data which belongs to the PhysioBank ATM of the MIT BIH Arrhythmia dataset [47, 48]. We used the MLII signal of the record 108 (9000 observations) to test the algorithm in this dataset.

All time series considered are shown in Figures 3 and 4.

## 5.2. Experimental design

The experimental design for the time series under study is presented in this subsection. We compare the following algorithms:

- An optimal algorithm which is able to obtain the minimum error segmentation for a given time series. We consider the method proposed by Salotti [49]. This method obtains optimal polygonal approximations of a digital curve when a prefixed starting point of the polygonal approximation is used. It is based on finding the shortest path in a graph using the  $A^*$ -algorithm. Its computational complexity is close to  $O(N^2)$ . However, in closed curves, in order to obtain the optimal polygonal approximation, all the points of the curve should be considered as starting points, and the computational complexity is close to  $O(N^3)$ . We consider the improved version proposed in [50], where the computational time is reduced by a 16%. This improved version was originally proposed for obtaining optimal polygonal approximations in closed curves, which is similar to the problem of time series segmentation, with two main differences: the first and the last points are fixed, thus its computational complexity is close to  $O(N^2)$ ; and the error is calculated in

the vertical line, instead of in a line perpendicular to the approximation. Both adaptations can be easily included to perform optimal time series segmentation.

- Two iterative versions of the Bottom-Up and Top-Down algorithms explained in Section 4.6 have been run, with the aim of obtaining an approximation of the time series with a predefined number of points or segments.
- A genetic algorithm (GA) has been run with crossover and mutation probabilities set to  $p_c = 0.8$  and  $p_m = 0.2$ , respectively.
- A basic particle swarm optimisation algorithm (PSO) is run with the following specific parameters: initial velocities of the particles are set to values close to zero [51], the inertia coefficient ( $w$ ) is set to 0.72, and the constant parameters ( $C_1$  and  $C_2$ ) are fixed to 1.49, as previously proposed in [39].
- The exploiter version of the barebones PSO (BBPSO) proposed by [31] has also been tested. Note that this version is better than BBPSO (see [31]), so we have not considered this last algorithm in our experiments.
- Finally, the DBBePSO proposed in this paper is also run, and, as mentioned before, the  $\lambda$  parameter is set to 1 at the beginning, and it linearly decreases to 0.1. No other parameters have to be set.

According to [52], a 40% of the in the GA, BBPSO, and the proposed DBBePSO are fine-tuned according to the method presented in Section 4.6 resulting in the HGA, HBBPSO, and HDBBePSO methods, respectively. For all algorithms, the population size is 100. The number of segments is set to a 2.5% of the total number of points of the time series. The stop criterion of all the algorithms is a maximum number of fitness evaluations, which is established based on the length of each time series,  $N$ , by considering the equation  $3.5N$ . Given the stochastic nature of the evolutionary algorithms, they have been run 30 times with different seeds. The error approximation results, measured in RMSE, and the computational time in seconds are analysed. Finally, some statistical tests are performed to determine the existence of significant differences in the results, which will be later detailed.

## 5.3. Discussion

RMSE results are shown in Table 1. For the deterministic algorithms (Salotti, Bottom-Up and Top-Down), there is a single result for each dataset-algorithm pair. In the case of the evolutionary algorithms (GA, PSO, BBPSO, DBBePSO and their hybrid versions), the table summarises the mean and the standard deviation of the 30 runs using different seeds. The mean ranking of each algorithm is also included, considering a 1 for the best method for each dataset and an 11 for the worst one. Firstly, we can observe that the proposed local search method improves the solutions of the evolutionary algorithms to a large extent. That is, hybrid algorithms reduce the approximation error of their corresponding standard evolutionary ones. In this

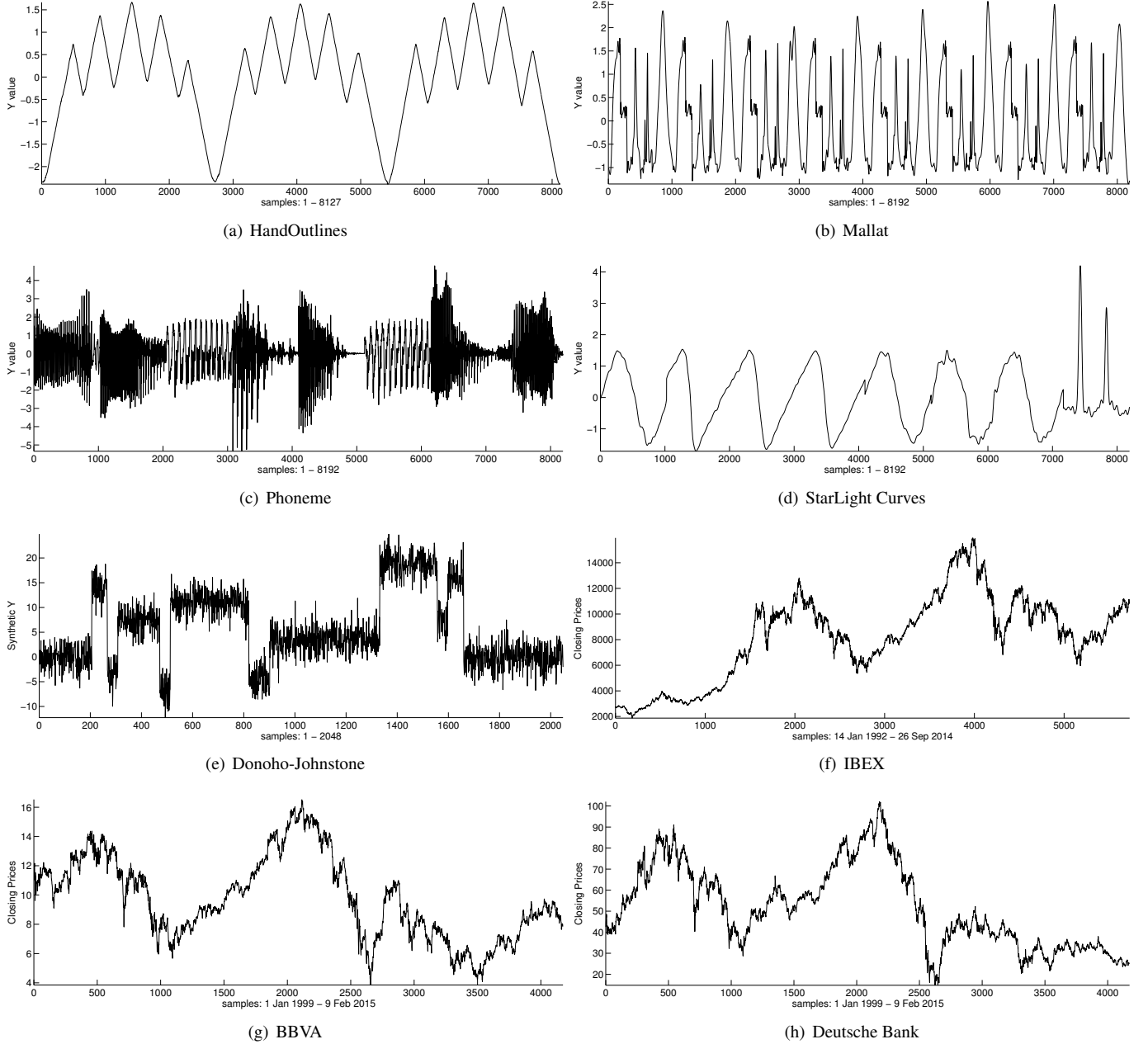


Figure 3: Time series considered for the experiments (1/2).

way, the mean ranking of the GA is improved from 9.03 to 4.03, the ranking of PSO decreases from 10.93 to 4.57, BBBePSO improves from 9.17 to 3.83, and the DBBePSO ranking is 7.53, this ranking being 2.43 in the corresponding hybrid version (HDBBePSO). Obviously, the best method in error terms for all databases is the optimal algorithm of Salotti. In general, if we do not consider the optimal algorithm, the best results are obtained with the HDBBePSO algorithm, with the lowest error for 10 out of 15 time series, and the second-best RMSE in the rest of series. HGA, HPSO and HBBBePSO results seem to be very similar in performance with a mean rank of 4.03, 4.57, and 3.83, respectively. Furthermore, the standard deviations of

HDBBePSO are the lowest ones, showing the robustness of the proposed method (the performance does not depend so much on the initialisation).

If we only observe standard evolutionary algorithms without considering hybrid versions, we can also conclude that the novel DBBePSO outperforms the rest of methods. Bottom-Up appear to be better finding low approximation error solutions when compared with all evolutionary algorithms except DBBePSO (again without considering hybrid versions). This is due to the bad performance of evolutionary methods in finding the precise optimum in high-quality areas, this reason motivating the use of hybrid algorithms. However, as can be seen,

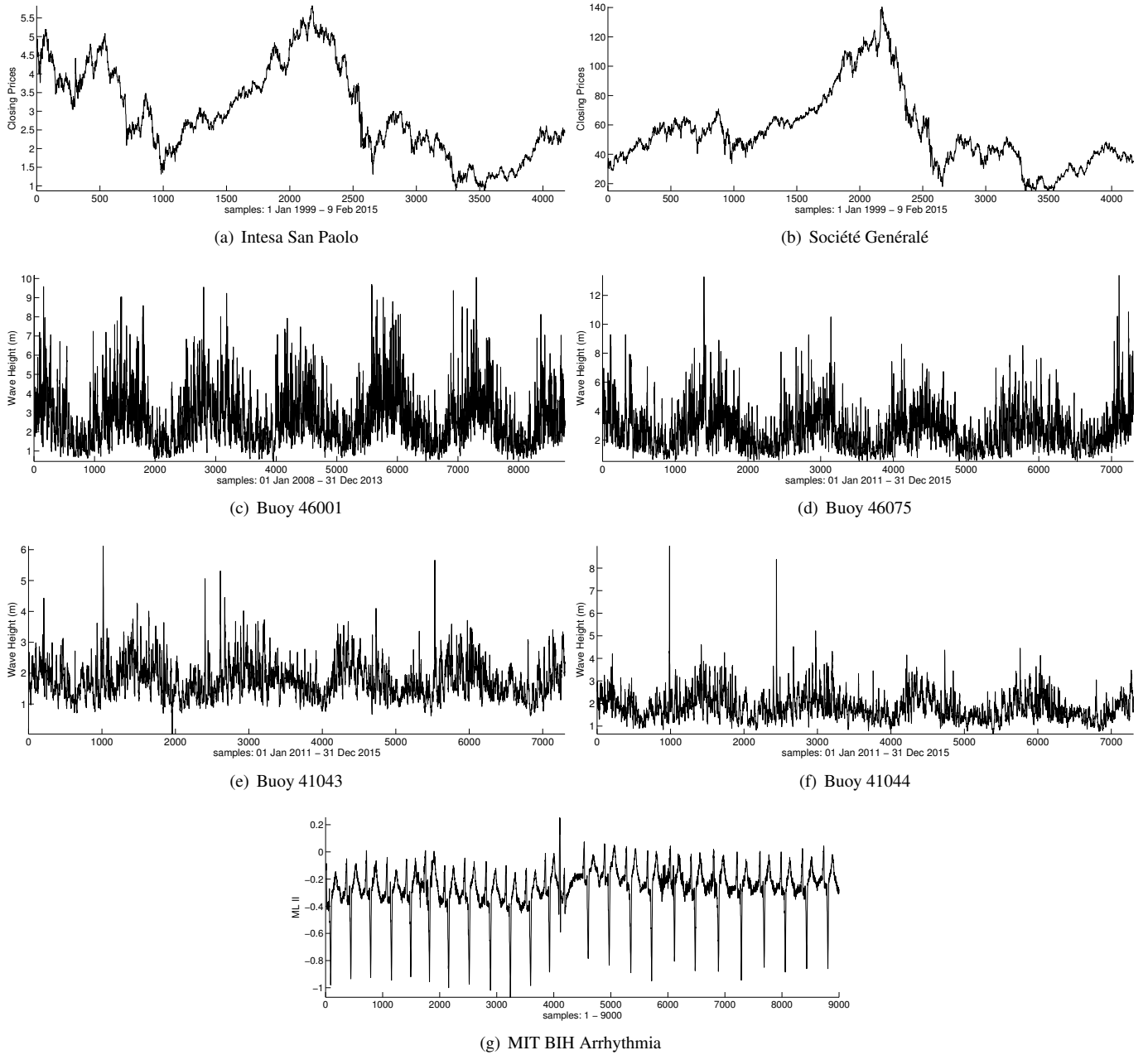


Figure 4: Time series considered for the experiments (2/2).

the dynamic adaptation of the exploration and exploitation of DBBePSO reduces this problem, obtaining the same error approximation that Top-Down and a slightly worse error, but comparable, with respect to Bottom-Up. Moreover, this problem is completely solved with the hybridisation proposed in this paper, which results, in the experiments, in the lowest error approximations, improving Top-Down and Bottom-Up methods to a great extent.

It is known that an important inconvenient of evolutionary algorithms (based on populations of solutions) is their higher computational cost when compared to algorithms based on a single solution which are not optimal. Table 2 summarises

the runtime for the deterministic algorithms and the means and standard deviations of the runtimes of 30 repetitions of the evolutionary ones, measured in seconds<sup>2</sup>. As can be seen, the worst computational times are obtained by the Salotti's method showing the necessity of using non optimal algorithms in order to derive good solutions in acceptable computational time. Specifically, when compared to HDBBePSO, the computational cost of Salotti's method is approximately nine times higher in the worst case and twice in the best one. The rest of the results confirm that the fastest algorithms are Bottom-Up and Top-Down.

<sup>2</sup>All the experiments were run using an Intel(R) Xeon(R) CPU E5-2620 v3 at 2.40 GHz with 32 GB of RAM



However, we should take into account that the approximation error obtained by these methods is clearly worse than that obtained by hybrid methodologies (specially, by HDBBePSO, see Table 1). Obviously, the hybrid versions of the algorithms are slightly costlier than their pure evolutionary alternatives. PSO is faster than the rest of the evolutionary methods but the second fastest method is DBBePSO, while being much better obtaining lower RMSE.

In order to analyse the results from the point of view of statistical hypothesis contrasts, a set of statistical tests have been used. Given that Salotti's algorithm is the optimal method, it does not make sense to include it in the statistical tests (it will be always the best performing method, at the cost of much higher computational resources). Also, the hybrid methods are always better and slightly costlier than the pure evolutionary ones. For these reasons, for the statistical tests, we only consider the deterministic methods and the hybrid versions of the algorithms, that is, Top-Down, Bottom-Up, HGA, HPSO, HBBBePSO and HDBBePSO. Firstly, we analyse the RMSE results. To do so, a Friedman test [53] has been considered using the different RMSE rankings, which shows that, for a level of significance  $\alpha = 5\%$ , the confidence interval is  $C_0 = (0, F_{0.05} = 2.35)$ , and the F-distribution statistical value is  $F^* = 22.19$ . Consequently, the test rejects the null-hypothesis, which states that all algorithms perform equally in mean ranking of RMSE, that is, the algorithm effect is statistically significant. Due to this rejection, we consider the best performing method in RMSE as control method for a post-hoc test [54], comparing this algorithm with the rest of methods. It has been noted that comparing all algorithms to a given one (control method) is more sensitive than comparing all algorithms to each other.

The Holm's test compares the  $i$ -th and  $j$ -th algorithms with the following statistic:

$$z = \frac{\bar{r}_i - \bar{r}_j}{\sqrt{\frac{k(k+1)}{6N}}},$$

where  $\bar{r}_i$  is the mean ranking of the  $i$ -algorithm,  $k$  is the number of algorithms, and  $N$  is the number of datasets. With the value of  $z$ , we find the probability of a normal distribution and compared it with a level of significance  $\alpha$ . Holm's test adjusts the value for  $\alpha$  to compensate multiple comparisons, using a procedure that sequentially tests the hypotheses ordered by their significance. The ordered  $p$ -values are denoted by  $p_1, p_2, \dots, p_k$ , so that  $p_1 < p_2 < \dots < p_k$ . The test compares each  $p_i$  with  $\alpha_i^* = \alpha/(k - i)$ , starting with the most significant  $p$ -value. If  $p_1$  is lower than  $\alpha/(k - 1)$ , the corresponding hypothesis is rejected, and then we compare  $p_2$  with  $\alpha/(k - 2)$ , and so on. When a certain null hypothesis is accepted the remaining ones are also accepted.

The results of the Holm's test are shown in Table 3. When using HDBBePSO as control algorithm (CA), Holm's test shows that  $p_i < \alpha_i^*$  in all cases, for  $\alpha = 0.05$ , confirming that there are statistically significant differences favouring HDBBePSO.

In the same way, to determine the existence of statistical significance of the rank differences in runtime (seconds) for the six algorithms and all databases, we perform another Friedman

test with their mean runtime rankings. We observe that, for a level of significance of 5%, the F-distribution statistical value is  $F^* = 201.33$  with a confidence interval of  $C_0 = (0, F_{0.05} = 2.35)$ , rejecting the null-hypothesis and concluding that the differences are statistically significant. Then, we apply the Holm's test, considering HDBBePSO, again, as the control algorithm. The results are shown in Table 4. Using HDBBePSO as CA, Bottom-Up and Top-Down are significantly better in mean run time than the proposed algorithm (marked with “(-)” in Table 4). This is because the optimisation of Bottom-Up and Top-Down is based on a single solution and the methods are not optimal, while the evolutionary approaches are based on populations. Finally, with respect the remaining methods (HGA, HPSO, and HBBBePSO), there are no statistically significant differences in runtime, but HDBBePSO outperforms them in quality of solutions.

## 6. Conclusions

This paper proposes a novel algorithm for time series segmentation based on reducing the number of points of the time series by minimising the approximation error of the linear interpolation of each segment. The contributions include the adaptation of the particle swarm optimisation algorithm (PSO) and its exploiter barebones variant (BBBePSO) for time series segmentation, along with the improvement of them using a dynamic adaptation of the exploration and exploitation importances (dynamic BBBePSO, DBBePSO). All algorithms are hybridised with a local search which combines the Bottom-Up and Top-Down strategies. The proposed method is then compared with other state-of-the-art algorithms: a genetic algorithm (GA), a standard particle swarm optimisation (PSO), the exploiting barebones PSO (BBBePSO), all their hybrid versions, the traditional Top-Down and Bottom-Up procedures, and Salotti's optimal algorithm.

The results conclude that the hybrid versions (HGA, HPSO, HBBBePSO, HDBBePSO) improve the solutions obtained by their standard versions (GA, PSO, BBBePSO, DBBePSO), showing that the hybridisation proposed is suitable for this type of problems. Salotti's algorithm is the best method in terms of RMSE, but the computational cost is much higher than that of the rest of algorithms. Furthermore, without considering Salotti's method, HDBBePSO results in the best results, obtaining the lowest approximation error, where the differences are found to be statistically significant. These results conclude that the dynamic adaptation of the BBBePSO allows the algorithm to escape the initial local optima and converge to optimal solutions at the end of the evolution. The algorithm proposed is statistically lower than traditional approaches (Top-Down and Bottom-Up), but their solutions are much worse.

