Classification of time series generation processes using experimental tools: a survey and proposal of an automatic and systematic approach

Renato P. Ishii*
Department of Computing Systems,
Federal University of Mato Grosso do Sul,
Campinho Grande-MS Caixa Postal 549, 79070-900, Brazil
Fax: +55-67-3345-7518
E-mail: renato@facom.ufms.br
*Corresponding author

Ricardo A. Rios and Rodrigo F. Mello
Department of Computer Science,
Institute of Mathematics and Computer Sciences,
São Carlos-SP 13566-590, Brazil
Fax: +55-16-3373-9633
E-mail: rios@icmc.usp.br
E-mail: mello@icmc.usp.br

Abstract: By modelling the outputs produced by real world systems, we can study and, therefore, understand how they work and behave under different circumstances. This is especially interesting to support the prediction of future behaviour and, consequently, decision-making, what is particularly required in certain application domains. In order to proceed with such modelling, we organise system outputs as time series and study how those series were generated. The study of the time series generation process typically requires specialists and also detailed information on how and where data was obtained from. However, none of them may be available in certain circumstances. Such limitations motivated this paper which presents a survey of techniques commonly used to evaluate and classify time series generation processes and, most importantly, a novel automatic and systematic approach to conduct such task with a minimum of human intervention and subjectivity. By using such approach, researchers can select adequate techniques to model time series, reducing the modelling time and improving the chances to obtain higher accuracy.

Keywords: time series; modelling; decision-making; stochasticity; determinism; linearity; stationarity; experimental tools; time series classification; automatic approach; survey.


Biographical notes: Renato P. Ishii is with the College of Computing, Department of Computing Systems, Federal University of Mato Grosso do Sul, Campo Grande, Brazil. He completed his PhD at the University of São Paulo, São Carlos, in 2010. His research interests include autonomic computing, cluster and grid computing, bioinspired computing and time series evaluation.

Ricardo A. Rios is a PhD candidate at the Institute of Mathematics and Computer Sciences, Department of Computer Science, University of São Paulo, São Carlos, Brazil. He finished his Master at the Federal University of São Carlos, São Carlos, in 2008. His research interests include grid computing, forecasting, time series analysis and bioinspired computing.

Rodrigo F. Mello is currently a Faculty Member at the Institute of Mathematics and Computer Sciences, Department of Computer Science, University of São Paulo, São Carlos, Brazil. He completed his PhD at the University of São Paulo, São Carlos in 2003. His research interests include autonomic computing, grid computing, load balancing, scheduling and bioinspired computing.
1 Introduction

The experimental data analysis has attracted the attention of several researchers, who have been attempting to develop methods and techniques to model and understand systems behaviour. In general, by modelling the outputs generated by a system, it is possible to estimate its generation rule (or function), that defines the system behaviour and, consequently, the values of its observations. After knowing such rule, we can take more accurate decisions, simulate and predict situations (Mello and Yang, 2009).

The outputs produced by real world systems present a strong temporal dependency, i.e., the correlation among adjacent observations is best explained in terms of the dependency level (partial or full) (Shumway and Stoffer, 2006). This dependency strongly reduces the modelling accuracy of conventional techniques. In order to overcome this problem, a new area was developed called time series analysis (Box et al., 1994), in which data are commonly organised in terms of variables and their observations over time.

Before modelling, nevertheless, it is necessary to understand the implicit features embedded in data such as the stationarity, linearity, and stochasticity. A time series is said to be stationary when its observations are in a particular state of statistical equilibrium, i.e., they evolve over time around a constant average and variance (Box et al., 1994). On the other side, in linear time series, observations are composed of a linear combination of past occurrences and noises. Finally, stochastic time series are formed by random observations and relations, which follow probability density functions and may change over time (Box et al., 1994).

The identification of these features, however, is not a simple task and requires not only specialists’ opinions but also additional information on how series are obtained (what supports the characterisation of the generation process). The main problem with this approach is the subjectivity imposed by specialists, which can lead to failures. Moreover, during time series modelling, mainly when considering those obtained from real world systems, researchers neither have further information nor specialists available to support this task (Shumway and Stoffer, 2006).

These issues have been constantly limiting the appropriate modelling of time series, mainly when researchers are not specialists in time series analysis or belong to other areas not related to statistics or mathematics, such as pharmaceutical research, biotechnology, medicine, public health, agriculture, and climate modelling.

The need for specialists and the lack of a step-by-step analysis on how to evaluate time series generation process have motivated the development of methods and techniques to understand implicit features embedded in temporal data. In this sense, for example, the stochasticity level of time series was studied by Marwan et al. (2007) and Marwan (2008), whereas the linearity was analysed using White neural network (WNN) (White, 1990). Finally, an example of technique proposed to study the influences of stationarity in time series is found in Provanzale et al. (1992). In spite of such techniques are independently found in literature, there is no work that unifies them all into a single approach to systematise and automatise the evaluation of time series generation processes.

In that sense, this paper presents a survey of techniques commonly used to evaluate and classify time series generation processes and most importantly a novel automatic and systematic approach to conduct such task with a minimum of human intervention and subjectivity. Our approach combines dynamical systems, statistical tools and tests to support such task. As main motivation, we intend to improve the selection of modelling tools, and therefore, the accuracy of time series modelling.

We conducted two sets of experiments in order to evaluate our approach. In the first, we considered synthetic time series, obtained from mathematical and statistical models. This set is very important as we previously know time series embedded features and, consequently, can draw useful conclusions on our approach. In the second, we considered real world time series, presenting unknown generation processes. Having no previous information on data, we decided to evaluate the accuracy of our approach by predicting observations. In summary, results confirm a significant improvement in modelling accuracy.

This paper is organised as follows: background information and related work are presented in Section 2; Section 3 presents our classification approach for time series generation processes as well as a survey of modelling techniques; Section 4 presents experiments on synthetic data, in order to validate our approach; experiments are performed on real world data and results analysed in Section 5; Section 6 shows accuracy results when predicting observations. Finally, this paper presents concluding remarks and references.

2 Background

Time series have been frequently used to organise and represent observations collected over time. By modelling time series, we can study, analyse and predict the behaviour of systems. Time series are composed of sequential data (or observations) gathered from one (univariate) or more (multivariate) variables over time. Univariate series are composed of a single variable \( X \), which presents observations in the form \( (x_0, x_1, \ldots, x_{n-1}) \) (considering time interval \( t \in [0, n - 1] \)). On the other hand, in a multivariate series \( X \), \( k \) variables are observed at every time instant \( t \). This series can be described as \( (x_{1t}, x_{2t}, \ldots, x_{kt}) \), with \( t \in [0, n - 1] \).

Besides this intuitive definition, a time series \( X_t \) (the index \( t \) denotes the time instant) can be formally defined by the sum of three non-observable components: \( X_t = \Gamma_t + S_t + \varepsilon_t \), in which \( \Gamma_t \) represents the trend, \( S_t \) is the seasonality and \( \varepsilon_t \) is a random component (Morettin and Toloi, 2004). The component \( \Gamma_t \), or trend, represents the variations of the series behaviour. The component \( S_t \), or seasonality, indicates whether the series behaviour tends to repeat itself within time intervals \( \Delta t \). These components are called non-observable because they are not obtained from
the monitored system, however, they can be inferred when considering the relation among data over time (Morettin and Tolo, 2004).

The study of time series involves the understanding of the generation process. By knowing such process, we are able to develop adequate techniques to model it. There are some aspects to consider when studying a time series. The first evaluates whether a series is deterministic or stochastic. A deterministic series presents recurrent behaviour; this is, it repeats itself over time in the same or in different scales. The good point about those systems is that they can be modelled by using deterministic differential equations. Thus, we can model and predict next observations based on previous knowledge. However, even being deterministic, some of those systems are very sensitive to small perturbations and initial conditions, what makes impossible long-term predictions (Marwan et al., 2007). Nevertheless, even for these chaotic systems, short-term predictions are feasible. On the other hand, series can also be generated by stochastic systems, where future observations may depend on previous ones (such as in deterministic systems) but also on random effects. Statistical techniques are commonly used to model those systems, once they consider time-dependent and probabilistic relations among observations (Box et al., 1994).

By studying the determinism and stochasticity of series, we can better select techniques to model them. In this sense, researchers have been proposing and applying different techniques to evaluate such series features like the recurrence ratio found by RP, one can distinguish those two types of errors, however, that work arises due to inputs do not contain all variables influencing outputs. In that sense, the author observed that such lack of influence generates two kinds of errors: the inherently unavoidable errors due to the inexact relationship in between inputs and outputs, and the potentially avoidable approximation errors. White concludes that it is impossible to distinguish those two types of errors, however, that work provides a statistical hypothesis test to confirm that no approximation errors exist. Thus, that test makes possible to distinguish whether a technique is capable of exactly approximating a system subject to inherent noise or whether there is some non-linear structures neglected by the technique and, therefore, it generates potentially avoidable approximation errors. In that work, White considers a multilayer artificial neural network as the technique to evaluate time series linearity

After characterising a system as deterministic or stochastic, we can focus on which type of technique is better to model it. When the system is deterministic, we may consider chaos-theory-based approaches. However, when the system is stochastic, we would select statistical models. However, in this case, we still need to understand additional system properties to improve modelling, such as the series linearity and also its stationarity. In linear time series, observations are formed by a linear combination of past values and noises. Therefore, the linearity of a series is presented in the model, map or process that created it. On the other hand, non-linear series are formed by a non-linear combination of past occurrences and noises. Non-linear series are more difficult to be understood and modelled.

