Eiffel
GEOSS APPLICATIONS FOR CLIMATE CHANGE

REVEALING THE ROLE OF GEOSS AS THE DEFAULT DIGITAL PORTAL FOR BUILDING CLIMATE CHANGE ADAPTATION & MITIGATION APPLICATIONS

D4.1: Stochastic methods for temporal augmentation and quality improvement of time series datasets

Version 1.0
Date 30-Nov-2022

Editor Ioannis Tsoukalas (ICCS)
Authors Ioannis Tsoukalas (ICCS) and Christos Makropoulos (ICCS)
Reviewers Nikoforos Samarinas and Konstantinos Karyotis (IBEC)

Dissemination Level Public (PU)

Call H2020-LC-CLA-2020-2
Topic LC-CLA-19-2020
Type of Action Research and Innovation Action
Start Date 01 June 2021
Duration 36 months
Project Information https://cordis.europa.eu/project/id/101003518

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003518

The Members of the EIFFEL Consortium:

ID	Organisation	Short Name	Country
1	INSTITUTE OF COMMUNICATION AND COMPUTER SYSTEMS	ICCS	GREECE
2	ETHNIKO ASTEROSKOPEIO ATHINON	NOA	GREECE
3	PRODEVELOP SL	PRO	SPAIN
4	UNIVERSITAT POLITECNICA DE VALENCIA	UPV	SPAIN
5	DRAXIS ENVIRONMENTAL SA	DRAXIS	GREECE
6	STICHTING IHE DELFT INSTITUTE FOR WATER EDUCATION	IHE	NETHERLANDS
7	OPEN UNIVERSITEIT NEDERLAND	OUNL	NETHERLANDS
8	NOORD-BRABANT PROVINCIE	NOORD-BRABANT	NETHERLANDS
9	LIBRA MLI LTD	LIBRA	UNITED KINGDOM
10	DIABALKANIKO KENTRO PERIBALLONTOS	IBEC	GREECE
11	AUTORIDAD PORTUARIA DE BALEARES	BPA	SPAIN
12	UNIVERSIDAD AUTONOMA DE BARCELONA	UAB	SPAIN
13	PERIFEREIA ATTIKIS	ATTICA	GREECE
14	NATIONAL PAYING AGENCY	NPA	LITHUANIA
15	SCHWEIZERISCHES FORSCHUNGSINSTITUT FUER HOCHGEBIRGSKLIMA UND MEDIZIN IN DAVOS	PMOD WRC	SWITZERLAND
16	EUROPEAN CENTRE FOR MEDIUM-RANGE WEATHER FORECASTS	ECMWF	UNITED KINGDOM
17	SUOMEN YMPARISTOKESKUS	SYKE	FINLAND
18	RISA SICHERHEITSANALYSEN GMBH	RISA	GERMANY
19	EDGE IN EARTH OBSERVATION SCIENCES MONOPROSOPI IKE	EDGE	GREECE

Disclaimer

The information in this document is subject to change without notice. No warranty of any kind is made with regard to this document, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The Members of the EIFFEL Consortium shall not be held liable for errors contained herein or direct, indirect, special, incidental or consequential damages in connection with the use or performance of this material. The content of this document reflects only the authors' view. The European Commission and the Research Executive Agency are not responsible for any use that may be made of the information it contains.

Funded by the European Union

Document History

Version	Date	Editor	Comments
0.0	04.10.2022	Ioannis Tsoukalas	Template and table of contents
0.1	01.11.2022	Ioannis Tsoukalas	First draft
0.2	18.11.2022	Ioannis Tsoukalas, Dimitris Bliziotis, Nikoforos Samarinas	First review update
0.3	23.11.2022	Dimitris Bliziotis, Nikoforos Samarinas	Second stage review and formatting review
0.4	25.11.2022	Ioannis Tsoukalas	Second draft
1.0	30.11.2022	Ioannis Tsoukalas	Final draft and official release


Funded by the European Union

Executive summary

This document provides a detailed description of the methodologies developed within EIFFEL project aiming to address aspects related with the temporal augmentation and quality improvement of time series datasets (see WP4 and T4.1). More specifically, the proposed approaches consist of a toolkit designed to cope with three, common in time series modelling, challenges/problems, that is: 1) the generation of statistically consistent stochastic realizations, 2) infilling of time series missing values, and 3) Lower-scale extrapolation (i.e., downscaling) of timeseries statistics. Further to the above, the report demonstrates the application of the methods via representative datasets of a variety of variables, which are also of high interest for the project's pilots (e.g., precipitation, temperature, streamflow, air quality related quantities, etc.).

Funded by the European Union

Table of Contents

Funded by the European Union

List of figures

Funded by the European Union

Funded by the European Union

List of tables

Funded by the European Union

List of acronyms and abbreviations

Acronym	Meaning
AI	Artificial Intelligence
ALOS	Advance Land Observing satellite
CDF	Cumulative distribution function
DWD	Deutscher Wetterdienst
EO	Earth observation
GP	Gaussian process
ICDF	Inverse cumulative distribution function (also termed as quantile function)
KNMI	Koninklijk Nederlands Meteorologisch Instituut
MSE	Mean squared error
NA	No value is Available
PDF	Probability density function
RF	Random field
RV	Random variable

Funded by the European Union

1 Introduction

This report aims to provide a detailed description of the theoretical background and implementation aspects of methods/tools developed within T4.1 ("*Stochastic methods for temporal augmentation and quality improvement of time series datasets*" led by ICCS). The *Alpha* version of the methods (coded in R programming language) can be found in the project's repository and is employed for internal testing since MS9 (as dictated by the project's DOW for MS9's "*Means of verification*").

T4.1 is designed to address challenges, and common modelling tasks, related with the augmentation of the temporal dimension of time series data. It is reminded that T4.1 consists the design and development of theoretically justified methods and tools, based on statistical and probabilistic notions such as those of, time series analysis (e.g., Bras and Rodríguez-Iturbe 1985; Tsay 2013), stochastic processes (e.g., Papoulis 1991) and copulas (e.g., Sklar 1973; Embrechts et al. 2003; Nelsen 2007), to address three key challenges of CC-related time series datasets. More specifically, the developed methods/tools are coping with the following problems (P):

P.1: Infilling of time series missing values,
P.2: Generation of statistically consistent stochastic realizations for time series data, and
P.3: Lower-scale extrapolation of time series statistics (e.g., temporal downscaling of key statistical properties).

1.1 Context

1.1.1 Objectives

The deliverable D4.1 is directly linked with the following objective of EIFFEL project (according to the Grant Agreement[1]):

- (O2) EIFFEL will leverage techniques of Explainable AI to develop tangible indicators for CC impacts; it will also make use of super resolution, data fusion and stochastic modelling techniques to generate spatially and temporally explicit information from the untapped pool of GEOSS.

Since D4.1 is a method-oriented report, its outputs are indirectly related with all EIFFEL pilots, hence indirectly related with other objectives of the project.

1.1.2 Work plan

This report, Deliverable D4.1 corresponds to T4.1: Stochastic methods for augmenting the temporal resolution and quality of CC-related datasets (M3-M30) (Leader ICCS). It is part of WP4: Improving temporal, spatial resolution and data quality of CC related datasets (M3-M30) (Leader ICCS). The methods/tools developed within D4.1 could provide a valuable modelling toolkit for the EIFFEL pilots (i.e., WP7).

[1] See, Part B, Table 1.

Funded by the European Union

1.1.3 Milestones

D4.1 is linked to MS9[2] and MS10[3]

1.1.4 Deliverables

Due to its methodology/research-oriented nature D4.1 is to a large extend a standalone deliverable which details methods/tools related with the temporal augmentation of timeseries datasets. Hence the methods/tools described herein could be used by the EIFFEL pilots (i.e., WP7) in cases where such modelling activities are involved.

1.2 Intended readership and document structure

The dissemination level of this report is public. It is specifically intended for partners working on **WP4** (Improving temporal, spatial resolution and data quality of CC related datasets – Leader: ICCS) as well as **WP7** (EIFFEL Pilot demonstrations and impact assessment). The remainder of the document is structured as follows:

- **Section 2** provides the necessary theoretical background for the development of D4.1/T4.1 methods/tools.
- **Section 3** details the methodologies and implementation steps for: (§3.1) infilling of time series missing values, (§3.2) the generation of statistically consistent stochastic realizations, and (§3.3) Lower-scale extrapolation (i.e., downscaling) of timeseries statistics.
- **Section 4** utilizes more than 2400 time series datasets to demonstrate the functionality of the proposed methods (fulfilling this way the target set by the O2-related KPI-2.2[4]).

The document is summarized with the conclusions section (**Section 5**) and provides various Appendices functioning as supporting material.

[2] *Alpha* versions of D4.1 - D4.3 related components [M14 - Achieved].
[3] Final adjustments to the T4.1,4.2 algorithms after pilots first phase concludes [M30].
[4] KPI-2.2: At least 20 representative GEOSS time-series datasets spatiotemporally augmented as proof of concept; chosen collaboratively with CoPs stakeholders; Validation: WP4, WP7

Funded by the European Union

2 Notation, basic notions and introduction of key concepts

It is noted that the solutions delivered within this task (mainly those for P.1 and P.2) are built upon the concept of Gaussian copula (a notion closely related to that of Nataf's joint distribution (Nataf 1962; Der Kiureghian and Liu 1986; Tsoukalas et al. 2018a). Further to this it is noted that the methodology employed for P.3 – also in some extent linked with the above concept – is based on the notion of multi-scale modelling of stochastic processes.

With the above in mind, the following sections aim to provide an introduction to the theoretical concepts necessary for the development of the proposed methods to address P1-P3. Further implementation details and algorithmic step-by-step recipes, specifically designed P1-P3, are provided in Section 4.

It is also remarked that throughout this report, the underbar notation (e.g., $\underline{x}$) is used to denote a random variable (RV) or a stochastic process, while the italic typeface (e.g., x) is reserved to denote a realization of it (i.e., a non-random quantity/variable). Furthermore, unless stated otherwise, this report concerns univariate discrete-time processes with continuous or zero-inflated marginal distributions with finite variance as well as valid (i.e., positive definite) autocorrelation structures, which are also non-negative (since they are abundant in hydrometeorological processes).

2.1 Brief introduction to copulas

In simple terms, copulas are statistical tools that enable the construction of multivariate distribution models with arbitrary marginal distributions and given dependence structure (i.e., correlation), which in turn allows the modelling and simulation (unconditional and conditional) of non-Gaussian random variables and processes. For early-day developments, the interested reader is referred to the seminal works of Sklar (1973, 1959) as well as other authors (e.g., Fréchet 1951; Féron 1956; Dall'Aglio 1959; Nataf 1962; Mardia 1970), while more recent and general treatments on the topic are provided by Embrechts et al. (2003), Nelsen (2007) or Joe (Joe 2014).

In more detail, copulas have been introduced by Sklar (1973, 1959), more than half-a-century ago, and since then have found fruitful ground in a variety of scientific domains, including that of hydrology (e.g., Chen and Guo, 2019; Dupuis, 2007; Favre et al., 2004; Grimaldi and Serinaldi, 2006; Kossieris et al., 2019; Renard and Lang, 2007; Salvadori and De Michele, 2007, 2004; Tsoukalas et al., 2020, 2019, 2018a), since there exists no other theoretical framework that is as general and as flexible as copulas for multivariate modelling.

Among the several available copula models, this report is focused on the Gaussian copula since it is the only one (along with the student-t copula) that allows the straightforward modelling of more than two (2) random variables, which is typical the case operational interest.

To provide some context, and following the description of Tsoukalas (2018), let $\underline{\boldsymbol{x}} = [\underline{x}_1, \dots, \underline{x}_m]^{\mathrm{T}}$ denote a vector of m cross-correlated (yet, time-independent) random variables (RVs), indexed using i, each one characterized by an arbitrarily specified marginal distribution function $F_{\underline{x}_i}(x) \coloneqq P\{\underline{x}_i \leq x\}$, with finite variance; also referred to as cumulative distribution

Funded by the European Union

function (CDF). Let also $f_{\underline{x}_i}(x) \coloneqq \mathrm{d}F_{\underline{x}_i}(x)/\mathrm{d}x$ denote the corresponding univariate probability density function (PDF). Furthermore, let $\boldsymbol{R} \coloneqq \mathrm{Corr}\left[\underline{\boldsymbol{x}}, \underline{\boldsymbol{x}}^{\mathrm{T}}\right]$ denote their (target) correlation matrix $(m \times m)$.

Let also, $\underline{\boldsymbol{z}} = [\underline{z}_1, \dots, \underline{z}_m]^{\mathrm{T}}$ be a vector characterized by a m-dimensional multivariate standard normal distribution, i.e., $\underline{\boldsymbol{z}} \sim \mathcal{N}_m(\widetilde{\boldsymbol{\mu}}, \widetilde{\boldsymbol{\Sigma}})$, where $\widetilde{\boldsymbol{\mu}} \coloneqq \mathrm{E}\left[\underline{\boldsymbol{z}}\right] = \mathbf{0}^{\mathrm{T}}$ is the mean vector $(m \times 1)$ and $\widetilde{\boldsymbol{\Sigma}} \coloneqq \mathrm{Cov}\left[\underline{\boldsymbol{z}}, \underline{\boldsymbol{z}}^{\mathrm{T}}\right]$ is the covariance matrix $(m \times m)$, which has to be positive semi-definite and in the case of multivariate standard normal distribution is synonymous with its correlation matrix, $\widetilde{\boldsymbol{R}} \coloneqq \mathrm{Corr}\left[\underline{\boldsymbol{z}}, \underline{\boldsymbol{z}}^{\mathrm{T}}\right] = \widetilde{\boldsymbol{\Sigma}}$.

The multivariate standard normal CDF, $\mathcal{N}_m$ is denoted for simplicity as $\Phi_m\left(\boldsymbol{z}; \widetilde{\boldsymbol{R}}\right)$, while its multivariate PDF as $\varphi_m\left(\boldsymbol{z}; \widetilde{\boldsymbol{R}}\right)$. Notice that the mean, has been omitted for brevity. Apparently, each element of $\underline{\boldsymbol{z}}$ is also characterized by standard normal distribution, $\Phi(\cdot)$ with density $\varphi(\cdot)$, i.e., $\underline{z}_\xi \sim \mathcal{N}(0,1)$.

The main idea of Gaussian copula lies into establishing the multivariate joint distribution $F_{\underline{\boldsymbol{x}}}(\boldsymbol{x}) = F_{\underline{\boldsymbol{x}}}(x_1, \dots, x_m) = P\left(\underline{x}_1 \le x_1, \dots, \underline{x}_m \le x_m\right)$ of $\underline{\boldsymbol{x}}$ through the joint CDF of $\underline{\boldsymbol{z}}$. Particularly, by expressing each element of $\underline{\boldsymbol{z}}$ as,

$$\underline{z}_i = \Phi^{-1}\left(F_{\underline{x}_i}\left(\underline{x}_i\right)\right) \tag{2.1}$$

where $\Phi^{-1}(\cdot)$ denotes the quantile function, else known as inverse cumulative density function (ICDF), of the univariate standard normal distribution. It is straightforward to see that by employing the probability integral transformation to each marginal CDF we obtain $\underline{u}_i \coloneqq F_{\underline{x}_i}\left(\underline{x}_i\right)$ which is a uniformly distributed RV in $[0, 1]$ that denotes probability. See also, Papoulis (1991 p. 101). Nevertheless, through the rules of probability transformation, the joint distribution (CDF) of $\underline{\boldsymbol{x}}$ can be written as,

$$F_{\underline{\boldsymbol{x}}}(x_1, \dots, x_m) = \Phi_m\left(\Phi^{-1}\left(F_{\underline{x}_1}(x_1)\right), \dots, \Phi^{-1}\left(F_{\underline{x}_m}(x_m)\right); \widetilde{\boldsymbol{R}}\right) \tag{2.2}$$

which is identical with the definition provided for the Gaussian copula.

In brief copulas, denoted with $C(\cdot)$, are m-dimensional distribution functions on $[0, 1]^m$ with uniform marginal distributions. Sklar (1959), established the theory of copulas and provided their general properties. Among them, it has been shown that any multivariate joint distribution can be regarded as a copula function. Particularly, Sklar's theorem states that a multivariate distribution $F_{\underline{\boldsymbol{x}}}(\boldsymbol{x}) = F_{\underline{\boldsymbol{x}}}(x_1, \dots, x_m)$ with marginal CDFs $F_{\underline{x}_1}, \dots, F_{\underline{x}_m}$, assuming that they are with continuous and differentiable, can be written as,

$$F_{\underline{\boldsymbol{x}}}(x_1, \dots, x_m) = C\left(F_{\underline{x}_1}(x_1), \dots, F_{\underline{x}_m}(x_m)\right) \tag{2.3}$$

In this work we are focusing on the Gaussian copula $C^G(\cdot)$ which is defined as multivariate standard normal distribution with correlation matrix $\widetilde{\boldsymbol{R}}$ (e.g., Embrechts et al. 2003),

$$C^G(\boldsymbol{u}) = C(u_1, \dots, u_m) = \Phi_m\left(\Phi^{-1}(u_1), \dots, \Phi^{-1}(u_m); \widetilde{\boldsymbol{R}}\right) \tag{2.4}$$

which apparently, after some substitutions can be transformed in in Eq. (2.2).

Funded by the European Union

Generally, according to copula theory, assuming that both $F_{\underline{x}_i}$ and $C(\cdot)$ are differentiable, the joint PDF of $\underline{\boldsymbol{x}}$ can be written as,

$$f_{\underline{\boldsymbol{x}}}(x_1, \dots, x_m) = c\left(F_{\underline{x}_1}(x_1), \dots, F_{\underline{x}_m}(x_m)\right) \cdot \prod_{i=1}^{m} f_{\underline{x}_i}(x_i) \tag{2.5}$$

where $c(\cdot)$ denotes the joint PDF (referred also as copula density) of copula $C(\cdot)$ and it is given by,

$$c\left(F_{\underline{x}_1}(x_1), \dots, F_{\underline{x}_m}(x_m)\right) = c(u_1, \dots, u_m) = \frac{\partial^m C(u_1, \dots, u_m)}{\partial u_1 \cdots \partial u_m} \tag{2.6}$$

In the case of Gaussian copula the joint PDF of $\underline{\boldsymbol{x}}$ is given (cf. Liu and Der Kiureghian 1986),

$$f_{\underline{\boldsymbol{x}}}(x_1, \dots, x_m) = \frac{\varphi_m\left(\Phi^{-1}\left(F_{\underline{x}_1}(x_1)\right), \dots, \Phi^{-1}\left(F_{\underline{x}_m}(x_m)\right); \widetilde{\boldsymbol{R}}\right)}{\prod_{i=1}^{m} \varphi\left(\Phi^{-1}\left(F_{\underline{x}_i}(\underline{x}_i)\right)\right)} \cdot \prod_{i=1}^{m} f_{\underline{x}_i}(x_i) \tag{2.7}$$

From these equations it is clear that, copula theory, in general, as well as the Gaussian copula specifically, allow us to describe complex multivariate distributions using as individual components the marginal distributions $F_{\underline{x}_1}, \dots, F_{\underline{x}_m}$ and the copula $C(\cdot)$, which eventually allow the formulation of the joint distribution.