For a future line of work, other distributions instead the Gaussian distribution could be taken into account, for instance, the Weibull distribution. We also plan to extend this work using the original and the approximated time series in posterior tasks, such as clustering or classification, observing if the method reduces the noise of the time series. Moreover, linear regression

Algorithm	Salotti	Bottom-Up	Top-Down	GA	HGA	PSO	HPSO	BBcPSO	HBbPSO	DBbPSO	HDBbPSO
Hand Outlines	<b>0.004</b>	0.005	0.006	0.023 $\pm$ 0.003	0.005 $\pm$ 0.000	0.036 $\pm$ 0.014	0.005 $\pm$ 0.000	0.010 $\pm$ 0.001	0.005 $\pm$ 0.000	0.007 $\pm$ 0.000	<b>0.004 <math>\pm</math> 0.000</b>
Mallat	<b>0.072</b>	0.097	0.502	0.305 $\pm$ 0.016	0.111 $\pm$ 0.004	0.345 $\pm$ 0.056	0.110 $\pm$ 0.004	0.246 $\pm$ 0.016	0.106 $\pm$ 0.003	0.203 $\pm$ 0.010	0.104 $\pm$ 0.003
Phoneme	<b>0.746</b>	1.057	0.940	0.957 $\pm$ 0.011	0.857 $\pm$ 0.005	1.132 $\pm$ 0.024	0.859 $\pm$ 0.005	1.042 $\pm$ 0.027	0.859 $\pm$ 0.005	1.019 $\pm$ 0.025	0.858 $\pm$ 0.005
StarLightCurves	<b>0.011</b>	0.016	0.026	0.054 $\pm$ 0.004	0.017 $\pm$ 0.000	0.081 $\pm$ 0.029	0.017 $\pm$ 0.000	0.037 $\pm$ 0.002	0.017 $\pm$ 0.000	0.030 $\pm$ 0.002	0.017 $\pm$ 0.000
Donoho-Johnstone	<b>2.218</b>	2.639	3.466	2.961 $\pm$ 0.070	2.414 $\pm$ 0.033	3.545 $\pm$ 0.236	2.431 $\pm$ 0.035	3.030 $\pm$ 0.078	2.431 $\pm$ 0.035	2.896 $\pm$ 0.081	2.425 $\pm$ 0.035
IBEX	<b>149.962</b>	210.321	269.801	261.067 $\pm$ 8.732	180.941 $\pm$ 1.836	321.976 $\pm$ 29.118	181.408 $\pm$ 1.772	264.590 $\pm$ 11.581	180.431 $\pm$ 1.645	229.682 $\pm$ 9.437	179.365 $\pm$ 1.928
BBVA	<b>0.236</b>	0.320	0.405	0.431 $\pm$ 0.009	0.286 $\pm$ 0.003	0.464 $\pm$ 0.039	0.286 $\pm$ 0.003	0.404 $\pm$ 0.012	0.285 $\pm$ 0.003	0.356 $\pm$ 0.010	0.282 $\pm$ 0.003
DEUTSCHE	<b>1.421</b>	2.032	2.318	2.428 $\pm$ 0.093	1.673 $\pm$ 0.019	2.899 $\pm$ 0.227	1.677 $\pm$ 0.021	2.428 $\pm$ 0.112	1.675 $\pm$ 0.020	2.105 $\pm$ 0.087	1.658 $\pm$ 0.018
SAN PAOLO	<b>0.080</b>	0.112	0.136	0.154 $\pm$ 0.003	0.097 $\pm$ 0.001	0.163 $\pm$ 0.018	0.097 $\pm$ 0.001	0.138 $\pm$ 0.005	0.097 $\pm$ 0.001	0.120 $\pm$ 0.005	0.096 $\pm$ 0.001
SO Générale	<b>1.598</b>	2.292	2.472	2.663 $\pm$ 0.085	1.902 $\pm$ 0.020	3.168 $\pm$ 0.296	1.905 $\pm$ 0.021	2.708 $\pm$ 0.087	1.898 $\pm$ 0.022	2.378 $\pm$ 0.087	1.882 $\pm$ 0.023
B46001	<b>0.799</b>	1.088	1.011	1.137 $\pm$ 0.010	0.931 $\pm$ 0.005	1.261 $\pm$ 0.019	0.931 $\pm$ 0.005	1.166 $\pm$ 0.012	0.931 $\pm$ 0.005	1.117 $\pm$ 0.026	0.927 $\pm$ 0.005
B46075	<b>0.822</b>	1.145	1.056	1.182 $\pm$ 0.011	0.978 $\pm$ 0.007	1.334 $\pm$ 0.019	0.978 $\pm$ 0.007	1.224 $\pm$ 0.023	0.978 $\pm$ 0.007	1.191 $\pm$ 0.025	0.975 $\pm$ 0.007
B41043	<b>0.295</b>	0.426	0.449	0.478 $\pm$ 0.006	0.360 $\pm$ 0.004	0.528 $\pm$ 0.009	0.360 $\pm$ 0.004	0.483 $\pm$ 0.010	0.359 $\pm$ 0.004	0.451 $\pm$ 0.013	0.356 $\pm$ 0.004
B41044	<b>0.292</b>	0.419	0.425	0.476 $\pm$ 0.008	0.351 $\pm$ 0.003	0.531 $\pm$ 0.011	0.351 $\pm$ 0.003	0.485 $\pm$ 0.009	0.351 $\pm$ 0.003	0.453 $\pm$ 0.014	0.348 $\pm$ 0.003
Arhythmia	<b>0.022</b>	0.032	0.091	0.100 $\pm$ 0.004	0.038 $\pm$ 0.001	0.114 $\pm$ 0.008	0.038 $\pm$ 0.001	0.078 $\pm$ 0.004	0.037 $\pm$ 0.001	0.064 $\pm$ 0.003	0.037 $\pm$ 0.001
Mean rankings ( $\bar{r}$ )	<b>1.03</b>	5.50	7.93	9.03	4.03	10.93	4.57	9.17	3.83	7.53	2.43

Table 1: RMSE values obtained by all the algorithms in each time series. Salotti, Top-Down, and Bottom-Up are deterministic and they are run once, while GA, HGA, PSO, HPSO, BBcPSO, HBbPSO, DBbPSO, and HDBbPSO are run 30 times with different seeds due to their stochastic nature (Mean  $\pm$  Standard deviation). The mean rankings of all algorithms are also included.

Algorithm	Salotti	Bottom-Up	Top-Down	GA	HGA	PSO	HPSO	BBePSO	HBBePSO	DBBePSO	HDBBePSO
Hand Outlines	594.84	9.97	<b>9.61</b>	41.94 ± 2.25	58.06 ± 1.73	13.64 ± 0.78	24.26 ± 0.94	39.33 ± 3.89	51.32 ± 2.35	41.41 ± 1.347	52.39 ± 3.94
Mallat	556.55	22.40	<b>22.01</b>	55.53 ± 0.57	71.03 ± 0.94	25.26 ± 1.26	36.53 ± 1.29	53.77 ± 5.62	63.96 ± 2.80	56.01 ± 3.950	63.83 ± 0.45
Phoneme	720.58	21.48	<b>21.14</b>	56.97 ± 0.70	70.96 ± 0.90	25.16 ± 0.76	35.65 ± 0.67	55.61 ± 3.18	63.65 ± 3.02	55.20 ± 0.630	62.95 ± 0.60
StarLightCurves	680.64	<b>20.77</b>	21.86	55.76 ± 1.45	69.85 ± 2.25	26.65 ± 1.17	35.42 ± 0.81	56.20 ± 7.66	62.87 ± 0.62	54.35 ± 2.175	65.02 ± 3.16
Donoho-Johnstone	9.71	<b>1.08</b>	1.16	4.13 ± 0.21	6.22 ± 0.18	1.47 ± 0.16	3.45 ± 0.14	2.87 ± 0.10	4.96 ± 0.37	3.051 ± 0.271	4.95 ± 0.15
IBEX	205.40	<b>4.37</b>	4.55	22.17 ± 0.39	30.64 ± 0.90	6.87 ± 0.17	13.60 ± 0.50	19.02 ± 0.08	27.35 ± 2.29	20.88 ± 1.427	27.80 ± 2.41
BBVA	75.24	<b>2.32</b>	2.44	12.21 ± 0.21	17.94 ± 0.73	3.55 ± 0.04	8.18 ± 0.12	11.14 ± 1.42	15.52 ± 1.20	10.68 ± 0.295	15.32 ± 0.84
DEUTSCHE	72.39	<b>2.31</b>	2.77	12.11 ± 0.24	17.19 ± 0.19	3.59 ± 0.08	8.11 ± 0.08	11.06 ± 1.39	15.21 ± 1.28	10.91 ± 0.802	14.78 ± 0.28
SAN PAOLO	69.69	<b>2.31</b>	2.59	12.18 ± 0.30	17.10 ± 0.10	3.61 ± 0.11	8.19 ± 0.08	10.16 ± 0.74	14.44 ± 0.09	11.05 ± 0.785	14.63 ± 0.09
SO Généralé	73.62	<b>2.31</b>	2.65	12.12 ± 0.20	17.12 ± 0.10	3.60 ± 0.09	8.14 ± 0.09	9.93 ± 0.64	14.43 ± 0.07	11.01 ± 0.781	14.63 ± 0.11
B46001	807.24	<b>10.64</b>	12.01	48.99 ± 1.10	63.07 ± 0.80	16.95 ± 0.49	27.66 ± 0.53	51.52 ± 4.50	55.88 ± 0.64	48.59 ± 2.472	57.65 ± 3.22
B46075	520.19	<b>7.18</b>	7.78	33.75 ± 0.25	45.05 ± 0.38	11.37 ± 0.34	19.52 ± 0.44	30.70 ± 3.00	40.66 ± 2.05	33.76 ± 2.959	40.17 ± 0.21
B41043	423.73	<b>7.19</b>	8.11	33.88 ± 0.19	44.96 ± 0.36	11.28 ± 0.30	19.32 ± 0.37	37.12 ± 0.07	41.48 ± 3.66	33.48 ± 2.663	39.96 ± 0.20
B41044	478.03	<b>7.40</b>	8.58	33.86 ± 0.60	42.15 ± 1.52	11.22 ± 0.18	19.43 ± 0.68	33.77 ± 3.96	43.31 ± 4.60	36.21 ± 4.428	40.75 ± 2.56
Arrhythmia	714.73	<b>11.32</b>	11.98	51.41 ± 1.03	60.38 ± 0.48	18.19 ± 0.47	28.76 ± 1.27	53.29 ± 5.29	61.69 ± 8.03	49.98 ± 3.301	59.70 ± 4.93
Mean rankings ( $\bar{r}$ )	11.00	<b>1.20</b>	1.80	6.53	9.87	3.00	4.13	5.67	8.73	5.67	8.40

Table 2: Computational time in seconds obtained by all the algorithms in each time series. Salotti, Top-Down, and Bottom-Up are deterministic and they are run once, while GA, HGA, PSO, HPSO, BBePSO, HBBePSO, DBBePSO, and HDBBePSO are run 30 times with different seeds due to their stochastic nature (Mean ± Standard deviation). The mean rankings of all algorithms are also included.

CA:HDBBePSO		RMSE	
i	$\alpha_{0.05}^*$	Algorithm	$p_i$
1	0.010	Top-Down	0.000 (*)
2	0.013	Bottom-Up	0.000 (*)
3	0.017	HPSO	0.002 (*)
4	0.025	HGA	0.021 (*)
5	0.050	HBBBePSO	0.045 (*)

Table 3: Results of the Holm test using HDBBePSO as control algorithm (CA) when comparing its average RMSE to those of Top-Down, Bottom-Up, HGA, HPSO, and HBBBePSO: corrected  $\alpha$  values, compared methods and  $p$ -values, all of them ordered by the number of comparison (i). CA results statistically better than the compared algorithm are marked with (\*).

CA:HDBBePSO		Run time (s)	
i	$\alpha_{0.05}^*$	Algorithm	$p_i$
1	0.010	Bottom-Up	0.000 (-)
2	0.013	Top-Down	0.000 (-)
3	0.017	HGA	0.032
4	0.025	HPSO	0.040
5	0.050	HBBBePSO	0.626

Table 4: Results of the Holm test using HDBBePSO as control algorithm (CA) when comparing its average runtime to those of Top-Down, Bottom-Up, HGA, HPSO, and HBBBePSO: corrected  $\alpha$  values, compared methods and  $p$ -values, all of them ordered by the number of comparison (i). CA results statistically worse than the compared algorithm are marked with (-).

could be also considered instead of linear interpolation, or even using polynomials with degree greater than one.

## Acknowledgement

This work has been subsidized by the projects TIN2017-85887-C2-1-P, TIN2014-54583-C2-1-R and TIN2015-70308-REDT of the Spanish Ministry of Economy and Competitiveness (MINECO), and FEDER funds (FEDER EU). Antonio M. Durán-Rosal's research has been subsidized by the FPU Pre-doctoral Program of the Spanish Ministry of Education, Culture and Sport (MECD), grant reference FPU14/03039.

## References

- [1] P. Esling, C. Agon, Time-series data mining, *ACM Computing Surveys (CSUR)* 45 (1) (2012) 12.
- [2] C. H. Fontes, H. Budman, A hybrid clustering approach for multivariate time series—a case study applied to failure analysis in a gas turbine, *ISA transactions* 71 (2017) 513–529.
- [3] M. Pérez-Ortiz, A. Durán-Rosal, P. Gutiérrez, J. Sánchez-Monedero, A. Nikolaou, F. Fernández-Navarro, C. Hervás-Martínez, On the use of evolutionary time series analysis for segmenting paleoclimate data, *Neurocomputing*. Available online 18 September 2017. doi:https://doi.org/10.1016/j.neucom.2016.11.101.
- [4] W. Deng, G. Wang, A novel water quality data analysis framework based on time-series data mining, *Journal of Environmental Management* 196 (2017) 365–375.
- [5] X. Gong, Y.-W. Si, S. Fong, R. P. Biuk-Aghai, Financial time series pattern matching with extended ucr suite and support vector machine, *Expert Systems with Applications* 55 (2016) 284–296.
- [6] T. Guyet, H. Nicolas, Long term analysis of time series of satellite images, *Pattern Recognition Letters* 70 (2016) 17–23.
- [7] A. Bagnall, J. Lines, J. Hills, A. Bostrom, Time-series classification with COTE: the collective of transformation-based ensembles, *IEEE Transactions on Knowledge and Data Engineering* 27 (9) (2015) 2522–2535.
- [8] J. Zhao, L. Itti, Classifying time series using local descriptors with hybrid sampling, *IEEE Transactions on Knowledge and Data Engineering* 28 (3) (2016) 623–637.
- [9] M.-Y. Chen, B.-T. Chen, A hybrid fuzzy time series model based on granular computing for stock price forecasting, *Information Sciences* 294 (2015) 227–241.
- [10] B. Sun, H. Guo, H. R. Karimi, Y. Ge, S. Xiong, Prediction of stock index futures prices based on fuzzy sets and multivariate fuzzy time series, *Neurocomputing* 151 (2015) 1528–1536.
- [11] A. Nikolaou, P. A. Gutiérrez, A. Durán, I. Dicaire, F. Fernández-Navarro, C. Hervás-Martínez, Detection of early warning signals in paleoclimate data using a genetic time series segmentation algorithm, *Climate Dynamics* 44 (7–8) (2015) 1919–1933.
- [12] L. N. Ferreira, L. Zhao, Time series clustering via community detection in networks, *Information Sciences* 326 (2016) 227–242.
- [13] X. Wang, A. Mueen, H. Ding, G. Trajcevski, P. Scheuermann, E. Keogh, Experimental comparison of representation methods and distance measures for time series data, *Data Mining and Knowledge Discovery* 26 (2) (2013) 275–309.
- [14] H. Kaya, Ş. Gündüz-Öğüdücü, A distance based time series classification framework, *Information Systems* 51 (2015) 27–42.
- [15] A. M. Durán-Rosal, P. A. Gutiérrez, S. Salcedo-Sanz, C. Hervás-Martínez, A statistically-driven coral reef optimization algorithm for optimal size reduction of time series, *Applied Soft Computing* 63 (2018) 139–153.
- [16] F.-L. Chung, T.-C. Fu, V. Ng, R. W. Luk, An evolutionary approach to pattern-based time series segmentation, *Evolutionary Computation, IEEE Transactions on* 8 (5) (2004) 471–489.
- [17] V. S. Tseng, C.-H. Chen, P.-C. Huang, T.-P. Hong, Cluster-based genetic segmentation of time series with dwt, *Pattern Recognition Letters* 30 (13) (2009) 1190–1197.
- [18] J. Abonyi, B. Feil, S. Nemeth, P. Arva, Modified gath–geva clustering for fuzzy segmentation of multivariate time-series, *Fuzzy Sets and Systems* 149 (1) (2005) 39–56.
- [19] Y. Gorshkov, I. Kokshenev, Y. Bodyanskiy, V. Kolodyazhnyi, O. Shylo, Robust recursive fuzzy clustering-based segmentation of biological time series, in: *Evolving Fuzzy Systems, 2006 International Symposium on*, IEEE, 2006, pp. 101–105.
- [20] E. Fuchs, T. Gruber, J. Nitschke, B. Sick, On-line motif detection in time series with swiftmotif, *Pattern Recognition* 42 (11) (2009) 3015 – 3031.
- [21] A. M. Durán-Rosal, M. de la Paz-Marín, P. A. Gutiérrez, C. Hervás-Martínez, Identifying market behaviours using european stock index time series by a hybrid segmentation algorithm, *Neural Processing Letters* (2017) 1–24.
- [22] A. Durán-Rosal, J. Fernández, P. Gutiérrez, C. Hervás-Martínez, Detection and prediction of segments containing extreme significant wave heights, *Ocean Engineering* 142 (2017) 268–279.
- [23] S. Aghabozorgi, A. S. Shirkhorshidi, T. Y. Wah, Time-series clustering—a decade review, *Information Systems* 53 (2015) 16–38.
- [24] S. Rani, G. Sikka, Recent techniques of clustering of time series data: a survey, *International Journal of Computer Applications* 52 (15).
- [25] J. Oliver, C. Forbes, Bayesian approaches to segmenting a simple time series, *Tech. Rep. 14/97*, Monash University, Department of Econometrics and Business Statistics (1997). URL <http://EconPapers.repec.org/RePEc:msh:ebswps:1997-14>
- [26] J. J. Oliver, R. A. Baxter, C. S. Wallace, Minimum message length segmentation, in: X. Wu, R. Kotagiri, K. Korb (Eds.), *Research and Development in Knowledge Discovery and Data Mining*, Vol. 1394 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, 1998, pp. 222–233.
- [27] E. J. Keogh, S. Chu, D. Hart, M. Pazzani, Segmenting Time Series: A Survey and Novel Approach, in: M. Last, A. Kandel, H. Bunke (Eds.), *Data Mining In Time Series Databases*, Vol. 57 of *Series in Machine Perception and Artificial Intelligence*, World Scientific Publishing Company, 2004, Ch. 1, pp. 1–22.

- [28] K. Chakrabarti, E. Keogh, S. Mehrotra, M. Pazzani, Locally adaptive dimensionality reduction for indexing large time series databases, *ACM Transactions on Database Systems (TODS)* 27 (2) (2002) 188–228.
- [29] J. Lin, E. Keogh, S. Lonardi, B. Chiu, A symbolic representation of time series, with implications for streaming algorithms, in: *Proceedings of the 8th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery*, ACM, 2003, pp. 2–11.
- [30] J. Kennedy, R. Eberhart, Particle swarm optimization, in: *Neural Networks, 1995. Proceedings., IEEE International Conference on*, Vol. 4, 1995, pp. 1942–1948.
- [31] J. Kennedy, Bare bones particle swarms, in: *Swarm Intelligence Symposium, 2003. SIS '03. Proceedings of the 2003 IEEE*, 2003, pp. 80–87.
- [32] K. Chau, Particle swarm optimization training algorithm for anns in stage prediction of shing mun river, *Journal of Hydrology* 329 (3) (2006) 363 – 367.
- [33] M. Zhang, M. Xin, J. Yang, Adaptive multi-cue based particle swarm optimization guided particle filter tracking in infrared videos, *Neurocomputing* 122 (Supplement C) (2013) 163 – 171, advances in cognitive and ubiquitous computing.
- [34] D. Tungadio, B. Numbi, M. Siti, A. Jimoh, Particle swarm optimization for power system state estimation, *Neurocomputing* 148 (Supplement C) (2015) 175 – 180.
- [35] C. R. Houck, J. A. Joines, M. G. Kay, J. R. Wilson, Empirical investigation of the benefits of partial lamarckianism, *Evol. Comput.* 5 (1) (1997) 31–60.
- [36] N. L. J. Ulder, E. H. L. Aarts, H.-J. Bandelt, P. J. M. v. Laarhoven, E. Pesch, Genetic local search algorithms for the travelling salesman problem, in: *Proceedings of the 1st Workshop on Parallel Problem Solving from Nature, PPSN I*, Springer-Verlag, London, UK, UK, 1991, pp. 109–116.
- [37] A. Kolen, E. Pesch, Genetic local search in combinatorial optimization, *Discrete Applied Mathematics* 48 (3) (1994) 273 – 284.
- [38] J. A. Joines, M. G. Kay, Utilizing hybrid genetic algorithms, in: *Evolutionary Optimization*, Vol. 48 of *International Series in Operations Research & Management Science*, Springer US, 2002, pp. 199–228.
- [39] M. Clerc, J. Kennedy, The particle swarm-explosion, stability, and convergence in a multidimensional complex space, *IEEE transactions on Evolutionary Computation* 6 (1) (2002) 58–73.
- [40] M. G. Omran, A. Engelbrecht, A. Salman, Barebones particle swarm for integer programming problems, in: *IEEE Swarm Intelligence Symposium., 2007*, pp. 170–175.
- [41] V. K. Koumoussis, C. P. Katsaras, A saw-tooth genetic algorithm combining the effects of variable population size and reinitialization to enhance performance, *IEEE Transactions on Evolutionary Computation* 10 (1) (2006) 19–28.
- [42] Y. Chen, E. Keogh, B. Hu, N. Begum, A. Bagnall, A. Mueen, G. Batista, The ucr time series classification archive, [www.cs.ucr.edu/~eamonn/time\\_series\\_data/](http://www.cs.ucr.edu/~eamonn/time_series_data/) (July 2015).
- [43] D. L. Donoho, I. M. Johnstone, Ideal spatial adaptation by wavelet shrinkage, *Biometrika* 81 (3) (1994) 425–455.
- [44] D. L. Donoho, I. M. Johnstone, G. Kerkycharian, D. Picard, Wavelet shrinkage: Asymptopia?, *Journal of the Royal Statistical Society. Series B (Methodological)* 57 (2) (1995) 301–369.
- [45] D. L. Donoho, I. M. Johnstone, Adapting to unknown smoothness via wavelet shrinkage, *Journal of the American Statistical Association* 90 (1995) 1200–1224.
- [46] <http://www.ndbc.noaa.gov/>, National buoy data center, National Oceanic and Atmospheric Administration of the USA (NOAA), 2015.
- [47] G. Moody, R. Mark, The impact of the MIT-BIH arrhythmia database, *Engineering in Medicine and Biology Magazine*, IEEE 20 (3) (2001) 45–50.
- [48] A. L. Goldberger, L. A. N. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C.-K. Peng, H. E. Stanley, Physiobank, physiotoolkit, and physionet: Components of a new research resource for complex physiologic signals, *Circulation* 101 (23) (2000) e215–e220.
- [49] M. Salotti, An efficient algorithm for the optimal polygonal approximation of digitized curves, *Pattern Recognition Letters* 22 (2) (2001) 215–221.
- [50] A. Carmona-Poyato, E. Aguilera-Aguilera, F. Madrid-Cuevas, M. Marín-Jiménez, N. Fernández-García, New method for obtaining optimal polygonal approximations to solve the min- $\epsilon$  problem, *Neural Computing and Applications* 28 (9) (2017) 2383–2394.
- [51] A. Engelbrecht, Particle swarm optimization: Velocity initialization, in: *Evolutionary Computation (CEC), 2012 IEEE Congress on*, IEEE, 2012, pp. 1–8.
- [52] A. M. Durán-Rosal, P. A. Gutiérrez-Peña, F. J. Martínez-Estudillo, C. Hervás-Martínez, Time series representation by a novel hybrid segmentation algorithm, in: *11th International Conference on Hybrid Artificial Intelligence Systems*, Springer, 2016, pp. 163–173.
- [53] M. Friedman, A comparison of alternative tests of significance for the problem of m rankings, *The Annals of Mathematical Statistics* 11 (1) (1940) 86–92.
- [54] J. Demšar, Statistical comparisons of classifiers over multiple data sets, *Journal of Machine Learning Research* 7 (Jan) (2006) 1–30.