In order to determine whether a time series was generated by a linear process, Hegger et al. (1998) employ a method to evaluate the linearity of series through surrogate data (Ghez and Vaienti, 1992) which, based on the original series, generates surrogate series and compares them against the original one. Those surrogate series are typically generated using a stationary linear process. Then, a hypothesis test is set, in which the null hypothesis assumes that there is no significant difference in between a set of surrogate series and the original one (therefore, the original series is assumed to be linear). The alternative hypothesis may be accepted when there is a significant difference, thus, the original series is said to be non-linear. Galka and Ozaki (2001) also study and compare different approaches to evaluate series linearity. That work considers surrogate series and evaluates hypotheses by using the following tests: the irreversibility test (Diks et al., 1995), the local constant predictor (Schreiber and Schmitz, 1997) and their own method. They concluded that each test presents better results to different scenarios.

Besides the application of the surrogate data method, the linearity of time series was still analysed by White (1989), who studied the outputs of multilayer feedforward networks. The author states that there is a lack of exact relation in between network inputs and outputs and that arises due to inputs do not contain all variables influencing outputs. In that sense, the author observed that such lack of influence generates two kinds of errors: the inherently unavoidable errors due to the inexact relationship in between inputs and outputs, and the potentially avoidable approximation errors. White concludes that it is impossible to distinguish those two types of errors, however, that work provides a statistical hypothesis test to confirm that no approximation errors exist. Thus, that test makes possible to distinguish whether a technique is capable of exactly approximating a system subject to inherent noise or whether there is some non-linear structures neglected by the technique and, therefore, it generates potentially avoidable approximation errors. In that work, White considers a multilayer artificial neural network as the technique to evaluate time series linearity

Lee et al. (1993) compare the multilayer neural network test proposed by White (1989) against others: the Keenan test, Tsay test, White dynamic information matrix test, McLeod-Li test, Ramsey Reset test, Brock-Dechert-Scheinkman test and the Bispectrum test. Experimental results confirm that no test was capable of outperforming all others. Nevertheless, the multilayer neural network test performed as well as other tests. In some scenarios, it outperformed others though.

After studying the series linearity, we still need to evaluate the series stationarity. The observations in stationary series are organised in a particular state of
statistical equilibrium (Box et al., 1994), i.e., they evolve randomly over time, around a constant mean and variance (Morettin and Toloi, 2004). Strictly stationary series are those whose properties are not affected according to changes in the origin of time, i.e., when the probability density function associated with n observations \((x_0, x_1, \ldots, x_{n-1})\) does not change after the displacement of observations in time under a constant \(k(x_0+k, x_1+k, \ldots, x_{n-1+k})\) (Box et al., 1994). Besides that, stochastic series \(\{X_t, t \in \mathbb{N}\}\) are called weakly stationary or second order stationary, when the average and variance are constant \((E(X_t) = \mu \text{ and } var(X_t) = \sigma^2)\) respectively and the covariances are defined by \(cov(X_t, X_{t+\tau}) = \gamma(\tau)\) (Box et al., 1994). The modelling and analysis of time series usually deals with strictly stationary and weakly stationary time series as simply stationary (Morettin and Toloi, 2004).

When time series do not satisfy the conditions of stationarity, they are called non-stationary stochastic series. Those series usually present evolutionary behaviour, due to variations in the trend component, and they are typically modelled by explosive processes (Morettin and Toloi, 2004). In those processes, the average of the observed values tends to increase (or decrease) as new values over time are considered. As an example of a process with evolutionary behaviour, consider a system that models the growth of a population given by \(y_t = \kappa \cdot x_t + \varepsilon_t\). In this model, \(x_t\) represents the size of the population at time instant \(t\), \(\varepsilon_t\) is a random component and \(\kappa\) is a constant to determine the evolution trend of the population size over time. One way of dealing with trends in those time series is by using linear filters, which convert the series outputs \(x_t\) in \(y_t\). Then, we define \(z_t = x_t - y_t\), which is a trend-free series (Morettin and Toloi, 2004).

By classifying the stationarity, we can also improve the technique selection mechanism. Furthermore, when neglecting the existence of non-stationarity, we may produce errors when computing other dynamical system measurements such as the Lyapunov exponent, the correlation dimension and entropies (Yu et al., 1998).

Two techniques are commonly applied to evaluate stationarity: the auto-correlation function (ACF) (Box et al., 1994) and the space-time separation plot (STP) (Provanzale et al., 1992). The ACF is a well-known technique and widely used to analyse the stationarity in time series. In general, it correlates the series with itself, evaluating the similarity in between observations as a function of the time separation between them. The ACF supports the finding of repeated patterns. On the other side, the STP, proposed by Provanzale et al. (1992), works similarly to identify temporal correlations in time series. Although, the resultant graphics generated by STP are easier analysed than ACF, both approaches can be used to evaluate how patterns repeat themselves over time. Scale (or amplitude) variations in such measurements indicate the series integrate previous observations to generate next ones, what depicts non-stationarity.

The application of these techniques to evaluate stationarity in time series is commonly used in several works and their efficiency is confirmed in several works (Li et al., 2009; Provanzale et al., 1992; Yu et al., 1998). Li et al. (2009) employ the ACF to test the stationarity of long-range dependent network traffic and, therefore, improve the analysis of traffic series. Provanzale et al. (1992) employ the STP to support their work on evaluating and distinguishing low-dimensional chaos and random behaviour in time series. Yu et al. (1998) extend the work by Provanzale et al. (1992) by proposing the space time-index plot, in which density distributions of observations are normalised according to time indices. Those works attempt to improve time series modelling by assessing their stationarity.

3 Classification of time series generation processes

After presenting basic concepts on time series (Section 2), here we organise all tests according to a tree view (Figure 1) and also indicate models for every branch. At the first level, we have time series. The second level separates deterministic from stochastic series. When a series is deterministic, it can be studied using chaos-theory tools. Such branch considers the false nearest neighbours (Kennel et al., 1992) to compute the embedded dimension and the auto-mutual information (Fraser and Swinney, 1986) to obtain the separation dimension, this is, the time delay or separation in between consecutive observations. After obtaining both, the time series can be reconstructed in its phase space (Marwan et al., 2007). This space only considers states (observations or set of observations) which lead to further states. The time is then removed from that space, thus, we can make the regression of such states function in order to understand series tendencies. In the deterministic branch, as presented in Figure 1, there is only one level because deterministic time series can be modelled by using differential equations or maps, which represent the generation process, making unnecessary further evaluations.¹

Still in the second level, we have the stochastic type of series. When under that branch, series still must be evaluated in order to select the best modelling approach. Thus, we proceed with the third level which verifies series linearity. Non-linear series can be addressed by using non-linear modelling approaches such as radial basis function (RBF) approximation (Mitchell, 1997), artificial neural networks (Haykin, 2008), the autoregressive conditional heteroskedasticity (ARCH) model, or using filters to remove non-linear components (Shumway and Stoffer 2006) (transforming the series in a linear one) and also by considering other polynomial approaches (Casdagli, 1989). When linear, we still need to evaluate whether the series is stationary or not. When the series is stochastic, linear and stationary, it can be modelled by using techniques such as autoregressive models (ARs), moving averages (MAs) or autoregressive moving average (ARMA) models (Box et al., 1994). When the series is stochastic, linear and
non-stationary, it can be modelled by using autoregressive integrated moving average (ARIMA) models (Box et al., 1994).

**Figure 1** Classification of time series generation processes and adequate modelling tools

Every modelling technique we indicated (AR, MA, ARMA, ARIMA, chaos theory tools, etc.) was designed to address a specific type of series. For example, the AR considers a linear sum of past observations to model time series (Box et al., 1994). That is why it is indicated to deal with stochastic linear stationary series. The MA model considers a linear sum of past random components or noise, thus, it is also indicated to stochastic linear stationary series (Box et al., 1994). ARMA is composed by AR and MA, thus modelling observations and noise over time. Given its design, ARMA is also indicated to stochastic linear stationary series (Box et al., 1994).

On the other hand, ARIMA models the relation of past observations (AR), relation of past random components or noise (MA) as well as making the integration (making possible to follow changes in trend) of such values to model time series. This is why ARIMA is indicated to approach stochastic linear non-stationary series (Box et al., 1994).

Stochastic non-linear time series are probably the most difficult branch to model. They can be modelled by using techniques which follow such non-linear behaviour of past observations, noise and its integration or by filtering such non-linearity, translating the series into a linear form. On the other hand, there are algorithms and techniques which directly deal with non-linear time series like the Polynom algorithm (Casdagli, 1989). This algorithm is a polynomial predictor, whose base function is defined by $\pi_i f_N : \mathbb{R}^m \rightarrow \mathbb{R}$, $i = 1, ..., m$, in which $m$ represents the variables of degree (at most $d$). In that approach, free parameters $\left( \frac{m + d}{m} \right)$ of the polynomial are chosen in order to reduce equation (1) (Casdagli, 1989).

\[
\sum_{n=1}^{N} (\pi_i x_{n+1} - \pi f_N (x_n))^2
\]

The main objective of the Polynom algorithm is to provide predictors $f_N$ which converage to the original generation function $f$ as soon as $N$ and $d$ increase according to the Weierstrass method (Casdagli, 1989)2 of an approximation theorem.

Thus, we indicate modelling techniques (AR, MA, ARMA, ARIMA, chaos theory tools, Polynom algorithm, etc.) for every branch of the tree we propose (Figure 1), according to the time series features. Consequently, this classification approach is mandatory to understand the time series generation processes and, therefore, improve modelling accuracy. We indicated such techniques due to they are commonly applied in most works available in literature. However, a researcher can replace them by others considered more suitable, but we claim this classification provided by our approach still must be considered, since it is mandatory to understand generation processes and, therefore, improve modelling accuracy.

4 Evaluating time series generation processes

This section presents our approach to classify time series generation processes. Once classified (as presented in Figure 1), we can select adequate modelling techniques. In order to evaluate the proposed approach, we selected a set of five time series to be evaluated, which were generated according to the processes present in Table 1. These synthetic time series were chosen because they are commonly used in the time series analysis area (Box et al., 1994; Shumway and Stoffer, 2006) and their generation rules are previously known. The main objective of these experiments is to evaluate the precision of our approach in classifying correctly a time series according to its generation process, i.e., for example, by analysing the time series created by ARIMA process, it is expected that it is classified as linear, non-stationary, and stochastic.