2.2 Parameter estimation for the Gaussian copula

Nevertheless, as evidenced by Eq. (2.2) and Eq. (2.7) in the case of Gaussian copula the joint distribution of $\underline{\boldsymbol{x}}$ depends on the correlation matrix $\widetilde{\boldsymbol{R}}$ of $\underline{\boldsymbol{z}}$ and not directly on $\boldsymbol{R}$ of $\underline{\boldsymbol{x}}$.

To elaborate, let us consider the inverse (yet equivalent) case where $\underline{\boldsymbol{x}}$ is obtained through $\underline{\boldsymbol{z}}$ via the following mapping equation:

$$\underline{x}_i = F_{\underline{x}_i}^{-1}\left(\Phi(\underline{z}_i)\right) \tag{2.8}$$

where $F_{\underline{x}_i}^{-1}$ is the ICDF of variable $\underline{x}_i$. It is noted that similar to the previous case (i.e., Eq. (2.1)) $\underline{u}_i := \Phi(\underline{z}_i)$ is also a RV uniformly distributed in $[0, 1]$ that denotes probability. A direct outcome of Eq. (2.8) is that for two variables $\underline{x}_i$ and $\underline{x}_j$ their correlation is given by:

$$\rho_{i,j} := \mathrm{Corr}[\underline{x}_i, \underline{x}_j] = \frac{\mathrm{E}[\underline{x}_i\, \underline{x}_j] - \mathrm{E}[\underline{x}_i]\, \mathrm{E}[\underline{x}_j]}{\sqrt{\mathrm{Var}[\underline{x}_i]\, \mathrm{Var}[\underline{x}_j]}} \tag{2.9}$$

where $\mathrm{E}[\underline{x}_i], \mathrm{E}[\underline{x}_j]$ and $\mathrm{Var}[\underline{x}_i], \mathrm{Var}[\underline{x}_j]$ are the mean and variance of $\underline{x}_i$ and $\underline{x}_j$ respectively, which are known since the associated marginal distributions are already specified (and have finite variance, otherwise the Pearson correlation coefficient cannot be defined), while $\mathrm{E}[\underline{x}_i\, \underline{x}_j]$ is given by,

$$\begin{aligned} \mathrm{E}[\underline{x}_i\, \underline{x}_j] &= \mathrm{E}\left[F_{\underline{x}_i}^{-1}\left(\Phi(\underline{z}_i)\right)\, F_{\underline{x}_j}^{-1}\left(\Phi(\underline{z}_j)\right)\right] \\ &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} F_{\underline{x}_i}^{-1}(\Phi(z_i))\, F_{\underline{x}_j}^{-1}\left(\Phi(z_j)\right) \varphi_2(z_i, z_j;\, \tilde{\rho}_{i,j}) \mathrm{d}z_i \mathrm{d}z_j \end{aligned} \tag{2.10}$$

Funded by the European Union

where $\varphi_2(z_i, z_j;\ \tilde{\rho}_{i,j})$ is the bivariate standard normal PDF.
By substituting Eq. (2.10) to Eq. (2.9) we obtain,

$$\rho_{i,j} = \frac{\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} F_{\underline{x}_i}^{-1}(\Phi(z_i))\, F_{\underline{x}_j}^{-1}\left(\Phi(z_j)\right)\varphi_2(z_i, z_j;\ \tilde{\rho}_{i,j})\mathrm{d}z_i\mathrm{d}z_j - \mathrm{E}[\underline{x}_i]\,\mathrm{E}[\underline{x}_j]}{\sqrt{\mathrm{Var}[\underline{x}_i]\,\mathrm{Var}[\underline{x}_j]}} \tag{2.11}$$

which for simplicity, let us rewrite it as,

$$\rho_{i,j} = \mathcal{F}\left(\tilde{\rho}_{i,j}\middle| F_{\underline{x}_i}, F_{\underline{x}_j}\right) \tag{2.12}$$

where $\mathcal{F}(\cdot)$ denotes an arbitrary function, which has the meaning that each target $\rho_{i,j}$ is a function of $\tilde{\rho}_{i,j}$ and the given marginal distributions $F_{\underline{x}_i}$ and $F_{\underline{x}_j}$.

In order to identify the values of $\tilde{\rho}_{i,j}$ that result in the target values $\rho_{i,j}$ Eq. (2.12) have to be inverted. i.e.,

$$\tilde{\rho}_{i,j} = \mathcal{F}^{-1}\left(\rho_{i,j}\middle| F_{\underline{x}_i}, F_{\underline{x}_j}\right) \tag{2.13}$$

In general, Eq. (2.12), and thus Eq. (2.13), do not have a general closed-form solution, with the exception of few special cases (Li and Hammond 1975; Cario and Nelson 1997; Crouse and Baraniuk 1999; Xiao 2014), yet it can be approximate with high accuracy using appropriate techniques. Herein, and unless stated otherwise, we employ a Monte-Carlo based approach, which has been proved effective and efficient (Tsoukalas et al. 2018a).
All the above highlight that the link between the target correlations $\rho_{i,j}$ of $\boldsymbol{R}$ with the corresponding elements $\tilde{\rho}_{i,j}$ of $\widetilde{\boldsymbol{R}}$. An apparent approach could be setting $\widetilde{\boldsymbol{R}} \equiv \boldsymbol{R}$, however, both theoretical and empirical evidence have indicated that this assumption will result in misspecification of the Gaussian copula model and lead to systematically underestimating correlations. The theoretical justification of this behaviour stems from the Pearson correlation coefficient itself, since it is not invariant under non-linear monotonic transformations, such as those imposed by the ICDFs (Embrechts et al. 1999 p. 8). Therefore, and except the trivial case of normal marginal distribution, in order to eliminate biases, it necessary to *a priori* identify the values of $\tilde{\rho}_{i,j}$.

2.3 Derivation of the conditional distribution

This section extends the rationale of Gaussian copula for the derivation of conditional distributions (Tsoukalas 2018) of RVs (and processes) with pre-specified distributions and correlation matrix.
Similarly to the previous sections, let $\underline{\boldsymbol{x}} = [\underline{x}_1, \dots, \underline{x}_m]^{\mathrm{T}}$ be a m-dimensional vector of RVs, with known distributions $F_{\underline{x}_1}, \dots, F_{\underline{x}_m}$ and correlation matrix $\boldsymbol{R}$, partitioned in a n-dimensional column-vector $\underline{\boldsymbol{x}}_1^* = [\underline{x}_1, \dots, \underline{x}_n]^{\mathrm{T}}$ and in a $(m-n)\times 1$ column-vector $\underline{\boldsymbol{x}}_2^* = [\underline{x}_{n+1}, \dots, \underline{x}_m]^{\mathrm{T}}$. Let also $\boldsymbol{h} = [x_{n+1}, \dots, x_m]^{\mathrm{T}}$ denote a vector of realizations of $\underline{\boldsymbol{x}}_2^*$ on which we wish to condition the derivation of the distribution of $\underline{\boldsymbol{x}}_1^*\,|\underline{\boldsymbol{x}}_2^* = \boldsymbol{h}$.
In order to derive the conditional distribution it suffice to derive the one of the auxiliary RVs $\underline{\boldsymbol{z}}$. This can be done by using well-known properties of the auxiliary multivariate standard normal

Funded by the European Union

distribution (e.g., Eaton 1983). Particularly, let the auxiliary m-dimensional vector $\underline{\boldsymbol{z}} = [\underline{z}_1, \dots, \underline{z}_m]^{\mathrm{T}}$ with $\underline{\boldsymbol{z}} \sim \mathcal{N}_m(\boldsymbol{0}, \widetilde{\boldsymbol{R}})$ be similarly partitioned in $\underline{\boldsymbol{z}}_1^* = [\underline{z}_1, \dots, \underline{z}_n]^{\mathrm{T}}$ and $\underline{\boldsymbol{z}}_2^* = [\underline{z}_{n+1}, \dots, \underline{z}_m]^{\mathrm{T}}$ with sizes $n \times 1$ and $(m-n) \times 1$ respectively. This allow us to partition the equivalent correlation matrix $\widetilde{\boldsymbol{R}}$ as follows (it is also noted that, $\widetilde{\boldsymbol{R}}_{12} = \widetilde{\boldsymbol{R}}_{21}^{\mathrm{T}}$),

$$\widetilde{\boldsymbol{R}} = \begin{bmatrix} \widetilde{\boldsymbol{R}}_{11} & \widetilde{\boldsymbol{R}}_{12} \\ \widetilde{\boldsymbol{R}}_{21} & \widetilde{\boldsymbol{R}}_{22} \end{bmatrix} \text{with sizes} \begin{bmatrix} n \times n & n \times (m-n) \\ (m-n) \times n & (m-n) \times (m-n) \end{bmatrix}$$

Furthermore, if $\underline{\boldsymbol{z}}_2^* = \widetilde{\boldsymbol{h}} = \left[\Phi^{-1}\left(F_{\underline{x}_{n+1}}(h_{n+1})\right), \dots, \Phi^{-1}\left(F_{\underline{x}_m}(h_m)\right)\right]^{\mathrm{T}}$ then the conditional distribution of $\underline{\boldsymbol{z}}_1^* | \underline{\boldsymbol{z}}_2^* = \widetilde{\boldsymbol{h}}$ is also multivariate normal, i.e., $P\left(\underline{\boldsymbol{z}}_1^* \le \boldsymbol{z}_1^* \middle| \underline{\boldsymbol{z}}_2^* = \widetilde{\boldsymbol{h}}\right) \sim \mathcal{N}_n(\overline{\widetilde{\boldsymbol{\mu}}}, \overline{\widetilde{\boldsymbol{\Sigma}}})$, where $\overline{\widetilde{\boldsymbol{\mu}}}$ and $\overline{\widetilde{\boldsymbol{\Sigma}}}$ are given by:

$$\overline{\widetilde{\boldsymbol{\mu}}} = \widetilde{\boldsymbol{R}}_{12}\widetilde{\boldsymbol{R}}_{22}^{-1}\widetilde{\boldsymbol{h}} \tag{2.14}$$

$$\overline{\widetilde{\boldsymbol{\Sigma}}} = \widetilde{\boldsymbol{R}}_{11} - \widetilde{\boldsymbol{R}}_{12}\widetilde{\boldsymbol{R}}_{22}^{-1}\widetilde{\boldsymbol{R}}_{21} \tag{2.15}$$

and denote respectively the conditional mean vector and covariance matrix. The matrix $\overline{\widetilde{\boldsymbol{\Sigma}}}$ can be easily calculated by exploiting the fact that it is Schur complement of $\widetilde{\boldsymbol{R}}_{22}$ in $\widetilde{\boldsymbol{R}}$. This allows the calculation of $\overline{\widetilde{\boldsymbol{\Sigma}}}$ via the inversion of the matrix $\widetilde{\boldsymbol{R}}$, the subsequent removal of columns and vectors that correspond to the variables conditioned upon (i.e., $\underline{\boldsymbol{z}}_2^*$), and finally $\overline{\widetilde{\boldsymbol{\Sigma}}}$ is obtained by the inversion of the remaining matrix.

Nevertheless, since Eq. (2.1) holds true, and similar to Eq. (2.2), the conditional CDF of $\underline{\boldsymbol{x}}_1^* | \underline{\boldsymbol{x}}_2^* = \boldsymbol{h}$ can be written as,

$$F_{\underline{\boldsymbol{x}}_1^* | \underline{\boldsymbol{x}}_2^* = \boldsymbol{h}}(\boldsymbol{x}_1^*) = P\left(\underline{\boldsymbol{x}}_1^* \le \boldsymbol{x}_1^* \middle| \underline{\boldsymbol{x}}_2^* = \boldsymbol{h}\right) = \Phi_{n;\overline{\widetilde{\boldsymbol{\mu}}},\overline{\widetilde{\boldsymbol{\Sigma}}}}\left(\Phi^{-1}\left(F_{\underline{x}_1}(x_1)\right), \dots, \Phi^{-1}\left(F_{\underline{x}_n}(x_n)\right); \overline{\widetilde{\boldsymbol{\mu}}}, \overline{\widetilde{\boldsymbol{\Sigma}}}\right) \tag{2.16}$$

where $\Phi_{n;\overline{\widetilde{\boldsymbol{\mu}}},\overline{\widetilde{\boldsymbol{\Sigma}}}}(\cdot)$ denotes the multivariate joint CDF of $\mathcal{N}_n(\overline{\widetilde{\boldsymbol{\mu}}}, \overline{\widetilde{\boldsymbol{\Sigma}}})$. Finally, it is noted that simulation from the latter conditional distribution (i.e., generation of conditional stochastic realizations) boils down to simulation from a conditional Gaussian distribution and subsequent mapping to the target domain (i.e., through Eq. (2.8)).

2.4 Admissible marginal distributions

The enhanced flexibility provided by copulas is highlighted by the fact that they can be utilized with any marginal distribution model, such as those listed in **Error! Reference source not found.** (typically used to model positive precipitation amounts), as well as with more complex models such as zero-inflated (ZI) distributions. It is noted that the latter type of models has been proven particularly useful for the probabilistic description of intermittent process, such as, precipitation (e.g., Bell, 1987; Kossieris et al., 2019; Lanza, 2000; Tsoukalas, 2022; Tsoukalas et al., 2020), since it allows the simultaneous description of both states of the process (i.e., no precipitation, and positive precipitation amounts, accounting this way for the extremes of both tails (i.e., minima and maxima). The CDF of a zero-inflated distribution is given by,

Funded by the European Union

$$F_{\underline{x}}(x) = \begin{cases} p_0, & x = 0 \\ p_0 + (1 - p_0)G_{\underline{x}}(x), & x > 0 \end{cases} \quad (2.17)$$

where $p_0 \coloneqq P\{X = 0\}$ denotes the probability of observing a zero value (i.e., no precipitation), and $G_{\underline{x}}(x) \coloneqq F_{\underline{x}|\underline{x}>0}(x) = P(\underline{x} \leq x | x > 0)$ stands for the continuous distribution part, that entails values greater than zero (i.e., positive precipitation amounts).

Table 1. Typical marginal distribution models for precipitation data.

Name	**CDF** (b: scale parameter | a_i: shape parameter)	**Support**
Gamma	$F_{\mathcal{G}}(x\|b,a) = \frac{1}{\Gamma(b)}\gamma\left(b, \frac{x}{a}\right)$	$x > 0$
Weibull	$F_{\mathcal{W}} = 1 - \exp\left(-\left(\frac{x}{b}\right)^a\right)$	$x \geq 0$
Generalized Gamma (Stacy, 1962)	$F_{GG}(x\|b,a_1,a_2) = F_{\mathcal{G}}(\left(\frac{x}{b}\right)^{a_2}, 1, \frac{a_1}{a_2})$	$x > 0$
Exponentiated Weibull (Choudhury, 2005)	$F_{\mathcal{GW}} = \left(1 - \exp\left(-\left(\frac{x}{b}\right)^{a_1}\right)\right)^{a_2}$	$x > 0$
Pareto II (Lomax)	$F_{\mathcal{P}II}(x\|b,a) = 1 - \left(1 + \frac{x}{b}\right)^{-a}$	$x \geq 0$
Burr type XII (Burr, 1942)	$F_{\mathcal{B}r_{\mathrm{XII}}}(x\|b,a_1,a_2) = 1 - \left(1 + \left(\frac{x}{b}\right)^{a_1}\right)^{-a_2}$	$x > 0$

Note: several example applications of these distribution can be found in the domain of statistical/stochastic hydrology, or say, water resources in general (e.g., Ganora and Laio, 2015; Hao and Singh, 2009; Kossieris et al., 2019; Koutsoyiannis, 2020; Shao et al., 2004; Tsoukalas et al., 2020, 2019).

To better illustrate the capabilities of copulas let us consider two hypothetical examples, both involving the Gaussian copula. The first (Figure 1) regards the establishment of the joint CDF (Figure 1c) of two random variables (RVs), both described by Gamma distribution with identical parameters. On the other hand, the second example (Figure 2), while also entailing two RVs, it regards RVs with different marginal distributions (i.e., Gamma and Log-Normal) and concerns the derivation of the conditional CDF using copulas. As shown in Figure 2 (panel A), the copula-based framework suffices to establish the conditional p-quantiles of interest (i.e., 1 and 99%), as well as to derive the conditional PDF of $\underline{x}_1|\underline{x}_2 = x_2$ for various values of conditioning (i.e., $x_2 = 45$ and $x_2 = 65$).

Funded by the European Union

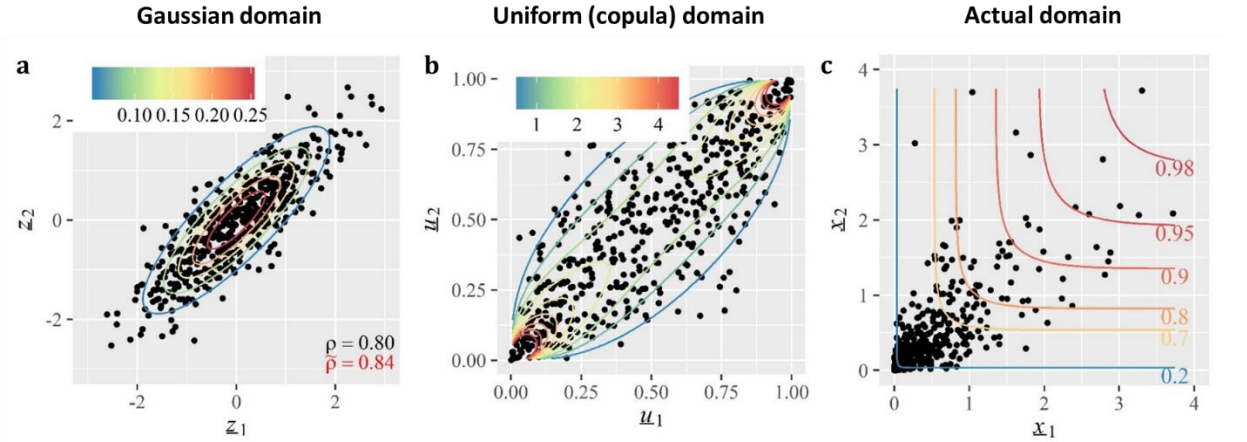

Figure 1. Hypothetical example of two correlated RVs ($\rho = 0.8$), each modeled by a Gamma distribution with parameters $a = 0.5$ and $b = 1$. From left to right, the subplots depict the joint PDF in the a) Gaussian and b) copula (i.e., uniform) domain, as well as c) the joint CDF in the actual domain.