### 4.3. Multiobjective time series segmentation

#### Main publication associated to this section:

- **A. M. Durán-Rosal**, P. A. Gutiérrez, F. J. Martínez-Estudillo, and C. Hervás-Martínez. “Simultaneous optimisation of clustering quality and approximation error for time series segmentation”, *Information Sciences*, Vol. 442-443, May, 2018, pp. 186-201. JCR(2017): 4.305 Position: 12/148 (Q1). DOI: 10.1016/j.ins.2018.02.041

#### Other publication associated to this section:

- **A. M. Durán-Rosal**, P. A. Gutiérrez, F. J. Martínez-Estudillo, and C. Hervás-Martínez. “Multiobjective time series segmentation by improving clustering quality and reducing approximation error”. XII Congreso Español de Metaheurísticas, Algoritmos Evolutivos y Bioinspirados (MAEB 2017). 2017. pp. 920-922.  
URL: <http://mic2017.upf.edu/proceedings/>

These works explore the optimisation of both the clustering quality and the error approximation considering a novel multiobjective algorithm for segmenting time series.

#### 4.3.1. Simultaneous optimisation of clustering quality and approximation error for time series segmentation

Previous works of time series segmentation are aimed to optimise only one objective, i.e. the segment clustering quality for discovering useful patterns or the reduction of the number of points of the time series to simplify them. Our main hypothesis is that both objectives are conflicting, that is, the optimisation of one harms the other. Up to the author knowledge, there are no previous works optimising both objectives in the same algorithm.

In this work, we tackle the problem of the optimisation of both objectives by developing a novel evolutionary multiobjective time series segmentation algorithm called GMOTSS. The algorithm is based on the NSGA-II, which was previously proposed to solve MOPs. NSGA-II has been modified and adapted taking into account all the considerations for time series segmentation. In this way, the GMOTSS algorithm is specifically designed to find the cut points of the segmentation, taking into account both objectives. Then, the user can choose the most appropriate solution from a Pareto front of different segmentations.

The algorithm is endowed with nine clustering validity indexes and an error fitness function. Each clustering quality index is combined in the algorithm with the error fitness, and an experimental validation is used to find the best clustering index based on Pareto

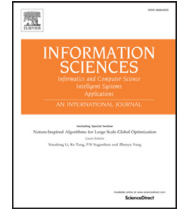
front evaluation metrics. After that, the algorithm is tested in a synthetic time series and four datasets with different scope and length.

The results obtained by GMOTSS are compared against other state-of-the-art methods, such as a mono-objective version of the algorithm using a linear combination of the objectives as fitness function, the algorithm proposed in Section 4.1.4, and the Growing Window, Bottom-Up and SWAB segmentation algorithms for reducing the number of points. The resulting segmentations are a good approximation of the original time series, but they also show a good level of similarity according to the groups discovered by the clustering. The main hypothesis of the work is also corroborated because it is clearly shown that both objectives are conflicting. Finally, the proposed algorithm shows a good trade-off for both objectives when compared with the previously cited algorithms.



Contents lists available at ScienceDirect

Information Sciences

journal homepage: [www.elsevier.com/locate/ins](http://www.elsevier.com/locate/ins)

# Simultaneous optimisation of clustering quality and approximation error for time series segmentation



Antonio Manuel Durán-Rosal<sup>a,\*</sup>, Pedro Antonio Gutiérrez<sup>a</sup>,  
Francisco José Martínez-Estudillo<sup>b</sup>, César Hervas-Martínez<sup>a</sup>

<sup>a</sup> Department of Computer Science and Numerical Analysis, University of Córdoba, Rabanales Campus, Albert Einstein building, Córdoba 14071, Spain

<sup>b</sup> Department of Quantitative Methods, Universidad Loyola Andalucía, Escritor Castilla Aguayo 4, Córdoba 14004, Spain

## ARTICLE INFO

### Article history:

Received 19 October 2016

Revised 9 January 2018

Accepted 17 February 2018

Available online 21 February 2018

### Keywords:

Time series segmentation

Multiobjective optimisation

Clustering

Evolutionary computation

## ABSTRACT

Time series segmentation is aimed at representing a time series by using a set of segments. Some researchers perform segmentation by approximating each segment with a simple model (e.g. a linear interpolation), while others focus their efforts on obtaining homogeneous groups of segments, so that common patterns or behaviours can be detected. The main hypothesis of this paper is that both objectives are conflicting, so time series segmentation is proposed to be tackled from a multiobjective perspective, where both objectives are simultaneously considered, and the expert can choose the desired solution from a Pareto Front of different segmentations. A specific multiobjective evolutionary algorithm is designed for the purpose of deciding the cut points of the segments, integrating a clustering algorithm for fitness evaluation. The experimental validation of the methodology includes three synthetic time series and three time series from real-world problems. Nine clustering quality assessment metrics are experimentally compared to decide the most suitable one for the algorithm. The proposed algorithm shows good performance for both clustering quality and reconstruction error, improving the results of other mono-objective alternatives of the state-of-the-art and showing better results than a simple weighted linear combination of both corresponding fitness functions.

© 2018 Elsevier Inc. All rights reserved.

## 1. Introduction

Time series are an important class of temporal data objects collected chronologically. The corresponding databases are often large, high in dimensionality and require continuous updating. Thus, their intrinsic characteristics make them difficult to analyse. In this context, dimensionality reduction, similarity measurement, segmentation, visualisation and mining methods (such as hidden pattern discovery, clustering, classification or rule discovery) are part of time series research [16,25,35].

The segmentation task aims at creating an accurate approximation of the time series, by reducing its dimensionality while retaining the essential features. The objective of this task is to minimise the reconstruction error of a reduced representation with respect to the original time series. Segmentation tasks do not only reduce storage space but also increase the performance of data mining techniques. According to the literature review, current time series compression techniques require expert understanding of the time series, and appropriate threshold values need to be adjusted in order to reduce

\* Corresponding author.

E-mail address: [i92duroa@uco.es](mailto:i92duroa@uco.es) (A.M. Durán-Rosal).



*Study the past if you would define the future.*

Confucius

# 5

## Prediction

This chapter includes some contributions of the Thesis related to the topic of prediction in time series, including a two-stage algorithm for the detection and prediction of extreme events in wave height time series and for the problem of the fog prediction in the airport of Valladolid.

### **Main publications associated to this chapter:**

- **A. M. Durán-Rosal**, J. C. Fernández, P. A. Gutiérrez, and C. Hervás-Martínez. “Detection and prediction of segments containing extreme significant wave heights”, *Ocean Engineering*, Vol. 142, September, 2017, pp. 268-279. JCR(2017): 2.214 Position: 2/14 (Q1). DOI: 10.1016/j.oceaneng.2017.07.009
- **A. M. Durán-Rosal**, J. C. Fernandez, C. Casanova-Mateo, J. Sanz-Justo, S. Salcedo-Sanz, and C. Hervás-Martínez. “Efficient Fog Prediction with Multi-objective Evolutionary Neural Networks”, *Applied Soft Computing*, Vol. 70, September, 2018, pp. 347-358. JCR(2017): 3.907 Position: 17/132 (Q1). DOI: 10.1016/j.asoc.2018.05.035

### **Other publications associated to this chapter:**

- M. Dorado-Moreno, **A. M. Durán-Rosal**, D. Guijo-Rubio, P. A. Gutiérrez, L. Prieto, S. Salcedo-Sanz, and C. Hervás-Martínez. “Multiclass prediction of wind power ramp events combining reservoir computing and support vector machines”. *Conferencia*

de la Asociación Española para la Inteligencia Artificial (CAEPIA 2016). 2016. pp. 300-309. DOI: 10.1007/978-3-319-44636-3\_28

- **A. M. Durán-Rosal**, J. C. Fernández, P. A. Gutiérrez, and C. Hervás-Martínez. “Hybridization of neural network models for the prediction of extreme significant wave height segments”. 2016 IEEE Symposium Series on Computational Intelligence (IEEE SSCI2016). 2016. pp. 1-8. DOI: 10.1109/SSCI.2016.7850144

As stated before, the prediction in time series is usually accomplished by considering standard statistical procedures, such as AR models and their variants. We propose to transform the prediction problems into ML classification tasks. For example, for the prediction of extreme wave height, we develop a two-stage algorithm, where the first part is the detection of extreme wave height (in a similar manner to the algorithm shown in Section 4.1.2) and the second part consists in an MOEA for training ANN models. Also, for the prediction of fog events in airports, we manually construct a database for a 6-hour resolution prediction, using multiobjective algorithms. As can be seen, in both, we use multiobjective optimisation, with the aim to optimise the global accuracy and the accuracy of the worst classified class, given the important imbalanced nature of the datasets. The two main publications are now presented in the different sections of this chapter.

### 5.1. Detection and prediction of segments containing extreme significant wave heights

The detection and prediction of extreme events in SWH time series is an essential challenge for oceanographic purposes, e.g. for long-term future operational environment of marine and coastal structures. The following paper presents a methodology which is organised in two well-defined stages: detection and prediction of those periods which contain wave heights which are very large with respect to other closer in time (segments containing very high SWH, SSWH).

Firstly, the methodology is based on an HA (a combination of a GA and an LS based on a likelihood ratio-test) with the purpose of detecting and defining the periods of time corresponding to those events. The detection consists in finding time series subsequences which present similar behaviour with the objective to determine a clustering able to group together these extreme events. For that, this first stage includes a modified deterministic  $k$ -means algorithm. This first stage allows us to study the nature of SSWH.

Secondly, once the detection (segmentation) is made, the algorithm automatically transforms the segmented time series in a sequence of labels, and, then, a database for the prediction stage is built. Each pattern of this dataset is formed by the characteristics of the

### 5.1. Detection and prediction of segments containing extreme significant wave heights 79

three previous segments (a total of 15 inputs), and the output is a binary label reflecting whether the next segment is an SSWH or not. We consider an MOEA for training ANNs given that the resulting dataset is imbalanced. That is, the number of SSWH events is much lower than the number of non SSWH events, in such a way that we can not only optimise the global accuracy, but the accuracy per the minority class also needs to be optimised.

The methodology is tested in two real-world time series of SWH collected in the Gulf of Alaska. This is compared against five state-of-the-art algorithms including LR, simple LR, SVM, and two DTs, C4.5 and RandomForest. Also, the cost-sensitive versions of these algorithms have been taken into account, which are able of considering the imbalanced distribution of the dataset.

Results confirm that our methodology can make a reasonable prediction of SSWH events without losing accuracy in the minority class. Although the cost-sensitive methods consider the imbalanced distribution of the classes, our methodology outperforms them. Also, the standard versions of the algorithms lead to the best global accuracy results, but they behave poorly in the prediction of the minority class, which corresponds with the SSWH events (i.e. the most important class).



# Detection and prediction of segments containing extreme significant wave heights



A.M. Durán-Rosal\*, J.C. Fernández, P.A. Gutiérrez, C. Hervás-Martínez

Department of Computer Science and Numerical Analysis, University of Córdoba, Rabanales Campus, Albert Einstein Building, 3rd Floor, 14071, Córdoba, Spain

## ARTICLE INFO

### Keywords:

Time series segmentation  
Multiobjective evolutionary algorithm  
Local search  
Prediction  
Detection  
Extreme significant wave height  
Minimum sensitivity

## ABSTRACT

This paper presents a methodology for the detection and prediction of Segments containing very high Significant Wave Height (SSWH) values in oceans. This kind of prediction is needed in order to account for potential changes in a long-term future operational environment of marine and coastal structures. The methodology firstly characterizes the wave height time series by approximating it using a sequence of labeled segments, and then a binary classifier is trained to predict the occurrence of SSWH periods based on past height values. A genetic algorithm (GA) combined with a likelihood-based local search is proposed for the first stage (detection), and the second stage (prediction) is tackled by an Artificial Neural Network (ANN) trained with a Multiobjective Evolutionary Algorithm (MOEA). Given the unbalanced nature of the dataset (SSWH is rarer than non SSWH), the MOEA is specifically designed to obtain a balance between global accuracy and individual sensitivities for both classes. The results obtained show that the GA is able to group SSWH in a specific cluster of segments and that the MOEA obtains ANN models able to perform an acceptable prediction of these SSWH.

## 1. Introduction

Large ocean waves pose significant risks to ships and offshore structures. The development of offshore installations for oil and gas extraction requires knowledge of the wave fields and any potential changes in them. Moreover, in order to accurately predict the long-term energy resource and performance of ocean wave energy converters, long-term prediction of extreme wave heights is particularly important. Additionally, high ocean waves represent significant risks in ship movements and port activity, and a reliable measurement of these extreme and critical events is crucial from the point of view of navigation and civil protection.

In recent years, different statistical and mathematical methods have been proposed for calculating and predicting Significant Wave Height (SWH) Mahjoobi et al. (2008), Mahjoobi and Mosabbe (2009). SWH can be defined either in the temporal domain or in the frequency domain. In the former case, it is noted  $H_{1/3}$  and is defined as the average height of the highest one-third of wave heights, measured from the time series of free surface by up or down-crossing. In the latter case, it is noted  $H_{m0}$  and is defined from the frequency spectrum. In deep water,  $H_{1/3}$  and  $H_{m0}$  are quite close (less than 5% of difference) and they are generally confused in the generic term  $H_s$ . For this reason, even if the definitions of wave height are formally expressed, it is advisable to

use the generic term  $H_s$  or simply SWH. According to the National Data Buoy Center (NDBC) and the National Oceanic and Atmospheric Administration (NOAA), SWH is the average trough to crest in meters of the highest one-third of all the wave heights during a 20-min sampling period (2016). NOAA uses hydrographic stations and ocean buoys with special sensors to collect data, and this paper uses this source of information. There are other statistical measures of the wave height, such as the Root Mean Square (RMS) wave height, which is defined as the squared root of the average of the squares of all wave heights and is approximately equal to SWH divided by 1.4 Holthuijsen (2007).

Recently, a more specific field, the determination and prediction of Extreme SWH (ESWH), has gained significant attention. In general, the previously proposed methods are based on considering the probability distributions of the Extreme Values (EV) of SWH. For example, the work of Muraleedharan et al. Muraleedharan et al. (2016) proposes the use of quantile regression to model the ESWH distribution, as an alternative to fitting EV distributions based on the tails of data samples. Another popular methodology is the Peaks Over Threshold (POT) Davison and Smith (1990) (i.e. considering only those values of the time series higher than a predefined threshold, that is, those values which are a sample of exceedances), which has been used as a standard approach for these predictions Caires and Sterl (2005), Viselli et al.

\* Corresponding author.

E-mail address: [i92duroa@uco.es](mailto:i92duroa@uco.es) (A.M. Durán-Rosal).

## 5.2. Efficient fog prediction with multi-objective evolutionary neural networks

As stated before, aviation is the transportation system most strongly impacted by the adverse weather conditions. Concerning airport operations, there are some factors which reduce visibility, and one of the most important ones is fog. This factor has a significant impact on safety and efficiency of airport-related operations, such as taxiing, take-off and landing.

In this work, we study, analyse and construct a model for fog prediction in the airport of Valladolid. Given the location of this airport, fog formation is commonly produced in the cold months of the year, resulting in the problems previously explained. Physical data is collected from different sensors situated in many parts of the airport, considering as the most important ones wind speed, wind direction, temperature, humidity and pressure. The RVR is a meteorological variable defined as the range over which the pilot of an aircraft on the central line of a runway can see the runway surface markings or the lights delineating the runway or identifying its centre line. RVR is a critical parameter for several operations, so our objective is to predict fog formation using the RVR as the decision variable with a 6-hour horizon with respect the data collected from the sensors.

In this way, we consider fog events when RVR is less than 1990 metres, so the problem is cast into a binary classification where the output is a 1 when fog is present ( $RVR < 1990m$ ), 0 otherwise. We use ANNs with SU, PU or RBF, to make a prediction model, and given the imbalanced nature of the dataset (the number of fog events is much lower), we decide to use an MOEA for optimising their parameters and structure. This algorithm takes into account the traditional global accuracy and the minimum sensitivity (the lowest percentage of patterns correctly classified as belonging to each class with respect to the total number of examples in the corresponding class).

The best model is obtained when using PUs as basis functions showing that a simple model with only two hidden neurons can predict this kind of events with a six-hour resolution. This model is also discussed through the importance of each variable, confirming that the model agrees with the physical properties of fog formation.



# Efficient fog prediction with multi-objective evolutionary neural networks

A.M. Durán-Rosal<sup>a,\*</sup>, J.C. Fernández<sup>a</sup>, C. Casanova-Mateo<sup>b,c</sup>, J. Sanz-Justo<sup>b</sup>,  
S. Salcedo-Sanz<sup>d</sup>, C. Hervás-Martínez<sup>a</sup>

<sup>a</sup> Department of Computer Science and Numerical Analysis, University of Córdoba, Córdoba, Spain

<sup>b</sup> LATUV: Remote Sensing Laboratory, University of Valladolid, Valladolid, Spain

<sup>c</sup> Department of Civil Engineering: Construction, Infrastructure and Transport, Polytechnic University of Madrid, Madrid, Spain

<sup>d</sup> Department of Signal Processing and Communications, University of Alcalá, Alcalá de Henares, Spain

## ARTICLE INFO

### Article history:

Received 3 October 2017

Received in revised form 8 March 2018

Accepted 22 May 2018

Available online 28 May 2018

### Keywords:

Multiobjective Evolutionary Algorithm

Neural networks

Fog events prediction

Time series

## ABSTRACT

This paper proposes the application of novel artificial neural networks with evolutionary training and different basic functions (sigmoidal, product and radial), for a real problem of fog events classification from meteorological input variables. Specifically, a Multiobjective Evolutionary Algorithm is considered as artificial neural network training mechanism in order to obtain a binary classification model for the detection of fog events at Valladolid airport (Spain). The evolutionary neural models developed are based on two-dimensional performance measures: traditional accuracy and the minimum sensitivity, as the lowest percentage of examples correctly predicted as belonging to each class with respect to the total number of examples in the corresponding class. These performance measures are directly related to features associated with any classifier: its global performance and the rate of the worst classified class. These two objectives are usually in conflict when the optimization process tries to construct models with a high classification rate level in the generalization dataset, and also with a good classification level for each class or minimum sensitivity. A sensitivity analysis of the proposed models is carried out, and thus the subjacent relations between the input variables and the output classification target can be better understood.

© 2018 Elsevier B.V. All rights reserved.

## 1. Introduction

Aviation is perhaps the transportation system most strongly impacted by adverse weather conditions. Severe weather phenomena affect almost all phases of flight, inducing air traffic flow disruptions, increasing the workload of air traffic controllers and pilots and eventually resulting in flight delays, diversions and cancellations. Regarding airport operations, there are some atmospheric phenomena, such as sandstorms, duststorms, snowfall or heavy rain that can degrade visibility. Nevertheless, fog formation is possibly the most important and frequent one [1]. Fog is a hydrometeor defined as a collection of liquid water droplets or ice crystals suspended in the air, forming a low level cloud with its base very close or in contact with the ground. Foggy conditions at airports can have a significant impact on both the safety and the

efficiency of airport-related operations (taxiing, take-off and landing). It has an increasing impact on both high-demand airports and local/regional airport networks [2]. Moreover, low visibility atmospheric conditions have been unfortunately a crucial factor in some historical accidents and incidents (e.g. on March 1977, fog was one of the determining factors that caused the worst accident in aviation history where two Boeing 747s collided at Los Rodeos airport – Tenerife, Spain). Among all the meteorological data collected at airports to support air navigation and airport operations in dealing with low-visibility atmospheric conditions, the Runway Visual Range (RVR) is the most significant one. The RVR is a meteorological variable defined as the range over which the pilot of an aircraft on the central line of a runway can see the runway surface markings or the lights delineating the runway or identifying its centre line [3]. RVR is a critical parameter for several operations. For example: low-visibility approach procedures are dependent on whether RVR falls within particular ranges: in fact, airports activate the so-called Low Visibility Procedures (LVP) when the RVR value is low enough so as to hamper safe operations [4]. Thus, an efficient methodology to predict whether or not RVR value will fall within low-visibility conditions is more than desirable.

\* Corresponding author at: Department of Computer Science and Numerical Analysis, University of Córdoba, Rabanales Campus, Albert Einstein Building, 3rd Floor, 14014, Córdoba, Spain.

E-mail address: [i92duroa@uco.es](mailto:i92duroa@uco.es) (A.M. Durán-Rosal).

*No human investigation can be called real science if it cannot be demonstrated mathematically.*

Leonardo da Vinci

# 6

## Statistical distribution-based learning

This chapter presents two works related to the determination of the statistical distribution of time series to guide posterior operations.

### **Main publications associated to this chapter:**

- **A. M. Durán-Rosal**, M. Carbonero, P. A. Gutiérrez, and C. Hervás-Martínez. “On the use of a mixed distribution to fix the threshold for Peak-Over-Threshold wave height estimation”, Coastal Engineering, 2019. JCR(2017): 2.674 Position: 21/128 (Q1). Under Review.
- D. Guijo-Rubio, **A. M. Durán-Rosal**, A. M. Gómez-Orellana, P. A. Gutiérrez, and C. Hervás-Martínez. “Distribution-based discretisation and ordinal classification applied to wave height prediction”. 19th International Conference on Intelligence Data Engineering and Automated Learning (IDEAL2018). 2018. pp 171-179. DOI: 10.1007/978-3-030-03496-2\_20

The first publication proposes a new theoretical distribution and the method to fit the parameters of this distribution in SWH time series, while the second proposed a new way to discretise the time series using the best-fitted statistical distribution selected using objective criteria, for the posterior ordinal classification of the segments produced by this discretisation. These proposals are now presented in the different sections of this chapter.