<table>
<thead>
<tr>
<th>Gen. process</th>
<th>Function</th>
<th># observ.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random walk</td>
<td>$x_i = x_{i-1} + \varepsilon_i$</td>
<td>1,000</td>
</tr>
<tr>
<td>AR</td>
<td>$x_i = c + \sum_{j=1}^{p} \phi_j x_{i-j} + \varepsilon_i$</td>
<td>1,000</td>
</tr>
<tr>
<td>ARIMA</td>
<td>$x_i = c + \sum_{j=1}^{p} \phi_j x_{i-j} + \sum_{j=1}^{q} \theta_j \varepsilon_{i-j}$</td>
<td>1,000</td>
</tr>
<tr>
<td>Logistic map</td>
<td>$x_{i+1} = \alpha x_i (1 - x_i)$</td>
<td>1,000</td>
</tr>
<tr>
<td>Lorenz</td>
<td>Dataset lor63.dat*</td>
<td>20,000</td>
</tr>
</tbody>
</table>

Source: *http://people.virginia.edu/~smb3u/psych611/lor63.dat
4.1 Stochasticity evaluation

As presented in Figure 1, the first test to be conducted aims at distinguishing deterministic and stochastic time series. In deterministic time series, observations are strictly dependent on past ones, i.e., there is a recurrent behaviour which can be unfolded and modelled. Stochastic time series usually present some dependency on past observations and also random components. These components follow probability density functions, which may change over time, difficulting data modelling.

The verification of time series stochasticity and determinism levels can be performed throughout a fundamental property of dynamical systems, called recurrence. This property allows us to characterise the system behaviour in phase space, i.e., in a reconstructed space with a higher number of dimensions, also called state space (Marwan et al., 2007). Formally, a dynamical system is given by a (phase) space, a continuous or discrete time and a time-evolution law. The space elements represent possible system states. Every state in that space is defined by a vector of \( d \) components [such as presented in equation (2)] in which time is removed and only the order of state visitation is maintained.

\[
\bar{v}(t) = (v_1(t), v_2(t), ..., v_d(t))^T
\]

The time-evolution law is a rule that allows to determine the state of the system at every instant \( t \). For time-continuous systems, the time evolution is given by a set of differential equations (Marwan et al., 2007).

When considering experimental scenarios, typically not every relevant component is known to construct state vectors. In such circumstance, the phase space needs to be reconstructed. A method frequently used for that is the Takens’ (1981) immersion theorem, presented in equation (3), in which \( m \) is the embedding dimension and \( \tau \) is the time delay (or separation dimension). All \( \tilde{e}_j \) are unit vectors and span an orthogonal coordinate system \((\tilde{e}_i \cdot \tilde{e}_j = \delta_{ij})\) where \( \delta_{ij} \) is the Kronecker delta function.\(^3\)

Takens’ (1981) immersion theorem guarantees the existence of a diffeomorphism in between the original and the reconstructed spaces. According to that theorem, all behaviour embedded into a series \( x_0, x_1, ..., x_{n-1} \) can be unfolded by reconstructing it in a multidimensional space \( x(m, \tau) = (x_0, x_{\tau+1}, ..., x_{(m-1)\tau}) \). This reconstruction is conducted by using a set of techniques, in this paper we consider the false nearest neighbours (Kennel et al., 1992) and the auto-mutual information (Fraser and Swinney, 1986) ones to obtain the embedding and the separation (also referred as time delay) dimensions, respectively. After obtaining such dimensions, we can reconstruct the phase space and study all system states and their relations (Dodonov and de Mello, 2010).

\[
\bar{v}_j = \sum_{j=1}^{m} x_j + (j-1)\tau \tilde{e}_j
\]

As an example, Figure 2 presents the observations of the Rössler system over time, which was generated by using equation (4) (considering \( a = 0.15, b = 0.20, c = 10 \)). By only observing the time series, we cannot conclude whether it is easy or difficult to model it. However, when reconstructing it in phase space, such as depicted in Figure 3, we observe the states recurrence over time. After the reconstruction as presented in Figure 3, we can make the regression of the trajectory in phase space and, thus, determine future observations based on past ones.

![Figure 2](image1.png)  
**Figure 2** Observations of the Rössler system over time (see online version for colours)

![Figure 3](image2.png)  
**Figure 3** Segment of the phase space trajectory of the Rössler system

A graphical representation of the RP is obtained through the matrix of recurrences such as presented in Figure 4. A phase space vector (trajectory point) at time \( j \) which falls into the \( \varepsilon \)-neighbourhood (grey circle in Figure 3 – defined as \( \varepsilon = 5 \),...
Classification of time series generation processes using experimental tools

In this example) of another phase space vector at time \( i \) is considered as a recurrence point (black point on the trajectory in Figure 3). This is marked with a black point in RP at coordinate \((i, j)\). A phase space vector outside the neighbourhood (empty circle in Figure 3) leads to a white point in RP.

\[
\begin{align*}
x &= -y - z \\
y &= x + ay \\
z &= b + cz
\end{align*}
\]

Before reconstructing a time series into the phase space, we need to verify if such series is either deterministic or stochastic. One of the most common approaches to test whether a series is deterministic is by verifying its property of recurrence. This property can be evaluated by using the RP (Marwan et al., 2007; Marwan, 2008). RP evaluates all possible states that can be visited by a trajectory in phase space. In that sense, there is a recurrence if a trajectory point and its neighbours are close enough, i.e., the distance among these points is lower than a threshold \( \varepsilon \), which defines the boundary for neighbourhood. In other words, considering a two-dimensional space, Figure 5, when a distance \( d \) in between points is lower than \( \varepsilon \), then such points are considered neighbours.

**Figure 4** RP of the Rössler system

\[ R_{i,j} = \begin{cases} 1 & : \tilde{x}_i \approx \tilde{x}_j \\ 0 & : \tilde{x}_i \neq \tilde{x}_j \end{cases}, \quad i, j = 1, \ldots, N \] (5)

\( \tilde{x}_i \approx \tilde{x}_j \) represents that a point is close enough to another, around a maximum distance \( \varepsilon \). Parameter \( \varepsilon \) is essential once systems often do not recur exactly to the same state (or point in trajectory), but within a neighbourhood. Matrix \( R_{i,j} \) compares system states at times \( i \) and \( j \). When states are similar, a value one is set as the matrix element, i.e., \( R_{i,j} = 1 \), on the other hand, when states are different, the corresponding element is set to zero (\( R_{i,j} = 0 \)).

RP is formally defined in equation (6), where \( N \) is the number of observations with \( \tilde{x}_i \in \mathbb{R}^d \) and \( \varepsilon \) is a distance threshold which defines the recurrence in a neighbourhood. It is important to select an adequate value for \( \varepsilon \), because it defines the neighbourhood of a point in the trajectory and consequently how the recurrence will be verified in series. Zbilut and Webber (1992) suggest 10% on the average of observations as a good estimation to \( \varepsilon \). \( \Theta(.) \) is the Heaviside function (i.e., \( \Theta(x) = 0 \) if \( x < 0 \), and \( \Theta(x) = 1 \) otherwise) and \( || \cdot || \) is a norm.

\[
R_{i,j}(\varepsilon) = \Theta\left(\varepsilon - \left|\tilde{x}_i - \tilde{x}_j\right|\right), \quad i, j = 1, \ldots, N
\] (6)

For \( \varepsilon \)-recurrent states, i.e., states which are within a \( \varepsilon \)-neighbourhood, it is possible to introduce the following notion:

\( \tilde{x}_i \approx \tilde{x}_j \Longleftrightarrow R_{i,j} = 1. \)

**Figure 6** One diagonal in RP corresponds to a trajectory section (dashed line) which remains close enough to the other section (solid line) around a \( \varepsilon \)-neighbourhood

In this work, we consider the Euclidean norm to calculate the distance in between a point and its neighbourhood. RP stores recurrence values into vectors, which are used to compose a matrix. Formally, this recurrence matrix is defined as:

In summary, both axes of RP charts represent time. Moreover, elements of the recurrence matrix form textures or patterns which describe series behaviour. Besides that when \( R_{i,i} = 1 \), RP forms a main diagonal line called line of identity (LOI). Moreover, RP is symmetrical by definition with respect to the main diagonal, thus \( R_{i,j} = R_{j,i} \) (Marwan et al., 2007). The length of diagonal lines determines the time interval in which trajectory segments
evolve within a neighbourhood $\varepsilon$. The vertical or horizontal lines correspond to time intervals in which the series state does not change (or present very few modifications). Isolated points represent rare states which occur for a very short period of time.

Based on RP textures, Zbilut and Webber (1992) propose several measurements (RQA or recurrence quantification analysis measurements) to quantify the visual structures, such as: the rate or percentage of recurrence (RR), degree of determinism (DET), maximum length of a diagonal line ($L_{\text{max}}$), degree of divergence (DIV) and the Shannon entropy of the probability density function of diagonal lines (ENTR). All those measurements are based on the recurrence state density and the structures formed by diagonal and vertical lines. Such measurements (computed by means of trajectory matrices along the LOI) produce a time-dependent behaviour which allows the analysis and study of series transitions. The first measure we present is the recurrence rate (RR) which quantifies the recurrence state density. This measure is defined in equation (7), in which $N$ represents the number of observations.

$$
RR(\varepsilon) = \frac{1}{N^2} \sum_{i,j=1}^{N} R_{i,j}(\varepsilon)
$$

As $N \to \infty$, RR represents the probability state $x_i$ recurs in a $\varepsilon$-neighbourhood of phase space. Table 2 shows RR for the five time series we considered (defined in Section 4). Note that, by incrementing $\varepsilon$, RR also increases because the Euclidean space becomes wider and, thus, the probability of recurrent states is also greater.