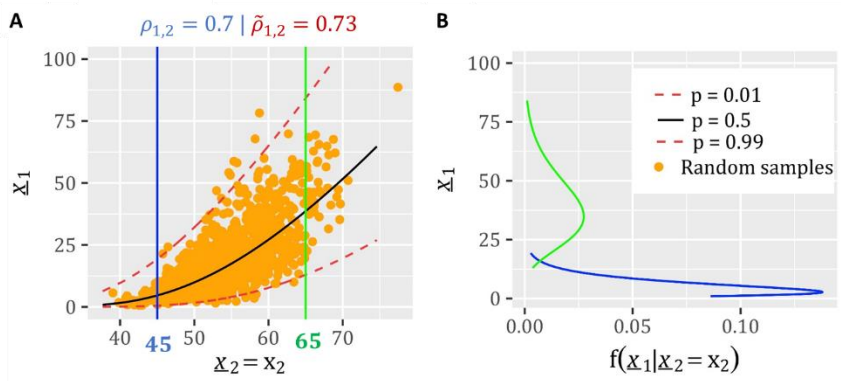

Figure 2. Hypothetical example of two correlated RVs ($\rho = 0.7$), with $\underline{x}_1 \sim Gamma(b = 10, a = 2)$ and $\underline{x}_2 \sim \mathcal{L}og\mathcal{N}ormal(0.10,\ 4)$. The subplots depict, (A) the conditional quantiles in the actual domain and (B) conditional PDF of $\underline{x}_1|\underline{x}_2$ for $\underline{x}_2 := x_2 = 45$ (blue line) and $\underline{x}_2 := x_2 = 65$ (green line).

2.5 Basic properties of stochastic processes at a single and multiple temporal scales

The following paragraphs aim to provide a brief introduction to the (temporal) multi-scale properties of stochastic processes, which then will be used for the temporal augmentation task of lower-scale extrapolation of time series statistics (see section 3.3).

To provide context, let $\{\underline{x}_t\}_{t \in Z^{\geq}}$, where t denotes the time index, be a discrete-time stationary process with arbitrary, continuous or zero-inflated, marginal distribution, $F_{\underline{x}}(x; \boldsymbol{\theta}) := \mathrm{P}\{\underline{x} \leq x\}$, where $\boldsymbol{\theta}$ is a vector that denotes the distribution's parameters. Let also the autocorrelation structure (ACS) of the process be denoted by, $\rho_\tau := \mathrm{Corr}[\underline{x}_t, \underline{x}_{t+\tau}]$, where $\tau \in \{0, \pm 1, \pm 2, \dots\}$

Funded by the European Union

stands for the time lag. Hereafter, and without loss of generality the parameter vector $\boldsymbol{\theta}$ will be omitted, when possible, for the sake of simplicity.

Let us assume that $\underline{x}_t$ refers to a process at a basic time scale $k = 1$. The process can be aggregated to any coarser time scale $k \in \{2,3,\dots\}$ by the following operation:

$$\underline{X}_l^{(k)} := \sum_{t=(l-1)k+1}^{kl} \underline{x}_t \tag{2.18}$$

where l is the new time index. The discrete-time *averaged process* $\underline{x}_l^{(k)}$ can be obtained from the aggregated process by $\underline{x}_l^{(k)} = \underline{X}_l^{(k)}/k$. Also note that $\underline{X}_l^{(1)} \equiv \underline{x}_l^{(1)} \equiv \underline{x}_t$.

As in the basic time scale, at each time scale k, the *averaged* process $\underline{x}_l^{(k)}$ has a marginal distribution $F_{\underline{x}^{(k)}}(x)$ and an autocovariance structure $c_\tau^{(k)} := \mathrm{Cov}\left[\underline{x}_l^{(k)}, \underline{x}_{l+\tau}^{(k)}\right]$, highlighting its scaling behaviour. Interestingly, some (low-order) statistical quantities can be analytically estimated for any level of temporal aggregation (Koutsoyiannis 2010, 2017). In particular, the mean of averaged processes is $\mu^{(k)} = \mu^{(1)} = \mu$, while its variance is given by,

$$\gamma^{(k)} := \left(c_0^{(1)} k + 2\sum_{\tau=1}^{k-1}(k-\tau)\, c_\tau^{(1)}\right)/k^2 \tag{2.19}$$

where $c_\tau^{(k)} := \mathrm{Cov}\left[x_l^{(k)}, x_{l+\tau}^{(k)}\right]$ stands for the auto-covariance function of a discrete-time averaged process at any time scale k. Note also that the variance of the discrete-time aggregated and averaged process, defined as $\Gamma^{(k)} := \mathrm{Var}\left[X_l^{(k)}\right]$ and $\gamma^{(k)} := \mathrm{Var}\left[x_l^{(k)}\right]$ respectively, are linked via $\gamma^{(k)} = \Gamma^{(k)}/k^2$. Also, the inverse operation also holds true,

$$c_\tau^{(k)} := \frac{1}{k^2}\left(\frac{\Gamma^{(|\tau+1|k)} - \Gamma^{(|\tau-1|k)}}{2} - \Gamma^{(|\tau|k)}\right) \tag{2.20}$$

Similarly, it is straightforward to express the autocorrelation function at any scale k by $\rho_\tau^{(k)} = c_\tau^{(k)}/\gamma^{(k)}$.

Beyond these quantities, the majority of high-order statistics cannot be analytically estimated solely from the characteristics of the processes at the basic scale, however it is interesting to note that such scaling laws also hold for intermittent processes such as precipitation (Koutsoyiannis 2006; Tsoukalas 2022). In more detail, intermittent processes are characterized by a scaling behaviour of the probability of zero values, i.e., $p_0^{(k)} = 1 - p_1^{(k)} = \mathrm{P}\{\underline{x}^{(k)} = 0\}$. For simplicity let $p_1 = p_1^{(1)}$ and $p_0 = p_0^{(1)}$. Arguably, the mean and variance at any scale k provide no information about the degree of intermittency at scales of aggregation $k > 1$. This is also apparent by the need of $p_0^{(k)}$ to estimate the mean ($\mu_p^{(k)}$) and the variance ($\gamma_p^{(k)}$) of positive values at scale k. The formulas for the estimation of these quantities at scales of aggregation $k > 1$ are given by (Tsoukalas 2022),

$$\mu_p^{(k)} = \frac{\mu}{p_1^{(k)}} \tag{2.21}$$

Funded by the European Union

$$\gamma_p^{(k)} = \frac{\gamma^{(k)} p_1^{(k)} + \mu^2 p_1^{(k)} - \mu^2}{\left(p_1^{(k)}\right)^2} \tag{2.22}$$

Solving the above equation for $\gamma^{(k)}$ yields,

$$\gamma^{(k)} = \frac{\mu^2 - \mu^2 p_1^{(k)} + \gamma_p^{(k)} \left(p_1^{(k)}\right)^2}{p_1^{(k)}} \tag{2.23}$$

which highlights the link between the variance of the whole process (including zeros) with the mean and the variance of positive values, as well as with probability of zero at scale k. Of course, when $p_1^{(k)} = 1$ (i.e., $p_0^{(k)} = 0$), $\gamma_p^{(k)}$ and $\gamma^{(k)}$ are identical quantities.

3 Recipes for typical temporal augmentation problems

Building upon the theoretical developments described in the previous section, the following three sub-sections provide algorithmic recipes for the three temporal augmentation problems of interest. That is:

P.1: Infilling of time series missing values,
P.2: Generation of statistically consistent stochastic realizations for time series data, and
P.3: Lower-scale extrapolation of time series statistics (e.g., temporal downscaling of key statistical properties).

3.1 Infilling of time series missing values

Basic component of the proposed approach is the use of the Gaussian copula and conditional distributions which in turn allow the conditional stochastic simulation of non-Gaussian random variables and processes. Due to its generality, the approach is considered ideal for both physical and non-physical variables, as well for those exhibiting intermittency, such as rainfall at fine time scales (since zero-inflated distribution can be employed).

In general, the problem of infilling time series missing data can be viewed as a prediction problem of continuous variables using as predictands nearby (in space/time) data (hence in essence constitutes a regression problem). In light of the above, stochastic simulation from conditional distributions appears to be an effective way to cope with this challenge. It is also reminded that key target of every infilling method is to provide an estimate of the missing datum using an appropriate statistical quantity (conditional to other data), Q, which expresses some (conditional) measure of central tendency (typicaly the mean or the median are employed).

The following sub-sections detail two (2) algorithmic recipes for the infilling of time series missing values. The 1st recipe is ideal when data are available/of interest at only one station. While the 2nd one can be used when we have available time series data at multiple locations, and we wish to infill the missing values using information provided by data at nearby locations.

Funded by the European Union

3.1.1 Algorithmic recipe #1

Let us assume that we are given a time series $x(t)$, where $t \in \{1, \dots, t, \dots T\}$, and T denotes the total number of time steps, which has k known and n missing values at time steps t_{**} and t_* respectively. Therefore, let us denote the known values as $y(t_{**})$ and missing ones by $x(t_*)$. Hence our target is to sample from the conditional distribution $\underline{\boldsymbol{x}}_*|\underline{\boldsymbol{x}}_{**} = \boldsymbol{x}_{**}$ (note that we now use vector/bold typeface notation to denote that the infilling will be performed in a vectorized manner).

For the infilling of missing values $x(t_*)$ we perform the following steps (for simplicity the description is focused on the stationary case, yet its application for the cases of cyclical stationarity, cyclostationarity, or even non-stationarity is rather straightforward):

Calibrating/training the model

Step 1. Fit a suitable marginal distribution $F_{\underline{x}}\ \forall\ i \in (1, T)$, using the non-missing data.

Step 2. Using the empirical correlation coefficients, $\hat{\rho}_\tau$, estimated up to maximum lag, $\tau_{\max}$, fit a suitable correlation structure $\rho(\tau, \boldsymbol{\theta})$, where $\boldsymbol{\theta}$ is a vector containing the model's parameters (e.g., for the power exponential model: $\rho(\tau, \boldsymbol{\theta}) = \exp\left(-\left(\frac{\tau}{\lambda}\right)^{\alpha}\right)$, $\boldsymbol{\theta} = [\lambda, \alpha]$)by minimizing the following norm (sum squared difference):

$$\arg\min_{\boldsymbol{\theta}} \left(\sum_{\tau=1}^{\tau_{\max}} (\rho(\tau, \boldsymbol{\theta}) - \hat{\rho}_\tau)^2 \right)$$

Step 3. Using the information provided by the two previous steps estimate the equivalent correlation structure $\tilde{\rho}(\tau, \boldsymbol{\theta}) = \mathcal{F}^{-1}\left(\rho(\tau, \boldsymbol{\theta})|F_{\underline{x}}\right)$ up to lag T, and next transform it to the equivalent correlation matrix $\widetilde{\boldsymbol{R}}(= \widetilde{\boldsymbol{\Sigma}})$ (i.e., the Gaussian copula parameter)**.**

Step 4. Estimate the parameters $\overline{\widetilde{\boldsymbol{\mu}}}$ and $\overline{\widetilde{\boldsymbol{\Sigma}}}$ using Eq. (2.14) and (2.15).

Step 5. Estimate the (lower triangular) matrix $\boldsymbol{B}$ for which it holds that $\boldsymbol{B}\boldsymbol{B}^{\mathrm{T}} = \overline{\widetilde{\boldsymbol{\Sigma}}}$ (typically using Cholesky decomposition).

Step 7. Estimate the vector $\widetilde{\boldsymbol{x}}_{**} = \left[\Phi^{-1}\left(F_{\underline{x}}(x_{**,1})\right), \dots, \Phi^{-1}\left(F_{\underline{x}}(x_{**,k})\right)\right]^{\mathrm{T}}$, on which we wish to condition the generation of RVs realizations.

Simulation/prediction using the model

Step 8. Estimate the vector $\boldsymbol{z} = \overline{\widetilde{\boldsymbol{\mu}}} + \boldsymbol{B}\boldsymbol{\varepsilon}$, where $\boldsymbol{\varepsilon} = [\varepsilon_1, \dots, \varepsilon_n]^{\mathrm{T}}$ and $\underline{\varepsilon_i} \sim \mathcal{N}(0,1)$, and next, map the vector $\boldsymbol{z}$ to the target domain through the ICDF $F_{\underline{x}}^{-1}$. In particular, the vector of conditional realizations $\boldsymbol{x}_* \sim \underline{\boldsymbol{x}}_*|\underline{\boldsymbol{x}}_{**} = \boldsymbol{x}_{**}$ is given by $\boldsymbol{x} = \left[F_{\underline{x}_1}^{-1}(\Phi(z_1)), \dots, F_{\underline{x}_n}^{-1}(\Phi(z_n))\right]^{\mathrm{T}}$

Step 9. Repeat Step 8, N times (typically set between 2000 and 5000), in order to obtain the matrix $\boldsymbol{X}_* = \left[\boldsymbol{x}_{*,1}^{\mathrm{T}}, \dots, \boldsymbol{x}_{*,N}^{\mathrm{T}}\right]^{\mathrm{T}}$, where each column contains N conditional realizations of the random vector $\underline{\boldsymbol{x}}_* = \left[\underline{x}_{*,1}, \dots, \underline{x}_{*,n}\right]^{\mathrm{T}}$.

Step 10. For each one of the random variables estimate the conditional statistic of interest that expresses a measure of central tendency (e.g., average or median operator). In particular, if we opt for the median, which is the suggested one, it is estimated as follows:

Funded by the European Union

$$Q_{\underline{x}_\nu} \coloneqq \mathrm{Med}_{X_\nu} = \begin{cases} X_{\left[\frac{N+1}{2}\right],\nu} & \text{If } N \text{ is odd.} \\ \dfrac{X_{\left[\frac{N}{2}\right],\nu} + X_{\left[\frac{N}{2}+1\right],\nu}}{2} & \text{If } N \text{ is even.} \end{cases}$$

Where $X_{[\cdot],\nu}$ is the order sample of column ν, and N the sample size.

3.1.2 Algorithmic recipe #2

As in the previous paragraph, the description of the recipe is focused on the stationary case, yet again its application for other cases is rather straightforward. For instance, in the case of monthly stationarity, one has to repeat the exact same procedure 12 times (i.e., one for each month).
Let us assume that we have time series data (of course of the same process) at m locations (i.e., gauge stations), and that each location, indexed by $i \in (1, \dots, m)$, corresponds to coordinates $s(i)7 = [s_X(i), s_Y(i)]$, where $s_X(i)$ and $s_Y(i)$ are the pair's values at the cartesian system.
Furthermore, let us assume that at some time step $t_* \in (1, \dots, t, \dots, T)$ (e.g., day t_*, from the total T days/time steps) $n(< m)$ stations have missing values and k have values, which are denoted by the vector $\boldsymbol{y}(t_*) = [y_1(t_*), \dots, y_k(t_*)]^{\mathrm{T}}$. To infill the missing values of the n stations, which are symbolized by the vector $\boldsymbol{x}(t_*) = [x_1(t_*), \dots, x_n(t_*)]^{\mathrm{T}}$, the following steps are employed:

Calibrating/training the model

Step 1. Determine the marginal distributions $F_{\underline{x}_i}\ \forall\ i \in (1, n)$ and $F_{\underline{y}_i} \forall\ i \in (1, k)$ for the RVs of the vectors $\underline{\boldsymbol{x}} = \left[\underline{x}_1, \dots, \nu, \dots, \underline{x}_n\right]^{\mathrm{T}}$ and $\underline{\boldsymbol{y}} = \left[\underline{y}_1, \dots, \kappa, \dots, \underline{y}_k\right]^{\mathrm{T}}$.

Step 2. Estimate the correlation matrix $\boldsymbol{R}$.

Step 3. Estimate the equivalent correlation matrix $\widetilde{\boldsymbol{R}}$ (i.e., the Gaussian copula parameter). To estimate the elements of this matrix (i.e., $\tilde{\rho}_{i,j}$) employ the relationship $\tilde{\rho}_{i,j} = \mathcal{F}^{-1}\left(\rho_{i,j} | F_{\underline{x}_i}, F_{\underline{x}_j}\right)$.

Step 4. Estimate the matrix $\boldsymbol{D}$, with dimensions $m \times m$, which contains the (Euclidean) distances $d(i,j) = \|\boldsymbol{s}(i) - \boldsymbol{s}(j)\| = \sqrt{\left(s_X(i) - s_X(j)\right)^2 + \left(s_Y(i) - s_Y(j)\right)^2}$ of all stations/locations, which correspond to correlations $\tilde{\rho}_{i,j}$ and $\rho_{i,j}$.

Step 5. Using the (empirically-derived) equivalent correlation coefficients, fit a theoretical correlation structure $\rho(d(i,j), \boldsymbol{\theta})$, where $\boldsymbol{\theta}$ is a vector containing the model's parameters (e.g., for the power exponential model: $\rho(d, \boldsymbol{\theta}) = \exp\left(-\left(\frac{d}{\lambda}\right)^{\alpha}\right)$, $\boldsymbol{\theta} = [\lambda, \alpha]$) using an appropriate minimization norm, such as the following one (in this instance, the mean squared difference of the off-diagonal elements of matrix $\widetilde{\boldsymbol{R}}$ and the theoretical model):

$$\arg\min_{\boldsymbol{\theta}} \left(\frac{1}{m} \sum_{i=2}^{m} \sum_{j=1}^{m-1} \left(\rho(d(i,j), \boldsymbol{\theta}) - \tilde{\rho}_{i,j}\right)^2 \right)$$

Simulation/prediction using the model (using the data of time step t_*)

Funded by the European Union

Step 6. Estimate the vector $\widetilde{\boldsymbol{y}} = \left[\Phi^{-1}\left(F_{\underline{y}_1}(y_1)\right), \ldots, \Phi^{-1}\left(F_{\underline{y}_k}(y_k)\right)\right]^{\mathrm{T}}$, on which we wish to condition the generation of RVs realizations.

Step 7. Estimate the parameters $\overline{\widetilde{\boldsymbol{\mu}}}$ and $\overline{\widetilde{\boldsymbol{\Sigma}}}$ using Eq. (2.14) and (2.15).