### 6.1. On the use of a mixed distribution to fix the threshold for peak-over-threshold wave height estimation

Modelling the distribution of SWH time series and, specifically, modelling extreme distributions, where the higher waves are much less frequent than the lower ones, is an open research studied by many authors, as we stated in the Introduction section. This has been tackled from the point of view of the POT methodologies, where modelling is based on those values higher than a threshold. The main problem is that the threshold is usually predefined by the user, and the rest of the values are ignored.

In this paper, we propose a new methodology (showing the proposal and the associated theoretical discussion) to estimate the distribution of the whole time series, that is, taking into account both extreme and normal values. This methodology starts with the main hypothesis that time series presenting extreme values can be modelled by a normal distribution in combination with a uniform one. In this way, the assumption consists in considering that extreme wave heights are normally distributed, and they are added as to standard values from a uniform distribution, considering the values from this distribution as part of the problem and never as noise. In the results, it is statistically shown that the distribution of this kind of time series can be adjusted by the proposed methodology.

Once the whole distribution is determined, it is used to fix the threshold for the POT approaches. For this, we use the percentiles 95 %, 97.5 % and 99 % as different values for the threshold, and then, the distribution of those values which are over these percentiles are adjusted using the EVT distributions: GPD, Gamma distribution and Weibull distribution. The best-fitted distribution is selected according to the values of AIC and BIC objective criteria.

The methodology is tested in nine real-world time series collected from buoys situated in the Gulf of Alaska (id numbers 46001 and 46075), in Puerto Rico (41043, 41044, 41046, 41047, 41048 and 41049), and in Spain (SIMAR-44). The methodology can fit the distribution of these time series, which is corroborated by a Kolmogorov-Smirnov test, and the best fit distribution of the extreme values situated over the threshold fixed by this methodology is the GPD.



# On the use of a mixed distribution to fix the threshold for Peak-Over-Threshold wave height estimation

Antonio M. Durán-Rosal<sup>a,\*</sup>, Mariano Carbonero<sup>b</sup>, Pedro Antonio Gutiérrez<sup>a</sup>, César Hervás-Martínez<sup>a</sup>

<sup>a</sup>Department of Computer Science and Numerical Analysis, University of Córdoba, Córdoba, Spain

<sup>b</sup>Department of Quantitative Methods, Universidad Loyola Andalucía, c/ Escritor Aguayo, 4, Córdoba, Spain

---

## Abstract

Modelling extreme values distributions, such as wave height time series where the higher waves are much less frequent than the lower ones, has been tackled from the point of view of the Peak-Over-Threshold (POT) methodologies, where modelling is based on those values higher than a threshold. This threshold is usually predefined by the user, while the rest of values are ignored. In this paper, we propose a new method to estimate the distribution of the complete time series, including both extreme and regular values. This methodology assumes that extreme values time series can be modelled by a normal distribution in a combination of a uniform one. The resulting theoretical distribution is then used to fix the threshold for the POT methodology. The methodology is tested in nine real-world time series collected in the Gulf of Alaska, Puerto Rico and Gibraltar (Spain), which are provided by the National Data Buoy Center (USA) and Puertos del Estado (Spain). By using the Kolmogorov-Smirnov statistical test, the results confirm that the time series can be modelled with this type of mixed distribution. Based on this, the return values and the confidence intervals for wave height in different periods of time are also calculated.

## Keywords:

Wave height time series, modelling mixed distribution, forecasting, method of moments

---

## 1. Introduction

Significant wave height forecasting is an important task for designing coastal and off-shore structures [1]. In this sense, the incorporation of wave models into numerical weather prediction models can improve atmospheric forecasts [2]. The development of offshore installations for oil and gas extraction and for renewable energy exploitation requires knowledge of the wave fields and any potential changes in them. One of the main problems is that the knowledge of the maximum peak-to-trough wave height is not usually available although largest waves have the greatest impact on ships and offshore structures [3].

The importance of time series data mining has been increasing exponentially in the last decade [4, 5]. They are present in different fields of application, e.g. climate [6], hydrology [7], GPU deep learning [8] and much more. In addition, they are used for different research objectives, such as classification [9], tipping point detection [10], forecasting [11], etc.

Basically, a time series can be defined as temporal data collected in different periods of time. In this sense, the observation of a random variable in regular periods of time can lead to the introduction of noise. That is, if the period between two consecutive observations is much lower than the real cadence of the

phenomenon under investigation, a high number of observed values will be very close to the average value of the characteristic studied.

In the context of oceanography and specifically, in the determination of extreme wave height values, if we consider a buoy collecting the wave height value every four hours, then a high proportion of values close to the average wave height will be recorded. This results in the fact that extreme wave heights, which are probably the most interesting ones, will be outnumbered by a set of very similar values without special interest. These non-informative observations have a distorting effect on the measures that could be taken to analyse the variable, because they do not significantly change the mean value but reduce the deviation, increasing the sample size.

Consequently, wave height extreme values will change from being more or less infrequent to atypical or outliers, with the drawbacks that this means for its analysis and prediction. The presence of these extreme values produces a denaturalization of the standard wave height probability distribution. For this reason, it is necessary to define thresholds of wave height from which the extreme wave distributions are considered, where large time series are needed, given that the number of these events every year is very low and depends on the oceanic position of the buoy.

Statistical methods to determine extreme wave heights using the Peaks-Over-Threshold approach (POT) have been significantly improved for several years. Mathiesen et al. [12] use the POT method along with a Weibull distribution estimated by a maximum likelihood procedure. This is applied to the pre-

---

\*Corresponding author at: Department of Computer Science and Numerical Analysis, University of Córdoba, Rabanales Campus, Albert Einstein Building 3rd Floor, 14071 Córdoba, Spain. Tel.: +34 957 218 349; Fax: +34 957 218 630. E-mail addresses: i92duroa@uco.es, mcarbonero@uloyola.es, pagutierrez@uco.es, chervas@uco.es

diction of significant wave heights associated with high return periods, considering that 100 years or more is enough for the extensive use of ocean's resources. In 2001, Coles [13] introduced the GPD-Poisson by fitting a Generalized Pareto Distribution (GPD), which was also used later on [14, 15].

In 2011, Mazas and Hamm [16] proposed the determination of extreme wave heights using a POT approach, where a double threshold  $(u_1, u_2)$  is presented. A low value  $u_1$  is set to select both weak and strong storms. Then, a second higher threshold  $(u_2)$  has to be determined to decide which storms have a statistically extreme behaviour. Tree probability distributions of extreme values are used to determine  $u_2$ : GPD-Poisson, Weibull and Gamma distributions. To select the best-fitting distribution, two objective criteria based on likelihood (Bayesian Information Criterion [17], BIC, and Akaike Information Criterion [18], AIC) are used.

More recently, Petrov et al. [19] presented a maximum entropy (MaxEnt) method for the prediction of extreme significant wave heights, comparing it with the state of the art methodologies of the Extreme Value Theory (EVT): the GPD and the Generalized Extreme Value distribution (GEV). According to the definition of the MaxEnt principle, the distribution that provides the highest entropy is selected to give more information among all other possible distributions that satisfy the proposed constraints.

As can be seen, all methods are based on selecting a threshold and modelling the distribution of the wave heights over this threshold. Thus, the main problem is how to select this threshold in order to avoid information loss. For that, it could be interesting to model the complete time series with both regular and extreme values and to use this theoretical distribution to fix the threshold for the POT approach. In this paper, we propose a new methodology to determine the distribution of the extreme wave heights considering that the normally distributed extreme wave heights are added as to regular values from a uniform distribution. The reason for choosing a uniform distribution is that, outside a range around the mean, all observations of wave height should be assumed to be part of the problem and never noise. This makes us discard the normal distribution as a contamination distribution. After that, using the estimated theoretical mixed distribution, we set the threshold for the POT methodologies. In this way, we fit several distributions of the values over this threshold and select the best-fitting distribution according to the BIC and AIC criteria.

The novel contributions of this work to applied energy issues are:

- In atmospheric time series, such as wave height [20], wind power [21] or fog formation in airports [22, 23], there are many values close to the average. This makes that extreme values of time series, which are the most interesting ones, are hidden by uninteresting values. For this reason, these values have a distorting effect on extreme values. In this paper, we show that regular values do not significantly change the mean value of the time series, but they reduce the deviation by increasing the sample size.

- We propose a new methodology which, up to the author knowledge, has not been applied before to energy time series. This methodology is able to determine the distribution of the complete time series, taking into account that wave height time series distribution is a mixture of a normal distribution of extreme values and noise from a uniform distribution.
- For adjusting the four parameters needed to define the mixed distribution, we used the method of moments [24], given that our methodology uses the raw time series.
- When the mixed distribution is estimated, this methodology is used to determine the threshold needed for POT approaches. We assume that using the extreme values situated over a percentile of the theoretical mixed distribution is more reliable than using a predefined value adjusted by a trial and error process. In this way, our methodology is applied to obtain return values for 1, 2, 5, 10, 20, 50 and 100 years for nine real-world wave height time series, using three different percentiles from the mixed distribution.

The rest of the paper is organized as follows: Section 2 briefly explains the Extreme Value Theory, while Section 3 introduces the proposed methodology. In Section 4, the combination of both perspectives is presented. Section 5 shows the considered time series, the experimental setting and the statistical analysis of the results obtained. Finally, the paper is concluded in Section 6.

## 2. Extreme Value Theory

Extreme Value Theory (EVT) is associated to the maximum sample  $M_n = \max(X_1, \dots, X_n)$ , where  $(X_1, \dots, X_n)$  is a set of independent random variables with common distribution function  $F$ . In this case, the distribution of the maximum observation is given by  $Pr(M_n < x) = F^n(x)$ . The hypothesis of independence when the  $X$  variables represent the wave height over a determined threshold is quite acceptable, because, for oceanographic data, it is common to adopt a POT scheme which selects extreme wave height events that are approximately independent [25]. Also, in [26], authors affirm that "The maximum wave heights in successive sea states can be considered independent, in the sense that the maximum height is dependent only on the sea state parameters and not in the maximum height in adjacent sea states". This  $M_n$  variable is described with one of the three following distributions: Gumbel, Frechet, and Weibull.

One methodology in EVT is to consider wave height time series with the annual maximum approach (AM), where  $X$  represents the wave height collected on regular periods of time of one year, and  $M_n$  is formed by the maximum values of each year. The statistical behaviour of AM can be described by the distribution of the maximum wave height in terms of Generalized Extreme Value (GEV) distribution [27]:

$$G(x) = \begin{cases} \exp \left\{ - \left[ 1 + \xi \left( \frac{x-\mu}{\sigma} \right) \right]^{\frac{1}{\xi}} \right\}, & \xi \neq 0, \\ \exp \left\{ - \exp \left( - \left( \frac{x-\mu}{\sigma} \right) \right) \right\}, & \xi = 0, \end{cases} \quad (1)$$

where:

$$0 < x < 1 + \xi \left( \frac{x - \mu}{\sigma} \right), \quad (2)$$

where  $-\infty < \mu < \infty$ ,  $\sigma > 0$  and  $-\infty < \xi < \infty$ . As can be seen, the model has three parameters: location ( $\mu$ ), scale ( $\sigma$ ), and shape ( $\xi$ ).

The estimation of the return values, corresponding to the return period ( $T_p$ ), are obtained by inverting Eq. 1:

$$z_p = \begin{cases} \mu - \frac{\sigma}{\xi} \left[ 1 - \{-\log(1 - p)\}^{-\xi} \right], & \xi \neq 0, \\ \mu - \sigma \log \{-\log(1 - p)\}, & \xi = 0, \end{cases} \quad (3)$$

where  $G(z_p) = 1 - p$ . Then,  $z_p$  will be exceeded once per  $1/p$  years, which corresponds to  $T_p$ .

The alternative method in the EVT context is the Peak-Over-Threshold (POT), where all values over a threshold predefined by the user are selected to be statistically described instead of only the maximum values [28, 29]. POT method has become a standard approach for these predictions [12, 29, 30]. Furthermore, several improvements over the basic approach have been proposed by various authors since then [31, 32, 33, 19, 25].

The POT method is based on the fact that if the AM approach uses a GEV distribution (Eq. 1), the peaks over a high threshold should result in the related approximated distribution: the Generalized Pareto Distribution (GPD). The GPD fitted to the tail of the distribution gives the conditional non-exceedance probability  $P(H_{max} \leq x | H_{max} > u)$ , where  $u$  is the threshold level. The conditional distribution function can be calculated as:

$$P(X \leq x | X > u) = \begin{cases} 1 - \left( 1 + \xi^* \left( \frac{x-u}{\sigma^*} \right) \right)^{\frac{1}{\xi^*}}, & \xi^* \neq 0, \\ 1 - \exp \left( - \left( \frac{x-u}{\sigma^*} \right) \right), & \xi^* = 0. \end{cases} \quad (4)$$

There is consistency between the GEV and GPD models, meaning that the parameters can be related to  $\xi^* = \epsilon$  and  $\sigma^* = \sigma + \xi(u - \mu)$ . The parameters  $\sigma$  and  $\xi$  are the scale and shape parameters, respectively. When  $\xi \geq 0$ , the distribution is referred to as long tailed. When  $\xi < 0$ , the distribution is referred to as short tailed. The methods used to estimate the parameters of the GPD and the selection of the threshold will be now discussed.

The use of the GPD for modelling the tail of the distribution is also justified by asymptotic arguments in [13]. In this paper, author confirms that it is usually more convenient to interpret extreme value models in terms of return levels, rather than individual parameters. In order to obtain these return levels, the exceedance rates of thresholds have to be determined as  $P(X > u)$ . In this way, using Eq. 4 ( $P(X > x | X > u) = P(X > x)/P(X > u)$ ) and considering that  $z_N$  is exceeded on average every  $N$  observations, we have:

$$P(X > u) \left[ 1 + \xi^* \left( \frac{z_N - u}{\sigma^*} \right) \right]^{-\frac{1}{\xi^*}} = \frac{1}{N}. \quad (5)$$

Then, the  $N$ -year return level  $z_N$  is obtained as:

$$z_N = u + \frac{\sigma^*}{\xi^*} \left[ (N * P(X > u))^{\xi^*} - 1 \right]. \quad (6)$$

There are many techniques proposed for the estimation of the parameters of GEV and GPD. In [19], authors applied the maximum likelihood methodology (ML) described in [13]. However, the use of this methodology for two parameter distributions (i.e. Weibull or Gamma) has a very important drawback: these distributions are very sensitive to the distance between the high threshold ( $u_2$ ) and the first peak [16]. For this reason, ML could be used with two-parameter distribution when  $u_2$  reaches a peak. As this peak is excluded, the first value of the exceedance is as far from  $u_2$  as possible. A solution would be to use the three-parameter Weibull and Gamma distributions. However, ML estimation of such distributions is very difficult, and the algorithms usually fit two-parameter distributions inside a discrete range of location parameters [34].

### 3. Methodology

As stated before, in this paper, we present a new methodology to model this kind of time series considering not only extreme values but also the rest of observations. In this way, instead of selecting the maximum values per a period (usually a year) or defining thresholds in the distribution of these extreme wave heights, we model the distribution of all wave heights, considering that it is a mixture formed by a normal distribution and a uniform distribution. The motivation is that the uniform distribution is associated to regular wave height values and contaminate the normal distribution of extreme values. This theoretical mixed distribution is used then to fix the threshold for the estimation of the POT distributions.

Let us consider as a sequence of independent random variables,  $(X_1, \dots, X_n)$  of wave height data. These data follow an unknown continuous distribution. We assume that this distribution is a mixture of two independent distributions:  $Y_1 \sim N(\mu, \sigma)$  and  $Y_2 \sim U(\mu - \delta, \mu + \delta)$ , where  $N(\mu, \sigma)$  is a Gaussian distribution,  $U(\mu - \delta, \mu + \delta)$  is a uniform distribution,  $\mu$  is the common mean of both distributions,  $\sigma$  is the standard deviation of  $Y_1$ , and  $\delta$  is the radius of  $Y_2$ . Then  $X = \gamma Y_1 + (1 - \gamma) Y_2$ , being  $\gamma$  the probability that an observation comes from the normal distribution.

For the estimation of the values of these four parameters ( $\mu, \sigma, \delta, \gamma$ ), the standard statistical theory considers the least squares methods, the method of moments and the maximum likelihood (ML) method. In this context, Mathiesen et al. [12] found that the least squares methods are sensitive to outliers, although Goda [35] recommended this method with modified plotting position formulae. The author also proposed the method of moments as first approximation, because this method gives too much bias for the typical samples sizes using AM or POT models. However, it is not our case because our methodology uses all the values of the wave height time series.

The ML method is commonly used in metocean applications [25], due to its asymptotic properties of being unbiased and efficient. However, the ML estimators do not achieve these asymptotic properties until they are applied to large sample sizes. Hosking and Wallis [36] showed that the ML estimators are non-optimal for sample sizes up to 500, with higher bias

and variance than other estimators, such as moments and probability weighted-moments estimators. Furthermore, the use of ML estimation for two-parameter distributions such as Weibull and Gamma distributions has the drawback [16] previously discussed. Besides, the ML estimation is known to provide poor results when the maximum is at the limit of the interval of validity of one of the parameters. On the other hand, the estimation of the GPD parameters is subject of ongoing research. A quantitative comparison of recent methods for estimating the parameters was presented by Kang and Song [37]. In our case, having to estimate four parameters, we have decided to use the method of moments, for its analytical simplicity.

Considering  $\phi$  as the probability density function (pdf) of a standard normal distribution  $N(0, 1)$ , the pdf of  $Y_1$  is defined as:

$$f_1(x) = \frac{1}{\sigma} \phi(z_x), \quad z_x = \frac{x - \mu}{\sigma}, \quad x \in \mathbb{R}. \quad (7)$$

The pdf of  $Y_2$  is:

$$f_2(x) = \frac{1}{2\delta}, \quad x \in (\mu - \delta, \mu + \delta). \quad (8)$$

Consequently, the pdf of  $X$  is:

$$f(x) = \gamma f_1(x) + (1 - \gamma) f_2(x), \quad x \in \mathbb{R}. \quad (9)$$

To infer the values of the four parameters of the wave height time series  $(\mu, \sigma, \delta, \gamma)$ , we define, for any symmetric random variable with respect to the mean  $\mu$  with pdf  $g$  and finite moments, a set of functions in the form:

$$U_k(x) = \int_{|t-\mu| \geq x} |t - \mu|^k g(t) dt, \quad x \geq 0, \quad k = 1, 2, 3, \dots, \quad (10)$$

or because of its symmetry:

$$U_k(x) = 2 \int_{x+\mu}^{\infty} (t - \mu)^k g(t) dt, \quad k = 1, 2, 3, \dots \quad (11)$$

These functions are well defined for the same moments of the variable  $x$ , because:

$$U_k(x) < \int_{-\infty}^{\infty} |t - \mu|^k g(t) dt < \infty, \quad k = 1, 2, 3, \dots \quad (12)$$

Particularly, for the normal and uniform distributions, all the moments are finite, and the same happens for all the  $U_k(x)$  functions. This function measures, for each pair of values  $x$  and  $k$ , the bilateral tail from the value  $x$  of the moment with respect to the mean of order  $k$  of the variable. It is, therefore, a generalization of the concept of probability tail, which is obtained for  $k = 0$ .

Now, if we denote the corresponding moments for the distributions  $Y_1$  and  $Y_2$  by  $U_{k,1}(x)$  and  $U_{k,2}(x)$ , it is verified that:

$$U_k(x) = \gamma U_{k,1}(x) + (1 - \gamma) U_{k,2}(x). \quad (13)$$

Then, to calculate the function  $U_k(x)$ , we just need to calculate the functions  $U_{k,1}(x)$  and  $U_{k,2}(x)$ .

### 3.1. Calculation $U_k$ for the uniform distribution ( $U_{k,2}$ )

From the definition of  $f_2(x)$  and  $U_k(x)$ , if  $x$  does not exceed  $\delta$ :

$$U_{k,2}(x) = 2 \int_{\mu+x}^{\mu+\delta} (t - \mu)^k \frac{1}{2\delta} dt = \frac{(t - \mu)^{k+1}}{(k+1)\delta} \Big|_{\mu+x}^{\mu+\delta} = \frac{\delta^{k+1} - x^{k+1}}{(k+1)\delta}, \quad (14)$$

then,

$$U_{k,2}(x) = \begin{cases} \frac{\delta^{k+1} - x^{k+1}}{(k+1)\delta} & 0 \leq x \leq \delta, \\ 0 & x > \delta. \end{cases} \quad (15)$$

### 3.2. Calculation $U_k$ for the normal distribution ( $U_{k,1}$ )

From the definition of the  $f_1(x)$  and  $U_k(x)$ , we have:

$$U_{k,1}(x) = \frac{2}{\sigma} \int_{\mu+x}^{\infty} (t - \mu)^k \phi\left(\frac{t - \mu}{\sigma}\right) dt. \quad (16)$$

Let the variable  $u$  be in the form  $u = \frac{t - \mu}{\sigma}$ , then:

$$U_{k,1}(x) = 2 \int_{\frac{x}{\sigma}}^{\infty} (u\sigma)^k \phi(u) du = \sigma^k \Upsilon_k\left(\frac{x}{\sigma}\right), \quad (17)$$

where  $\Upsilon_k = 2 \int_x^{\infty} (u)^k \phi(u) du$ .  $\Upsilon_k(z)$  is the  $U_k$  function calculated for a  $N(0, 1)$  distribution, which will be then updated with values of  $k = 1, 2, 3$ .

#### 3.2.1. Proposition I

The following equations are verified:

$$\Upsilon_1(x) = 2 \int_x^{\infty} u \phi(u) du = 2\phi(x), \quad (18)$$

$$\Upsilon_2(x) = 2 \int_x^{\infty} u^2 \phi(u) du = 2(1 - \Phi(x) + x\phi(x)), \quad (19)$$

$$\Upsilon_3(x) = 2 \int_x^{\infty} u^3 \phi(u) du = 2(2 + x^2)\phi(x), \quad (20)$$

where  $\Phi$  is the cumulative distribution function (CDF) of the  $N(0, 1)$  distribution. See Appendix A for the demonstration of proposition I.

