Chapter 2
Incomplete Time Series: Imputation through Genetic Algorithms

Juan Carlos Figueroa-García, Dusko Kalenatic, and César Amilcar López

Abstract. Uncertainty in time series can appear in many ways, and its analysis can be performed based on different theories. An important problem appears when time series is incomplete since the analyst should impute those observations before any other analysis.

This chapter focuses on designing an imputation method for multiple missing observations in time series through the use of a genetic algorithm (GA), which is designed for replacing these missed observations in the original series. The flexibility of a GA is used for finding an adequate solution to a multicriteria objective, defined as the error between some key properties of the original series and the imputed one. A comparative study between a classical estimation method and our proposal is presented through an example.

1 Introduction and Motivation

The analysis of time series includes the handling of nonlinear behavior, heteroscedasticity and incomplete series. Data loss is an important problem for univariate time series analysis since most of the available estimation methods require either complete information or covariates to estimate missing observations. Moreover, when the series has a large number of missing observations or there is a subset of missing observations in a row, then classical estimation methods cannot produce a reasonable solution, so the use of GAs arises as an alternative for problems involving multiple missing data.

Juan Carlos Figueroa-García · Cesar Amilcar López-Bello
Universidad Distrital Francisco José de Caldas,
Bogotá - Colombia
e-mail: {jcfigueroag,clopezb}@udistrital.edu.co

Dusko Kalenatic
Universidad de La Sabana, Chía - Colombia
e-mail: duskokalenatic@yahoo.com

W. Pedrycz & S.-M. Chen (Eds.): Time Series Analysis, Model. & Applications, ISRL 47, pp. 31–52.
DOI: 10.1007/978-3-642-33439-9_2

Thus, the scope of this chapter is to present an evolutionary algorithm for imputing all missing observations of an incomplete time series. The main focus is to preserve some key properties of available data after imputation.

Nowadays, evolutionary algorithms are efficient computational intelligence tools which provide fast and efficient exploration of the search space of complex problems. To do so, a multi-criteria fitness function derived from the autocorrelation function, mean and variance of the series, is minimized.

This chapter is divided into six sections, Section 1 presents the Introduction and Motivation; in Section 2 some useful statistical measures for time series analysis are introduced; in Section 3 the proposed genetic algorithm is described and its methodological issues are presented; in Section 4, we apply the genetic algorithm to a weather prediction case; Section 5 presents a statistical analysis to verify the obtained results; and finally in Section 6, some concluding remarks of the proposal are presented.

1.1 A Review

The missing data problem is mainly presented in financial and biological time series. In fact, it is an uncontrollable phenomenon which conduces to get biased results on posterior analysis such as identification and prediction.

There exist some methods to impute missing data, some of them based on optimal estimators, as the *EM Algorithm* proposed by Dempster [16] and Gaetana & Yao [22], and its modifications. Other approaches are based on averages, expected values or simple prediction structures, and some advanced methods are based on both covariates and additional information of the series, which leads to new directions to estimate those missed observations. For further information see González, M. Rueda & A. Arcos [24], Qin, Zhang, Zhu, Zhang & Zhang [41] Ibrahim & Molenberghs [30], Tsiatis [46], Chambers & Skinner [15] and Hair, Black, Babin & Anderson [26].

The mathematical treatment of time series is different to multivariate or longitudinal data since it has some special properties such as autocorrelated structures, trend and/or seasonal components and ergodic behavior. Basically, a time series is analyzed for forecasting, so an incomplete series does not allow to obtain the best predictors. Most of classical estimation methods do not provide good results when there are no covariates, complementary information, multiple missing observations in multiple locations, or even when the time series is volatile.

A univariate time series has no covariates for prediction, and in most cases there is no additional information available. If the time series has multiple missing data, then it is impossible to obtain its decomposition into Autorregressive (AR) and Moving Average (MA) processes.

Some applications of GAs to missing data problems were reported by Figueroa-García, Kalenatic & Lopez [19, 20], who used GAs to weather time series; Mussa Abdella & Tshilidzi Marwala [2], who used neural networks

trained by genetic algorithms to impute missing observations in databases; Siripitayananon, Hui-Chuan & Jin Kang-Ren [44] treated the missing data problem by using Neural Networks, Parveen & Green [39] solved a similar problem using Recurrent networks and Broersen, de Waele & Bos [12] found an optimal method to estimate missing data by means of autoregressive models and its spectral behavior.