Step 8. Estimate the (lower triangular) matrix $\boldsymbol{B}$ for which it holds that $\boldsymbol{B}\boldsymbol{B}^{\mathrm{T}} = \overline{\widetilde{\boldsymbol{\Sigma}}}$ (typically using Cholesky decomposition).

Step 9. Estimate the vector $\boldsymbol{z} = \overline{\widetilde{\boldsymbol{\mu}}} + \boldsymbol{B}\boldsymbol{\varepsilon}$, where $\boldsymbol{\varepsilon} = [\varepsilon_1, \ldots, \varepsilon_n]^{\mathrm{T}}$ and $\underline{\varepsilon_i} \sim \mathcal{N}(0,1)$, and next, map the vector $\boldsymbol{z}$ to the target domain through the ICDF $F_{\underline{x}_i}^{-1}$. In particular, the vector of conditional realizations $\boldsymbol{x} \sim \underline{\boldsymbol{x}}|\underline{\boldsymbol{y}} = \boldsymbol{y}$ is given by $\boldsymbol{x} = \left[F_{\underline{x}_1}^{-1}(\Phi(z_1)), \ldots, F_{\underline{x}_n}^{-1}(\Phi(z_n))\right]^{\mathrm{T}}$

Step 10. Repeat Step 9, N times (typically set between 2000 and 5000), in order to obtain the matrix $\boldsymbol{X} = [\boldsymbol{x}_1^{\mathrm{T}}, \ldots, \boldsymbol{x}_N^{\mathrm{T}}]^{\mathrm{T}}$, where each column contains N conditional realizations of the random vector $\underline{\boldsymbol{x}} = \left[\underline{x}_1, \ldots, \nu, \ldots, \underline{x}_n\right]^{\mathrm{T}}$.

Step 8. For each one of the random variables estimate the conditional statistic of interest that expresses a measure of central tendency (e.g., average, or median operator). In particular, if we opt for the median, which is the suggested one, it is estimated as follows:

$$Q_{\underline{x}_\nu} \coloneqq \mathrm{Med}_{X_\nu} = \begin{cases} X_{\left[\frac{N+1}{2}\right],\nu} & \text{If } N \text{ is odd.} \\ \dfrac{X_{\left[\frac{N}{2}\right],\nu} + X_{\left[\frac{N}{2}+1\right],\nu}}{2} & \text{If } N \text{ is even.} \end{cases}$$

Where $X_{[\cdot],\nu}$ is the order sample of column ν, and N the sample size.

3.2 Generation of statistically consistent stochastic realizations for time series data

The theoretical basis provided by the Gaussian copula (see sections 2.1-2.4) allows the straightforward simulation of stochastic processes with any marginal distribution (with finite variance) and (valid) dependence structure, expressed via the well-known Pearson correlation coefficient. Such simulation models can be used for the generation of statistically-consistent synthetic realizations of time series data, under a variety of assumptions for the temporal dynamics, such as stationarity, cyclo-stationarity and cyclical stationarity (Tsoukalas et al. 2018a, b, 2019, 2020; Kossieris et al. 2019). It is also noted that non-stationary models can also be developed (Tsoukalas 2018), yet such an application requires knowledge of the evolution of the temporal dynamics of the process, which is rarely the case, and thus the employment of such models warrants extremes caution.

In more detail, and by using a similar rationale with random variables, it is possible to establish stochastic processes with any target marginal distribution and correlation structure through the mapping (similar to Eq. (*2.8*)) of an appropriately specified auxiliary (stationary or cyclostationary) standard Gaussian process (Gp) with zero mean and unit variance, to which an *equivalent* correlation structure is pre-assigned. As shown in section 2, the mapping operation is typically a non-linear function, often implemented through the inverse cumulative distribution

Funded by the European Union

function (ICDF). These approaches can be viewed as Gaussian copula-based schemes (since they rely on the mapping of a Gaussian process) or non-linear versions of the classic (i.e., Gaussian) linear stochastic schemes (Tsoukalas et al. 2018c). It is remarked that approaches sharing a similar rationale, have been for decades used within the domains of operations research (e.g., Cario and Nelson 1996; Biller and Nelson 2003) and probabilistic engineering mechanics (e.g., Grigoriu 1998; Deodatis and Micaletti 2001). However, their employment within hydrological/earth observation sciences was, until recently, formally unexplored, yet, *post factum* linked with other approaches in hydrological domain (Tsoukalas et al. 2018a).

In the following sub-sections, and following the general guidelines provided by Tsoukalas (2018) and Tsoukalas et al. (2019) we detail two (2) algorithmic recipes for the establishment of such models, and thus the generation of statistically-consistent realizations of time series data. The 1st recipe is ideal for the generation of synthetic realizations when univariate data are available, and of interest. While the 2nd one can be used when we have available time series data at multiple locations, and we wish to generate synthetic data at some other location in the region (as an analogy one may consider that of spatial interpolation using conditional non-Gaussian random fields).

3.2.1 **Algorithmic recipe #1**

The proposed stochastic simulation recipe is synopsized by the following steps (also graphically depicted in Figure 3).

Step 1. Make an assumption about the temporal dynamics of the process (i.e., stationary or cyclostationary, cyclical stationarity), accounting for process properties and the time scale of simulation.

Step 2. Based on the available information (e.g., historical data), as well as expert-knowledge, assign appropriate target marginal distribution(s) and identify the target temporal correlation structure.

Step 3. Select a suitable linear stochastic model to simulate the auxiliary Gaussian process (Gp), for instance an autoregressive (AR) or a moving average (MA) model.

Step 4. Estimate the equivalent correlation coefficients for all pairs of variables that are required by the parameter estimation procedure of the auxiliary model (i.e., the Gp).

Step 5. Estimate the parameters of the Gp model through the equivalent correlation coefficients.

Step 6. Generate a synthetic time series by employing the Gp (say, $\underline{z}_t$) – see 1st row of Figure 3.

Step 7. Map the auxiliary (i.e., Gaussian) time series to the copula domain, say $\underline{u}_t$, to obtain a realization with uniform marginal distribution - see 2nd row of Figure 3.

Step 8. Map, using the inverse of the target distribution (i.e., the ICDF), the uniform time series to the actual domain in order to attain a realization of the target process (say, $\underline{x}_t$) - see 3rd row of Figure 3.

Funded by the European Union

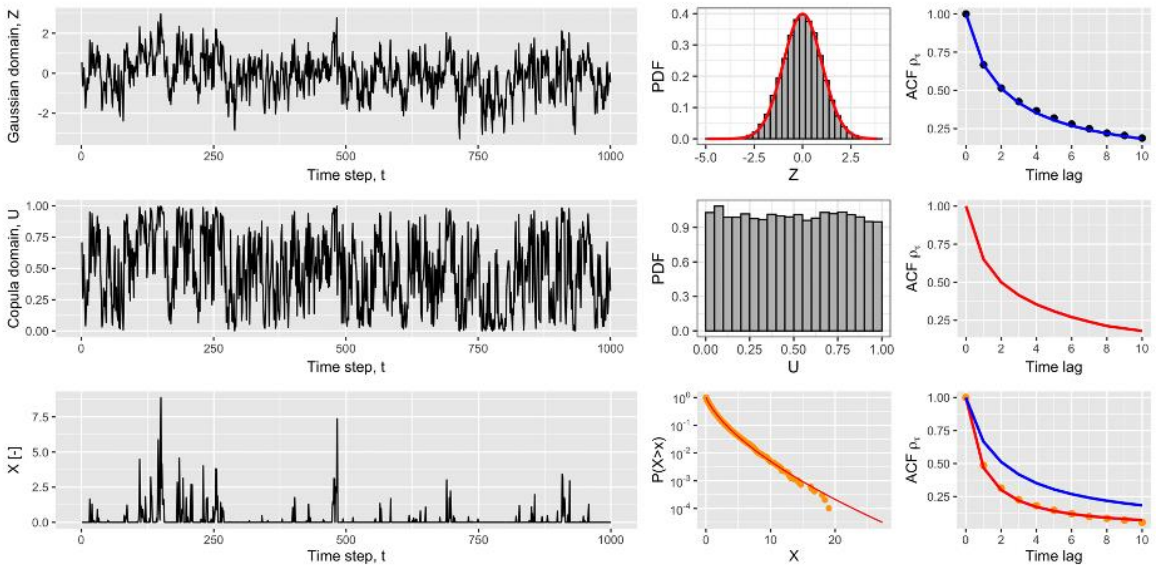

Figure 3. Step-by-step illustration of the stochastic simulation approach used to generate (a stationary) statistically consistent synthetic time series data. The first row illustrates the generation of a Gaussian realization with the equivalent correlation structure, the second row its transformation to the copula domain (i.e., with uniform marginal distribution), and the third row its mapping to the target domain (in this case mapped using a zero-inflated distribution model).

3.2.2 **Algorithmic recipe #2**

This recipe is essentially identical with the recipe provided in section 3.1.2 for the infilling of missing values using information from nearby stations, with the only differences being: a) the assumption of a homogenous random field and b) the removal of the last step (i.e., the operator quantifying central tendency). This recipe can be viewed also as an approach for spatial interpolation and/or a conditional simulation approach for non-Gaussian random fields. For completeness the recipe is provided below:

As in the previous paragraph, the description of the recipe is focused on the stationary case, yet again its application for other cases is rather straightforward. For instance, in the case of monthly stationarity, one has to repeat the exact same procedure 12 times (i.e., one for each month).

Let us assume that we have a grid of points, say a grid with m cells, and that each cell is indexed by $i \in (1, \dots, m)$, that corresponds to coordinates $\mathrm{s}(i) = [s_X(i), s_Y(i)]$, where $s_X(i)$ and $s_Y(i)$ are the pair's values at the cartesian system.

Furthermore, let us assume that at some time step $t_* \in (1, \dots, t, \dots, T)$ (e.g., day t_*, from the total T days/time steps) $n(< m)$ cells have missing values and k have values, which are denoted by the vector $\boldsymbol{y}(t_*) = [y_1(t_*), \dots, y_k(t_*)]^{\mathrm{T}}$. To generate conditional stochastic realization to the n cells, which are symbolized by the vector $\boldsymbol{x}(t_*) = [x_1(t_*), \dots, x_n(t_*)]^{\mathrm{T}}$, the following steps are employed:

Funded by the European Union

Calibrating/training the model

Step 1. Determine the marginal distributions $F_{\underline{x}}\ \forall\ i \in (1,m)$ for the RVs of the vectors $\underline{\boldsymbol{x}} = \left[\underline{x}_1, \ldots, v, \ldots, \underline{x}_n\right]^{\mathrm{T}}$ and $\underline{\boldsymbol{y}} = \left[\underline{y}_1, \ldots, \kappa, \ldots, \underline{y}_k\right]^{\mathrm{T}}$. Due to homogeneity of the field, the distribution is identical for cells (i.e., $F_{\underline{x}} \equiv F_{\underline{y}}$).

Step 2. Estimate the correlation matrix $\boldsymbol{R}$.

Step 3. Estimate the equivalent correlation matrix $\widetilde{\boldsymbol{R}}$ (i.e., the Gaussian copula parameter). To estimate the elements of this matrix (i.e., $\tilde{\rho}_{i,j}$) employ the relationship $\tilde{\rho}_{i,j} = \mathcal{F}^{-1}\left(\rho_{i,j} | F_{\underline{x}}, F_{\underline{x}}\right)$.

Step 4. Estimate the matrix $\boldsymbol{D}$, with dimensions $m \times m$, which contains the (Euclidean) distances $d(i,j) = \|\boldsymbol{s}(i) - \boldsymbol{s}(j)\| = \sqrt{\left(s_X(i) - s_X(j)\right)^2 + \left(s_Y(i) - s_Y(j)\right)^2}$ of all stations/locations, which correspond to correlations $\tilde{\rho}_{i,j}$ and $\rho_{i,j}$.

Step 5. Using the (empirically-derived) equivalent correlation coefficients, fit a theoretical correlation structure $\rho(d(i,j), \boldsymbol{\theta})$, where $\boldsymbol{\theta}$ is a vector containing the model's parameters (e.g., for the power exponential model: $\rho(d, \boldsymbol{\theta}) = \exp\left(-\left(\frac{d}{\lambda}\right)^{\alpha}\right)$, $\boldsymbol{\theta} = [\lambda, \alpha]$) using an appropriate minimization norm, such as the following one (in this instance, the mean squared difference of the off-diagonal elements of matrix $\widetilde{\boldsymbol{R}}$ and the theoretical model):

$$\arg\min_{\boldsymbol{\theta}} \left(\frac{1}{m} \sum_{i=2}^{m} \sum_{j=1}^{m-1} \left(\rho(d(i,j), \boldsymbol{\theta}) - \tilde{\rho}_{i,j}\right)^2 \right)$$

Simulation/prediction using the model (using the data of time step t_*)

Step 6. Estimate the vector $\widetilde{\boldsymbol{y}} = \left[\Phi^{-1}\left(F_{\underline{x}}(y_1)\right), \ldots, \Phi^{-1}\left(F_{\underline{x}}(y_k)\right)\right]^{\mathrm{T}}$, on which we wish to condition the generation of RVs realizations.

Step 7. Estimate the parameters $\overline{\widetilde{\boldsymbol{\mu}}}$ and $\overline{\widetilde{\boldsymbol{\Sigma}}}$ using Eq. (2.14) and (2.15).

Step 8. Estimate the (lower triangular) matrix $\boldsymbol{B}$ for which it holds that $\boldsymbol{B}\boldsymbol{B}^{\mathrm{T}} = \overline{\widetilde{\boldsymbol{\Sigma}}}$ (typically using Cholesky decomposition).

Step 9. Estimate the vector $\boldsymbol{z} = \overline{\widetilde{\boldsymbol{\mu}}} + \boldsymbol{B}\boldsymbol{\varepsilon}$, where $\boldsymbol{\varepsilon} = [\varepsilon_1, \ldots, \varepsilon_n]^{\mathrm{T}}$ and $\underline{\varepsilon_i} \sim \mathcal{N}(0,1)$, and next, map the vector $\boldsymbol{z}$ to the target domain through the ICDF $F_{\underline{x}}^{-1}$. In particular, the vector of conditional realizations $\boldsymbol{x} \sim \underline{\boldsymbol{x}} | \underline{\boldsymbol{y}} = \boldsymbol{y}$ is given by $\boldsymbol{x} = \left[F_{\underline{x}}^{-1}(\Phi(z_1)), \ldots, F_{\underline{x}}^{-1}(\Phi(z_n))\right]^{\mathrm{T}}$

Step 10. Repeat Step 9, N times, that is equal to the number of conditional realizations we wish to generate.

3.3 Lower-scale extrapolation of time series statistics

This section regards the so-called *statistics' downscaling* problem, which more intuitively can be viewed as a lower-scale extrapolation problem of time series statistics, using information that is only available at coarser temporal scales.

Following the notation introduced earlier, let $\underline{x}_t$ denote a stationary, and (possibly) non-Gaussian, stochastic process, and also let us denote any statistical quantity $m(k)$ of the process

Funded by the European Union

as a function of scale k, which is given by $m(k) \coloneqq S\left(\underline{x}_l^{(k)}\right)$, where $S(\cdot)$ is a function (e.g., a statistical estimator) applied to the statistical quantity of interest. For instance, $S[\cdot]$ may represent a moment (e.g., variance) of the process, or a quantile (e.g., median or 95%).
To better showcase the problem posed, let us rephrase it via the following question:

> *Given that a set of statistical quantities, $\{m(i), m(i \times 2), \dots, m(i \times n)\}$, is known at time scales i up to $i \times n$, where $n = 1, 2, \dots$ is an integer index, downscale (reconstruct) the statistical quantity $\hat{m}(j)$ at a finer time scale j, where $j \in [1, i)$ and is an integer.*

A graphical representation of the above question accompanied with the proposed methodological framework is given in Figure 4. It is noted that the proposed method is demonstrated in Section 4.3using daily precipitation data.

The proposed approach, which has been recently applied in water demand data (Kossieris et al. 2021), builds upon the temporal scaling behaviour exhibited by stochastic processes (see section 2.5) and provides a simple and parsimonious approach summarised by the following steps:

Step 0. Considering the resolution of the available time-series data, define the lowest resolution of interest j, and without the loss of generality, treat it as the basic scale, and hence $k = j = 1$ (blue dashed vertical line in Figure 4). For instance, when a time series of daily resolution is available, and the lowest resolution of interest is assumed to be 1 hour, i.e., $k = j = 1$, then all coarser-scale processes are constructed based on the latter resolution. For example, under this assumption, the time scales of 2, 3, and 10 days derive according to Eq. (2.18) for $k = 48$, $k = 72$ and $k = 240$, respectively.

Step 1. Given the observed time series, which now corresponds to time scale $k = i$ (black dashed horizontal line in XX), estimate the statistical quantity of interest at coarser time scales $i \times n$, where $n = 1, 2, \dots n_{max}$, i.e., estimate $\{m(i), m(i \times 2), \dots, m(i \times n_{max})\}$.
Continuing the example from *Step 0*, and hence $i = 10$, estimate the set of quantities $\{m(10), m(10 \times 2), \dots, m(10 \times n_{max})\}$ (red dots in **Error! Reference source not found.**).

Step 2a. Select a parametric function $H(k; \boldsymbol{\theta})$, where $\boldsymbol{\theta}$ is a vector of parameters, suitable to model the statistical quantity of interest, and hence provide estimates $\hat{m}(k) = H(k; \boldsymbol{\theta})$.

Step 2b. Fit the selected function $\hat{m}(k) = H(k; \boldsymbol{\theta})$ on the set of known statistics $\{m(i), m(i \times 2), \dots, m(i \times n_{max})\}$. The fitted function is displayed via a black solid line in Figure 4.
This implies solving an optimisation problem with the following formulation (of course, other alternative error metrics could be employed):

$$\arg\min_{\boldsymbol{\theta}} \sum_{k=i}^{i\, n_{\max}} \left(1 - \frac{\hat{m}(k; \boldsymbol{\theta})}{m(k)}\right)^2 \tag{3.1}$$

It is noted that the latter objective function is known as the sum of the squared relative difference between observed and modelled quantities.

Step 3. Using the fitted model, estimate the statistical quantity of interest $\hat{m}(j)$ at the finer time scale $j \in [1, i)$. Estimations at finer scales are represented by the blue dashed line in Figure 4.