### 3.3. Sample estimates of $U_k$

For each value of  $k$  and  $x \geq 0$ , the sample estimator of  $U_k$  obtained by the method of moments is:

$$u_k(x) = \frac{1}{n} \sum_{|x_i - \mu| \geq x} |x_i - \mu|^k, \quad (21)$$

which has the properties described in the following propositions.

#### 3.3.1. Proposition II

The estimator  $u_k(x)$  is an unbiased estimator of  $U_k(x)$ . See Appendix B for the demonstration of proposition II.

### 3.3.2. Proposition III

The estimator  $u_k(x)$  is a consistent estimator of  $U_k(x)$ . See Appendix C for the demonstration of proposition III.

### 3.4. Estimation of $\sigma$ , $\delta$ , and $\gamma$ parameters

Applying the method of moments, we have the following three-equation system:

$$U_k(0) = u_k(0), k = 1, 2, 3. \quad (22)$$

The reason for choosing the origin is that it has the maximum amount of information about the  $u_k(x)$  functions defined in Eq. 21. If a nonzero  $x$  value is chosen, the estimate will discard all observations in the interval  $(\mu - x, \mu + x)$ . Substituting equations 15, A.11, A.12 and A.13 in Eq. 13, the resulting equation system is:

$$\gamma U_{1,1}(0) + (1 - \gamma)U_{1,2}(0) = \gamma\sigma\sqrt{\frac{2}{\pi}} + (1 - \gamma)\frac{\delta}{2} = u_1(0), \quad (23)$$

$$\gamma U_{2,1}(0) + (1 - \gamma)U_{2,2}(0) = \gamma\sigma^2 + (1 - \gamma)\frac{\delta^2}{3} = u_2(0), \quad (24)$$

$$\gamma U_{3,1}(0) + (1 - \gamma)U_{3,2}(0) = \gamma\sigma^3 2\sqrt{\frac{2}{\pi}} + (1 - \gamma)\frac{\delta^3}{4} = u_3(0), \quad (25)$$

where the solution must satisfy:  $\hat{\sigma}, \hat{\delta} > 0$  and  $\gamma \in [0, 1]$ .

### 3.5. Adjustment to the mixed distribution

To contrast if the obtained estimators are valid, we could see if the set of observations  $\{x_1, \dots, x_n\}$  fit the pdf of the final distribution:

$$\hat{f}(x) = \hat{\gamma}\hat{f}_1(x) + (1 - \hat{\gamma})\hat{f}_2(x), x \in \mathbb{R}, \quad (26)$$

where:

$$\hat{f}_1(x) = \frac{1}{\hat{\sigma}}\phi\left(\frac{x - \hat{\mu}}{\hat{\sigma}}\right), x \in \mathbb{R}, \quad (27)$$

and:

$$\hat{f}_2(x) = \frac{1}{2\hat{\delta}}, x \in (\hat{\mu} - \hat{\delta}, \hat{\mu} + \hat{\delta}). \quad (28)$$

For this purpose, a test that can be used is the Kolmogorov-Smirnov test. The one-sample Kolmogorov-Smirnov test [38] is commonly used to examine whether samples come from a specific distribution function by comparing the observed cumulative distribution function with an assumed theoretical distribution. The Kolmogorov-Smirnov statistic  $Z$  is computed from the largest difference (in absolute value) between the observed and theoretical cumulative distribution. In this way,  $Z$  is the greatest vertical distance between empirical distribution function  $S(x)$  and the specified hypothesized distribution function  $F^*(x)$ , which can be calculated as:

$$Z = \max_x |F^*(x) - S(x)|, \quad (29)$$

where the null hypothesis is  $H_0 : F(x) = F^*(x)$  for all  $-\infty < x < \infty$ , and the alternative hypothesis is  $H_1 : F(x) \neq F^*(x)$  for at least one value of  $x$ ,  $F(x)$  being the true distribution. If  $Z$  exceeds the  $1 - \alpha$  quantile value ( $Q(1 - \alpha)$ ), then we reject  $H_0$  at the level of significance of  $\alpha$ . When the number of observations  $n$  is large, the  $Q(1 - \alpha)$  value can be approximated as [39]:

$$Q(1 - \alpha) = \frac{\sqrt{-0.5 \log(\frac{\alpha}{2})}}{\sqrt{n}}. \quad (30)$$

## 4. Using the theoretical mixed distribution to fix the threshold of the POT approaches

In this paper, when the mixed distribution is estimated, we use it to set the threshold for estimating the POT distributions. We assume that using the points which are situated over a percentile of the theoretical mixed distribution is more reliable than using a threshold value predefined by a trial and error procedures. In our work, we consider the 95%, 97.5% and 99% percentiles as possible thresholds.

In this way, a new sample of independent random variables is defined by  $Z = (z_1, z_2, \dots, z_M)$ , where  $Z = X > u$ ,  $u$  being the threshold and  $M$  being the number of exceedances. In this work, three distributions are fitted for the threshold exceedance distribution:

- The first one is the GPD [40], whose cumulative function is defined in Eq. 4.
- The second distribution is the Gamma distribution, with the following cumulative function:

$$F(z; \xi, \sigma) = \frac{\gamma(\xi, \frac{z}{\sigma})}{\Gamma(\xi)}, \quad (31)$$

where  $\gamma$  is the lower incomplete gamma function, and  $\Gamma$  is the Gamma function.

- Finally, the Weibull distribution is also considered:

$$F(z; \xi, \sigma) = 1 - \exp\left[-\left(\frac{z}{\sigma}\right)^\xi\right]. \quad (32)$$

These three distributions are adjusted to the exceedances using the Maximum Likelihood Estimator (MLE) [12]. After that, we select the best fit based on two objective criteria: BIC [17] and AIC [18]. On the one hand, BIC minimizes the bias between the fitted model and the unknown true model:

$$\text{BIC} = -2 \ln L + k_p \ln M, \quad (33)$$

where  $L$  is the likelihood of the fit,  $M$  is the sample size (in our case, the number of exceedances) and  $k_p$  the number of parameters of the distribution. On the other hand, AIC gives the model providing the best compromise between bias and variance:

$$\text{AIC} = -2 \ln L + 2k_p. \quad (34)$$

Both criteria need to be minimized.

When the best-fitted distribution is obtained, the return period  $T$  ( $H_{ST}$ ) is calculated, and then the confidence intervals are computed. As can be seen in the experimental section, the GPD is the best distribution for all cases. The quantile for the GPD is:

$$H_{ST} = \mu + \frac{\sigma}{\xi} \left[ 1 - (\lambda T)^{-\xi} \right], \quad (35)$$

where  $\lambda$  is the number of exceedances per year.

Finally, confidence intervals are also computed. For that, many authors use the classical asymptotic method [13]. However, Mathiesen et al. advocate the use of Monte-Carlo (MC) simulation techniques. A robust way is to use parametric bootstrap methods [15]. Also, Mackay and Johanning [26] proposed a storm-based MC method for calculating return periods of individual wave and crest heights. In the MC method, a random realisation of the maximum wave height in each sea state is simulated from the metocean parameter time series, and the GPD is fitted to storm peak wave heights exceeding some threshold. Mackay and Johanning [26] showed that using  $n = 1000$  is sufficient to obtain a stable estimation, although in our case, we have considered  $n = 100000$  following the work of [16]. In [16], as in our work, authors used the MC simulation method, and, after 100000 iterations, the 90% confidence interval is obtained using the percentiles  $[H_{ST,5\%}; H_{ST,95\%}]$  of the 100000  $H_{ST}$  values obtained with the procedure.

## 5. Experimental results and discussion

This section describes the time series used in our work, shows the experimental setting and presents the results validating the proposed methodology.

### 5.1. Wave height time series

As stated before, the objective of this work is to model wave height time series where extreme values are present. For this reason, we evaluate the performance of the proposed methodology in several real-world wave height time series from different locations:

- Gulf of Alaska: two time series of significant wave height collected from the National Data Buoy Center of the USA [41] in the Gulf of Alaska have been used. The buoys have the registration numbers 46001 and 46075. For the two buoys, one value every six hours is considered. The buoy 46001 is an offshore buoy placed in the coordinates 56.23N 147.95W, and data from 1st January 2008 to 31st December 2013 is considered, with a total of 8767 observations. On the other hand, 46075 is an offshore buoy whose coordinates are 53.98N 160.82W and data from 1st January 2011 to 31st December 2015 are collected in this buoy (7303 observations).
- Puerto Rico: a total of six offshore buoys from Puerto Rico have been selected in our experiments to evaluate the proposed methodology. These buoys also belong to the NDBC of the USA, with registration ids 41043, 41044, 41046, 41047, 41048 and 41049. One value every six

hours is considered, and data from 1st January 2011 to 31st December 2015 are used (7303 observations for each one). The geographical coordinates for each buoy are 21.13N 64.86W, 21.58N 58.63W, 23.83N 68.42W, 27.52N 71.53W, 31.86N 69.59W, and 27.54N 62.95W, respectively.

- Spain: this dataset comes from the SIMAR-44 hindcast database provided by Puertos del Estado (Spain). The point is placed in the Strait of Gibraltar, whose coordinates are 36N 6W. One value every three hours is considered in this dataset from 1st January 1959 to 31 December 2000, forming a set of 122278 observations. Note that, it is the largest time series in our experiments. Given that the time series includes 42 years, we can estimate long return periods of wave height.

The summary of the information for each time series can be seen in Table 1 which includes the type of buoy, the location, the geographical coordinates, the number of observations, the mean values of the time series ( $H_s$ ), and the maximum values of each one. The map location can be observed in Figure 1, while the representation of the time series are shown in Figure 2.

### 5.2. Experimental design

The experimental design for the time series under study is presented in this subsection. We divide the experiments in two stages:

- Firstly, the methodology is tested on the raw time series presented in the previous subsection. The algorithm estimates the parameters of the mixed distribution ( $\mu, \sigma, \delta, \gamma$ ) for each wave height time series, and then, the Kolmogorov-Smirnov test is applied to check if the estimated distribution corresponds to the empirical distribution of the data. It is important to mention that the Kolmogorov-Smirnov test is applied considering  $n = 50$ , which is an acceptable value for the Eq. 30, that is, we calculate the CDF of the estimated theoretical function and the empirical one in 50 intervals. Graphically, in this paper, we show the comparison between the theoretical distribution (estimated) and the empirical one (Figure 3).
- Secondly, as we stated in previous sections, we use the theoretical mixed distribution to establish the threshold. In this sense, we delete the values below the threshold, and we fit the GPD, Gamma and Weibull distributions with the remaining values (those which are higher than the threshold). Based on two objective criteria, BIC and AIC, we select the best-fitted distribution and, finally, the return values of this distribution for the following return periods in years  $T = (1, 2, 5, 10, 20, 50, 100)$  are calculated.

### 5.3. Discussion

The estimates and the Kolmogorov-Smirnov test results are shown in Table 2. As can be seen, the estimation of the  $\mu$  parameter is the same than the mean value of the time series (see

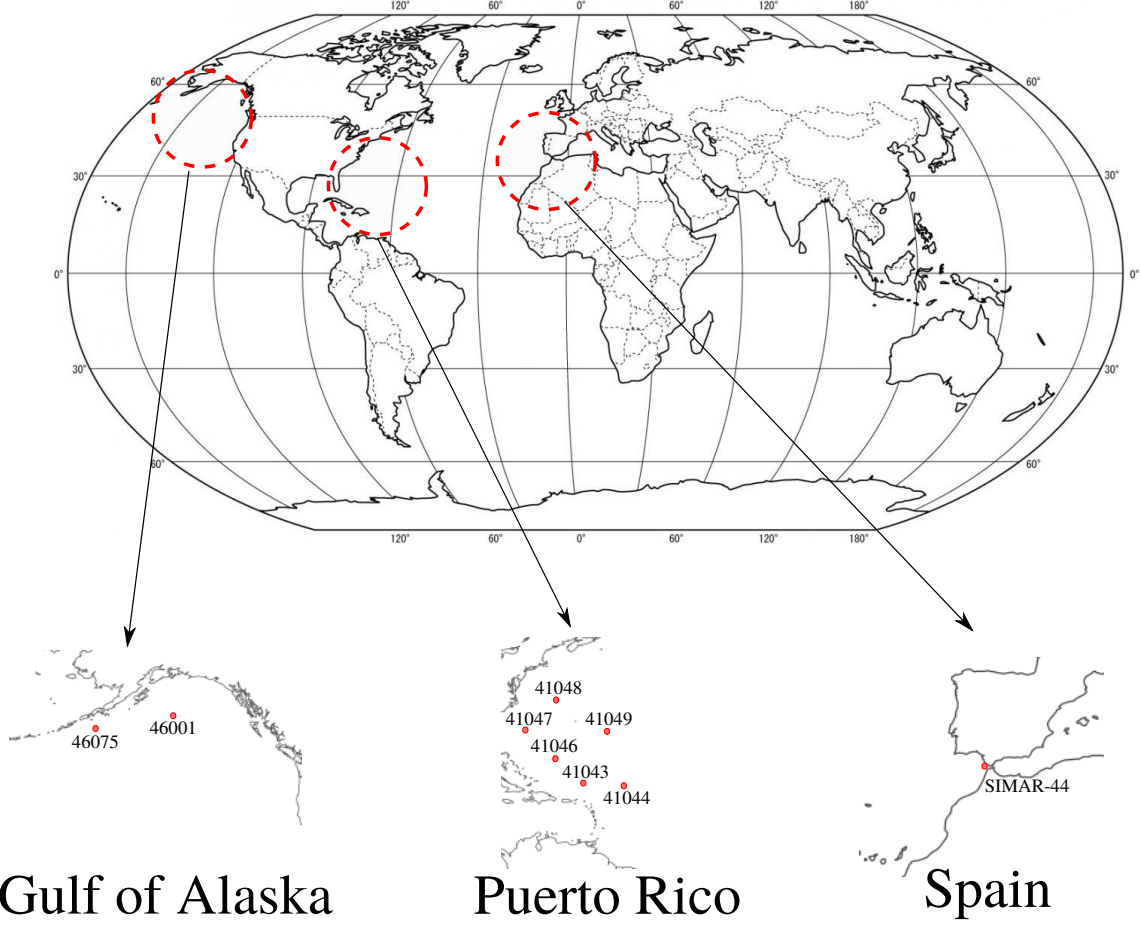


Figure 1: Locations of the different buoys considered for the experimentation.

Id	Type	Location	Coordinates	# Observations	Average Hs (m)	Max Hs (m)
46001	Offshore	Alaska	56.23N 147.95W	8767	2.65	10.17
46075	Offshore	Alaska	53.98N 160.82W	7303	2.72	13.39
41043	Offshore	Puerto Rico	21.13N 64.86W	7303	1.76	6.12
41044	Offshore	Puerto Rico	21.58N 58.63W	7303	1.84	8.98
41046	Offshore	Puerto Rico	23.83N 68.42W	7303	1.71	7.85
41047	Offshore	Puerto Rico	27.52N 71.53W	7303	1.63	8.51
41048	Offshore	Puerto Rico	31.86N 69.59W	7303	1.85	12.07
41049	Offshore	Puerto Rico	27.54N 62.95W	7303	1.78	10.96
SIMAR-44	Coastal	Spain	36.00N 6.00W	122278	1.09	8.60

Table 1: Characteristics of the time series recorded for every buoy.

Table 1), because we have used the sample mean as estimator (see Section 3).  $\sigma$  estimation seems to be very high with respect to the mean. It makes sense given that the estimation is made with approximately 7000 points, the variance needing to be high.  $\delta$  has values in the interval (0.74,1.80) because there is wave height data that, although not very small, contaminates the normal distribution (in intervals of three months, the parameter value is lower).  $\gamma$ , which is the probability that an observation comes from the normal distribution, is very low. Again, this makes sense because of the high amount of data which are not

extreme values and represent regular waves (uniform distribution). The Kolmogorov-Smirnov test does not reject the null hypothesis for all cases,  $Z < Q(1 - \alpha)$ , confirming that the estimated parameters of the mixed distribution correspond to the empirical values. For this reason, we can accept the theory proposed in this paper as a good method to estimate the theoretical distribution in wave height time series. Note that the  $Z$  values are lower in those time series whose mean value is higher, so the wave height time series collected from buoys 46001 and 46075 are better adjusted with this distribution, while the Spanish time

series results in a worse fit. The results of the Kolmogorov-Smirnov test can be complementary analysed with the representation of the empirical and theoretical distribution, as can be observed in Figure 3. The graphs show how the estimated theoretical distributions are adapted to the empirical distributions in each database.

For the second experiment, Table 3 shows the values of the BIC and AIC criteria when the GPD, Gamma and Weibull distribution are fitted using the values over the threshold determined by the percentiles 95%, 97.5% and 99% of the theoretical mixed distribution. The number of POTs ( $M$ ) and the number of peaks per year ( $\lambda$ ) are also included. As can be seen, the higher the percentile, the lesser number of peaks per year, because the number of POTs will be much lower. The results confirm that the best fitted distribution for all databases and for all percentiles is the GPD.

There exist a perfect correlation between the values of BIC and AIC for the three percentiles (0.977, 0.998 and 1.000, respectively), for the three distributions and the nine time series. For this reason, we focus on the percentile 95% and the BIC criterion, given that  $M$  and  $\lambda$  is higher with this percentile. For instance, in buoy 46001, the BIC value for the GPD is 622.72, a 69.8% lower than the value for the Gamma distribution, and a 73.5% lower than the value for the Weibull distribution. These results differ from those obtained by [16] for the SIMAR-44 time series, where GPD gives poor results with respect to these criteria when compared to Gamma; but it is important to mention that we use a 3-parameter GPD instead of a 2-parameter one.

Finally, the return values and the confidence intervals for each dataset considering the different thresholds are summarized in Table 4. We have considered return periods of  $T \in \{1, 2, 5, 10, 20, 50, 100\}$  years. If we compare the obtained return values and the confidence intervals with respect to the ones obtained by Mazas and Hamm [16], for SIMAR-44 time series, we can see that the results are not the same due to the differences in the thresholds, and because they consider 44 years instead of 42, as the first and the last year are used although they are not complete. We agree with the authors in that work in the sense that choosing the right threshold is not always a straightforward issue. For example, if we consider the percentile 97.5% of the theoretical mixed distribution, the return values and the confidence intervals are quite similar to the ones obtained by Mazas (with the slight differences commented above). With respect to the values obtained for the rest of the buoys, up to our knowledge, there are not other reference values. These estimations are approximate, given the reduced length of the time series (six years for buoy 46001 and five for the other buoys). If we compare them with the extreme values that appear in Table 1, we can see that, for the buoys 46075, 41043, 41046, the confidence intervals for the 95% percentile tend to contain these values more frequently, for the buoys 41047, 41048, 41049 and SIMAR-44, the confidence intervals are more adjusted, and, for the buoys 46001 and 41044, there are no confidence intervals that contain them.

## 6. Conclusions

This paper proposes a novel methodology for wave height time series modelling based on the assumption that, given a time series where the high waves are less common than lower ones, its distribution can be modelled as a mixture of a normal distribution with a uniform distribution. The methodology is based on the method of moments, and we use it to establish the threshold for the distribution estimation of the values over a peak methodology (POT). The automatic determination of this threshold is an important task, given that the alternative is to use a trial and error method which, as several authors agree, can be problematic and quite subjective. The whole approach is tested on nine real-world time series collected from the Gulf of Alaska (46001 and 46075), from Puerto Rico (41043, 41044, 41046, 41047, 41048 and 41049), and from Spain (SIMAR-44). For SIMAR-44, we compare our return periods with those obtained by Mazas and Hamm. The return periods obtained for the rest buoys can be considered as an initial approximation given the reduced length of the time series.

The experimentation is divided into two stages: the first one analysed the estimation of the distribution in the nine time series, showing that the estimated theoretical distribution fits the empirical one. These results are corroborated by a Kolmogorov-Smirnov test where  $Z < Q(1 - \alpha)$  in all databases. For the second experiment, we use the percentiles 95%, 97.5% and 99% of the estimated theoretical distribution as possible thresholds for the POT distribution estimation. Results show that the best-fitted distribution for the POT is the Generalized Pareto Distribution in all cases, showing their return periods and confidence intervals.

A future line of work could approach the segmentation of the time series based on the percentiles of the obtained distribution and perform a posterior prediction of the segments obtained. We also plan to extend this work using time series from different fields and more advanced methods for forecasting, such as artificial neural networks.

## Acknowledgement

This work has been subsidized by the projects TIN2017-85887-C2-1-P and TIN2017-90567-REDT of the Spanish Ministry of Economy and Competitiveness (MINECO), and FEDER funds (EU). Antonio M. Durán-Rosal's research has been subsidized by the FPU Predoctoral Program of the Spanish Ministry of Education, Culture and Sport (MECD), grant reference FPU14/03039.

## Appendix A. Demonstration of proposition I

The three equations can be obtained using integration by parts, but it is easier to derive the functions  $\Upsilon_k(x)$  to check the result. For the definition of the functions, for each value of  $k$ , we have:

$$\Upsilon'_k(x) = \frac{\partial \Upsilon_k(x)}{\partial x} = -2x^k \phi(x). \quad (\text{A.1})$$



Id	$\hat{\mu}$	$\hat{\sigma}$	$\hat{\delta}$	$\hat{\gamma}$	$Z$	$Q(1 - \alpha)$
46001	2.652597	2.082763	1.708683	0.296738	0.081194	0.192065
46075	2.724890	2.522156	1.799095	0.189406	0.080575	0.192065
41043	1.762838	0.956801	0.743943	0.224906	0.086916	0.192065
41044	1.836434	1.449356	0.795858	0.077810	0.107365	0.192065
41046	1.705895	1.236797	0.793447	0.170138	0.099714	0.192065
41047	1.633332	1.853012	0.893645	0.113544	0.110250	0.192065
41048	1.849044	2.435167	1.158171	0.109262	0.119285	0.192065
41049	1.777286	2.023050	0.998251	0.091232	0.132657	0.192065
SIMAR-44	1.093372	1.580551	0.748225	0.125561	0.142356	0.192065

Table 2: Parameter estimation and Kolmogorov-Smirnov test results.