### Table 2  
Recurrence rate for the five synthetic time series

<table>
<thead>
<tr>
<th>Time series</th>
<th>$\varepsilon = 0.1$</th>
<th>$\varepsilon = 0.5$</th>
<th>$\varepsilon = 1.0$</th>
<th>$\varepsilon = \frac{\mu}{10}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random walk</td>
<td>0.0</td>
<td>0.0001</td>
<td>0.0011</td>
<td>0.0121</td>
</tr>
<tr>
<td>AR(1)</td>
<td>0.0</td>
<td>0.0002</td>
<td>0.0061</td>
<td>0.0</td>
</tr>
<tr>
<td>ARIMA(1,1,1)</td>
<td>0.0</td>
<td>2.2\times 10^{-6}</td>
<td>0.0792</td>
<td>1.3\times 10^{-5}</td>
</tr>
<tr>
<td>Logistic map</td>
<td>0.0003</td>
<td>0.1368</td>
<td>0.8570</td>
<td>5.1\times 10^{-5}</td>
</tr>
<tr>
<td>Lorenz attractor</td>
<td>1.1\times 10^{-5}</td>
<td>0.0003</td>
<td>0.0013</td>
<td>2.0\times 10^{-6}</td>
</tr>
</tbody>
</table>

The measurements based on diagonal line structures allow to quantify the determinism degree (or predictability) of time series. Processes, whose behaviour is uncorrelated, weakly correlated or stochastic, present none or a low number of diagonal lines. On the other hand, deterministic processes present long diagonal lines and textures. After this definition, we present the measurement called DET, equation (8), in which $P(\varepsilon, l)$ is the histogram of diagonal lines of length $l$ as defined in equation (9).

$$
DET = \frac{\sum_{l=1}^{L_{\text{max}}} lP(\varepsilon, l)}{\sum_{l=1}^{\infty} lP(\varepsilon, l)}
$$

$$
P(\varepsilon, l) = \sum_{i,j=1}^{N} \{1 - R_{i-1,j-1}(\varepsilon)\}(1 - R_{i+l,j-l}(\varepsilon))
$$

For $l_{\text{min}} = 1$, the determinism is always 1. In that sense, the selection of $l_{\text{min}}$ must consider histogram $P(\varepsilon, l)$, which may become sparse when $l_{\text{min}}$ is very large, and thus, the representativeness of DET decays. Figure 7 illustrates how a histogram $P(\varepsilon, l)$ is obtained, considering the diagonal length of RP, and Table 3 presents DET for the five synthetic time series we considered.

### Table 3  
Degrees of determinism for the five synthetic time series with $L_{\text{min}} = 2$

<table>
<thead>
<tr>
<th>Time series</th>
<th>$\varepsilon = 0.1$</th>
<th>$\varepsilon = 0.5$</th>
<th>$\varepsilon = 1.0$</th>
<th>$\varepsilon = \frac{\mu}{10}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random walk</td>
<td>0.0</td>
<td>0.0001</td>
<td>0.0011</td>
<td>0.0121</td>
</tr>
<tr>
<td>AR(1)</td>
<td>0.0</td>
<td>0.0002</td>
<td>0.0061</td>
<td>0.0</td>
</tr>
<tr>
<td>ARIMA(1,1,1)</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0434</td>
<td>0.0</td>
</tr>
<tr>
<td>Logistic map</td>
<td>0.2426</td>
<td>0.4750</td>
<td>0.9911</td>
<td>0.0869</td>
</tr>
<tr>
<td>Lorenz attractor</td>
<td>0.9515</td>
<td>0.9488</td>
<td>0.9500</td>
<td>0.9633</td>
</tr>
</tbody>
</table>

A diagonal line of length $l$ represents the time interval in which two trajectories remain close enough to each other. In fact, those lines are related to the trajectory divergence. The average length of diagonals is defined in equation (10). Consecutively, we can define the maximum length of a diagonal line as presented in equation (11). This measurement represents the longest period of time in which two trajectories remain $\varepsilon$-equidistant, thus $N_l = \sum_{l>l_{\text{min}}} P(\varepsilon, l)$ is the total length of the diagonal.
The next measurement is the DIV, which is obtained by applying the inverse of $L_{\text{max}}$ as defined in equation (12). The faster the trajectory segments diverge, the shorter are diagonal lines and higher is the value of DIV. Table 4 presents DIV for the five synthetic time series under study.

$$\text{DIV} = \frac{1}{L_{\text{max}}}$$  \hspace{1em} (12)

### Table 4 DIV for the five synthetic time series

<table>
<thead>
<tr>
<th>Time series</th>
<th>$\varepsilon = 0.1$</th>
<th>$\varepsilon = 0.5$</th>
<th>$\varepsilon = 1.0$</th>
<th>$\varepsilon = \frac{\mu}{10}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random walk</td>
<td>0.0</td>
<td>0.5</td>
<td>0.25</td>
<td>0.0102</td>
</tr>
<tr>
<td>AR(1)</td>
<td>0.0</td>
<td>1.0</td>
<td>0.5</td>
<td>0.0</td>
</tr>
<tr>
<td>ARIMA(1,1,1)</td>
<td>0.0</td>
<td>1.0</td>
<td>0.5</td>
<td>1.0</td>
</tr>
<tr>
<td>Logistic map</td>
<td>0.3333</td>
<td>0.1</td>
<td>0.0074</td>
<td>0.5</td>
</tr>
<tr>
<td>Lorenz attractor</td>
<td>0.0071</td>
<td>0.0038</td>
<td>0.0037</td>
<td>0.0101</td>
</tr>
</tbody>
</table>

Besides these measurements, we can also quantify the complexity embedded into time series by computing the Shannon’s entropy (ENTR) as presented in equation (13), in which $p(l) = \frac{P(l)}{N_l}$. The entropy reflects the RP complexity in terms of diagonal line structures, i.e., when states are not correlated, ENTR is low, what indicates lower complexity. Table 5 presents the ENTR for the five synthetic time series.

$$\text{ENTR} = -\sum_{l=1}^{N_l} p(l) \log p(l).$$  \hspace{1em} (13)

### Table 5 Entropy for the five synthetic time series

<table>
<thead>
<tr>
<th>Time series</th>
<th>$\varepsilon = 0.1$</th>
<th>$\varepsilon = 0.5$</th>
<th>$\varepsilon = 1.0$</th>
<th>$\varepsilon = \frac{\mu}{10}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random walk</td>
<td>0.0</td>
<td>0.0</td>
<td>0.585</td>
<td>1.684</td>
</tr>
<tr>
<td>AR(1)</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>ARIMA(1,1,1)</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>Logistic map</td>
<td>1.673</td>
<td>1.955</td>
<td>2.891</td>
<td>0.0</td>
</tr>
<tr>
<td>Lorenz attractor</td>
<td>2.538</td>
<td>2.481</td>
<td>2.498</td>
<td>1.771</td>
</tr>
</tbody>
</table>

Now consider the time series generated using the AR(1) process (Section 4). By analysing RQA measurements (Table 7), we confirm that it presents low recurrence rate (RR). Therefore, this series has low DET, high DIV and low complexity (ENTR). Furthermore, RP chart (Figure 9) also confirms the low frequency of diagonals. After those indices, this time series can also be classified as stochastic.

### Table 6 RQA measurements for the random walk series with time delay 20, embedding dimension 3 and $\varepsilon = 1.0$

<table>
<thead>
<tr>
<th>RR</th>
<th>$L_{\text{max}}$</th>
<th>DIV</th>
<th>$L_{\text{mean}}$</th>
<th>DET</th>
<th>ENTR</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.00115211</td>
<td>4</td>
<td>0.25</td>
<td>2.24</td>
<td>0.210526</td>
<td>0.585966</td>
</tr>
</tbody>
</table>

All previous measurements support the analysis of time series stochasticity. For example, time series generated by Random Walk and ARIMA(1,1,1) processes have low recurrence rate, which is indeed expected, assuming that they are derived from stochastic processes (see Table 1). On the other hand, the recurrence rate of the Lorenz attractor is low, which would indicate it is also stochastic; however, it is not, thus requiring additional measurements.

This relationship in between stochasticity and recurrence rate (RR) is confirmed when analysing other measurements such as for Random Walk series (summarised in Table 6). By verifying the values of all measurements, we conclude that this series has low DET, high DIV and low complexity (ENTR). Moreover, RP chart for the random walk (Figure 8) confirms the low frequency of diagonals. Given such scenario, we can classify it as stochastic.
After all previous evaluations, we conducted experiments considering other time series, generated by using deterministic processes. The first series was generated by using the Logistic map presented in Table 1. Based on RQA measurements (Table 9), we confirm that this series presents high recurrence rate (RR), high DET, low DIV and high complexity (ENTR). In the same manner, we analysed the RP chart (Figure 11) and confirmed the low frequency of diagonals, however having well-defined textures. We also observed that a uniform texture was formed, which characterises deterministic behaviour. Given such circumstances, we can classify the Logistic map series as deterministic.

Table 9  RQA measurements for the Logistic map with time delay 5, embedding dimension 3 and $\varepsilon = 1.0$

<table>
<thead>
<tr>
<th>RR</th>
<th>$L_{\text{max}}$</th>
<th>DIV</th>
<th>$L_{\text{mean}}$</th>
<th>DET</th>
<th>ENTR</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.857076</td>
<td>135</td>
<td>2.65431</td>
<td>7.9072</td>
<td>0.991169</td>
<td>1.388</td>
</tr>
</tbody>
</table>

Figure 11  RP for the logistic map

In the last experiment of this section, we considered the Lorenz attractor (Table 1). By analysing results (Table 10), we confirm this series presents low recurrence rate (RR), high DET, low DIV and high complexity (ENTR). By analysing the RP chart (Figure 12), we also confirm the high frequency of diagonals and well-defined textures. This is an interesting situation, in which the series is indeed deterministic however it presents low recurrence rate. This fact requires the analysis of all other measurements which indeed confirm it can be classified as deterministic.