Funded by the European Union

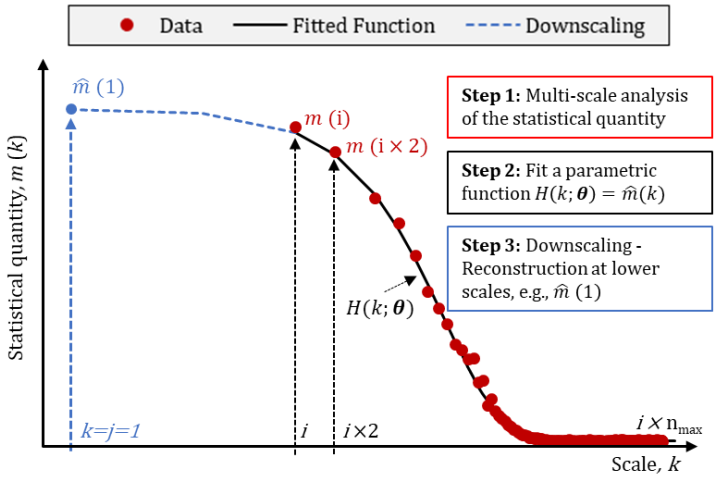

Figure 4. Graphical explanation of the methodological framework for the downscaling of statistical quantities at fine temporal scales. Source: Kossieris et al. (2021).

Evidently, the use of a parametric function to model the scaling behaviour of a statistical quantity lies at the core of the proposed methodology. As noted by as noted by Kossieris et al. (2021), key advantages of this approach are its parsimony (in terms of parameters) as well as its ability to perform extrapolations to lower and coarser scales.

Such an approach essentially consists an unsupervised approach for the statistics downscaling problem since it solely relies on coarser resolution estimations of the statistic of interest – hence not involving any prior training/fitting/learning task using the *labelled* (i.e., target/lower resolution) data. The form of the parametric model is related with the characteristics of the scaling behaviour exhibited by the statistical quantity of interest (e.g., variance or skewness), and thus herein we employ the following function:

$$m^{(k)} = m^{[1+(\xi^{-1/\eta}-1)(k-1)]^{\eta}} \tag{3.2}$$

where $m \coloneqq m^{(1)}$ is the statistical quantity at a basic scale (i.e., $k = 1$), whereas ξ and η are the model parameters varying in the interval $[0, 1]$. This model has been initially proposed by Koutsoyiannis (2006), for the multi-scale description of rainfall occurrence process, while recently it was implemented to model the statistics of water demand processes at multiple temporal scales (Kossieris et al. 2021), proving itself as a particularly flexible model. It is noted, that alternative models could be employed, requiring although to exhibit a particular monotonically decreasing *form*, such as that of survival functions of distribution models and functions of theoretical autocorrelation structures. See also the so-called *climacogram* (Koutsoyiannis 2010, 2017) which is a model particularly tailored for the multi-scale modelling of variance.

Beyond these, it is remarked that the proposed approach is focused on the multiscale modelling of the following statistics: (a) probability of zero value, (b) variance, (c) L-variation and (d) L-skewness. The rationale behind this selection lies on the fact that these statistics are particularly

Funded by the European Union

useful for the parameterization of typical stochastic models (e.g., those described in section 3.2). In detail, the first statistic quantifies the intermittent behaviour of a process, typically exhibited at fine time scales for the physical (e.g., rainfall), while the other three allow the fitting of a probability distribution model. Our selection, is opted towards, *L-moment statistics* (Greenwood et al. 1979; Hosking 1990) since they are proven to provide more reliable (high-order moment) estimations, compared to classical product-type moments. It is reminded that: L-variation, $\tau_2 = \lambda_2/\lambda_1$, and L-skewness, $\tau_3 = \lambda_3/\lambda_2$, defined as ratios of L-moments λ_i. L-variation takes values in the range [0, 1], while L-skewness is a dimensionless measure of asymmetry, analogous to that of skewness coefficient, and can take values in the range [−1, 1] – although it is typically limited in $[0, 1]$ for both physical (e.g., rainfall) and non-physical (e.g., water demand) processes since they rarely exhibit left-skewed behavior.

Funded by the European Union

4 Demonstration of the developed methods

For the demonstration of the capabilities of T4.1/D4.1 methods we employed a variety of variables/processes, which are of course selected in order to be of relevance to the EIFFEL pilots. In particular, this report extends the demonstration exercises showcased in the accompanying document of MS9 (see also Table 3) by putting special focus to the modelling of precipitation processes, since a) it is of interest for all EIFFEL pilots, and b) it is the most "modelling-demanding" variable since beyond the typical peculiarities of physical processes (e.g., non-Gaussianity, spatio-temporal dependencies, periodicities), it also exhibits intermittency, an aspect that makes such processes a particular modelling challenge. The nine (9) demonstrations carried out within this report are summarized below in Table 2.

Table 2. Summary of demonstration exercises showcased in section 4.

#	Demonstration	Variable involved	Dataset employed	Variable relevant to pilot No. #
D1	Infilling of time series missing values (recipe #1)	Streamflow	Monthly streamflow time series from Nile river (used as an example due to its iconic status)	All
D2	Infilling of time series missing values (recipe #1)	Temperature	Daily temperature time series at Paris station (obtained from KNMI)	All
D3	Infilling of time series missing values (recipe #1)	O_3	Hourly O_3 time series of Athens	Pilot #4
D4	Infilling of time series missing values (recipe #2)	Precipitation	Daily precipitation data from 102 stations in Netherlands (obtained from KNMI)	All and Pilot 1 in particular
D5	Generation of synthetic time series (recipe #1)	Streamflow	Monthly streamflow timeseries from Nile river (used as an example due to its iconic status)	All
D6	Generation of synthetic time series (recipe #1)	Precipitation	10-minute precipitation of Soltau, Germany (data obtained from IDW	All
D7	Generation of synthetic time series (recipe #2)	Precipitation	Daily precipitation data from 102 stations in Netherlands (obtained from KNMI)	All and Pilot 1 in particular
D8	Lower-scale extrapolation of time series statistics	Precipitation	Hourly precipitation data from 33 stations in Netherlands (obtained from KNMI)	All and Pilot 1 in particular

Funded by the European Union

D9	Lower-scale extrapolation of time series statistics	Precipitation	Daily gridded (with 0.25 resolution) precipitation data at about 2200 locations (E-OBS dataset of ECA&D) masking all 5 EIFFEL pilots	All

Furthermore, as mentioned earlier, beyond the demonstrations detailed in the above table and presented herein, it is reminded that the functionality of the proposed methods has been showcased at MS9 (i.e., the *Alpha* version of T4.1), and provided three (3) R scripts, as well as the complete code of T4.1/D4.1 methods, that demonstrate, in an interactive manner, using real-world time series data of various temporal resolutions (spanning from hourly to annual), the functionalities of the three computational/analysis engines. Table 3 provides a quick summary of the relevant scripts and the type variables involved.

Table 3. Synopsis of demonstration scripts for T4.1 toolkit Alpha version.

R script name	Description/demo	Time series data employed
`1_Demo_Infill.R`	Infilling of time series missing values	Streamflow (monthly), Temperature (monthly and daily), NO_2 and O_3 (hourly)
`2_Demo_synthetic_data.R`	Generation of synthetic time series data	Precipitation (annual, monthly, daily), Streamflow (annual and monthly), Temperature (annual, monthly and daily)
`3_Demo_Stats_downscaling.R`	Lower-scale extrapolation of time series statistics	Precipitation (downscaling of daily statistics to hourly statistics) – incl. verification.

4.1 Infilling of time series missing values

The methodology developed for the infilling of univariate time series missing values (i.e., algorithmic recipe #1) is demonstrated throughout demonstrations D1-D3 via three univariate time series of different type (i.e., streamflow, temperature and O_3) and of different temporal resolution (monthly, daily and hourly). In all cases, and to assess the method, we employed a complete time series dataset which has been artificially inflated by missing values (NAs) at random time steps. The degree of NAs inflation has been set in all cases equal to 20% of the total time series length.

D1 regards the iconic monthly streamflow time series dataset of Nile River at Aswan dam (1870-1945). As shown in Figure 5 the method is capable of infilling the time series missing values with remarkable success ($R^2 \sim 0.95$), relying only on information provided by the time series per se.

Funded by the European Union

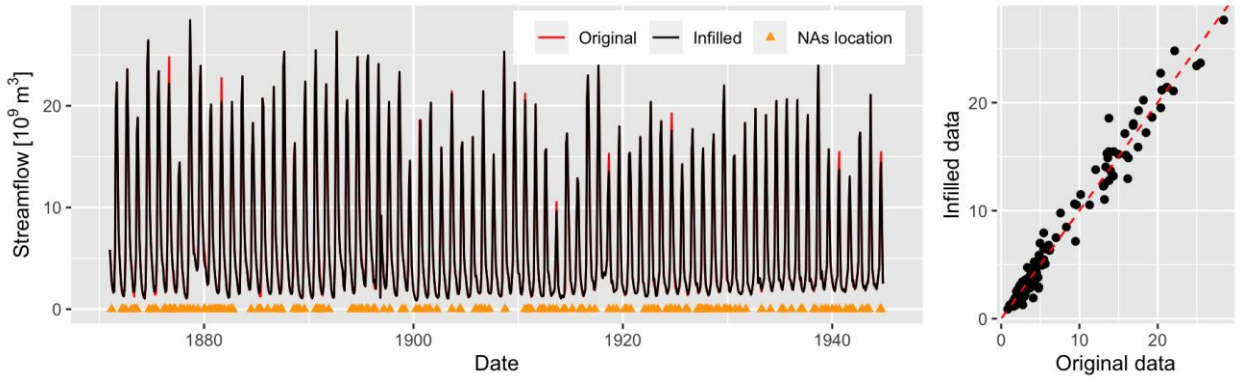

Figure 5. Demonstration of missing values imputation method using monthly streamflow data from the Nile station (1870 - 1945). (Left) Comparison between infilled and observed values. The yellow dots at the bottom depict the randomly selected time steps which assumed missing. (Right) Scatter plot depicting the observed/original and infilled values.

The performance of the method is similar for the cases of D2 (daily temperature time series at Paris station, GHCN-D station code: FR000007150 PARIS/LE_BOURGET, period: 1900-2000) and D3 (hourly O_3 data from the Athens pilot, provided by the project partner, period 2016-2018), highlighting that the method can perform equally well regardless of the temporal resolution of the time series. The results of D2 and D3 are synopsized in Figure 6 and Figure 7 respectively.

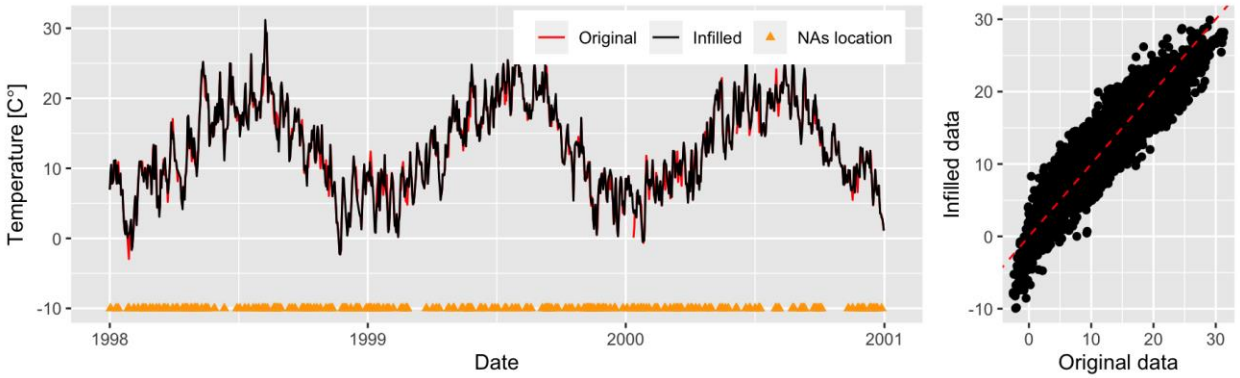

Figure 6. Demonstration of missing values imputation method using daily average temperature data from Paris station (GHCN-D station code: FR000007150 PARIS/LE_BOURGET, 1900-2000). (Left) Comparison between infilled and observed values for the period 1998-2000. The yellow dots at the bottom depict the randomly selected time steps which assumed missing. (Right) Scatter plot depicting the observed/original and infilled values.

Funded by the European Union

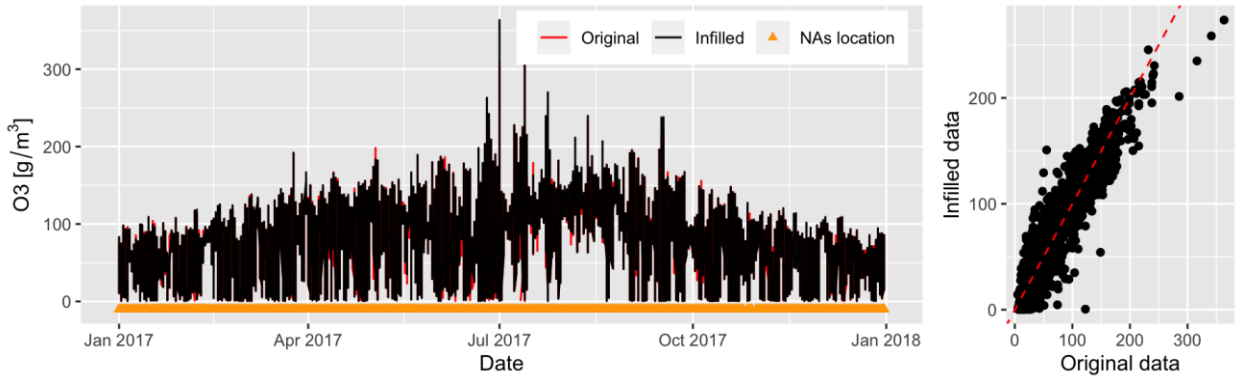

Figure 7. Demonstration of missing values imputation method using hourly O_3 data from the Athens pilot (provided by the project partner). (Left) Comparison between infilled and observed values for the period 2017. The yellow dots at the bottom depict the randomly selected time steps which assumed missing. (Right) Scatter plot depicting the observed and infilled values.

Moving to D4, this demonstration showcases the algorithmic recipe #2 for the infilling of missing values. It is reminded that this algorithmic recipe can be used when we have available time series data at multiple locations, and we wish to infill the missing values using information provided by data at nearby locations. Therefore, we employed 102 historical precipitation time series of daily resolution spanning across Netherlands (obtained from KNMI), and for the period 2000-2010. For their exact locations see Figure 8. Again, and prior employing the infilling method we invoked NAs at random time steps and locations. The number of NAs has been set equal to the 20% of the total length of the time series, and the employed distribution is the zero-inflated Generalized Gamma distribution. To obtain a better picture of the distribution of the missing values across the 102 locations, Figure 9 provides a time series of the NAs count per time step. It is observed that at each time step the average number of missing values is 20, while in some instances they exceed 30 (out of 102). Figure 10, via scatter plot among the true (i.e., the original) and infilled values, provides a quick summary of the method's application, highlighting the remarkable ability of the method to infill successfully the missing values of intermittent processes ($R^2 = 0.827$ and $\mathrm{MSE} = 3.89$), such as precipitation, without having the problem of predicting negative values (a property attributed to the use of conditional simulation in combination with zero-inflated distributions). It is noted that as for comparison purposes we employed the particularly popular (with about 3000 citations since 2018), `missForest` R package (Stekhoven and Buhlmann 2012), which is considered state-of-the-art for missing values imputation. The use of the latter package - using the exact same dataset - resulted in $R^2 = 0.808$ and $\mathrm{MSE} = 4.33$. It could be argued that this kind of performance reserves a place for the proposed method in the state-of-the-art of infilling methods.

Funded by the European Union

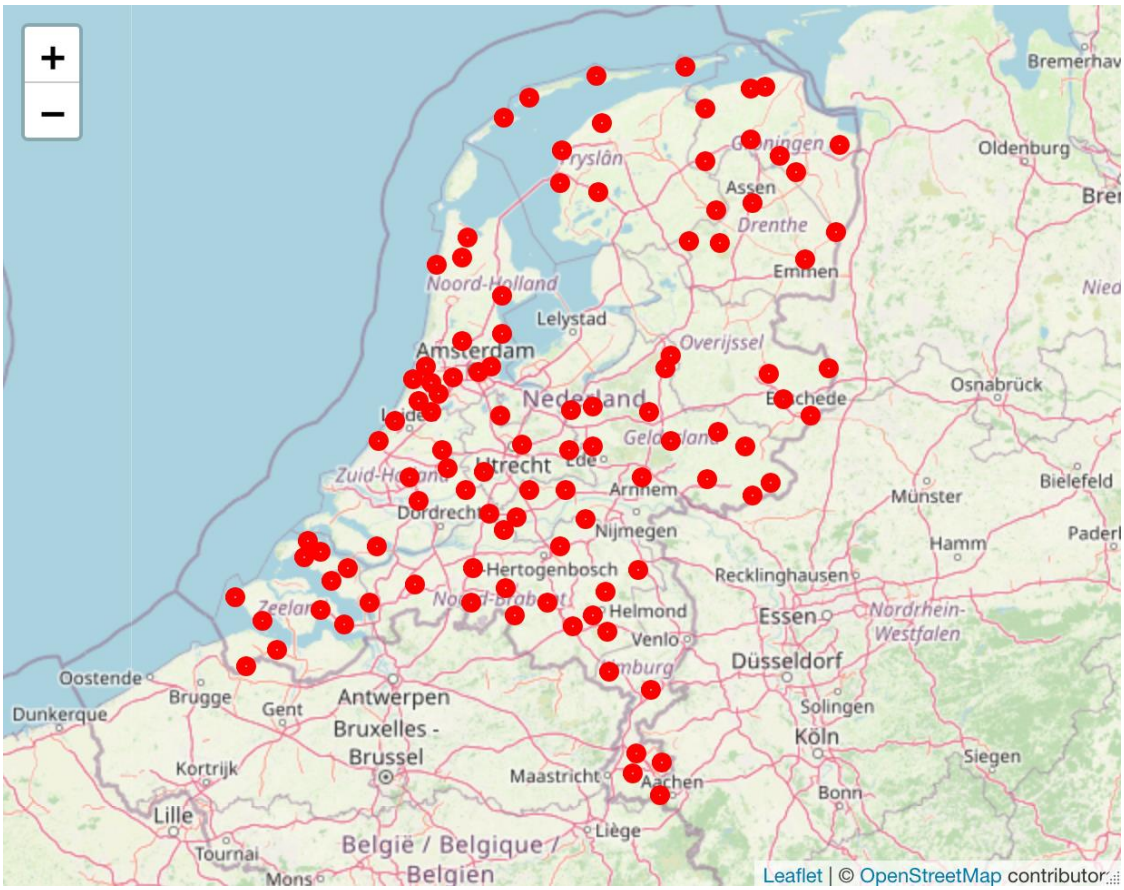

Figure 8. Map depicting the location of the 102 daily precipitation gauge stations employed in D4.