Id		Percentile 95%				Percentile 97.5%				Percentile 99%			
		$M$	$\lambda$	BIC	AIC	$M$	$\lambda$	BIC	AIC	$M$	$\lambda$	BIC	AIC
46001	GPD			662.72	653.33			786.42	774.61			313.09	303.98
	Gamma	806	134.33	2193.33	2183.95	379	63.17	1002.74	994.87	154	25.67	389.46	383.38
	Weibull			2497.93	2488.54			1146.18	1138.30			441.27	435.20
46075	GPD			1894.56	1880.56			818.69	807.23			290.39	281.88
	Gamma	786	157.20	2381.82	2372.49	337	67.40	1025.61	1017.97	126	25.20	389.93	384.26
	Weibull			2719.86	2710.53			1188.91	1181.27			458.29	452.62
41043	GPD			302.62	288.63			79.40	68.20			49.64	41.98
	Gamma	784	156.80	820.51	811.18	298	59.60	375.38	367.92	94	18.80	158.16	153.06
	Weibull			1307.63	1298.30			574.64	567.17			207.79	202.69
41044	GPD			346.78	332.89			320.77	307.14			50.51	42.41
	Gamma	758	151.60	1018.04	1008.78	694	138.80	947.71	938.63	110	22.00	249.05	243.65
	Weibull			1638.67	1629.41			1521.01	1511.93			328.35	322.95
41046	GPD			606.02	592.50			238.24	227.33			62.41	54.84
	Gamma	669	167.25	1040.63	1031.62	280	70.00	449.81	442.54	92	23.00	173.41	168.36
	Weibull			1399.50	1390.49			628.37	621.10			235.26	230.21
41047	GPD			1064.67	1051.34			580.17	568.91			185.58	177.85
	Gamma	629	157.25	1503.31	1494.42	316	79.00	775.16	767.65	97	24.25	253.51	248.36
	Weibull			1749.18	1740.29			910.31	902.80			295.82	290.67
41048	GPD			1776.19	1762.11			971.75	959.69			301.70	293.34
	Gamma	806	161.20	2320.91	2311.53	412	82.40	1231.09	1223.04	120	24.00	368.35	362.77
	Weibull			2626.43	2617.05			1392.24	1384.20			421.23	415.66
41049	GPD			1227.71	1213.61			895.71	882.74			226.25	218.09
	Gamma	811	162.20	1870.43	1861.03	558	111.60	1337.14	1328.49	112	22.40	324.23	318.80
	Weibull			2277.40	2268.00			1624.41	1615.76			378.43	372.99
SIMAR-44	GPD			16998.99	16976.38			8345.27	8325.29			2867.27	2850.61
	Gamma	13847	329.69	28375.75	28360.68	5768	137.33	12646.92	12633.60	1908	45.43	4089.08	4077.97
	Weibull			33396.55	33381.48			14701.35	14688.03			4842.63	4831.52

Table 3: BIC and AIC criterion for the estimated distributions of the POT method.

Taking into account that  $\frac{\partial \phi(x)}{\partial x} = -x\phi(x)$ , and  $\frac{\partial \Phi(x)}{\partial x} = \phi(x)$ :

$$\frac{\partial 2\phi(x)}{\partial x} = -2x\phi(x) = \Upsilon_1'(x), \quad (A.2)$$

$$\frac{\partial(2(2+x^2)\phi(x))}{\partial(x)} = 2(2x\phi(x) - (2+x^2)x\phi(x)) = -2x^3\phi(x) = \Upsilon_3'(x). \quad (A.4)$$

Therefore, the left and right sides of the previous equations differ in, at most, a constant. To verify that they are the same, we

$$\frac{\partial(2(1-\Phi(x)+x\phi(x)))}{\partial x} = 2(-\phi(x) + \phi(x) - x^2\phi(x)) = -2x^2\phi(x) = \Upsilon_2'(x), \quad (A.3)$$

Id	T	Percentile 95%		Percentile 97.5%		Percentile 99%	
		$Hs_T$	Confidence Interval	$Hs_T$	Confidence Interval	$Hs_T$	Confidence Interval
46001	100	23.50	18.25 - 32.75	20.65	15.17 - 32.21	28.71	18.17 - 62.30
	50	21.46	17.00 - 29.06	18.95	14.46 - 28.29	25.09	16.99 - 50.77
	20	18.97	15.47 - 24.49	16.87	13.31 - 23.42	21.01	15.12 - 37.18
	10	17.22	14.34 - 21.59	15.40	12.56 - 20.75	18.38	13.84 - 29.61
	5	15.60	13.18 - 19.17	14.03	11.70 - 18.04	16.09	12.60 - 24.10
	2	13.61	11.89 - 16.11	12.35	10.66 - 15.08	13.51	11.28 - 18.14
	1	12.22	10.88 - 14.15	11.18	9.93 - 13.15	11.84	10.32 - 14.79
46075	100	16.24	12.99 - 21.77	16.69	12.48 - 25.15	12.59	9.79 - 21.29
	50	15.39	12.49 - 19.95	15.78	12.11 - 22.95	12.22	9.67 - 19.40
	20	14.28	11.85 - 18.21	14.59	11.59 - 20.30	11.70	9.48 - 17.23
	10	13.44	11.34 - 16.68	13.70	11.15 - 18.38	11.27	9.40 - 15.90
	5	12.60	10.78 - 15.27	12.81	10.69 - 16.51	10.82	9.19 - 14.24
	2	11.49	10.06 - 13.58	11.64	10.00 - 14.26	10.18	8.94 - 12.58
	1	10.64	9.49 - 12.23	10.77	9.50 - 12.82	10.18	8.94 - 12.58
41043	100	6.47	5.38 - 8.34	4.68	4.04 - 5.93	4.58	3.99 - 6.48
	50	6.20	5.26 - 7.81	4.61	4.02 - 5.72	4.54	3.97 - 6.23
	20	5.85	5.02 - 7.10	4.50	3.97 - 5.46	4.48	3.96 - 5.90
	10	5.57	4.84 - 6.63	4.41	3.94 - 5.23	4.42	3.94 - 5.59
	5	5.29	4.66 - 6.21	4.30	3.89 - 5.03	4.35	3.93 - 5.33
	2	4.93	4.43 - 5.62	4.15	3.81 - 4.69	4.23	3.88 - 4.96
	1	4.64	4.24 - 5.21	4.02	3.73 - 4.46	4.13	3.84 - 4.66
41044	100	5.10	4.42 - 6.19	5.06	4.40 - 6.15	4.03	3.78 - 4.65
	50	4.99	4.36 - 5.96	4.95	4.32 - 5.94	4.02	3.78 - 4.61
	20	4.83	4.28 - 5.66	4.80	4.26 - 5.67	4.01	3.78 - 4.54
	10	4.70	4.21 - 5.45	4.68	4.18 - 5.43	4.00	3.78 - 4.49
	5	4.56	4.12 - 5.19	4.55	4.11 - 5.21	3.98	3.77 - 4.42
	2	4.36	4.00 - 4.87	4.35	3.99 - 4.89	3.95	3.76 - 4.30
	1	4.20	3.89 - 4.62	4.19	3.88 - 4.62	3.91	3.75 - 4.21
41046	100	7.53	6.01 - 10.21	6.50	5.13 - 9.55	4.87	4.26 - 6.83
	50	7.20	5.86 - 9.49	6.29	5.07 - 8.94	4.83	4.25 - 6.60
	20	6.75	5.62 - 8.63	6.00	4.96 - 8.11	4.77	4.24 - 6.22
	10	6.41	5.43 - 7.99	5.77	4.83 - 7.49	4.72	4.22 - 5.98
	5	6.06	5.21 - 7.35	5.53	4.72 - 6.96	4.65	4.20 - 5.70
	2	5.60	4.92 - 6.59	5.19	4.55 - 6.27	4.55	4.17 - 5.31
	1	5.24	4.68 - 6.04	4.92	4.41 - 5.76	4.45	4.14 - 5.05
41047	100	7.83	6.25 - 10.50	10.37	7.58 - 16.37	9.35	6.55 - 19.55
	50	7.57	6.14 - 9.99	9.85	7.41 - 15.03	8.98	6.45 - 17.26
	20	7.19	5.95 - 9.22	9.15	7.06 - 13.36	8.47	6.34 - 14.93
	10	6.89	5.78 - 8.63	8.61	6.82 - 11.91	8.06	6.21 - 13.08
	5	6.58	5.58 - 8.03	8.06	6.54 - 10.81	7.63	6.09 - 11.60
	2	6.13	5.32 - 7.33	7.33	6.15 - 9.27	7.04	5.85 - 9.66
	1	5.78	5.09 - 6.76	6.77	5.81 - 8.32	6.57	5.65 - 8.38
41048	100	10.09	8.17 - 13.15	12.93	9.78 - 19.31	16.06	10.43 - 34.91
	50	9.73	8.01 - 12.51	12.28	9.42 - 17.50	14.98	10.20 - 30.03
	20	9.22	7.72 - 11.53	11.41	8.99 - 15.61	13.59	9.74 - 24.07
	10	8.81	7.47 - 10.83	10.75	8.69 - 14.30	12.56	9.37 - 20.67
	5	8.39	7.22 - 10.09	10.07	8.30 - 12.97	11.54	8.89 - 17.51
	2	7.79	6.81 - 9.15	9.15	7.73 - 11.32	10.24	8.35 - 14.23
	1	7.31	6.48 - 8.41	8.44	7.34 - 10.10	9.28	7.85 - 11.99
41049	100	6.69	5.64 - 8.32	7.14	5.92 - 9.35	7.63	5.98 - 12.71
	50	6.53	5.57 - 8.02	6.96	5.84 - 8.89	7.48	5.93 - 11.75
	20	6.30	5.45 - 7.59	6.70	5.69 - 8.31	7.25	5.88 - 10.73
	10	6.11	5.35 - 7.27	6.48	5.57 - 7.88	7.05	5.83 - 9.94
	5	5.91	5.22 - 6.87	6.25	5.46 - 7.49	6.82	5.75 - 9.14
	2	5.61	5.02 - 6.41	5.91	5.24 - 6.88	6.48	5.61 - 8.19
	1	5.36	4.85 - 6.03	5.63	5.06 - 6.44	6.18	5.49 - 7.43
SIMAR-44	100	4.49	4.31 - 4.70	6.84	6.37 - 7.41	10.68	9.39 - 12.36
	50	4.43	4.25 - 4.63	6.64	6.20 - 7.16	10.03	8.96 - 11.51
	20	4.34	4.18 - 4.52	6.35	5.97 - 6.79	9.19	8.31 - 10.32
	10	4.26	4.11 - 4.42	6.12	5.78 - 6.51	8.56	7.84 - 9.50
	5	4.17	4.03 - 4.32	5.87	5.57 - 6.20	7.94	7.36 - 8.69
	2	4.02	3.90 - 4.16	5.51	5.26 - 5.78	7.14	6.70 - 7.69
	1	3.90	3.79 - 4.02	5.22	5.01 - 5.44	6.54	6.20 - 6.94

Table 4: Return values and confidence intervals for the GPD distribution considering  $T = (1, 2, 5, 10, 20, 50, 100)$  and the percentiles 95%, 97.5%, and 99%.

check the value  $x = 0$ :

$$\Upsilon_1(0) = 2 \int_0^\infty u \phi(u) du = \sqrt{\frac{2}{\pi}}, \quad (\text{A.5})$$

$$\Upsilon_2(0) = 2 \int_0^\infty u^2 \phi(u) du = 1, \quad (\text{A.6})$$

$$\Upsilon_3(0) = 2 \int_0^\infty u^3 \phi(u) du = 2\sqrt{\frac{2}{\pi}}, \quad (\text{A.7})$$

which match with the right sides of Eqs. 18, 19 and 20:

$$\Upsilon_1(0) = 2\phi(0) = \sqrt{\frac{2}{\pi}}, \quad (\text{A.8})$$

$$\Upsilon_2(0) = 2(1 - \Phi(0)) = 1, \quad (\text{A.9})$$

$$\Upsilon_3(0) = 2(2)\phi(0) = 2\sqrt{\frac{2}{\pi}}. \quad (\text{A.10})$$

Substituting these results in Eq. 17 we have:

$$U_{1,1} = \sigma \Upsilon_1\left(\frac{x}{\sigma}\right) = 2\sigma\phi\left(\frac{x}{\sigma}\right), \quad (\text{A.11})$$

$$U_{2,1} = \sigma^2 \Upsilon_2\left(\frac{x}{\sigma}\right) = 2\sigma^2 \left(1 - \Phi\left(\frac{x}{\sigma}\right) + \frac{x}{\sigma}\phi\left(\frac{x}{\sigma}\right)\right), \quad (\text{A.12})$$

$$U_{3,1} = \sigma^3 \Upsilon_3\left(\frac{x}{\sigma}\right) = 2\sigma^3 \left(2 + \left(\frac{x}{\sigma}\right)^2\right)\phi\left(\frac{x}{\sigma}\right). \quad (\text{A.13})$$

These functions will be the base to estimate the parameters of the distribution of variable  $X$ , except in the case of  $\mu$ , as we will comment later. The estimates will be made with the corresponding  $U_k$  sample estimates, defined in Section 3.3.

## Appendix B. Demonstration of proposition II

Firstly, we rewrite  $u_k$  in the form:

$$u_k(x) = \frac{1}{n} \sum_{i=1}^n |x_i - \mu|^k I(|x_i - \mu| \geq x), \quad (\text{B.1})$$

where  $I$  is the indicator function. Considering the previous expression, we check the condition of an unbiased estimator:

$$\begin{aligned} E(u_k(x)) &= \frac{1}{n} \sum_{i=1}^n E(|x_i - \mu|^k I(|x_i - \mu| \geq x)) = \\ &= E(|t - \mu|^k I(|t - \mu| \geq x)) = \\ &= \int_{|t-\mu| \geq x} |t - \mu|^k g(t) dt = U_k(x). \end{aligned} \quad (\text{B.2})$$

## Appendix C. Demonstration of proposition III

Considering again Eq. B.1 for the variance of  $u_k(x)$  we have:

$$\begin{aligned} V(u_k(x)) &= \\ &= \frac{1}{n^2} \sum_{i=1}^n V(|x_i - \mu|^k I(|x_i - \mu| \geq x)) = \frac{1}{n} V(|t - \mu|^k I(|t - \mu| \geq x)) = \\ &= \frac{1}{n} (E(|t - \mu|^{2k} I(|t - \mu| \geq x)) - E^2(|t - \mu|^k I(|t - \mu| \geq x))) = \\ &= \frac{1}{n} (U_{2k}(x) - U_k^2(x)) \xrightarrow{n \rightarrow \infty} 0, \end{aligned} \quad (\text{C.1})$$

taking into account that  $I^2\{.\} = I\{.\}$ .

### Appendix C.1. Parameter estimation of the mixed distribution of $X$

The estimates are based on the  $u_k(0)$  values, for  $k = 1, 2, 3$ , which estimate the corresponding population parameters.

#### Appendix C.1.1. Estimation of $\mu$

Given that the mean value of both distributions (uniform and normal) is the same, this value is not affected by the mixture. Therefore, the natural estimator is

$$\hat{\mu} = \bar{x} = \frac{1}{n} \sum_{i=1}^n x_i. \quad (\text{C.2})$$

## References

- [1] C. G. Soares, M. Scotto, Modelling uncertainty in long-term predictions of significant wave height, *Ocean Engineering* 28 (3) (2001) 329–342.
- [2] Ø. Sætra, J.-R. Bidlot, Assessment of the ECMWF Ensemble Prediction Sytem for Waves and Marine Winds, European Centre for Medium-Range Weather Forecasts, 2002.
- [3] X. Feng, M. Tsimplis, M. Yelland, G. Quartly, Changes in significant and maximum wave heights in the norwegian sea, *Global and Planetary Change* 113 (2014) 68–76.
- [4] P. Esling, C. Agon, Time-series data mining, *ACM Computing Surveys (CSUR)* 45 (1) (2012) 12.
- [5] C. H. Fontes, H. Budman, A hybrid clustering approach for multivariate time series—a case study applied to failure analysis in a gas turbine, *ISA transactions*.
- [6] M. Pérez-Ortiz, A. Durán-Rosal, P. Gutiérrez, J. Sánchez-Monedero, A. Nikolaou, F. Fernández-Navarro, C. Hervás-Martínez, On the use of evolutionary time series analysis for segmenting paleoclimate data, *Neurocomputing*. doi:<https://doi.org/10.1016/j.neucom.2016.11.101>.
- [7] W. Deng, G. Wang, A novel water quality data analysis framework based on time-series data mining, *Journal of Environmental Management* 196 (2017) 365–375.
- [8] I. M. Coelho, V. N. Coelho, E. J. da S. Luz, L. S. Ochi, F. G. Guimarães, E. Rios, A gpu deep learning metaheuristic based model for time series forecasting, *Applied Energy* 201 (2017) 412 – 418. doi:<https://doi.org/10.1016/j.apenergy.2017.01.003>. URL <http://www.sciencedirect.com/science/article/pii/S0306261917300041>
- [9] A. Bagnall, J. Lines, J. Hills, A. Bostrom, Time-series classification with COTE: the collective of transformation-based ensembles, *IEEE Transactions on Knowledge and Data Engineering* 27 (9) (2015) 2522–2535.
- [10] A. Nikolaou, P. A. Gutiérrez, A. Durán, I. Dicaire, F. Fernández-Navarro, C. Hervás-Martínez, Detection of early warning signals in paleoclimate data using a genetic time series segmentation algorithm, *Climate Dynamics* 44 (7-8) (2015) 1919–1933.
- [11] Y. Zhao, L. Ye, Z. Li, X. Song, Y. Lang, J. Su, A novel bidirectional mechanism based on time series model for wind power forecasting, *Applied Energy* 177 (2016) 793 – 803. doi:<https://doi.org/10.1016/j.apenergy.2016.03.096>. URL <http://www.sciencedirect.com/science/article/pii/S0306261916304263>
- [12] M. Mathiesen, Y. Goda, P. J. Hawkes, E. Mansard, M. J. Martín, E. Peltier, E. F. Thompson, G. Van Vledder, Recommended practice for extreme wave analysis, *Journal of hydraulic Research* 32 (6) (1994) 803–814.
- [13] S. Coles, J. Bawa, L. Trenner, P. Dorazio, An introduction to statistical modeling of extreme values, Vol. 208, Springer, 2001.
- [14] F. J. Méndez, M. Menéndez, A. Luceño, I. J. Losada, Estimation of the long-term variability of extreme significant wave height using a time-dependent peak over threshold (pot) model, *Journal of Geophysical Research: Oceans* 111 (C7).

- [15] P. Thompson, Y. Cai, D. Reeve, J. Stander, Automated threshold selection methods for extreme wave analysis, *Coastal Engineering* 56 (10) (2009) 1013–1021.
- [16] F. Mazas, L. Hamm, A multi-distribution approach to pot methods for determining extreme wave heights, *Coastal Engineering* 58 (5) (2011) 385–394. doi:<https://doi.org/10.1016/j.coastaleng.2010.12.003>. URL <http://www.sciencedirect.com/science/article/pii/S0378383910001845>
- [17] G. Schwarz, et al., Estimating the dimension of a model, *The annals of statistics* 6 (2) (1978) 461–464.
- [18] H. Akaike, Information theory and an extension of the maximum likelihood principle, in: *Selected Papers of Hirotugu Akaike*, Springer, 1998, pp. 199–213.
- [19] V. Petrov, C. G. Soares, H. Gotovac, Prediction of extreme significant wave heights using maximum entropy, *Coastal engineering* 74 (2013) 1–10.
- [20] A. Durán-Rosal, J. Fernández, P. Gutiérrez, C. Hervás-Martínez, Detection and prediction of segments containing extreme significant wave heights, *Ocean Engineering* 142 (2017) 268–279.
- [21] M. Dorado-Moreno, L. Cornejo-Bueno, P. Gutiérrez, L. Prieto, C. Hervás-Martínez, S. Salcedo-Sanz, Robust estimation of wind power ramp events with reservoir computing, *Renewable Energy* 111 (2017) 428–437.
- [22] D. Guijo-Rubio, P. Gutiérrez, C. Casanova-Mateo, J. Sanz-Justo, S. Salcedo-Sanz, C. Hervás-Martínez, Prediction of low-visibility events due to fog using ordinal classification, *Atmospheric Research* 214 (2018) 64–73.
- [23] A. Durán-Rosal, J. Fernández, C. Casanova-Mateo, J. Sanz-Justo, S. Salcedo-Sanz, C. Hervás-Martínez, Efficient fog prediction with multi-objective evolutionary neural networks, *Applied Soft Computing* 70 (2018) 347–358.
- [24] K. Bowman, L. Shenton, Estimation: Method of moments, *Encyclopedia of statistical sciences* 3.
- [25] P. Jonathan, K. Ewans, Statistical modelling of extreme ocean environments for marine design: a review, *Ocean Engineering* 62 (2013) 91–109.
- [26] E. Mackay, L. Johanning, Long-term distributions of individual wave and crest heights, *Ocean Engineering* 165 (2018) 164–183.
- [27] E. Mackay, L. Johanning, A generalised equivalent storm model for long-term statistics of ocean waves, *Coastal Engineering*.
- [28] A. C. Davison, R. L. Smith, Models for exceedances over high thresholds, *Journal of the Royal Statistical Society. Series B (Methodological)* (1990) 393–442.
- [29] J. Ferreira, C. G. Soares, An application of the peaks over threshold method to predict extremes of significant wave height, *Journal of Off-shore Mechanics and Arctic Engineering* 120 (3) (1998) 165–176.
- [30] S. Caires, A. Sterl, 100-year return value estimates for ocean wind speed and significant wave height from the era-40 data, *Journal of Climate* 18 (7) (2005) 1032–1048.
- [31] C. G. Soares, M. Scotto, Application of the largest-order statistics for long-term predictions of significant wave height, *Coastal Engineering* 51 (5-6) (2004) 387–394.
- [32] C. N. Stefanakos, G. A. Athanassoulis, Extreme value predictions based on nonstationary time series of wave data, *Environmetrics: The official journal of the International Environmetrics Society* 17 (1) (2006) 25–46.
- [33] G. Huerta, B. Sansó, Time-varying models for extreme values, *Environmental and Ecological Statistics* 14 (3) (2007) 285–299.
- [34] V. G. Panchang, R. C. Gupta, On the determination of three-parameter weibull mle's, *Communications in Statistics-Simulation and Computation* 18 (3) (1989) 1037–1057.
- [35] Y. Goda, *Random seas and design of maritime structures*, Vol. 33, World Scientific Publishing Company, 2010.
- [36] J. R. Hosking, J. R. Wallis, Parameter and quantile estimation for the generalized pareto distribution, *Technometrics* 29 (3) (1987) 339–349.
- [37] S. Kang, J. Song, Parameter and quantile estimation for the generalized pareto distribution in peaks over threshold framework, *Journal of the Korean Statistical Society* 46 (4) (2017) 487–501.
- [38] I. M. Chakravarty, J. Roy, R. G. Laha, *Handbook of methods of applied statistics*.
- [39] E. S. Pearson, H. O. Hartley, *Biometrika tables for statisticians*, Cambridge University Press, 1966.
- [40] J. Pickands, Statistical inference using extreme order statistics, *the Annals of Statistics* (1975) 119–131.
- [41] <http://www.ndbc.noaa.gov/>, National buoy data center, National Oceanic and Atmospheric Administration of the USA (NOAA), 2015.