Finally, we conclude that all measurements and analysis conducted using the RP support the characterisation of time series generation processes. Then, series with higher DET may be better modelled by using dynamical systems tools, while others are better represented by statistical ones.
4.2 Linearity evaluation

The study presented in previous section assists the selection in between deterministic and stochastic models. However, time series generation processes can still be classified in linear and non-linear, what also improves modelling. In linear series, observations are directly related to past ones linear and non-linear, what also improves modelling. In that sense, Lee et al. (1993) proposed a test (based on White, 1990), called WNN, to verify the linearity of processes. Observations of such series usually present high variations during collection, irreversibility, or even world systems, time series may be generated by non-linear system behaviour.

In several situations, especially when analysing real world systems, time series may be generated by non-linear processes. Observations of such series usually present high variations during collection, irreversibility, or even conditional variance (Morettin and Toloi, 2004). Non-linear series are usually represented by specific models or by applying filters to smooth and/or remove the non-linear component, transforming them into linear series (Morettin and Toloi, 2004).

In that sense, Lee et al. (1993) proposed a test (based on White, 1990), called WNN, to verify the linearity of time series by using an artificial neural network. The test considers a stochastic process $Z = \{Z_t, t = 1, \ldots, n\}$, $Z_t = (Y_t, X_t)$, in which $Y_t$ is a scalar and $X_t$ is a vector, and measures the conditional probability, i.e., the probability of $Y_t$ given $X_t$, denoted by $E(Y_t \mid X_t)$, to conclude about series linearity. Formally, this is represented by a regression function $g(x) = E(Y_t \mid X_t)$ which is found after training a multilayer artificial neural network. The objective of that test is to verify whether a WNN architecture is capable of mapping function $g(.)$. WNN is a feedforward artificial neural network with one hidden layer, such as presented in Figure 13.

<table>
<thead>
<tr>
<th>Table 10</th>
<th>RQA measurements for the Lorenz attractor with time delay 5, embedding dimension 3 and $\varepsilon = 1.0$</th>
</tr>
</thead>
<tbody>
<tr>
<td>RR: 0.00135383</td>
<td>$L_{\text{max}} = 266$</td>
</tr>
<tr>
<td>$L_{\text{mean}} = 7.45998$</td>
<td>DET = 0.95001</td>
</tr>
<tr>
<td>ENTR = 2.49831</td>
<td></td>
</tr>
</tbody>
</table>

In that neural network, input $x_i$, $i = 1, \ldots, n$ sends signals throughout connections which attenuate or amplify the original signal by a factor $\gamma_j$ (weights). An intermediate processing unit (or neuron) $j$ sums the received signals from input layer $\hat{x}^j$, where

$$\hat{x}(1, x_1, \ldots, x_n)^{\top}, \lambda_j \approx \{\lambda_{j0}, \lambda_{j1}, \ldots, \lambda_{jm}\}^{\top},$$

and produces an output $\psi(\hat{x}^j \gamma')$, in which activation function $\psi$ is a non-linear map $\mathbb{R}$ in $\mathbb{R}$. Activation $\psi$ is usually a cumulative distribution function (CDF) (in this paper, we considered the logistic non-linear function $\psi(\lambda) = (1 + e^{-\lambda})^{-1}$, $\lambda \in \mathbb{R}$). The output layer sums up received signals from the hidden layer, i.e., $\psi(\hat{x}^j \gamma')$, $j = 1, \ldots, m$, as presented in equation (14), where $\theta = (\beta, \gamma')$, $\beta = (\beta_0, \ldots, \beta_m)$ are the connection weights between the hidden and the output layers and $\gamma' = (\gamma_1, \ldots, \gamma_m)^{\top}$ represents the initial weights provided to the network.

$$f(x, \theta) = \beta_0 + \sum_{j=1}^{m} \beta_j \psi(\hat{x}^j) \quad m \in \mathbb{N}$$

(14)

For a given set of weights $\theta$, $Y_i = f(X_i, \theta) + [g(X_i - f(X_i, \theta)] + [Y_i - g(X_i)] = f(X_i, \theta) + \alpha(X_i, \theta) + \varepsilon_i = f(X_i, \theta) + \mu(X_i, \theta)$, where $\alpha(X_i, \theta) = g(X_i) - f(X_i, \theta)$ is the approximate error of the network output for $g(X_i)$. Term $\varepsilon_i = Y_i - g(X_i)$ consists in the random error, where $E(\varepsilon_i \mid X_i) = 0$ and, finally,
\[ \mu(X_t, \theta) = a(X_t, \theta) + \epsilon_t \] is the residual component, i.e., approximation and random errors.

WNN considers non-linear activation functions, therefore, when the network is capable of representing function \( g(.) \) (i.e., the approximation error is low), then a weight vector \( \theta \) is obtained such that \( g(X_t) = f(X_t, \theta) \) with probability 1, what indicates non-linear behaviour on the series. This means that after training the network, approximation errors tend to be reduced or removed. On the other hand, random errors (another residual component) may be present, what would indicate that the series presents some degree of linear behaviour.

Formally, we define a hypothesis test to verify the linearity of a series. The null hypothesis [equation (15)] assumes WNN was capable of representing linear approximation and random errors.

\[ H_0 : P[g(X_t) = f(X_t, \theta)] = 1 \text{ for some } \theta^* . \]  

\[ H_a : P[g(X_t) = f(X_t, \theta)] < 1 \text{ for all } \theta . \] 

WNN presents good results when \( \sum_{j=1}^{m} \beta_j \psi(\tilde{x}_{y_j}) \) is capable of extracting values with Chi-square error \( \epsilon^*_t = Y_t - \tilde{X}^* \theta^* \), where \( \theta^* \) is a vector of linear approximation parameters for \( E(y_t | X_t) \), given by \( \theta^* = (\beta_1^*, \gamma_1^*) \). Stinchcombe and White (1989) demonstrate that assuming \( \psi(x) \) as a logistic CDF, term \( \psi(\tilde{x}_{y_j}) \) is capable of extracting error values which can subsequently be applied in the hypothesis test.

By knowing WNN approximates non-linear functions, this network can be used to verify time series linearly. In order to demonstrate the WNN test, consider the five synthetic series presented in Table 1. For each series, the Chi-square test is applied to verify whether the WNN output satisfies the alternative hypothesis, \( H_a \), i.e., the series is linear. In this case, we considered the significance level \( \alpha = 0.05 \) or 5%. Thus, we have the following hypothesis:

\[ H_0 : \rho \text{-value} \leq 0.05 \text{ accepts } \rho \]  

\[ H_a : \rho \text{-value} > 0.05 \text{ not accepts } \rho \] 

Table 11 summarises results of the WNN test for the five synthetic series. Table rows present the \( \rho \)-value for WNN, the null hypothesis and the final classification (linear or non-linear). Results confirm what was indeed expected when considering such synthetic series.

### Table 11  WNN tests to verify linearity in synthetic time series

<table>
<thead>
<tr>
<th>Time series</th>
<th>( \rho )-value (WNN)</th>
<th>( H_0 )</th>
<th>Linearity</th>
</tr>
</thead>
<tbody>
<tr>
<td>AR(1)</td>
<td>0.8</td>
<td>Reject</td>
<td>Linear</td>
</tr>
<tr>
<td>ARIMA(1,1,1)</td>
<td>0.3</td>
<td>Reject</td>
<td>Linear</td>
</tr>
<tr>
<td>Random walk</td>
<td>0.5</td>
<td>Reject</td>
<td>Linear</td>
</tr>
<tr>
<td>Logistic map</td>
<td>&lt; 0.0001</td>
<td>Accept</td>
<td>Non-linear</td>
</tr>
<tr>
<td>Lorenz attractor</td>
<td>&lt; 0.0003</td>
<td>Accept</td>
<td>Non-linear</td>
</tr>
</tbody>
</table>

### 4.3 Stationarity evaluation

Besides previous tests, we carried on evaluating another important factor to model stochastic time series which is the stationarity. Stochastic stationary time series are characterised by the statistical equilibrium around average \( \mu \) and variance \( \sigma^2 \). There are two categories of stationary series: weakly stationary or second order and strictly stationary. A second order time series has average and variance constants. Besides such constants, strictly stationary series also present a constant autocovariance structure. If a series presents stationarity of covariance, such behaviour is repeated over time, i.e., autocovariance remains the same regardless of the temporal reference point.

Given such circumstances, the autocovariance depends only on the difference in between states, i.e., \( |x_{t_2} - x_{t_1}| \) and not on absolute values \( x_t \). Therefore, if a series is stationary, the magnitude of autocorrelation reduces quickly, however, when the series is non-stationary, the autocorrelation decreases gradually over time (Yu et al., 1998). Consider a discrete time series with observations \( x_1, x_2, ..., x_n \), and the pair of points \((x_1, x_2), (x_2, x_3), \ldots, (x_{n-1}, x_n)\) which can be represented as \((a - k)\) observations of the probability distribution function (PDF) \( f_{x_1}(x_1, x_2) \) similar in all \( k \Delta (k \in [1, n - k]) \) future instants. In general, an equilibrium condition implies that properties of the multivariate PDF associated with any set \( x_n, x_2, ..., x_n \) depends only on differences \( |x_{t_i} - x_{t_j}| \), defined in equation (17). For all sets of time series and for all \( k \), a stochastic process that satisfies the conditions of equation (17) is considered strictly stationary.

\[ F_{X_{t_1}...X_{t_n}}(x_1, x_2, ..., x_n) = F_{X_{t_1}...X_{t_n}}(x_1, x_2, ..., x_n) \] (17)

Two different points of view can be applied to verify the stationarity of time series: by considering dynamical systems or statistical tools. The dynamical system area attempts to identify the stationarity by applying techniques to evaluate trajectories and information complexity such as the Lyapunov exponent, correlations and entropies (e.g., Kolmogorov). The statistics area attempts to evaluate whether the probability structure of a series can be affected after temporal displacement.