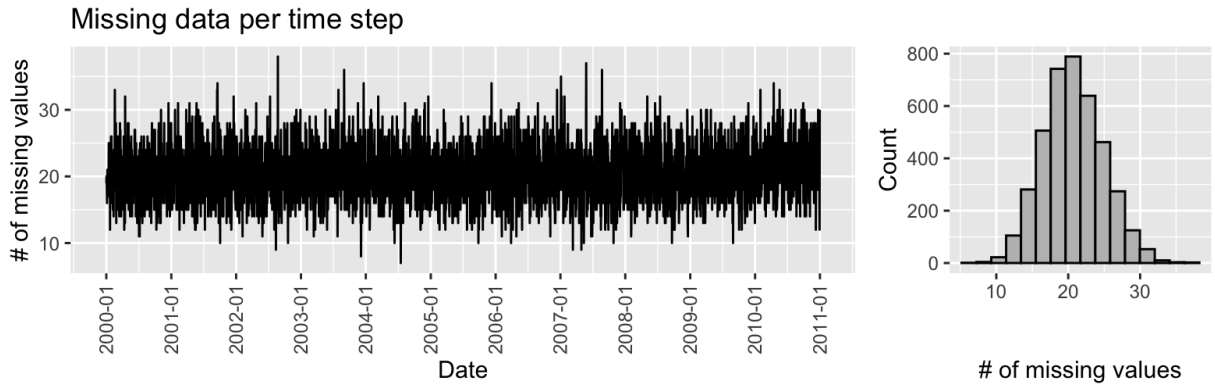

Figure 9. (Left)Total number of NAs invoked at each time step. (Right) Histogram of NAs count.

Funded by the European Union

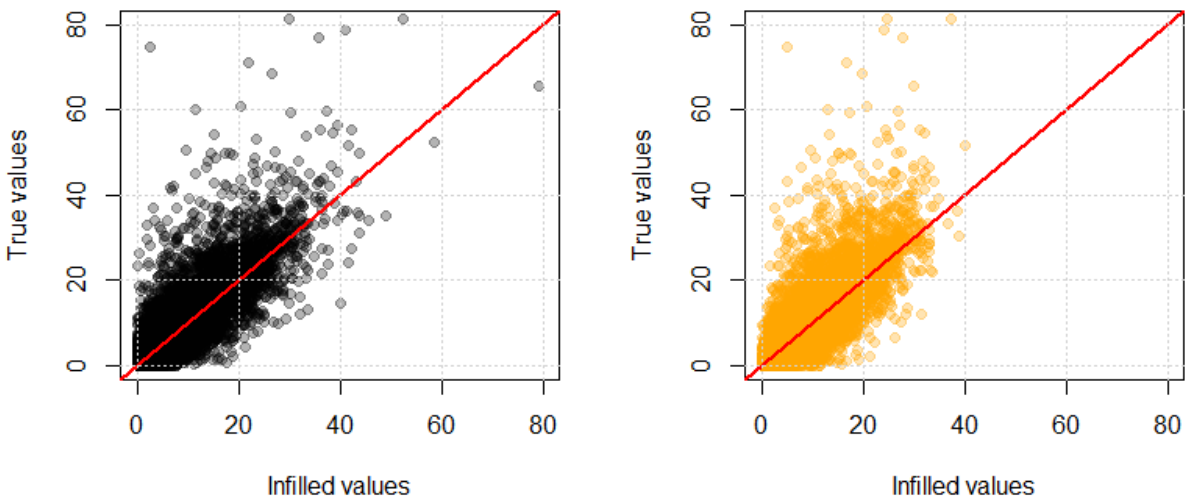

Figure 10. Comparison between the true and infilled values, obtained using the (left) proposed method and (right) missForest R package.

4.2 Generation of statistically consistent stochastic realizations for time series data

To demonstrate the synthetic data generation approach (algorithmic recipe #1) we started by D5 and by employing the iconic monthly streamflow time series dataset of Nile River at Aswan dam (1870-1945). Figure 11 provides a quick summary of this demonstration (which relies on a cyclostationary non-Gaussian stochastic model with either a Burr type XII or a Generalized Gamma distribution – see section 3.2 and 3.2.1). In particular, panel (a) depicts the historical Nile monthly streamflow series (March 1870 to December 1945), panel (b) the synthetically generated time series for a randomly selected window of 80 years. While, panel (c) provides a monthly-wise comparison of historical and simulated L-moments, as well as lag-1 month-to-month correlations coefficients, highlighting the ability of the proposed approach to generate synthetic time series with the desired (target) statistical/probabilistic characteristics.

In the same spirit, D6 demonstrates the same algorithmic recipe, yet in this case by employing a much more challenging (due to intermittency) historical dataset. In particular, as reference data we employ the historical data of 10-min precipitation from Soltau, Germany (data obtained from IDW, Station ID 4745), extending from 1999 to 2009. The simulation results of D6 (employing this time a zero-inflated Burr type XII distribution) are depicted in Figure 12, where panel (b) provides a sample of the generated synthetic time series (for a randomly selected window), which panels (c) and (d) provide a comparison among the historical and simulated distribution function and autocorrelation structure respectively. The latter are in perfect agreement, highlighting the statistical consistency of the synthetically-generated time series.

Funded by the European Union

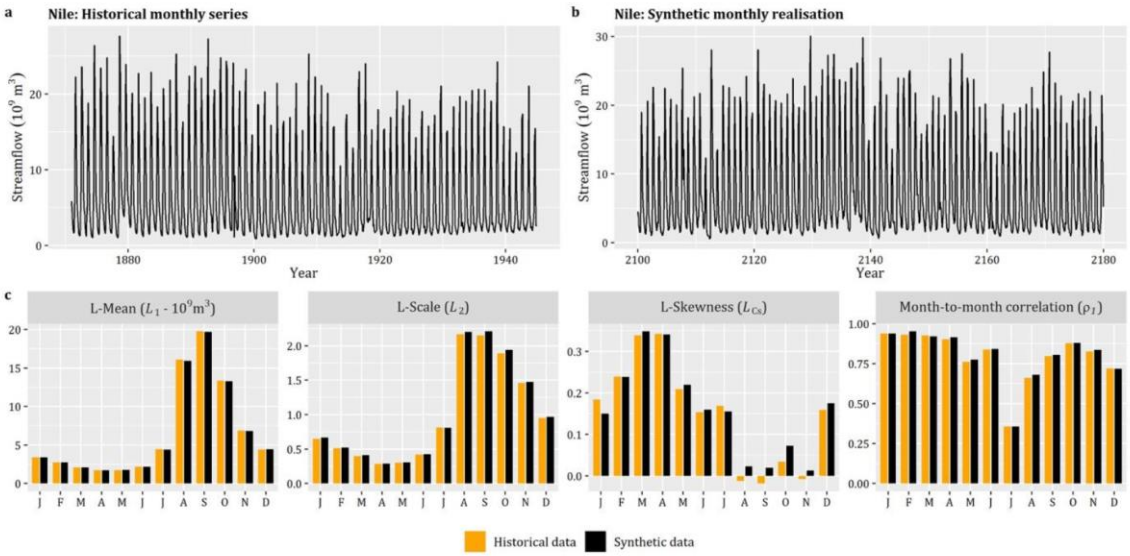

Figure 11. Demonstration of the synthetic data generation method. (a) Historical Nile monthly streamflow series (March 1870 to December 1945). (b) Synthetic time series (randomly selected window of 80 years). (c). Monthly-based comparison of historical and simulated L-moments, as well as lag-1 month-to-month correlations coefficients. Note: in this case a cyclostationary non-Gaussian stochastic model was employed.

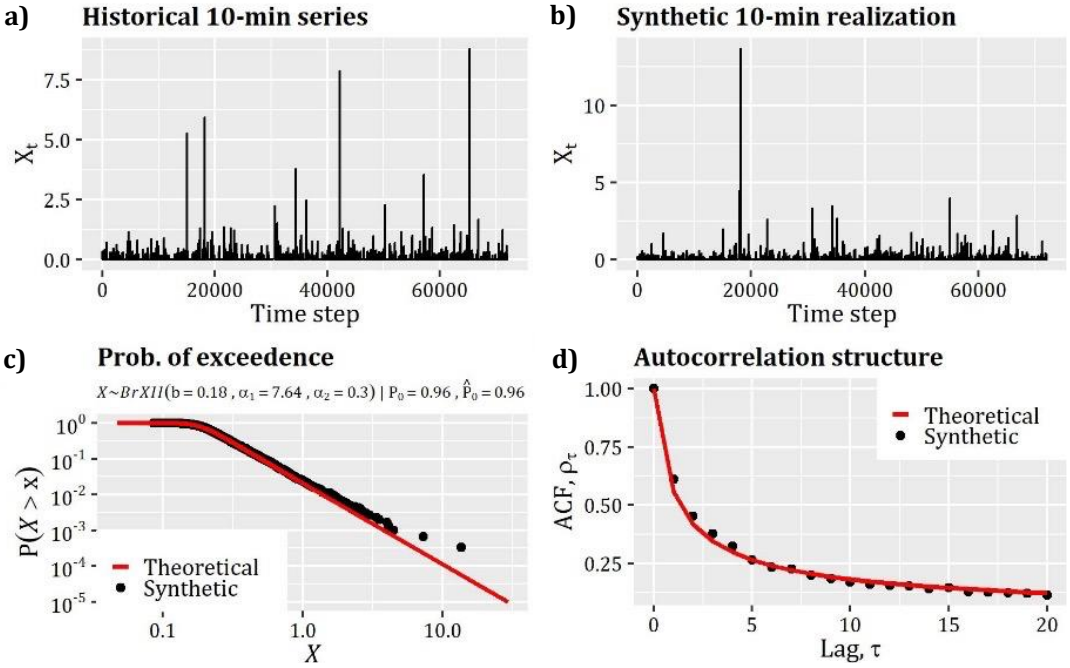

Figure 12. a) Historical 10-min rainfall from Soltau, Germany (data obtained from IDW, Station ID 4745), extending from 1999 to 2009. b) Sample of the generated synthetic time series (randomly selected window). Comparison of historical and simulated c) distribution function and d) autocorrelation structure.

Funded by the European Union

The D7 demonstration utilizes the algorithmic recipe #2 of section 3.2 and can be used when we have available time series data at multiple locations, and we wish to generate synthetic data at some other location in the region (as an analogy one may consider that of spatial interpolation using conditional non-Gaussian random fields). For this particular demo we employ the same data used in D4, that is 102 historical precipitation time series of daily resolution spanning across Netherlands (obtained from KNMI), and for the period 2000-2010. For their exact locations of the stations see Figure 8.

Going through the steps described in 3.2.2 we generate multiple conditional (on the observed values of the 102 stations) random fields (RFs) that span across the entire Netherlands (we assumed a discretized field with spatial resolution of about 5 km). To provide some visual results, Figure 14 depicts 30 consecutive snapshots of the RF that cover Netherlands, where we may observe that the simulated conditional RFs preserve the space-time dynamics of precipitation, thus being able to generate synthetic precipitation values at any ungauged location (an example of such time series is given in Figure 13, which corresponds to synthetically generated time series at coordinates: [5.685, 52.223]).

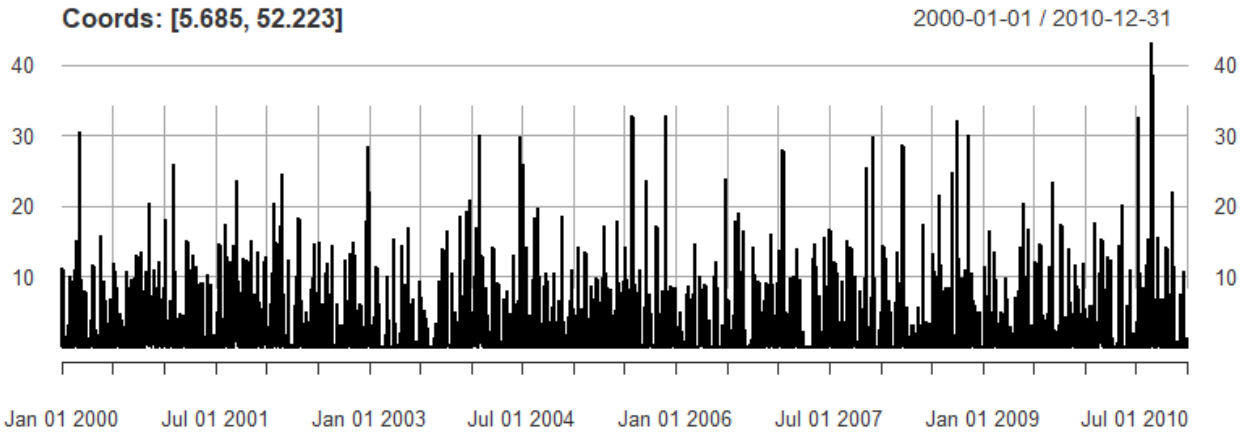

Figure 13. Synthetically generated time series (using a non-Gaussian conditional RF) at a randomly selected, ungagged location (coordinates: [5.685, 52.223]), preserving the temporal dynamics and intermittency dictated by the historical data.

Funded by the European Union

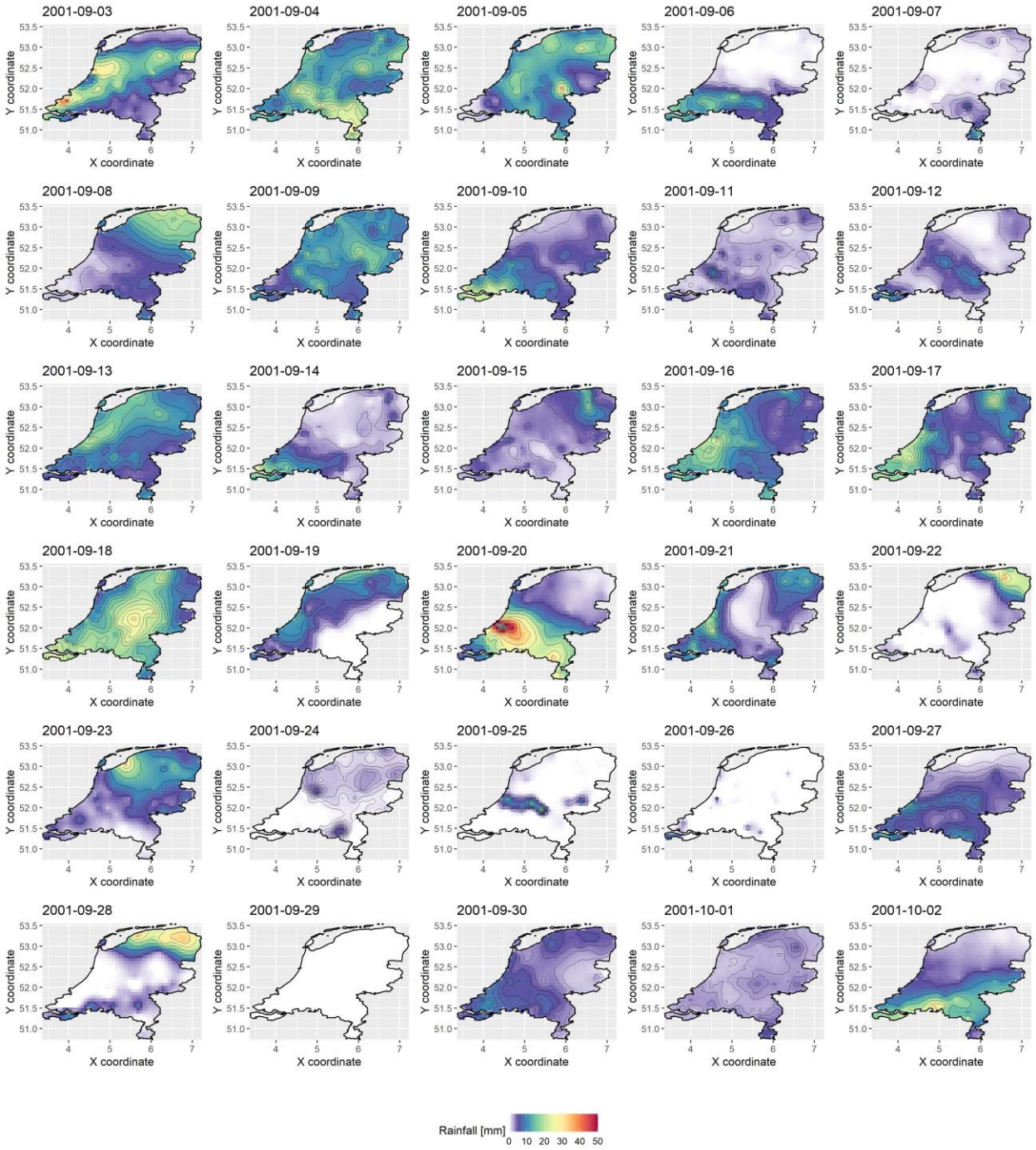

Figure 14. Snapshots of the simulated non-Gaussian random field, spanning across 30 (randomly selected) time steps. White cells represent cells with zero values (i.e., no precipitation), while blue colour palette is used to depict the non-zero values (light precipitation is depicted with light blue, while heavy precipitation with dark blue).

Funded by the European Union

4.3 **Lower-scale extrapolation of time series statistics**

To demonstrate the methodology of section 3.3 which allows the downscaling (i.e., extrapolation at lower time scales) of time series statistics we employ data[5] from 33 hourly time series of precipitation in Netherlands (see Figure 15 for their exact location). First, the data are aggregated to daily scale, and these time series are used to demonstrate the approach. The finer-resolution time series (i.e., hourly to 24 hours) are used to validate the lower-scale/downscaling method of section 3.3. Figure 16 provides an application example of the method for the downscaling of: a) probability of zero values (i.e., probability dry), b) variance, c) L-variation and d) L-skewness, using data from De Kooy gauging station.

The results from all 33 stations are summarized in Figure 17, where it is observed that the downscaling approach is capable of downscaling the statistics of daily precipitation down to hourly scale with acceptable accuracy (remember we only had, and used, a daily time series).