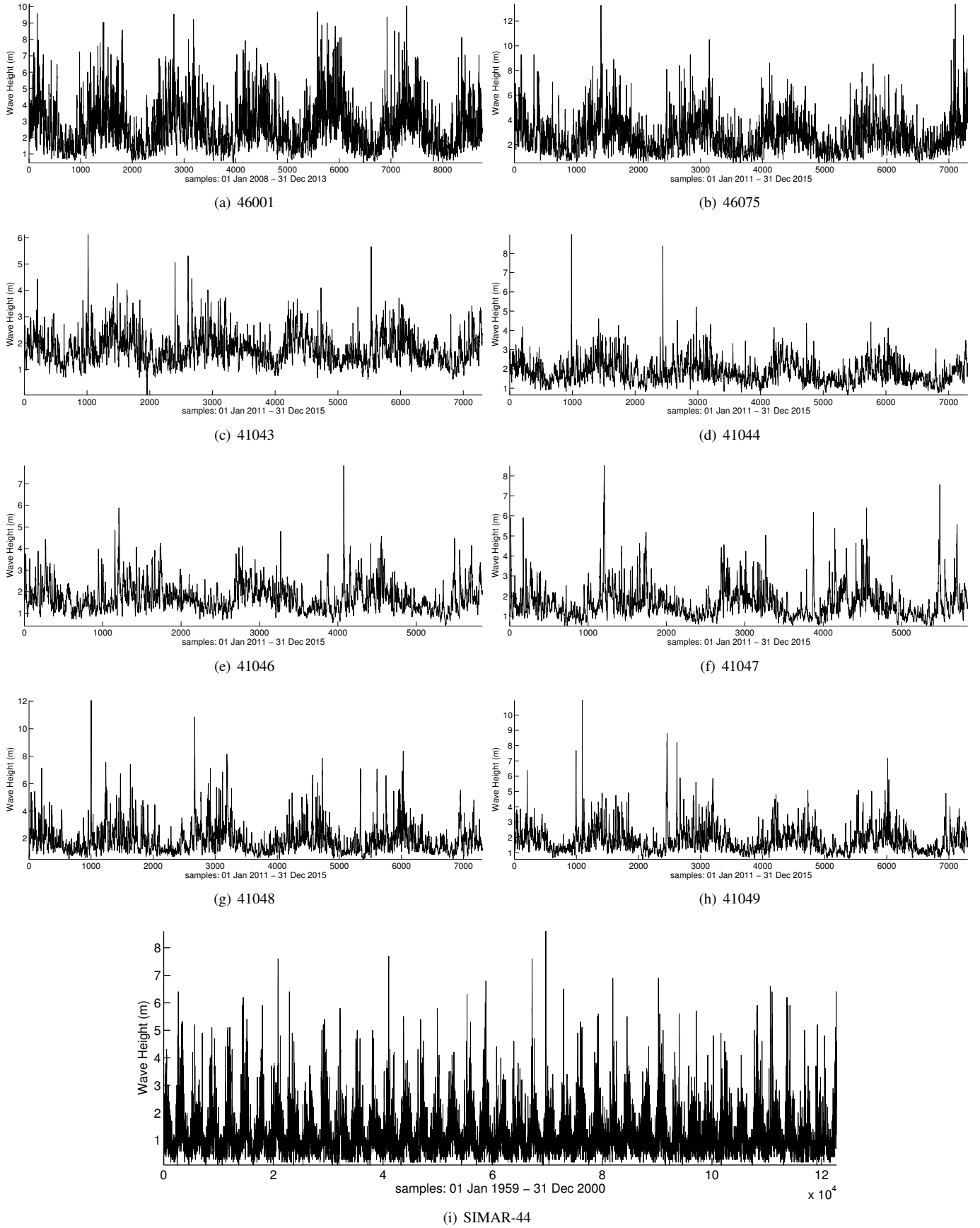


Figure 2: Graphical representation of the time series recorded for every buoy.

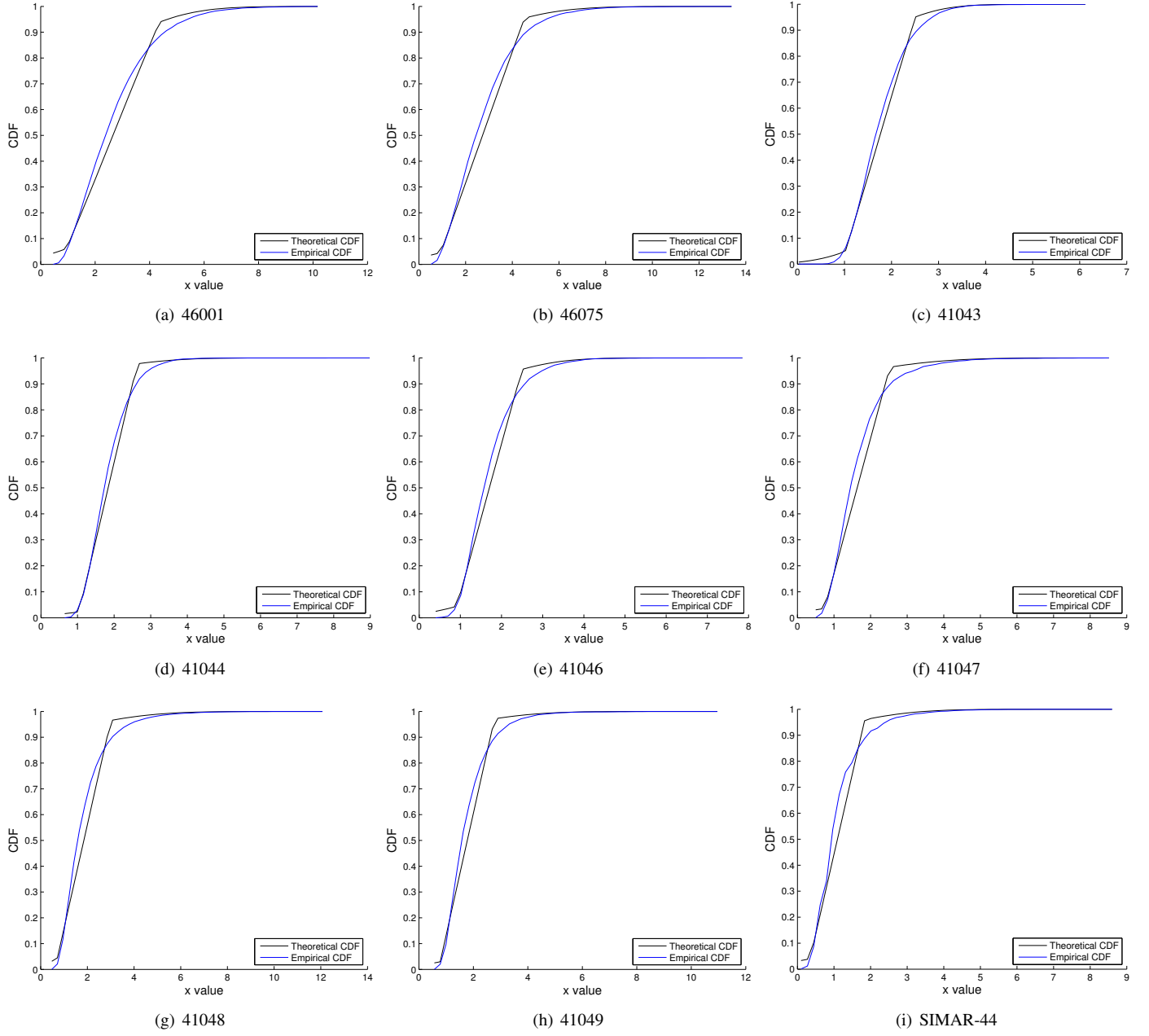


Figure 3: Estimated theoretical distribution versus empirical distribution in all wave height time series considered.

## 6.2. Distribution-based discretisation and ordinal classification applied to wave height prediction

The last work presented in this Thesis is related to the ordinal classification of segments with the aim to predict subsequences in a time series. To do so, the discretisation of the values of the time series is made as preprocessing of the time series. In many works, this discretisation is made according to the criterion of an expert, but, in this work, we propose to use the best-fitted distribution according to the time series. For this, we propose a methodology based on two phases.

In the first phase, we analyse the best-fitted distribution over the values of the time series. For that, we consider the GEV distribution, the normal distribution, the Weibull distribution and the Logistic distribution. Using training data, we apply an MLE method to adjust the parameters of the four distributions. Then, the best-fitted distribution is selected base on two objectives criteria, BIC and AIC. When the best distribution is adjusted, the corresponding 25 %, 50 % and 75 % are selected to be the thresholds ( $Q_1$ ,  $Q_2$  and  $Q_3$ ) to discretise the output variable in training and test sets.

In the second stage, we label the output in four categories:  $y_t \in C_1, C_2, C_3, C_4$ , where  $C_1$  ( $y_t \leq Q_1$ ) represents LOW wave height,  $C_2$  ( $y_t \in (Q_1, Q_2]$ ) represents AVERAGE wave height,  $C_3$  ( $y_t \in (Q_2, Q_3]$ ) represents BIG wave height, and, finally,  $C_4$  ( $y_t > Q_3$ ) represents HUGE wave height. The new dataset is defined as  $\mathbf{D} = (\mathbf{X}, \mathbf{Y}) = \{(\mathbf{x}_t, y_t)\}_{t=1}^n$ , where  $y_t$  is the target discretised category and  $\mathbf{x}_t$  is a set of inputs based on the previous events,  $\mathbf{x}_t = \{\mathbf{x}_{t-1}, y_{t-1}, \mathbf{x}_{t-2}, y_{t-2}, \dots, \mathbf{x}_{t-m}, y_{t-m}\}$ . As can be seen, there exists a natural order between the labels so that we can transform the prediction problem into an ordinal classification one. In this paper, we use the following ordinal classifiers: proportional odds models, kernel discriminant learning for ordinal regression and three variants of SVMs, specifically designed for ordinal classification.

The proposed method is tested in two real-world datasets showing that the best-fitted distribution is the GEV one. After that, the best classifier was the Reduction applied to SVM (REDSVM) which achieved the best performance in both.



# Distribution-Based Discretisation and Ordinal Classification Applied to Wave Height Prediction

David Guijo-Rubio<sup>(✉)</sup>, Antonio M. Durán-Rosal, Antonio M. Gómez-Orellana,  
Pedro A. Gutiérrez, and César Hervás-Martínez

Department of Computer Science and Numerical Analysis, Universidad de Córdoba,  
Córdoba, Spain

{dguijo, aduran, am.gomez, pagutierrez, chervas}@uco.es

**Abstract.** Wave height prediction is an important task for ocean and marine resource management. Traditionally, regression techniques are used for this prediction, but estimating continuous changes in the corresponding time series can be very difficult. With the purpose of simplifying the prediction, wave height can be discretised in consecutive intervals, resulting in a set of ordinal categories. Despite this discretisation could be performed using the criterion of an expert, the prediction could be biased to the opinion of the expert, and the obtained categories could be unrepresentative of the data recorded. In this paper, we propose a novel automated method to categorise the wave height based on selecting the most appropriate distribution from a set of well-suited candidates. Moreover, given that the categories resulting from the discretisation show a clear natural order, we propose to use different ordinal classifiers instead of nominal ones. The methodology is tested in real wave height data collected from two buoys located in the Gulf of Alaska and South Kodiak. We also incorporate reanalysis data in order to increase the accuracy of the predictors. The results confirm that this kind of discretisation is suitable for the time series considered and that the ordinal classifiers achieve outstanding results in comparison with nominal techniques.

**Keywords:** Wave height prediction · Distribution fitting  
Time series discretisation · Autoregressive models  
Ordinal classification

---

This work has been subsidised by the projects with references TIN2017-85887-C2-1-P and TIN2017-90567-REDT of the Spanish Ministry of Economy and Competitiveness (MINECO), FEDER funds, and the project PI15/01570 of the Fundación de Investigación Biomédica de Córdoba (FIBICO). David Guijo-Rubio's and Antonio M. Durán-Rosal's researches have been subsidised by the FPU Predoctoral Program (Spanish Ministry of Education and Science), grant references FPU16/02128 and FPU14/03039, respectively.



*You cannot discover new oceans unless you have  
the courage to lose sight of the shore.*

André Gide

# 7

## Discussion and conclusions

The final chapter of this Thesis includes the main conclusions resulting from the previous research. Also, opened research lines are outlined for future works.

### 7.1. Conclusions

This Thesis presents research performed on TSDM problems. As stated in the Motivation chapter, the Thesis has been organised in four main work lines: preprocessing, time series segmentation, prediction and distribution-based learning. In this section, we summarise the contributions grouped by these topics.

#### 7.1.1. Preprocessing

The Thesis contribution begins with Chapter 3, which includes a novel technique for the reconstruction of missing values in SWH time series collected from buoys placed in Alaska. State-of-the-art methodologies include ANN models, but up to the author's knowledge, there are no previous works where the basis functions of these ANNs are PUs. Moreover, the structure and connections of the PUNNs are optimised using an evolutionary strategy (EPUNN).

Specifically, the work produced within this topic is focused on the reconstruction

of wide gaps of missing information in different parts of the Gulf of Alaska. We use coastal buoys (id numbers 46061, 46076 and 46082) and offshore buoys (46001, 46078 and 46085). The work proposes a two-stage methodology. Firstly, the method calculates the correlation between complete parts of different time series. Then, those time series which are complete and for which the relation with the incomplete ones is higher than a predefined threshold, are used as independent variables for the reconstruction of the gaps with linear models, as the same way than the use of transfer functions and neighbour techniques. Once the first stage is completed, each time series is reconstructed again used ANN models optimised with an EA.

The results confirm that the proposed two-stage methodology outperforms previous linear models. Furthermore, EPUNN models result in the best basis functions, showing that they can reconstruct the missing part of the time series using a simple model with a hidden layer and two or three hidden neurons. These model can be rewritten as linear models of logarithmic functions. The main drawback of the method is that, for wave height over six metres, it can produce a slight underestimation of wave height.

According to the objectives established in the Chapter 2, Chapter 3 satisfies objectives 1, 2 and part of objective 8.

### 7.1.2. Segmentation

Time series segmentation is the main topic solved in Chapter 4. As we stated in that Chapter, the segmentation is an operation which consists in cutting the time series into several cut points with the aim of satisfying different objectives. The two main points of view include segmentation for discovering useful patterns in the time series and segmentation for the simplification of the time series by reducing the number of points.

Concerning the first objective, firstly, we propose a GA for the detection and design of early warning signals of TPs in paleoclimate data. Specifically, we use data from the GISP2 and the NGRIP  $\delta^{18}\text{O}$  time series with a 20-year resolution. The main objective of this work is to determine the nature of TPs called DO events with a GA in combination with a clustering technique. The method is able to find common characteristics that occur before a DO event, allowing us to construct an early warning signal corresponding to an increase in autocorrelation, variance and mean square error.

In the second proposal, the GA is hybridised with a proper likelihood-based segmentation assuming that points in the time series follow a beta distribution, designed explicitly for correctly representing extreme values. In this way, we use SWH time series where extreme values are present. The methodology can segment and group in a cluster those events which correspond with the extreme values, and the likelihood-based segmentation improves the quality of the clustering. Also, in this work, an empirical validation is

made to determine which is the best way to apply the LS in the GA, which turns to be the application of the LS to the best 20 % of the individuals in the last generation.

Financial time series are very important in TSDM. We also propose an HA similar to the previous one, but the likelihood-based segmentation assumes that time series are normally distributed. It does not make sense to consider extreme distributions in the segmentation of stock market data because there are no extreme values within them. The methodology is tested in European stock market indexes, specifically over closing prices, with the aim of determining common phases between them. The HA improves the results of the standard GA, and also, the analysis of the segmentation produced and the socio-economic phases and events of the literature is made. The segmentation produces five groups of segments: the first one corresponds with the broadening phases, the second is related to Wedges, the third one represents crashes in time series (Downtrend patterns), cluster 4 presents the lowest autocorrelation, and cluster 5 is associated with increasing periods (Uptrend). Concluding, this algorithm without prior information of the financial time series analysed, is able to automatically determine common phases in the time series related to the financial patterns defined in [10].

Concerning the segmentation for the reduction of the number of points of the time series, in this Thesis, the main contributions are organised in two international journals and three national/international conferences. The two main contributions propose two new bioinspired algorithms. In the first one, an SCRO is proposed with the aim to guide the search without the necessity of establishing any configuration parameter of the algorithm. In this way, the algorithm dynamically adapts the parameters according to the statistics of centralisation and dispersion of the fitness distribution. Furthermore, a new hybridisation procedure is made combining Bottom-Up and Top-Down algorithms. For the experimental validation, several time series collected from different sources and scopes, are used, and the results agree that the SCRO methodology significantly outperforms the rest of the state-of-the-art algorithms. When compared to standard CRO, SCRO gets better results, but they are not statistically significant. However, our method does not need to specify any parameter as we stated before.

The second idea is a modification of the BBPSO MH, that is, our proposal is a new DBBPSO that automatically adapts the cognitive and social components in the evolutionary process. In this way, the social component is higher at the beginning of the evolution, and the cognitive one is higher at the end. It causes a decrease of the error approximation, improving the results from the rest of state-of-the-art algorithms, and when compared to an optimal algorithm (Salotti) [78], our methodology finds solutions close to the optimal with a lower computational cost.

Finally, the optimisation of the two previous objectives is tackled using a new MOEA,

assuming that both objectives are conflicting. To represent both objectives, we consider the optimisation of the clustering quality of the segments and the error of approximation. Firstly, the most conflicting quality clustering index (when compared to mean squared error) is selected from a set of 9 possibilities, using four Pareto front evaluation metrics. In this way, the quality clustering index is decided to be the Silhouette index. Then, the methodology is compared against other state-of-the-art mono-objective algorithms given that there are no alternative multiobjective algorithms for time series segmentation. The results confirm that the objectives are in conflict, and our algorithm can show a good trade-off of the solutions. Also, when optimising clustering quality, we show that the number of segments is lower than when the optimisation is focused on the approximation error.

Summarising, Chapter 4 (time series segmentation proposals) is based on eight international journal papers and eight national/international conference papers. In this Chapter, we achieve the objectives 3, 4, 5, and part of objective 8.

### 7.1.3. Prediction

Prediction in time series has been barely studied in the literature and, in general, from a statistical point of view. The prediction in the literature is frequently tackled using real values. In this Thesis, we propose several ways to make predictions using higher levels of representation instead of the real values of the time series.

Firstly, we propose a novel methodology divided into two stages. For the first one, a GA in combination with a likelihood-based segmentation assuming a beta distribution is developed with the objective of detecting extreme values in SWH time series. Then, once the segmentation is done, the cluster which contains higher values in relation with other waves close in time is labelled as extreme event. The algorithm is able to create a database for the second stage. The second stage corresponds with the prediction of the extreme events previously detected. Each pattern of the database is made up of five characteristics of the three previous segments and the output represents whether the next segment is an extreme one or not (binary label). To achieve this prediction, and given the imbalanced nature of the dataset, we use an MOEA for training ANNs, with the aim to optimise the global accuracy and the accuracy of the worst classify class (minimum sensitivity). The method is tested using two time series collected in the Gulf of Alaska. Methodology shows a great performance when compared to other state-of-the-art machine learning algorithms (LR, simple LR, SVM, C4.5 DT, and Random Forest) and their cost-sensitive version (that is, the version of the algorithm which takes into account the imbalanced nature of the problem). To conclude with this work, the algorithm is validated with the AUC and MS metrics, which are better when evaluating imbalanced datasets.