One of the main dynamical system tools is the STP (Provanzale et al., 1992) which is a correlation test to evaluate whether series have geometric trajectories (called global) or fractal structures (called local). From a statistical point of view, we can consider the autocorrelation function to measure the degree of correlation in time series, i.e., we compute the separation distance of an observation at time instant \( t \) against another \( t + k \), where \( k \) is the time lag. Formally, the Autocorrelation Function, \( ACF(k) \) (Box et al., 1994), of a time series \( X \) with average \( \mu \) is defined in equation (18), in which \( E[.] \) is the expected average value of the expression and \( \sigma^2 \) is the variance of \( X \).

\[ ACF(k) = \frac{E[(X_t - \mu)(X_{t+k} - \mu)]}{\sigma^2} \] (18)
STP explicitly considers the time separation of states, i.e., dispersion (distance) between states versus their separation over time. It generates contours that determine the ranges of correlation for a given embedded dimension in a series. This correlation represents the probability that a pair of randomly selected states is lower than a distance \( r \) in phase space. This distance and the separation dimension reflect the position in neighbourhood at a given time instant.

In particular, every pair of states separated over a given time \( \Delta t \) is also separated in phase space by a distance \( ACF_p(\Delta t) \) and probability \( p \). Thus, contours map a fraction of the closest points (in terms of a distance \( r \)) to a given separation \( \Delta t \), which is equivalent to \( P(|x(t + \Delta t) - x(t)| < r) \) for an arbitrary \( t \).

In order to demonstrate the previous concepts on stationarity test, we apply STP and ACF on the five synthetic time series we considered (Section 4). We observe the autocorrelation function was evaluated under a set of different time lags. Analysing Figure 14(a), which presents the ACF from a random walk time series, we notice values slowly decay to zero. This behaviour is common in series having a well-defined trend or when there are dependencies among observations, which may change the average and variance over time, e.g., \( E(X_t) = t \cdot \mu \) and \( \text{var}(X_t) = t \cdot \sigma^2 \) (Box et al., 1994), respectively. Therefore, according to this assumption, we can classify this time series as non-stationary.

**Figure 14** STP test and autocorrelation function for random walk, (a) autocorrelation function (b) STP (see online version for colours)

Besides ACF, we can still apply STP in order to evaluate stationarity. Figure 14(b) presents the STP for the random walk series, in which the x-axis is the time separation and the y-axis represents the space separation. Every contour corresponds to a probability \( p \) which reveals time series periodicity. In Figure 14(b), we observe low correlation over time and space separations in interval \([0, 58]\). Considering any state over time, this figure presents high values on space, for any probability \( p \), what characterises a non-stationary time series.

Figure 15(a) presents the ACF for the AR(1) time series. We observe values orbit around 0, demonstrating low perturbation over time, which indicates this series is composed of random variables. Figure 15(b) presents the STP for the same series, in which the time separation has high correlation and space separation ranges mostly in \([0, 3.8]\). This time series has low values for any state and probability over time, what characterises a stationary time series.

**Figure 15** STP test and autocorrelation function for AR(1), (a) autocorrelation function (b) STP (see online version for colours)

**Figure 16** STP test and autocorrelation function for ARIMA(1,1,1), (a) autocorrelation function (b) STP (see online version for colours)
Figure 17 STP test and autocorrelation function for the logistic map, (a) autocorrelation function (b) STP (see online version for colours)

![Figure 17](image)

Figure 18 STP test and autocorrelation function for the Lorenz attractor, (a) autocorrelation function (b) STP (see online version for colours)

![Figure 18](image)

Figure 16(a) presents the ACF for the ARIMA(1,1,1) series, in which autocorrelations are different from zero, demonstrating abrupt changes over time. According to this ACF, we observe this process behaves as the Random Walk one, i.e., ACF values take considerable time to decay to zero and, furthermore, vary around it, indicating the average and variance depend on time. Therefore, we can classify this series as non-stationary. Figure 16(b) shows the STP to the same series, in which the space separation is in [0, 19]. Taking any value over time, we obtain high values in space, what characterises a non-stationary series.

Figure 17(a) presents the ACF for the logistic map, whose autocorrelations quickly decay to zero, indicating that it is stationary. Figure 17(b) shows the STP for the same series, in which the space separation is in [0, 0.95]. For any value over the time axis, we obtain low values in space, what depicts a stationary time series.

Figure 18(a) presents the ACF for the Lorenz attractor, whose autocorrelations are significantly different from zero. According to it, we observe this series behaves similarly to the random walk and ARIMA(1,1,1) ones, whose ACF values also take considerable time to decay to zero, indicating that it is non-stationary. Figure 18(b) shows the STP for the same series, in which the space separation is in [0, 80]. For any value we take over time, a high value is obtained in space, confirming it is non-stationary.

5 Experiments

After confirming the efficiency of our approach in classifying synthetic time series, we performed a study on real world datasets. As we were already conducting studies on computer data storages, we decided to consider the datasets provided by Storage Networking Industry Association (SNIA), which are representative of a large number of small to medium-size enterprise data centres (Narayanan et al., 2008). Those datasets were generated by standard data centre servers (enterprise servers at Microsoft Research at Cambridge) during one week of observation. Such data centre has 13 servers, 36 logical volumes and 179 hard disks. Table 12 presents the monitored servers.

<table>
<thead>
<tr>
<th>Server</th>
<th>Function</th>
<th># volumes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Usr</td>
<td>User home directories</td>
<td>3</td>
</tr>
<tr>
<td>proj</td>
<td>Project directories</td>
<td>5</td>
</tr>
<tr>
<td>prn</td>
<td>Print Server</td>
<td>2</td>
</tr>
<tr>
<td>hm</td>
<td>Hardware monitoring</td>
<td>2</td>
</tr>
<tr>
<td>rsrch</td>
<td>Research projects</td>
<td>3</td>
</tr>
<tr>
<td>proxy</td>
<td>Firewall/web proxy</td>
<td>2</td>
</tr>
<tr>
<td>src1</td>
<td>Source control</td>
<td>3</td>
</tr>
<tr>
<td>src2</td>
<td>Source control</td>
<td>3</td>
</tr>
<tr>
<td>stg</td>
<td>Web stage</td>
<td>2</td>
</tr>
<tr>
<td>ts</td>
<td>Terminal server</td>
<td>1</td>
</tr>
<tr>
<td>web</td>
<td>Web/SQL server</td>
<td>4</td>
</tr>
<tr>
<td>mds</td>
<td>Media server</td>
<td>2</td>
</tr>
<tr>
<td>wdev</td>
<td>Test web server</td>
<td>4</td>
</tr>
</tbody>
</table>

Table 12 Data centre: 13 servers, 36 logical volumes and 179 hard disks were monitored

Monitoring was conducted on every logical volume and all block-level read and write operations were captured. Traces were collected using the Event Tracing for Windows (ETW) (Narayanan et al., 2008) during a period of 168 hours (1 week). Each trace event describes one I/O request which includes timestamps, disk number, initial logical block number, number of blocks transferred and type of operation (read or write). According to our purpose, we filtered those events and only considered timestamps, number of blocks
transferred and type of operations. For the sake of simplicity, we considered three traces (logical volumes) in our experiments (boldface rows in Table 12): the proj volume with write-only operations, the hm volume with read-only and the mds volume with read-only operations.

In summary, our approach aims at classifying these three traces according to their stochasticity, linearity and stationarity in order to select most adequate models to represent them. Thus, we evaluated three issues following the tests described in Section 4. RP, WNN and STP were considered to evaluate the stochasticity, the linearity and the stationarity, respectively.

The first trace considered was proj, which contains write-only operations. RQA measurements for that trace are presented in Table 13. By analysing results, we observe this series presents low recurrence rate and that diagonals confirm it presents low divergence, complexity and determinism. Furthermore, when analysing the RP chart (Figure 19), we confirm the low frequency of diagonals. Given such circumstances, proj can be classified as a stochastic time series.

Table 13  RQA measurements for proj with time delay 6, embedding dimension 14 and $\varepsilon = 1.0$

<table>
<thead>
<tr>
<th>RR: 0.3749</th>
<th>$L_{\text{max}} = 21$</th>
<th>DIV = 0.04762</th>
</tr>
</thead>
<tbody>
<tr>
<td>$L_{\text{mean}} = 2.628$</td>
<td>DET = 0.6282</td>
<td></td>
</tr>
<tr>
<td>ENTR = 1.082</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 19  RP for proj

Table 14 presents RQA measurements for hm, which considers read-only operations. Based on results (Table 14), hm presents low recurrence rate. Diagonal measurements confirm this series presents moderate divergence, low complexity and DET. Moreover, by analysing the RP chart (Figure 20), we still confirm the low frequency of diagonals. Considering such analysis, hm can be classified as stochastic.

Table 14  RQA measurements for hm with time delay 4, embedding dimension 2 and $\varepsilon = 1.0$

<table>
<thead>
<tr>
<th>RR: 0.19511</th>
<th>$L_{\text{max}} = 3.1275$</th>
<th>DIV = 26</th>
</tr>
</thead>
<tbody>
<tr>
<td>$L_{\text{mean}} = 1.39192$</td>
<td>DET = 0.610596</td>
<td></td>
</tr>
<tr>
<td>ENTR = 0.04</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 20  RP for hm

Table 15 presents RQA measurements for mds, which considers read-only operations. By analysing results (Table 15), we conclude this series presents a very high recurrence rate and the formation of diagonals. Those diagonals confirm this series presents low divergence, moderate complexity and high DET. Furthermore, when analysing the RP chart (Figure 21), we also confirm the high frequency of diagonals. Given such circumstances, this series is classified as deterministic.

Then, we applied the WNN test (Section 4.2) to evaluate time series linearity. Table 16 summarises the WNN test and indicates the classification of the series under study. In this situation, two series are linear and one is not. However, even after applying the linearity test, we still need to identify the degrees of stationarity for each series. Thus, the STP test (Section 4.2) is applied.