The final demonstration, D9 regards the downscaling of statistics over country of all the project's pilots (i.e., Greece, Finland, Lithuania, Netherlands, and Spain). In particular, we implemented the methodology of section 3.3 for the downscaling of daily precipitation time series statistics using data provided by the European Climate Assessment & Dataset project, in particular using the latest version available, that is, E-OBSv26.0e[6]. This precipitation dataset is of daily resolution, hence downscaling its statistics to finer temporal scales, such as hourly, is significant value and use. To provide some examples, Figure 18 illustrates the probability of zero values (i.e., probability dry) of daily precipitation over the countries of the five EIFFEL pilots, while Figure 19 depicts the downscaled (using the method of section 3.3) probability of zero values (i.e., probability dry) of hourly precipitation over the same regions. Further to these, to provide a more complete overview we created online, interactive plots that depict our analysis.

The following table provides links and description to the analysis.

Table 4. Description and links to the analysis performed for D9.

Daily resolution	
Prob. of zero values	https://rpubs.com/johntt7/973834
L-variation	https://rpubs.com/johntt7/973833
L-skewness	https://rpubs.com/johntt7/973832
Hourly resolution (downscaled using the method of section 3.3)	
Prob. of zero values	https://rpubs.com/johntt7/973831
L-variation	https://rpubs.com/johntt7/973829
L-skewness	https://rpubs.com/johntt7/973823

[5] Obtained from: https://www.daggegevens.knmi.nl/klimatologie/uurgegevens.
[6] Obtained from: https://www.ecad.eu/download/ensembles/download.php.

Funded by the European Union

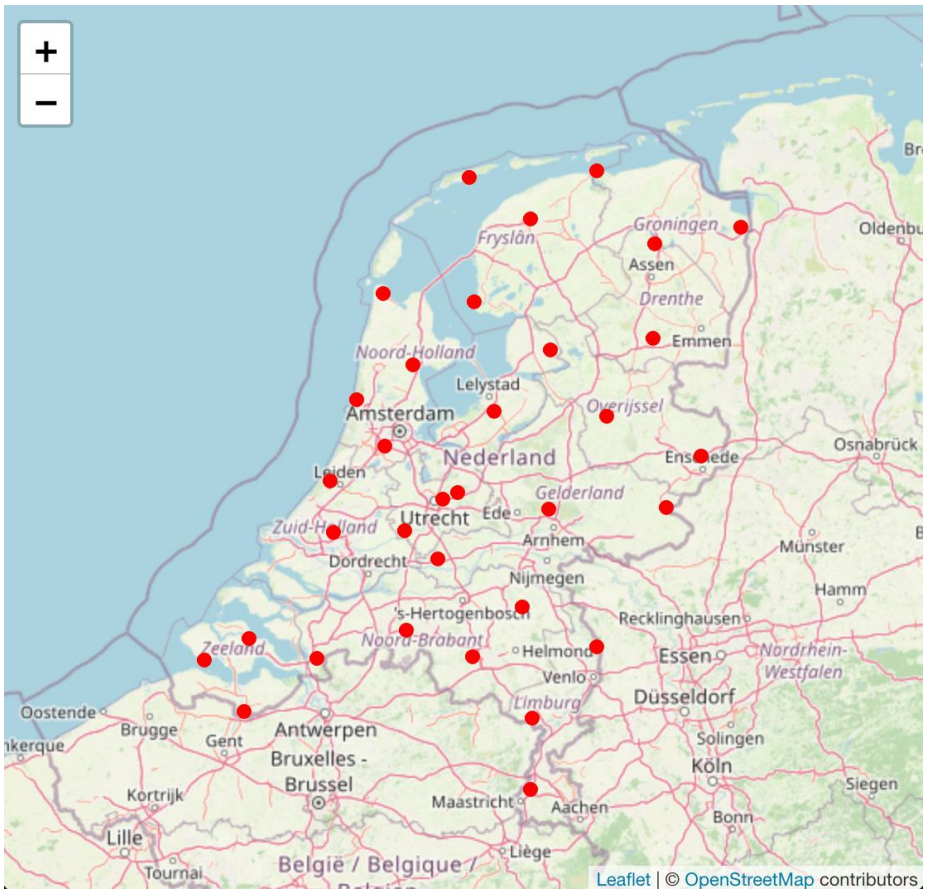

Figure 15. Map depicting the 33 hourly precipitation gauge stations employed in D8.

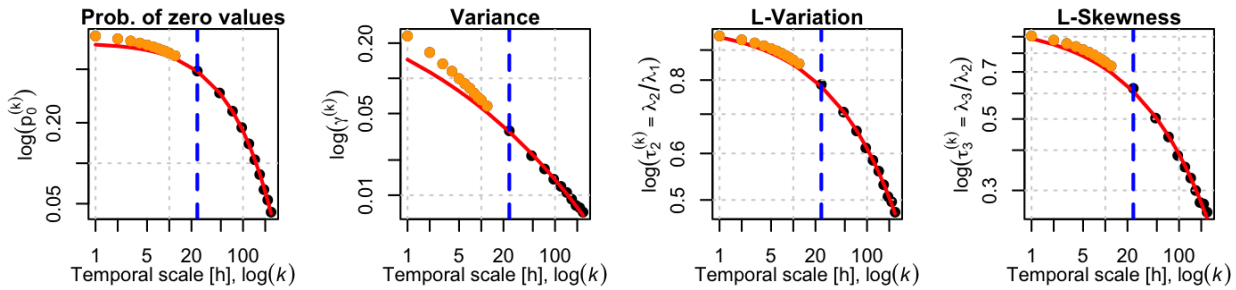

Figure 16. Demonstration of lower-scale extrapolation method for the downscaling of time series statistics (in this case, probability dry, variance, L-variation and L-skewness). The employed dataset regards daily precipitation at De Kooy, NL gauging station (obtained from KNMI climate explorer), whose statistics have been downscaled down to the temporal scale of 1 hour ($k = 1$).

Funded by the European Union

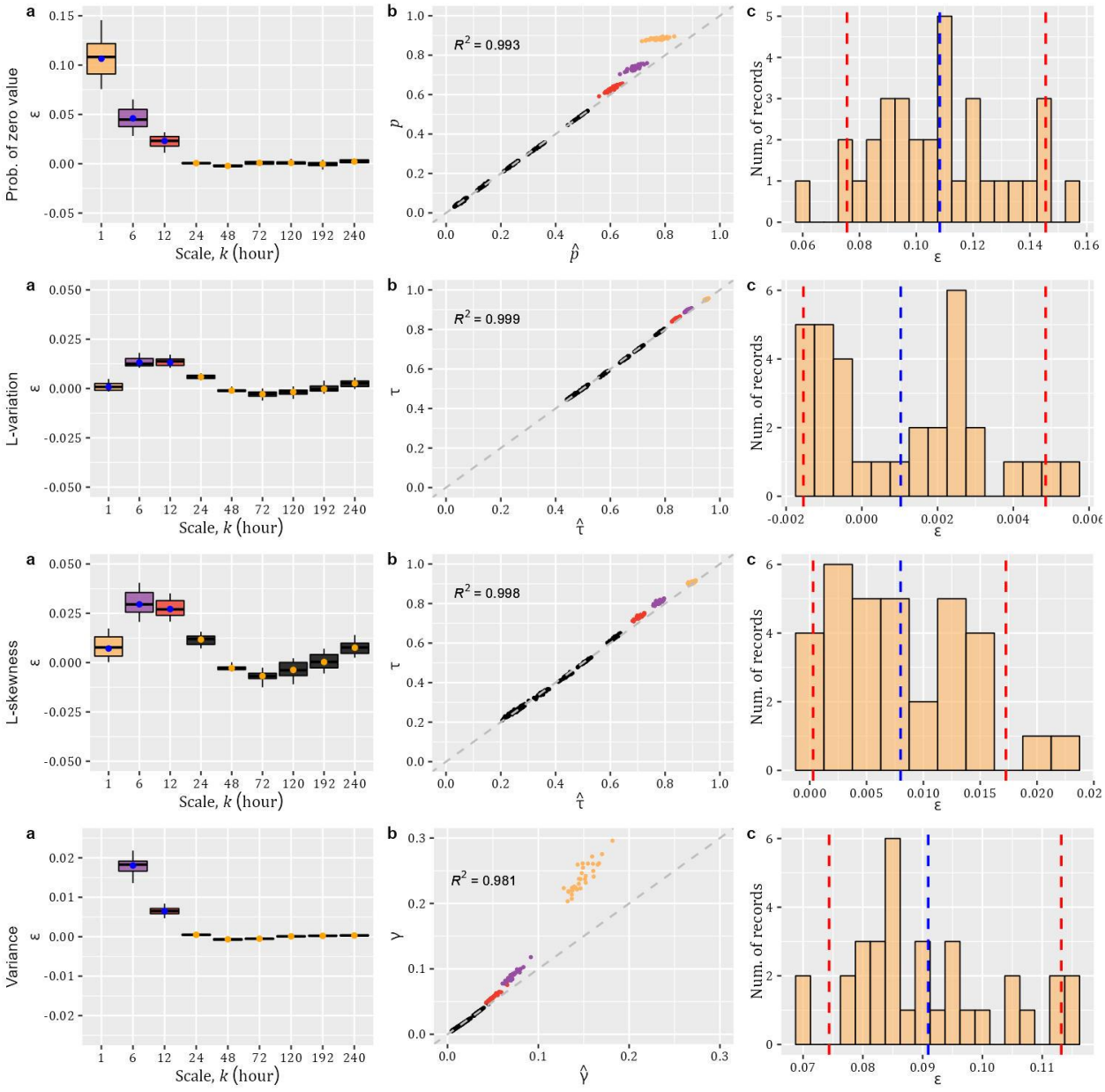

Figure 17. Summary of application of the downscaling approach to the 33 hourly precipitation station in Netherlands. Each row concerns a different statistic (i.e., probability of zero value, L-variation, L-skewness, and variance respectively). Note that $\epsilon = m^{(k)} - \widehat{m}^{(k)}$, where $m^{(k)}$ is the empirical statistic (probability of zero value, L-variation, L-skewness, or variance), and $\widehat{m}^{(k)}$ the statistic estimated by the model.

Funded by the European Union

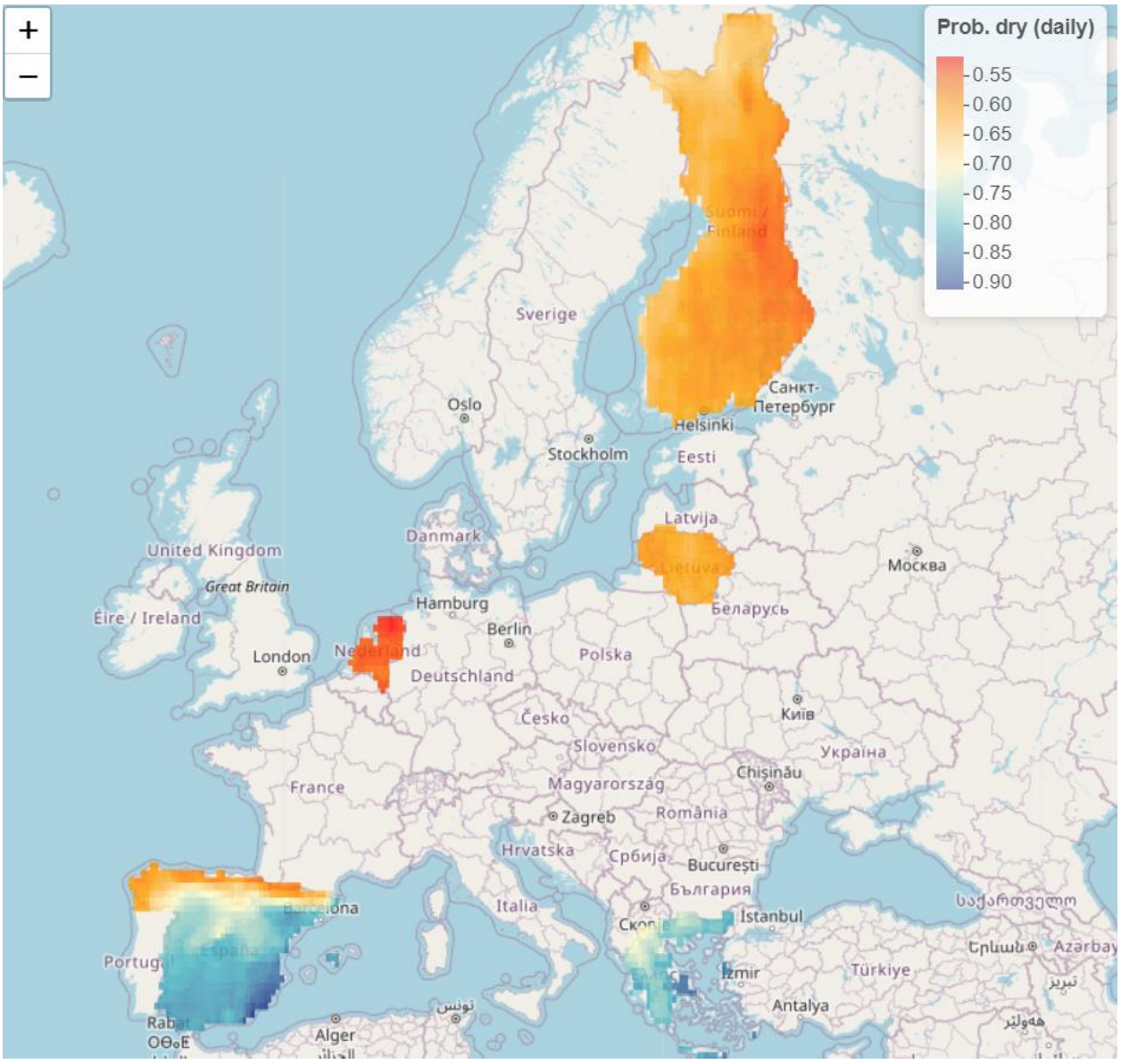

Figure 18. Probability of zero values (i.e., probability dry) of daily precipitation over the countries of the five EIFFEL pilots.

Funded by the European Union

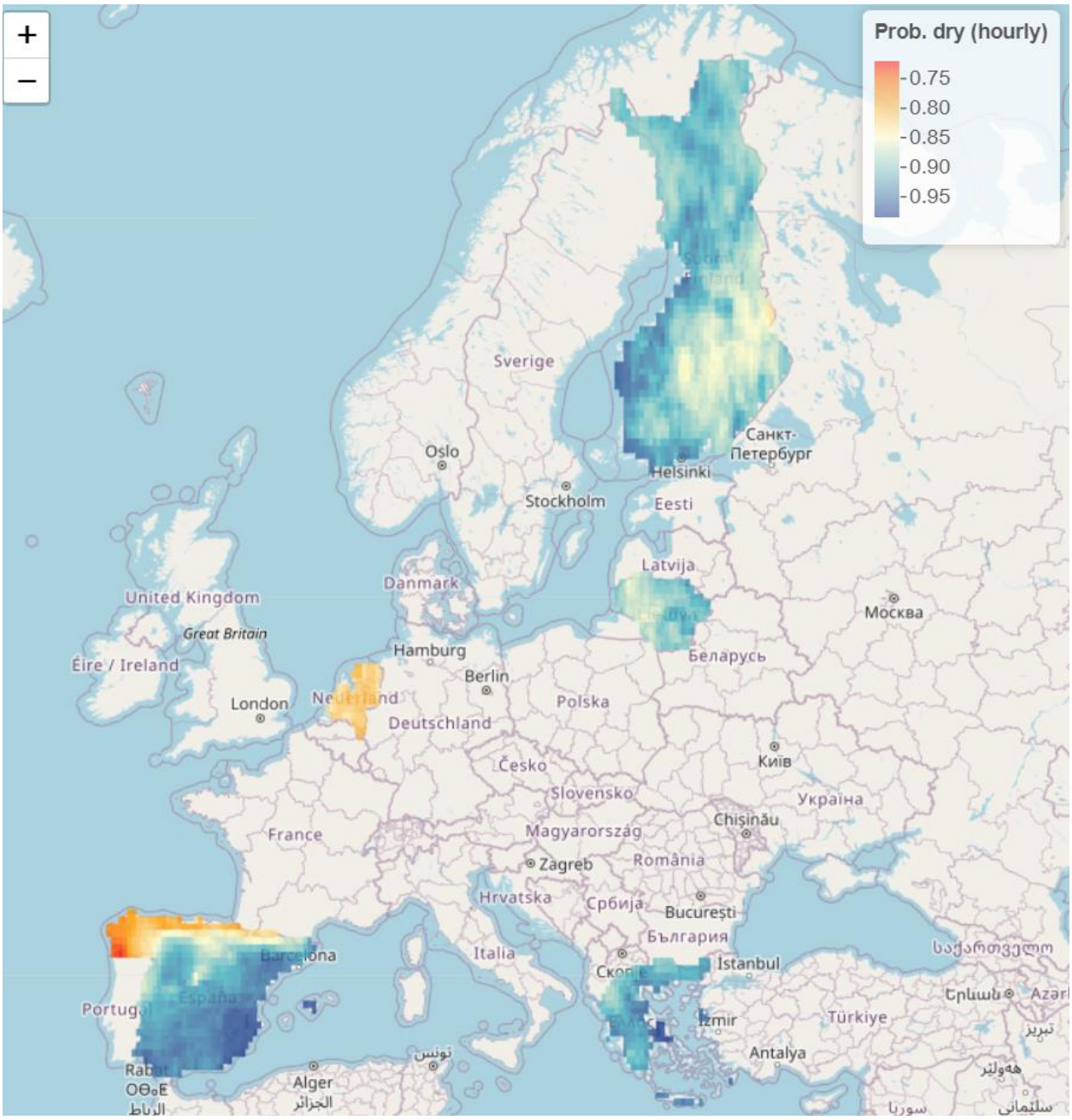

Figure 19. Downscaled (using the method of section 3.3) probability of zero values (i.e., probability dry) of hourly precipitation over the countries of the five EIFFEL pilots.

Funded by the European Union

5 Conclusions

The deliverable details the challenge of temporal augmentation of time series datasets by providing a suite of theoretically justified methods/tools for three common modelling challenges/problems. In particular D4.1/T4.1 copes with:

- The infilling of time series missing values,
- The generation of statistically consistent stochastic realizations for time series data, and
- The lower-scale extrapolation of time series statistics (e.g., temporal downscaling of key statistical properties).

Common characteristics of the developed methods are:

- All methods are built upon solid theoretical foundations, since they rely on statistical/stochastic concepts (such as copulas, and multi-scale properties of stochastic processes).
- The *mechanics* of all methods are easily interpretable and explainable, since the use of black-box models/mechanisms is fairly limited (if not existent at all) – an aspect in direct link with O2. Furthermore, the functionalities of the methods are closely related to the needs of the pilots and have horizontal applicability in different steps of pilots' methodological approaches.
- Given the provided theoretical background, all methods could be straightforwardly extended in various directions, such as, modelling of vector time series and spatial augmentation, etc.