The second prediction methodology has been applied to the field of fog formation

in a real-world problem extracted from the Valladolid airport. Given that aviation is one of the most affected phenomena by weather conditions, fog formation has a significant impact for operation procedures in the airport due to the low visibility produced when it is present. In airports, the decisions about low visibility are taken by the operators, and our goal is to automatically predict the occurrence of this weather condition with a 6-hour resolution. For that, we use different variables collected from sensors situated in the runway of the airport. To solve this problem, we use an MOEA for training ANNs with the aim to construct a model, compound by physical variables, with efficiently predicts the majority and the minority class in the dataset (normal or fog periods, respectively). The methodology is compared against other state-of-the-art algorithms, including a persistence model whose binary decision consists in the rule  $Y_t = Y_{t-1}$ . This algorithm can provide a high performance due to the consecutive constant values in the dataset, but it cannot support the decision in the airport because it does not take into account the variability and the nature of fog. For the experiments, our methodology outperforms the rest of methods. Moreover, we obtain a simple model which can be physically analysed from the point of view of the radiation fog formation mechanism. In this way, the model shows that an increment of the wind speed causes an increase of no fog events, and that the direction is slightly correlated with fog and also the velocity, but both variables do not have much importance in the model. When the temperature increases, the fog formation probability decreases, which is obvious. Finally, an increment of the air humidity results in an increase of fog probability, and an increment of the pressure decreases the probability of fog formation.

From the previous comments, we confirm that the achieved goals in the publications of Chapter 5 are objective 6 and part of objective 8.

#### 7.1.4. Statistical-distribution based learning

In Chapter 6, we explore the idea of fitting the statistical distribution of the time series for guiding posterior operations. That is, taking into account the best-fitted distribution for a given time series improve the performance of subsequent tasks.

In this Chapter, we first present a new theoretical way to determine the statistical distribution of wave height time series. In literature, authors usually fit the distribution of the values over a given threshold (POT approaches). The main problem is that the threshold needs to be specified, and the distribution of the whole time series is not used nor fitted. We propose not only the theory to determine the statistical distribution of the values situated over a threshold but also the distribution of the complete time series. The methodology is based on the idea that a wave height time series is sampled from a mixed distribution including normally distributed extreme values and uniformly distributed

values close to the mean of the time series. The use of the method of the moments for the determination of the four parameters related to this kind of distribution is suitable for this problem, which is shown by a Kolmogorov-Smirnov test of critical differences between real and theoretical distributions. The methodology, which is tested in different time series collected from Puerto Rico, Alaska and Spain (Gibraltar), is then used to theoretically define the threshold for the posterior POT analysis. Results confirm that the proposed methodology works well when defining the threshold, and the best statistical distribution over the values of this threshold is the GPD, which agrees with results of literature, but in a theoretical way.

The second methodology consists of an estimation of the best-fitted distribution selected from a set of distributions in time series. The selection of the best-fitted distribution is made using two criteria objectives, which are BIC and AIC. Both criteria agree that the best-fitted distribution is the GEV. Then, the percentiles of the estimated theoretical distribution are used for discretising the values of the time series, and this discretisation is used for inducing a classification problem. It can be seen that the resulting problem should be solved by an ordinal classification task, given that it comes from a discretisation made by consecutive percentile values. Results agree that the best predictor (classifier) is the REDSVM, which achieved the best performance in the two tested datasets.

This Chapter satisfies objective 7 and part of objective 8.

## 7.2. Generic discussion and future work

In the present Thesis, we have proposed several works concerning TSDM. A total of 12 international journal papers and 11 national/international conference papers summarise our proposals in: 1) preprocessing of time series for the reconstruction of missing values; 2) time series segmentation for discovering useful patterns, for reducing the number of points in the time series and for optimising both objectives at the same time; 3) prediction in time series using segments obtained in previous segmentation procedures; and 4) statistical distribution determination for guiding posterior tasks.

One of the contributions is focused on the production of new bioinspired algorithms, such as DBBePSO, new GAs and HAs, SCRO algorithms and EANNs. Also, the use of ML techniques in new areas of application is considered during the Thesis. The adjustment of the statistical distribution of the time series is used as a prerequisite to guide different operations.

All of these contributions in the different tasks have been applied to different real-world problems, which are the detection of TPs in paleoclimate data, the analysis of trends and phases in stock market indexes (financial data), the reconstruction of massive missing

data values in SWH time series, the detection of the highest wave heights in oceanographic data, the prediction of segments containing these wave heights in oceanographic data, predicting fog formation in airports and fitting statistical distribution for establishing the threshold for POT method in oceanographic data. So, from the previous comments, we can conclude that all the objectives presented in Chapter 2 have been addressed.

As future work, several promising lines can be introduced. Firstly, the reconstruction of missing values in time series could be tackled by considering multiobjective algorithms, where the optimisation will be done taking into account the reconstruction of values from 0 to 6m, but without losing precision in values over 6m. Secondly, a better optimisation of the segmentation methods for reducing the number of points of a time series could be introduced in order to minimise the error of approximation until reaching the error produced for optimal methods (which are much more costly).

The adjustment of the statistical distribution of the complete time series for guiding different operations is a recent research line, which entails an open challenge for future research lines. For example, the determination of the statistical distribution in other kinds of time series, such as in financial time series, could refine the likelihood-based segmentation performed assuming a distribution which has not been fitted.

Finally, based the knowledge and work done during the research stay, another current and future line could be the analysis of time series from the point of view of dynamical systems. It implies the study of the main anomaly detection methods in these systems, which includes one class SVM (OCSVM) and the support vector data descriptor (SVDD), whose aim is to differentiate normal states from those which are not. Specifically, the combination of echo state neural networks and the hidden Markov models (HMM) for the creation of dynamical systems which work in the state space should result in a better model than the AR HMM of the literature.





## References

- [1] Annex 3 to the Convention on International Civil Aviation: Meteorological Service for International Air Navigation, Eighteenth Edition 2013.
- [2] J. Abonyi, B. Feil, S. Nemeth, and P. Arva. Fuzzy clustering based segmentation of time-series. In *Advances in Intelligent Data Analysis V*, pages 275–285. Springer, 2003.
- [3] S. Aghabozorgi, A. S. Shirkhorshidi, and T. Y. Wah. Time-series clustering—a decade review. *Information Systems*, 53:16–38, 2015.
- [4] A. Altunkaynak. Prediction of significant wave height using geno-multilayer perceptron. *Ocean Engineering*, 58(0):144–153, 2013.
- [5] K. K. Andersen, N. Azuma, J.-M. Barnola, M. Bigler, P. Biscaye, N. Caillon, J. Chappellaz, H. B. Clausen, D. Dahl-Jensen, H. Fischer, et al. High-resolution record of northern hemisphere climate extending into the last interglacial period. *Nature*, 431(7005):147, 2004.
- [6] L. Bianchi, M. Dorigo, L. M. Gambardella, and W. J. Gutjahr. A survey on metaheuristics for stochastic combinatorial optimization. *Natural Computing*, 8(2):239–287, 2009.
- [7] C. M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.
- [8] T. Boneh, G. Weymouth, P. Newham, R. Potts, J. Bally, A. Nicholson, and K. Korb. Fog forecasting for Melbourne airport using a Bayesian decision network. *Weather and Forecasting*, 30(5):1218–1233, 2015.
- [9] S. Caires and A. Sterl. 100-year return value estimates for ocean wind speed and significant wave height from the era-40 data. *Journal of Climate*, 18(7):1032–1048, 2005.

- [10] F.-L. Chung, T.-C. Fu, V. Ng, and R. W. Luk. An evolutionary approach to pattern-based time series segmentation. *IEEE Transactions on Evolutionary Computation*, 8(5):471–489, 2004.
- [11] M. Clerc and J. Kennedy. The particle swarm-explosion, stability, and convergence in a multidimensional complex space. *IEEE Transactions on Evolutionary Computation*, 6(1):58–73, 2002.
- [12] C. A. C. Coello, G. B. Lamont, D. A. Van Veldhuizen, et al. *Evolutionary algorithms for solving multi-objective problems*, volume 5. Springer, 2007.
- [13] S. Coles, J. Bawa, L. Trenner, and P. Dorazio. *An introduction to statistical modeling of extreme values*, volume 208. Springer, 2001.
- [14] V. Dakos, S. R. Carpenter, W. A. Brock, A. M. Ellison, V. Guttal, A. R. Ives, S. Kefi, V. Livina, D. A. Seekell, E. H. Van Nes, et al. Methods for detecting early warnings of critical transitions in time series illustrated using simulated ecological data. *PLoS One*, 7(7):e41010, 2012.
- [15] A. C. Davison and R. L. Smith. Models for exceedances over high thresholds. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 393–442, 1990.
- [16] C. M. Dayton. Logistic regression analysis. *Stat*, pages 474–574, 1992.
- [17] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, April 2002.
- [18] R. Durbin and D. Rumelhart. Products units: A computationally powerful and biologically plausible extension to backpropagation networks. *Neural Computation*, 1(1):133–142, 1989.
- [19] D. Dutta and S. Chaudhuri. Nowcasting visibility during wintertime fog over the airport of a metropolis of India: decision tree algorithm and artificial neural network approach. *Natural Hazards*, 75:1349–1368, 2015.
- [20] R. D. Edwards, J. Magee, and W. Bassetti. *Technical Analysis of Stock Trends*. CRC Press, 10th edition, 2013.
- [21] M. Ester, H.-P. Kriegel, J. Sander, X. Xu, et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Kdd*, volume 96, pages 226–231, 1996.

- [22] D. Fabbian, R. De-Dear, and S. Lellyett. Application of artificial neural network forecasts to predict fog at canberra international airport. *Weather and Forecasting*, 22:372–381, 2007.
- [23] C. Faloutsos, M. Ranganathan, and Y. Manolopoulos. *Fast subsequence matching in time-series databases*, volume 23. ACM, 1994.
- [24] J. Ferreira and C. G. Soares. An application of the peaks over threshold method to predict extremes of significant wave height. *Journal of Offshore Mechanics and Arctic Engineering*, 120(3):165–176, 1998.
- [25] T.-C. Fu. A review on time series data mining. *Engineering Applications of Artificial Intelligence*, 24(1):164 – 181, 2011.
- [26] H. R. Glahn and D. A. Lowry. The use of model output statistics (MOS) in objective weather forecasting. *Journal of Applied Meteorology*, 11(8):1203–1211, 1972.
- [27] F. Glover. Tabu search—part i. *ORSA Journal on computing*, 1(3):190–206, 1989.
- [28] L. Gonzalez, J. G. Powell, J. Shi, and A. Wilson. Two centuries of bull and bear market cycles. *International Review of Economics & Finance*, 14(4):469 – 486, 2005.
- [29] J. D. Hamilton. *Time series analysis*, volume 2. Princeton university press Princeton, NJ, 1994.
- [30] S. Haykin. *Neural Networks: A Comprehensive Foundation*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 2nd edition, 1998.
- [31] M. A. Hearst, S. T. Dumais, E. Osuna, J. Platt, and B. Scholkopf. Support vector machines. *IEEE Intelligent Systems and their applications*, 13(4):18–28, 1998.
- [32] G. R. Herman and R. S. Schumacher. Using Reforecasts to Improve Forecasting of Fog and Visibility for Aviation. *Weather and Forecasting*, 31(2):467–482, 2016.
- [33] C. Hervás, P. A. Gutierrez, M. Silva, and J. M. Serrano. Combining classification and regression approaches for the quantification of highly overlapping capillary electrophoresis peaks by using evolutionary sigmoidal and product unit neural networks. *Journal of Chemometrics: A Journal of the Chemometrics Society*, 21(12):567–577, 2007.
- [34] G. Hinton and T. Sejnowski. *Unsupervised Learning: Foundations of Neural Computation*. Computational Neuroscience. Mit Press, 1999.
- [35] J. H. Holland. *Adaptation in natural and artificial systems*. MIT Press, Cambridge, MA, USA, 1992.

- [36] C. R. Houck, J. A. Joines, M. G. Kay, and J. R. Wilson. Empirical investigation of the benefits of partial lamarckianism. *Evolutionary Computation*, 5(1):31–60, 1997.
- [37] <http://www.ndbc.noaa.gov/wavecalc.shtml>. National Data Buoy Center: How are significant wave height, dominant period, average period, and wave steepness calculated? National Oceanic and Atmospheric Administration of the USA (NOAA), accessed June 29, 2016.
- [38] J. A. Joines and M. G. Kay. Utilizing hybrid genetic algorithms. In *Evolutionary Optimization*, volume 48 of *International Series in Operations Research & Management Science*, pages 199–228. Springer US, 2002.
- [39] P. Jonathan and K. Ewans. Statistical modelling of extreme ocean environments for marine design: a review. *Ocean Engineering*, 62:91–109, 2013.
- [40] J. Kennedy. Bare bones particle swarms. In *Proceedings of the 2003 IEEE Swarm Intelligence Symposium. SIS'03*, pages 80–87, 2003.
- [41] J. Kennedy and R. Eberhart. Particle swarm optimization. In *Proceedings of International Conference on Neural Networks. ICNN'95*, volume 4, pages 1942–1948, Nov 1995.
- [42] E. Keogh, S. Chu, D. Hart, and M. Pazzani. Segmenting time series: A survey and novel approach. In *Data mining in time series databases*, pages 1–21. World Scientific, 2004.
- [43] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by simulated annealing. *Science*, 220(4598):671–680, 1983.
- [44] A. Kolen and E. Pesch. Genetic local search in combinatorial optimization. *Discrete Applied Mathematics*, 48(3):273 – 284, 1994.
- [45] M. Koziara, J. Robert, and W. Thompson. Estimating marine fog probability using a model output statistics scheme. *Monthly Weather Review*, 111:2333–2340, 1983.
- [46] T. M. Lenton. Early warning of climate tipping points. *Nature Climate Change*, 1(4):201–209, 2011.
- [47] M. Levy, H. Levy, and S. Solomon. A microscopic model of the stock market: Cycles, booms, and crashes. *Economics Letters*, 45(1):103 – 111, 1994.
- [48] T. W. Liao. Clustering of time series data—a survey. *Pattern Recognition*, 38(11):1857–1874, 2005.

- [49] R. P. Lippmann. Pattern classification using neural networks. *IEEE Communications Magazine*, 27(11):47–50, 1989.
- [50] Y. Liu and R. H. Wisberg. Patterns of ocean current variability on the west florida shelf using the self-organizing map. *Journal of Geophysical Research*, 110:C06003, 2005.
- [51] E. Mackay and L. Johanning. A generalised equivalent storm model for long-term statistics of ocean waves. *Coastal Engineering*, 140:411–428, 2018.
- [52] E. Mackay and L. Johanning. Long-term distributions of individual wave and crest heights. *Ocean Engineering*, 165:164–183, 2018.
- [53] J. MacQueen et al. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, pages 281–297. Oakland, CA, USA, 1967.
- [54] A. Martínez-Estudillo, F. Martínez-Estudillo, C. Hervás-Martínez, and N. García-Pedrajas. Evolutionary product unit based neural networks for regression. *Neural Networks*, 19(4):477–486, 2006.
- [55] F. J. Martínez-Estudillo, C. Hervás-Martínez, P. A. Gutiérrez, and A. C. Martínez-Estudillo. Evolutionary product-unit neural networks classifiers. *Neurocomputing*, 72(1-3):548–561, 2008.
- [56] C. Marzban, S. Leyton, and B. Colman. Ceiling and visibility forecasts via neural networks. *Weather and Forecasting*, 22:466–479, 2007.
- [57] M. Mathiesen, Y. Goda, P. J. Hawkes, E. Mansard, M. J. Martín, E. Peltier, E. F. Thompson, and G. Van Vledder. Recommended practice for extreme wave analysis. *Journal of Hydraulic Research*, 32(6):803–814, 1994.
- [58] F. Mazas and L. Hamm. A multi-distribution approach to pot methods for determining extreme wave heights. *Coastal Engineering*, 58(5):385 – 394, 2011.
- [59] W. S. McCulloch and W. H. Pitts. A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, 5:115–133, 1943.
- [60] Y. Miao, R. Potts, X. Huang, G. Elliott, and R. Rivett. A fuzzy logic fog forecasting model for Perth Airport. *Pure and Applied Geophysics*, 169:110–1119, 2012.
- [61] F. V. Nelwamondo, S. Mohamed, and T. Marwala. Missing data: A comparison of neural network and expectation maximisation techniques. *Current Science*, 93(11):1514–1521, 2007.

- [62] D. J. Olive. *Linear regression*. Springer, 2017.
- [63] J. Oliver and C. Forbes. Bayesian approaches to segmenting a simple time series. Technical Report 14/97, Monash University, Department of Econometrics and Business Statistics, 1997.
- [64] J. J. Oliver, R. A. Baxter, and C. S. Wallace. Minimum message length segmentation. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 222–233. 1998.
- [65] M. G. Omran, A. Engelbrecht, and A. Salman. Barebones particle swarm for integer programming problems. In *IEEE Swarm Intelligence Symposium.*, pages 170–175, 2007.
- [66] A. R. Pagan and K. A. Sossounov. A simple framework for analysing bull and bear markets. *Journal of Applied Econometrics*, 18(1):23–46, 2003.
- [67] V. G. Panchang and R. C. Gupta. On the determination of three-parameter weibull mle’s. *Communications in Statistics-Simulation and Computation*, 18(3):1037–1057, 1989.
- [68] V. Petrov, C. G. Soares, and H. Gotovac. Prediction of extreme significant wave heights using maximum entropy. *Coastal Engineering*, 74:1–10, 2013.
- [69] K. B. Pratt. *Locating Patterns In Discrete Time-Series*. PhD thesis, University of South Florida, Department of Computer Science and Engineering, 2001.
- [70] T. Rakthanmanon, E. J. Keogh, S. Lonardi, and S. Evans. MDL-based time series clustering. *Knowledge and Information Systems*, 33(2):371–399, 2012.
- [71] S. Rao and S. Mandal. Hindcasting of storm waves using neural networks. *Ocean Engineering*, 32:667–684, 2005.
- [72] C. R. Reeves. *Modern Heuristic Techniques for Combinatorial Problems*. Oxford: Blackwell, 1993.
- [73] C. Román-Cascón, G. Steeneveld, C. Yagüe, M. Sastre, J. Arrillaga, and G. Maqueda. Forecasting radiation fog at climatologically contrasting sites: evaluation of statistical methods and WRF. *Quarterly Journal of the Royal Meteorological Society*, 142(695):1048–1063, 2016.
- [74] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning representations by back-propagating errors. *Nature*, 323(6088):533, 1986.

- [75] S. R. Safavian and D. Landgrebe. A survey of decision tree classifier methodology. *IEEE Transactions on Systems, Man, and Cybernetics*, 21(3):660–674, 1991.
- [76] S. Salcedo-Sanz. A review on the coral reefs optimization algorithm: new development lines and current applications. *Progress in Artificial Intelligence*, 6:1–15, 2017.
- [77] S. Salcedo-Sanz, J. Del Ser, I. Landa-Torres, S. Gil-López, and J. Portilla-Figueras. The coral reefs optimization algorithm: a novel metaheuristic for efficiently solving optimization problems. *The Scientific World Journal*, 2014, 2014.
- [78] M. Salotti. An efficient algorithm for the optimal polygonal approximation of digitized curves. *Pattern Recognition Letters*, 22(2):215–221, 2001.
- [79] R. J. Schalkoff. *Artificial Neural Networks*, volume 1. McGraw-Hill New York, 1997.
- [80] M. Schmitt. On the complexity of computing and learning with multiplicative neural networks. *Neural Computation*, 14:241–301, 2002.
- [81] N. Setiawan, P. Venkatachalam, and A. Hani. Missing attribute value prediction based on artificial neural network and rough set theory. In *International Conference on BioMedical Engineering and Informatics*, volume 1, pages 306–310, 2008.
- [82] W. Sturges. On interpolating gappy records for time-series analysis. *Journal of Geophysical Research: Oceans*, 88(C14):9736–9740, 1983.
- [83] B.-Y. Sun, J. Li, D. D. Wu, X.-M. Zhang, and W.-B. Li. Kernel discriminant learning for ordinal regression. *IEEE Transactions on Knowledge and Data Engineering*, 22:906–910, 2010.
- [84] R. Thompson. Spectral estimation from irregularly spaced data. *IEEE Transactions on Geoscience Electronics*, 9(2):107–110, 1971.
- [85] R. E. Thomson and W. J. Emery. *Data Analysis Methods in Physical Oceanography*. Elsevier, 2014.
- [86] V. S. Tseng, C.-H. Chen, P.-C. Huang, and T.-P. Hong. Cluster-based genetic segmentation of time series with DWT. *Pattern Recognition Letters*, 30(13):1190–1197, 2009.
- [87] N. L. J. Ulder, E. H. L. Aarts, H.-J. Bandelt, P. J. M. v. Laarhoven, and E. Pesch. Genetic local search algorithms for the travelling salesman problem. In *Proceedings of the 1st Workshop on Parallel Problem Solving from Nature*, PPSN I, pages 109–116, London, UK, UK, 1991. Springer-Verlag.

- [88] A. M. Viselli, G. Z. Forristall, B. R. Pearce, and H. J. Dagher. Estimation of extreme wave and wind design parameters for offshore wind turbines in the Gulf of Maine using a POT method. *Ocean Engineering*, 104:649–658, 2015.
- [89] A. S. Weigend. *Time series prediction: forecasting the future and understanding the past*. Routledge, 2018.
- [90] G. M. Weiss. Mining with rarity: a unifying framework. *ACM Sigkdd Explorations Newsletter*, 6(1):7–19, 2004.
- [91] X. Xi, E. Keogh, C. Shelton, L. Wei, and C. A. Ratanamahatana. Fast time series classification using numerosity reduction. In *Proceedings of the 23rd international conference on Machine learning*, pages 1033–1040. ACM, 2006.
- [92] H. Zhao, Z. Dong, T. Li, X. Wang, and C. Pang. Segmenting time series with connected lines under maximum error bound. *Information Sciences*, 345:1–8, 2016.
- [93] B. Zhou and J. Du. Fog prediction from a multimodel mesoscale ensemble prediction system. *Weather and Forecasting*, 25(1):303–322, 2010.