Table 15  RQA measurements for mds with time delay 5, embedding dimension 2 and $\varepsilon = 1.0$

<table>
<thead>
<tr>
<th>RR: 0.9088</th>
<th>$L_{\text{max}} = 525$</th>
<th>DIV = 0.001905</th>
</tr>
</thead>
<tbody>
<tr>
<td>$L_{\text{mean}} = 53.34$</td>
<td>DET = 0.9956</td>
<td></td>
</tr>
<tr>
<td>ENTR = 3.275</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Time series proj presents the ACF and STP charts presented in Figure 22. ACF demonstrates proj has a constant autocovariance structure which is maintained over time. STP results confirm this autocovariance structure, whose distance (y-axis) values attenuate slowly over time (x-axis). Every curve in STP chart represents a probability $p$ such as
defined in Section 4.2. By analysing results, we conclude \textit{proj} is stationary.

<table>
<thead>
<tr>
<th>Time series</th>
<th>(p)-value (WNN)</th>
<th>(H_0)</th>
<th>Linearity</th>
</tr>
</thead>
<tbody>
<tr>
<td>\textit{proj}</td>
<td>0.05205</td>
<td>Reject</td>
<td>Linear</td>
</tr>
<tr>
<td>\textit{hm}</td>
<td>0.5637</td>
<td>Reject</td>
<td>Linear</td>
</tr>
<tr>
<td>\textit{mds}</td>
<td>9.498e – 07</td>
<td>Accept</td>
<td>Non-linear</td>
</tr>
</tbody>
</table>

Time series \textit{hm} has the ACF and STP charts presented in Figure 23. The ACF values slowly decay at the beginning and present abrupt changes which are maintained over time, i.e., this does not present a constant autocovariance structure. STP results confirm high values of distance over time. Based on such analysis, we consider \textit{hm} as a non-stationary time series.

Time series \textit{mds} has the ACF and STP charts presented in Figure 24. The ACF values slowly decay at the beginning, which means that this series does not present a constant autocovariance structure maintained over time. STP results confirm this autocovariance structure in which present abrupt changes. By analysing results, we conclude \textit{mds} is non-stationary.

### 6 Evaluation

The process of modelling time series commonly involves two steps:

1. firstly, we need to understand the series generation process
2. then, we can select an adequate technique to model it.

Section 5 presents results, obtained from the application of our approach, to support the classification of time series generation processes from real world data. Now, we consider such classification to model series and evaluate prediction accuracy.
In order to validate the classification obtained in last section, we divided every time series in two parts:
1. the first with 80% of observations to adjust the model, that can also be referenced as training
2. the second, with 20%, to conduct predictions and test models.

The measurement considered to evaluate prediction results was the normalised root mean squared error (NRMSE), which is defined in equation (20), in which $x_{\text{max}}$ is the maximum and $x_{\text{min}}$ is the minimum observed values. NRMSE equation depends on equation (19), which presents the root mean squared error (RMSE) (Andersson and Woessner 1992), where $x_i$ is the expected value at instant $i$, $\hat{x}_i$ is the obtained value at time instant $i$ and $n$ is the number of observations predicted. Moreover, hypothesis tests were performed over all prediction results, obtained from NRMSE, considering the critical value of the statistic $z = 1.96$ for the significance level $\alpha = 0.05$.

\[
\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (x_i - \hat{x}_i)^2}
\]

\[
\text{NRMSE} = \frac{\text{RMSE}}{x_{\text{max}} - x_{\text{min}}}
\]

We considered the following modelling tools to make predictions: AR, autoregressive moving average model (ARMA), ARIMA model (Box et al., 1994), Polynom (Hegger et al., 1998) and RBF model (Hegger et al., 1998). The first three approach the problem in a statistical point of view, while the last two in a deterministic form. In this manner, according to the generation process classification (Section 4), we expect to select the most adequate modelling tool.

Time series $\text{proj}$ was the first to be studied. According to the tests presented in Section 5, it was classified as stochastic, linear and stationary. By considering such classification, this series (Figure 1) would be better modelled by using statistical tools, such as AR and ARMA models (Box et al., 1994). Results considering such models are presented in Figure 25, which confirm our hypothesis. In that sense, ARMA(1,2) was capable of modelling $\text{proj}$ with the smallest error among all evaluated approaches, indicating good prediction.

A hypothesis test was performed to confirm whether ARMA(1,2) is statistically better than others, i.e., $\mu(\text{NRMSE}_{\text{ARMA(1,2)}}) < \mu(\text{NRMSE}_{\text{AR(1)}})$, $\mu(\text{NRMSE}_{\text{ARMA(1,2)}}) < \mu(\text{NRMSE}_{\text{Polynom}})$, and $\mu(\text{NRMSE}_{\text{ARMA(1,2)}}) < \mu(\text{NRMSE}_{\text{RBF}})$.

In order to compare ARMA(1,2) to AR(1), we defined the following hypothesis:

\[
\left\{ \begin{array}{l}
H_0 : \mu(\text{NRMSE}_{\text{ARMA(1,2)}}) = \mu(\text{NRMSE}_{\text{AR(1)}}) \\
H_a : \mu(\text{NRMSE}_{\text{ARMA(1,2)}}) < \mu(\text{NRMSE}_{\text{AR(1)}})
\end{array} \right.
\]

Table 17 presents the average ($\mu$) and standard deviation ($\sigma$), in terms of NRMSE, for AR(1) and ARMA(1,2). We consider the statistic $z$ due to samples are independent.

**Figure 25** Time series $\text{proj}$: NRMSE

<table>
<thead>
<tr>
<th></th>
<th>ARMA(1,2)</th>
<th>AR(1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\mu$</td>
<td>0.2921915</td>
<td>0.3308134</td>
</tr>
<tr>
<td>$\sigma$</td>
<td>0.04644697</td>
<td>0.02452782</td>
</tr>
</tbody>
</table>

Replacing the obtained values (Table 17) in equation (21), we have:

\[
z = \frac{\mu(\text{NRMSE}_{\text{ARMA(1,2)}}) - \mu(\text{NRMSE}_{\text{AR(1)}})}{\sqrt{\frac{\sigma^2_{\text{ARMA(1,2)}}}{n_{\text{ARMA(1,2)}}} + \frac{\sigma^2_{\text{AR(1)}}}{n_{\text{AR(1)}}}}}
\]

\[
z = \frac{0.2921915 - 0.3308134}{\sqrt{\frac{0.04644697^2}{200} + \frac{0.02452782^2}{200}}} = -10.39867
\]

Using a two-tail test we have $p$-value $< 2.2 e - 16$, thus, we can reject $H_0$ and, conclude that there is enough evidence that the NRMSE is reduced when using ARMA(1,2) instead of AR(1).

In order to compare ARMA(1,2) to Polynom, we defined the following hypotheses:

\[
\left\{ \begin{array}{l}
H_0 : \mu(\text{NRMSE}_{\text{ARMA(1,2)}}) = \mu(\text{NRMSE}_{\text{Polynom}}) \\
H_a : \mu(\text{NRMSE}_{\text{ARMA(1,2)}}) < \mu(\text{NRMSE}_{\text{Polynom}})
\end{array} \right.
\]

Table 18 presents the NRMSE average ($\mu$) and standard deviation ($\sigma$) for ARMA(1,2) and Polynom.

**Table 17** NRMSE: $\mu$ and $\sigma$ for ARMA(1,2) and AR(1)

<table>
<thead>
<tr>
<th></th>
<th>ARMA(1,2)</th>
<th>AR(1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\mu$</td>
<td>0.2921915</td>
<td>0.3308134</td>
</tr>
<tr>
<td>$\sigma$</td>
<td>0.04644697</td>
<td>0.02452782</td>
</tr>
</tbody>
</table>

**Table 18** NRMSE: $\mu$ and $\sigma$ for ARMA(1,2) and Polynom

<table>
<thead>
<tr>
<th></th>
<th>ARMA(1,2)</th>
<th>Polynom</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\mu$</td>
<td>0.2921915</td>
<td>0.404553</td>
</tr>
<tr>
<td>$\sigma$</td>
<td>0.04644697</td>
<td>0.03574264</td>
</tr>
</tbody>
</table>
Replacing the obtained statistics (Table 18) in equation (21), we have:

\[
    z = \frac{0.2921915 - 0.404535}{\sqrt{0.04644967^2 + 0.03574264^2}} = -27.10868
\]

Using the two-tail test we have \( \rho \)-value < 2.2e - 16, thus, we can reject H0 and, conclude that there is enough evidence that ARMA(1,2) presents lower NRMSE than Polynom.

In order to compare ARMA(1,2) to RBF, we defined the following hypotheses:

\[
    \begin{align*}
    H_0 & : \mu(\text{NRMSE}_{\text{ARMA(1,2)}}) = \mu(\text{NRMSE}_{\text{RBF}}) \\
    H_u & : \mu(\text{NRMSE}_{\text{ARMA(1,2)}}) < \mu(\text{NRMSE}_{\text{RBF}})
    \end{align*}
\]

Table 19 presents the NRMSE average (\( \mu \)) and standard deviation (\( \sigma \)) for ARMA(1,2) and RBF.

| NRMSE: \( \mu \) and \( \sigma \) for ARMA(1,2) and RBF |
|-------------|-------------|
| ARMA(1,2)   | RBF         |
| \( \mu \)   | 0.2921915   | 0.4117446  |
| \( \sigma \) | 0.04644697  | 0.1409526  |

Replacing the obtained results (Table 19) in equation (21), we have:

\[
    z = \frac{0.5636638 - 0.7270487}{\sqrt{0.04644967^2 + 0.1409526^2}} = -11.36674
\]

Conducting the two-tail test, we obtained \( \rho \)-value < 2.2e - 16, thus, we can reject H0 and, conclude that there is sufficient evidence that ARMA(1,2) produces lower errors when compared to RBF.

We finally confirm that our classification approach supported the correct decision in selecting the adequate modelling technique for the time series \( \text{proj} \).