As supported by the demonstrations (utilizing more than 2400 time series datasets) showcased within this report, by using these tools, it should be possible to address common modelling tasks related with temporal augmentation of time series datasets, which could be related with the needs of the project's pilots, and beyond.

Finally, besides the methodology development aspect, D4.1/T4.1 builds a software tool (in R programming language) aiming to increase understanding, reusability and, in a final step, impact in the EO community. Therefore, both the code and documentation (incl. use case examples and tutorial) have been uploaded to the project's repository.

Funded by the European Union

6 References

Biller B, Nelson BL (2003) Modeling and generating multivariate time-series input processes using a vector autoregressive technique. ACM Trans Model Comput Simul 13:211–237. doi: 10.1145/937332.937333

Bras RL, Rodríguez-Iturbe I (1985) Random functions and hydrology. Addison-Wesley, Reading, Mass

Cario MC, Nelson BL (1997) Modeling and generating random vectors with arbitrary marginal distributions and correlation matrix. Technical Report, Department of Industrial Engineering and Management Sciences, Northwestern University, Evanston, Illinois

Cario MC, Nelson BL (1996) Autoregressive to anything: Time-series input processes for simulation. Oper Res Lett 19:51–58. doi: 10.1016/0167-6377(96)00017-X

Crouse M, Baraniuk RG (1999) Fast, exact synthesis of gaussian and nongaussian long-range-dependent processes. IEEE Trans Inf Theory

Dall'Aglio G (1959) Sulla compatibilità delle funzioni de ripartizione doppia

Deodatis G, Micaletti RC (2001) Simulation of highly skewed non-Gaussian stochastic processes. J Eng Mech 127:1284–1295

Der Kiureghian A, Liu P-L (1986) Structural reliability under incomplete probability information. J Eng Mech 112:85–104

Eaton ML (1983) Multivariate statistics: a vector space approach. JOHN WILEY SONS, INC, 605 THIRD AVE, NEW YORK, NY 10158, USA, 1983, 512

Embrechts P, Lindskog F, Mcneil A (2003) Modelling Dependence with Copulas and Applications to Risk Management. In: Handbook of Heavy Tailed Distributions in Finance. pp 329–384

Embrechts P, McNeil AJ, Straumann D (1999) Correlation and Dependence in Risk Management: Properties and Pitfalls. In: Dempster MAH (ed) Risk Management. Cambridge University Press, Cambridge, pp 176–223

Féron R (1956) Sur les tableaux de corrélation dont les marges sont données, cas de l'espace à trois dimensions. Publ Inst Stat Univ Paris 5:3–12

Fréchet M (1951) Sur les tableaux de corrélation dont les marges sont données. Ann Univ Lyon, 3^ e Ser Sci Sect A 14:53–77

Greenwood JA, Landwehr JM, Matalas NC, Wallis JR (1979) Probability weighted moments: Definition and relation to parameters of several distributions expressable in inverse form. Water Resour Res 15:1049–1054. doi: 10.1029/WR015i005p01049

Grigoriu M (1998) Simulation of stationary non-Gaussian translation processes. J Eng Mech 124:121–126

Hosking JRM (1990) L-Moments: Analysis and Estimation of Distributions Using Linear Combinations of Order Statistics. J R Stat Soc Ser B 52:105–124

Joe H (2014) Dependence modeling with copulas. CRC Press

Kossieris P, Tsoukalas I, Efstratiadis A, Makropoulos C (2021) Generic Framework for Downscaling

Funded by the European Union

Statistical Quantities at Fine Time-Scales and Its Perspectives towards Cost-Effective Enrichment of Water Demand Records. Water 13:3429. doi: 10.3390/w13233429

Kossieris P, Tsoukalas I, Makropoulos C, Savic D (2019) Simulating Marginal and Dependence Behaviour of Water Demand Processes at Any Fine Time Scale. Water 11:885. doi: 10.3390/w11050885

Koutsoyiannis D (2010) A random walk on water. Hydrol Earth Syst Sci 14:585–601. doi: 10.5194/hess-14-585-2010

Koutsoyiannis D (2017) Entropy Production in Stochastics. Entropy 19:581. doi: 10.3390/e19110581

Koutsoyiannis D (2006) An entropic-stochastic representation of rainfall intermittency: The origin of clustering and persistence. Water Resour Res 42:n/a-n/a. doi: 10.1029/2005WR004175

Li ST, Hammond JL (1975) Generation of Pseudorandom Numbers with Specified Univariate Distributions and Correlation Coefficients. IEEE Trans Syst Man Cybern SMC-5:557–561. doi: 10.1109/TSMC.1975.5408380

Liu PL, Der Kiureghian A (1986) Multivariate distribution models with prescribed marginals and covariances. Probabilistic Eng Mech 1:105–112. doi: 10.1016/0266-8920(86)90033-0

Mardia K V (1970) A Translation Family of Bivariate Distributions and Frechet's Bounds. Sankhya Indian J Stat Ser A 32:119–122

Nataf A (1962) Statistique mathematique-determination des distributions de probabilites dont les marges sont donnees. C R Acad Sci Paris 255:42–43

Nelsen RB (2007) An introduction to copulas. Springer Science & Business Media

Papoulis A (1991) Probability, Random Variables, and Stochastic Processes, Third edit. McGraw-Hill Series in Electrical Engineering. New York City, New York, USA: McGraw-Hill

Sklar A (1973) Random variables, joint distribution functions, and copulas. Kybernetika 9:449–460

Sklar M (1959) Fonctions de repartition an dimensions et leurs marges. Publ Inst Stat Univ Paris 8:229–231

Stekhoven DJ, Buhlmann P (2012) MissForest--non-parametric missing value imputation for mixed-type data. Bioinformatics 28:112–118. doi: 10.1093/bioinformatics/btr597

Tsay RS (2013) Multivariate Time Series Analysis: with R and financial applications. John Wiley & Sons, Hoboken, New Jersey

Tsoukalas I (2018) Modelling and simulation of non-Gaussian stochastic processes for optimization of water-systems under uncertainty. PhD Thesis, Department of Water Resources and Environmental Engineering, National Technical University of Athens (Defence date: 20 December 2018)

Tsoukalas I (2022) The tales that the distribution tails of non-Gaussian autocorrelated processes tell: efficient methods for the estimation of the k -length block-maxima distribution. Hydrol Sci J 67:898–924. doi: 10.1080/02626667.2021.2014056

Tsoukalas I, Efstratiadis A, Makropoulos C (2018a) Stochastic Periodic Autoregressive to Anything

Funded by the European Union

(SPARTA): Modeling and Simulation of Cyclostationary Processes With Arbitrary Marginal Distributions. Water Resour Res 54:161–185. doi: 10.1002/2017WR021394

Tsoukalas I, Efstratiadis A, Makropoulos C (2019) Building a puzzle to solve a riddle: A multi-scale disaggregation approach for multivariate stochastic processes with any marginal distribution and correlation structure. J Hydrol 575:354–380. doi: 10.1016/j.jhydrol.2019.05.017

Tsoukalas I, Kossieris P, Makropoulos C (2020) Simulation of Non-Gaussian Correlated Random Variables, Stochastic Processes and Random Fields: Introducing the anySim R-Package for Environmental Applications and Beyond. Water 12:1645. doi: 10.3390/w12061645

Tsoukalas I, Makropoulos C, Koutsoyiannis D (2018b) Simulation of Stochastic Processes Exhibiting Any-Range Dependence and Arbitrary Marginal Distributions. Water Resour Res 54:9484–9513. doi: 10.1029/2017WR022462

Tsoukalas I, Makropoulos C, Koutsoyiannis D (2018c) Simulation of stochastic processes exhibiting any-range dependence and arbitrary marginal distributions. Water Resour Res. doi: 10.1029/2017WR022462

Xiao Q (2014) Evaluating correlation coefficient for Nataf transformation. Probabilistic Eng Mech 37:1–6. doi: 10.1016/j.probengmech.2014.03.010

Funded by the European Union

Appendix A: description of the developed T4.1 R library

The following paragraphs consist an excerpt (updated when/if deemed necessary) from the accompanying document of MS9, which briefly described the associated D4.1/T4.1 R library.

Following the architectural specifications described in D2.3 of the project, we developed, in R programming language, a single interface/library for all D4.1/T4.1 tools where the user has the option to choose the one that wishes to employ (i.e., choose among the three methods/tools listed above). The rationale behind employing a single interface for the T4.1 tools is that all methods require the user to provide as an input a historical time series record, as well as the recognition of the potential synergies and continuity between the three methods (e.g., one may wish to: infill the missing values of the historical record, downscale its statistics, and then generate a stochastic realization of it). The overall architecture of the R library is visualized in **Figure 20** with the help of a simplified flowchart - for more details on the matter the interested reader is referred to D2.3. Direct dependencies of the developed R library are the following libraries: `anySim`, `condMVNorm`, `DEoptim`, `fGarch`, `fitdistrplus`, `homtest`, `lmom`, `lmomco`, `lubridate`, `Matrix`, `matrixStats`, `moments`, `nloptr`, `readr`, `stats` and `xts`. The above mentioned R libraries are open-source and freely available, as well as can be found/obtained from the Comprehensive R Archive Network (CRAN) and/or GitHub.

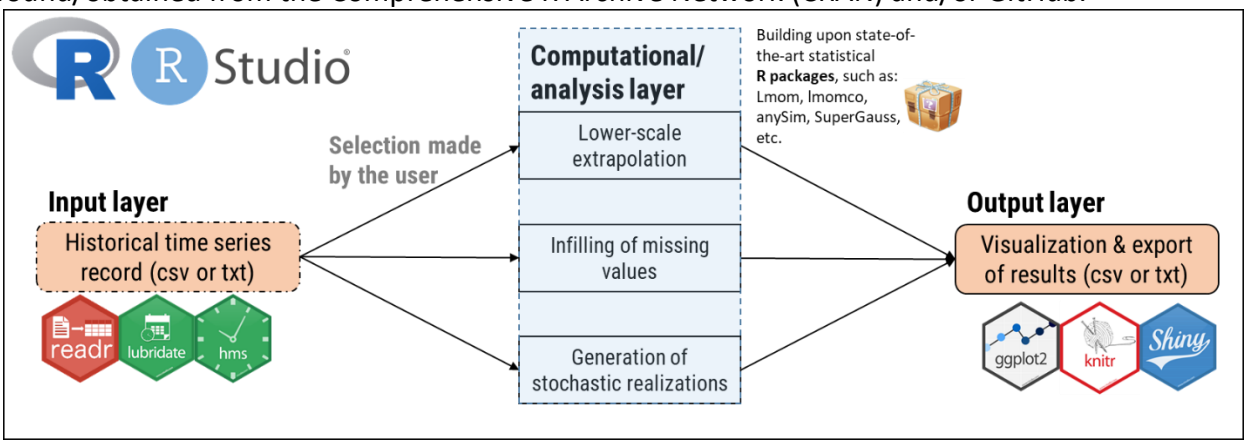

Figure 20. Flowchart illustrating the High Level Layered Architecture of the unified T4.1 system (obtained from D2.3).

During the development of the Alpha version of the tools special focus was given on the input and computational/analysis layer, while the output layer will be further developed during the upcoming months.

As far it concerns functionality, the developed R scripts can be organized in two main categories. The first are scripts that contain auxiliary/Utility functions (for a list see **Table 5**), such as functions to import and clean time series data (see `import_clean_functions.R`), while the second regards method/tool-related scripts (see the list and classification provided in **Table 6**) developed to consist the computational/analysis layer of T4.1 toolkit. On the other hand, a complete list, as well as a brief description of T4.1 library's R functions can be found in **Table 7**.

Funded by the European Union

Table 5. List of R scripts containing a variety of utility functions useful for developments of T4.1.

Utility functions
import_clean_functions.R
help_funs.R
distr_funs.R
cs_funs.R
error_funs.R
get_funs.R

Table 6. List, and classification, of R scripts according to their main functionality (i.e., infilling of time series missing values, generation of statistically consistent stochastic realizations for time series data, lower-scale extrapolation of time series statistics).

Infilling of missing values	Generation of stochastic realizations	Statistics downscaling
infill_rem_Cycle.R	fit_model_stationary.R	stats_over_scales.R
infill_hourlyData.R	fit_model_cyclical_stationary.R	stats_downscaler.R
infill_dailyData.R	fit_model_cyclostationary.R	stats_ObjFun_fitting.R
infill_monthlyData.R	simmodels_funs.R	

Table 7. R functions and brief description of T4.1 library.

R function	Brief Description
ACFVfromClimaco	Estimation of the autocorrelation function using as input the process's climacogram (i.e., variance over multiple temporal scales)
aggregation.proc	Temporal aggregation of time series
align.time.down	Auxiliary function to clean/adjust the time step of a time series
apply.periodly	Auxiliary function that facilitates some function at periodic manner
cgFHKC	Theoretical climacogram of the Filtered Hurst Kolmogorov process
countifGreater	Find the number of data points greater than a threshold value
countifSmaller	Find the number of data points smaller than a threshold value
countNA	Find the number of NA data points
csCAS	Theoretical autocorrelation of a Cauchy-type process (CAS)
csCASobj	Auxiliary function used to fit the Cauchy-type model to time series data
csPEXP	Theoretical autocorrelation of power exponential process (PEXP)
csPEXPobj	Auxiliary function used to fit the power exponential model to time series data
dgam	Probability density function of the Gamma distribution
DistrFit	Distribution fitting through quantiles matching
Distrfitobj	Auxiliary function for the distribution fitting through quantiles matching
dkappa	Probability density function of the Kappa distribution
egam	Fitting of the Gamma distribution using L-moments
ekappa	Fitting of the Kappa distribution using L-moments
elmom	Fitting an arbitrary distribution using L-moments

Funded by the European Union

fit_model_cyclical_stationary	Fitting a cyclically stationary non-Gaussian stochastic model
fit_model_cyclostationary	Fitting a cyclostationary non-Gaussian stochastic model
fit_model_stationary	Fitting a stationary non-Gaussian stochastic model
fitCS	Fitting of a theoretical correlation structure (CAS or PEXP) to time series data
FitGGBr	Fitting of the Generalized Gamma and Burr type XII distributions using L-moments
get_day	Auxiliary function to select data of a given day (i.e., 1,…,30)
get_hour	Auxiliary function to select data of a given hour (i.e., 0, …, 23)
get_month	Auxiliary function to select data of a given month (i.e., 1, …, 12)
import_fix_ts	Import and clean a csv file containing time series data
infill_dailyData	Infilling of missing values for time series data with daily temporal resolution
infill_hourlyData	Infilling of missing values for time series data with hourly temporal resolution
infill_monthlyData	Infilling of missing values for time series data with monthly temporal resolution
mburr	Moment-generating function of the Burr type XII distribution
mdagum	Moment-generating function of the Dagum distribution
mgengamma	Moment-generating function of the Generalized Gamma distribution
MHE	Error metric based on hyperbolic functions
MSE	Error metric identical to mean squared error
pburr	Cumulative distribution function of the Burr type XII distribution
pdagum	Cumulative distribution function of the Dagum distribution
pexpweibull	Cumulative distribution function of the Exponentiated Weibull distribution
pgam	Cumulative distribution function of the Gamma distribution
pgengamma	Cumulative distribution function of the Generalized Gamma distribution
pkappa	Cumulative distribution function of the Kappa distribution
qburr	Quantile function of the Burr type XII distribution
qdagum	Quantile function of the Dagum distribution
qexpweibull	Quantile function of the Exponentiated Weibull distribution
qgam	Quantile function of the Gamma distribution
qgengamma	Quantile function of the Generalized Gamma distribution
qkappa	Quantile function of the Kappa distribution
rburr	Random number generation for the Burr type XII distribution
rdagum	Random number generation for the Dagum distribution
rem_dailyCycle	Auxiliary function for the infilling of missing values tool
rem_hourlyCycle	Auxiliary function for the infilling of missing values tool
rem_monthCycle	Auxiliary function for the infilling of missing values tool
remove_leap	Remove leap years from xts objects
rexpweibull	Random number generation for the Exponentiated Weibull distribution
rgam	Random number generation for the Gamma distribution

Funded by the European Union

`rgengamma`	Random number generation for the Generalized Gamma distribution
`rkappa`	Random number generation for the Kappa distribution
`s2scor`	Estimation of season-to-season correlations
`sim_model_cyclical_stationary`	Simulation of non-Gaussian cyclically-stationary process
`sim_model_cyclostationary`	Simulation of non-Gaussian cyclostationary process
`sim_model_stationary`	Simulation of non-Gaussian stationary process
`statfunscale`	Theoretical scaling law model for statistics
`stats_downscaler`	Downscale the statistical quantity of interest
`stats_ObjFun_fitting`	Fitting of a theoretical scaling law model on observed statistics across multiple temporal scales
`stats_over_scales`	Estimate the statistical quantity of interest at multiple temporal scales
`StatsOfSeries`	Estimate key statistical quantities of time series data
`which.na`	Find the index of NA data points

All in all, it can be argued that the *Alpha* version of T4.1 library (see MS9) address all the requirements set in D2.2, at a satisfying level (about 78% completion; see **Table 8**), while there are already included *Alpha* version for all identified functional and non-functional requirements (see also **Table 9**).

Table 8. Progress of completion of each requirement specified in in relation to D2.2.

Requirement Identifier	Name	Status	Alpha Version
FN 4.1-1	Generation of statistically consistent synthetic time series	In progress / 55%	Included
FN 4.1-2	Temporal downscaling of statistical quantities	In progress / 55%	Included
FN 4.1-3	Infilling of time series missing values (imputation of missing values)	In progress / 50%	Included
FN 4.1-4	Stochastic methods for augmenting the temporal resolution - variables	In progress / 60%	Included
FN 4.1-5	Stochastic methods for augmenting the temporal resolution - process	In progress / 60%	Included
Note: The first 3 requirements are self-explanatory, while FN 4.1-4 and FN 4.1-5 are linked with non-functional requirements. The description of the first (i.e., FN 4.1-4) reads as follows: "*Suitable for both physical and non-physical processes (e.g., rainfall, temperature, water demand time series)*", while the description of the second (i.e., FN 4.1-5) reads as follows: "*Suitable for stationary and cyclo-stationary processes*".			

Funded by the European Union

Table 9. Progress of development and the requirements (related with D2.2 and D2.3).

Component Identifier	Name	Status	Requirement Identifier
Temporal Resolution Augmentation toolkit	Temporal Resolution Augmentation toolkit	In progress / 56% (derived as the average of the status listed in Table 3)	FN 4.1-1, FN 4.1-2, FN 4.1-3, FN 4.1-4, and FN 4.1-5

Funded by the European Union