Now, consider time series \( \text{hm} \) which is, according to our approach, classified as stochastic, linear and non-stationary. According to our proposed classification (see Figure 1), this series would be better modelled by statistical tools, such as ARIMA, which was confirmed by the results presented in Figure 26. ARIMA(2,1,2) was indeed capable of modelling \( \text{hm} \) at lower errors.

In addition, we conducted a statistical test over prediction results. A hypothesis test was performed in order to confirm if ARIMA(2,1,2) is statistically better than others, i.e., \( \mu(\text{NRMSE}_{\text{ARIMA(2,1,2)}}) < \mu(\text{NRMSE}_{\text{AR(1)}}) \) and \( \mu(\text{NRMSE}_{\text{ARIMA(2,1,2)}}) < \mu(\text{NRMSE}_{\text{Polynom}}) \). We did not compare ARIMA(2,1,2) against RBF due to the latter did not generate good prediction results even varying the number of centroids.

In order to compare ARIMA(2,1,2) against AR(1), we defined the following hypotheses:

\[
    \begin{align*}
    H_0 & : \mu(\text{NRMSE}_{\text{ARIMA(2,1,2)}}) = \mu(\text{NRMSE}_{\text{AR(1)}}) \\
    H_u & : \mu(\text{NRMSE}_{\text{ARIMA(2,1,2)}}) < \mu(\text{NRMSE}_{\text{AR(1)}})
    \end{align*}
\]

Table 20 presents the NRMSE average (\( \mu \)) and standard deviation (\( \sigma \)) for ARIMA(2,1,2) and AR(1).

| NRMSE: \( \mu \) and \( \sigma \) for ARIMA(2,1,2) and AR(1) |
|-----------------|-----------------|
| ARIMA(2,1,2)    | AR(1)           |
| \( \mu \)   | 0.5636638   | 0.7270487 |
| \( \sigma \) | 0.2929577 | 2.399923 |

In order to compare ARIMA(2,1,2) to Polynom, we can defined the following hypotheses:

\[
    \begin{align*}
    H_0 & : \mu(\text{NRMSE}_{\text{ARIMA(2,1,2)}}) = \mu(\text{NRMSE}_{\text{Polynom}}) \\
    H_u & : \mu(\text{NRMSE}_{\text{ARIMA(2,1,2)}}) < \mu(\text{NRMSE}_{\text{Polynom}})
    \end{align*}
\]

Table 21 presents the NRMSE average (\( \mu \)) and standard deviation (\( \sigma \)) for ARIMA(2,1,2) and Polynom.

| NRMSE: \( \mu \) and \( \sigma \) for ARIMA(2,1,2) and Polynom |
|-----------------|-----------------|
| ARIMA(2,1,2)    | Polynom         |
| \( \mu \)   | 0.5636638 | 0.7864895 |
| \( \sigma \) | 0.2929577 | 01344380 |

Replacing the obtained results (Table 21) in equation (21), we have:
Then, we conducted a two-tail test and obtained $\rho$-value $< 2.2e - 16$, thus, we can reject $H_0$ and, conclude that there is enough evidence that ARIMA(2,1,2) produces a better model than Polynom.

After all these evaluations, we confirm that our approach correctly classified the generation process for $hm$ therefore improving model accuracy.

Finally, we consider time series $mds$ which is, according to our approach, classified as deterministic, non-linear and non-stationary. Following the proposed classification (see Figure 1), this series is better modelled by using chaos-theory or dynamical system tools. Figure 27 confirms our hypothesis, in which this series is better modelled by using chaos-theory tools (unfolding the behaviour in phase space and modelling using RBF). The Phase Space/RBF approach was capable of modelling $mds$ at lower errors, thus, obtaining good prediction results.

In addition, we conducted statistical tests over prediction results. Hypothesis tests were performed to confirm whether the Phase Space/RBF approach is statistically better than others, i.e., $\mu(\text{NRMSE}_{\text{Phase Space/RBF}}) < \mu(\text{NRMSE}_{\text{AR(1)}})$ and $\mu(\text{NRMSE}_{\text{Phase Space/RBF}}) < \mu(\text{NRMSE}_{\text{Polynom}})$ and $\mu(\text{NRMSE}_{\text{Phase Space/RBF}}) < \mu(\text{NRMSE}_{\text{ARMA(5,5)}})$.

In order to compare Phase Space/RBF to AR(1), we defined the following hypotheses:

$$H_0 : \mu(\text{NRMSE}_{\text{Phase Space/RBF}}) = \mu(\text{NRMSE}_{\text{AR(1)}})$$
$$H_a : \mu(\text{NRMSE}_{\text{Phase Space/RBF}}) < \mu(\text{NRMSE}_{\text{AR(1)}})$$

Table 22 presents the NRMSE average ($\mu$) and standard deviation ($\sigma$) for Phase Space/RBF and AR(1).

Replacing the obtained results (Table 22) in equation (21), we have:

$$z = \frac{0.2979402 - 0.6365301}{\sqrt{0.7498916^2 + 1.175815^2}} = -11.63121$$

By using a two-tail test, we obtained $\rho$-value $< 2.282e - 06$, thus, we can reject $H_0$ and, conclude that there is enough evidence that the phase space/RBF is better than AR(1).

In order to compare phase space/RBF to Polynom, we defined following hypotheses:

$$H_0 : \mu(\text{NRMSE}_{\text{Phase Space/RBF}}) = \mu(\text{NRMSE}_{\text{Polynom}})$$
$$H_a : \mu(\text{NRMSE}_{\text{Phase Space/RBF}}) < \mu(\text{NRMSE}_{\text{Polynom}})$$

Table 23 presents the NRMSE average ($\mu$) and standard deviation ($\sigma$) for the Phase Space/RBF and Polynom.

Replacing the obtained results (Table 23) in equation (21), we have:

$$z = \frac{0.2979402 - 0.8305897}{\sqrt{0.6365301^2 + 0.1102992^2}} = -11.63121$$

Using a two-tail test, we obtained $\rho$-value $< 2.2e - 16$, thus, we can reject $H_0$ and, conclude that there is enough evidence that the Phase Space/RBF produces lower prediction errors than Polynom.

In order to compare the Phase Space/RBF against ARMA(5,5), we defined the following hypotheses:

$$H_0 : \mu(\text{NRMSE}_{\text{Phase Space/RBF}}) = \mu(\text{NRMSE}_{\text{ARMA(5,5)}})$$
$$H_a : \mu(\text{NRMSE}_{\text{Phase Space/RBF}}) < \mu(\text{NRMSE}_{\text{ARMA(5,5)}})$$

Table 24 presents NRMSE average ($\mu$) and standard deviation ($\sigma$) for the Phase Space/RBF and ARMA(5,5).

Replacing the obtained results (Table 24) in equation (21), we have:
The two-tail test was conducted and we obtained ρ-value = 3.130 – 14, thus, we can reject $H_0$ and, conclude there is enough evidence that the Phase Space/RBF approach is better than ARMA(5,5), what again confirms our classification approach.

7 Concluding remarks

Time series can be used to organise and represent data as they are generated over time. By modelling those series, we can understand the behaviour of real world systems and, therefore, predict observations. Such prediction supports decisions beforehand what is relevant in different domains. However, modelling is related to specific aspects of the time series such as stochasticity, linearity and stationarity. Moreover, it is difficult to perceive such aspects, thus, this analysis usually requires specialists. In our point of view, the major problem with this approach is the subjectivity imposed in selecting an appropriate time series model.

In order to avoid this drawback, some studies apply several modelling techniques on time series and select the one with best results at all. Results, in general, are analysed through residue analysis, i.e., the difference in between the one with best results at all. Results, in general, are analysed through residue analysis, i.e., the difference in between the predicted value and the expected one.

The main problem of these approaches is that the specialist may be available or there is enough time or resources to assess many modelling techniques. These drawbacks motivated this paper which presents a survey on techniques to classify time series generation processes and a novel automatic and systematic approach to support such task. After such classification, the proposed approach is used to select modelling techniques. By classifying a time series, the modelling process becomes more effective because less representative models are not regarded, moreover, this approach also minimises human intervention. In that sense, this approach is helpful to end-users and also to specialists who need support to select modelling techniques.

The effectiveness of our approach was confirmed through experiments using synthetic and real world datasets. Firstly, we evaluated a set of time series generated using well-known mathematical and statistical tools. Then, we evaluated our approach to select modelling techniques for real world data. By conducting hypotheses tests, we confirmed this new approach selects the most appropriate set of modelling techniques.

Acknowledgements

This paper is based upon work supported by CNPq-Universal (National Counsel of Technological and Scientific Development), under grant no. 470739/2008-8, FUNDECT (Foundation for Support to Education, Science and Technology Development of the Mato Grosso do Sul State) no. 23/200.402/2008, CAPES (Coordination for the Improvement of Higher Level – or Education – Personnel) and FAPESP (São Paulo Research Foundation), Brazil. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of CNPq, FUNDECT, CAPES and FAPESP.

References


Notes

1. Further evaluations are only necessary when the generation process is not fully or at least satisfactorily understood considering a given domain.

2. Let $f$ be a real-valued and continuous function in $[a, b]$ and for every $\epsilon > 0$, there is a polynomial $p$ in $[a, b]$ such that, $|f(x) - P(x)| < \epsilon$, for every $x \in [a, b]$. This is, any continuous function in a closed and limited interval can be uniformly approximated in that interval by using polynomials at any precision degree (Jeffreys and Jeffreys, 1988).

3. $d(i, j) = \{1 \mid i \neq j, 0 \mid i = j\}$

4. Real-non-negative function defined on a vector space which satisfies the triangle inequality.

5. It is relevant to mention that we assume a time series is deterministic only if $\text{DET} \geq 95\%$, otherwise we conclude it tends to be stochastic.

6. We executed experiments using PolyNeom and RBF versions made available in TISEAN 3.0.1 – http://www.mpipks-dresden.mpg.de/tisean/Tisean 3.0.1

7. ARMA(1,2) was executed using the auto.arima tool provided by R statistical software (Hyndman and Khandakar, 2008), which estimates the order of the Arima model.