Other interesting works are proposed by Nelwamondo, Golding and Marwala [37] who use dynamic programming to train neural networks; Kalra and Deo [32] who applied genetic algorithms for imputing missing data in biological systems; Zhong, Lingras and Sharma [48] who compared different imputation techniques for traffic problems; Londhe [35] who design a real-time framework for impute missed observations of wave measures; Ssali and Marwala [45] proposed a theoretical approach based on computational intelligence tools and decision trees to missing data imputation; JiaWei, TaoYang and YanWang [31] used fuzzy clustering for array problems; Abdella and Marwala [1] provide basic key features for implementing neural networks and genetic algorithms in missing data problems; and Eklund [18] computed the confidence interval of missed observations for spatial data problems.

Given this background, we present an evolutive algorithm for imputing multiple missing observations applied to a study case with multiple missing observations, where classical algorithms cannot solve the problem properly. Now, some basic definitions about time series are provided in next section.

2 Statistical Definitions for Time Series Analysis

The main purpose of a statistical analyst when analyzing a time series is to extract information about its behavior in order to make a decision based on the available information so far.

Now, a classical scenario starts from the definition of some basic statistical measures which represent the properties of the series before using any forecasting method. This reasoning is based on the concept of a *stochastic time series process*, which is defined as follows.

Definition 2.1. *Consider a set of observations of the variable x, where $x \in S$ and S is a metric space in which x is measured. This set x is said to be a Stochastic Process $\{X_t\}$ if it is a random sequence of observations recorded at a specific time t, $t \in T$ where T is the time space described by its probability density function (pdf). The pdf is a function in the form $f(X;\theta|S,\omega)$ where ω is the probability space of $f(X;\theta)$ and θ is a vector of parameters that characterizes its behavior.*

Remark 2.1. *Indeed, a stochastic process $\{X_t\}$ has the following property: Its pdf can vary at different times t_1 and t_2, although the metric space S is the same at all instants $t \in T$, then the probability that a specific value x occurs at different times x_{t_1} and x_{t_2}, is different.*

As usual, the most important order statistics for obtaining optimal models as ARIMA (Autoregressive, Co-integrated and Moving Average), ARCH (Autoregressive Conditional Heteroscedastic) and GARCH (Generalized Autoregressive Conditional Heteroscedastic) are the mean, the variance and the autocorrelation function, defined as follows

Definition 2.2. *The expected value of a random variable $E(X_t)$ is a measure of concentration of $\{X_t\}$ in ω defined as:*

$$E(X) = \int_{-\infty}^{\infty} x\ f(x;\theta)\ d(x) \tag{1}$$

Let $\{x_1, x_2, \cdots, x_n\}$ be observations of a time series. An unbiased estimator of $E(X_t)$ assuming large samples is the sample mean:

$$\bar{x} = \sum_{t=1}^{n} \frac{x_t}{n} \tag{2}$$

where n is the sample size.

Definition 2.3. *The variance of a random variable $Var(X)$ is a measure of form of $\{X_t\}$ in ω and is defined as:*

$$\mathrm{Var}(X_t) = \int_{-\infty}^{\infty} (x - E(X))^2\ f(x;\theta)\ d(x) \tag{3}$$

An unbiased estimator of $Var(X_t)$ assuming large samples is:

$$\mathrm{Var}(X) = \sum_{t=1}^{n} \frac{(x_t - \bar{x})^2}{n-1} \tag{4}$$

On the other hand, a *time series model* is a model that tries to infer some key properties of the series. According to Brockwell and Davis [10, 11], Hamilton [27] and Anderson [3], a time series model is

Definition 2.4. *A time series is a set of observations x_t, each one being recorded at a specific time t. A time series model for the observed data $\{x_t\}$ is a specification of joint distribution (or possibly the means and covariances) of a sequence of random variables $\{X_t\}$ for which $\{x_t\}$ is postulated to be a realization.*

The *sample autocovariance* and *sample autocorrelation* of the series is a linear relation between the variable at a specific time $\{x_t\}$ to itself at a lag h, $\{x_{t+h}\}$. Graybill & Mood [36], Wilks [47], Huber [29], Grimmet [25], Ross [42], Brockwell and Davis [10, 11], and Harville [28] defined them as follows

Definition 2.5. *The sample autocovariance function* $\hat{\gamma}(h)$ *is:*

$$\hat{\gamma}(h) = \sum_{t=1}^{n-|h|} \frac{(x_{t+|h|} - \bar{x})(x_t - \bar{x})}{n}, \ -n < h < n \tag{5}$$

Definition 2.6. *The sample autocorrelation function* $\hat{\rho}(h)$ *is:*

$$\hat{\rho}(h) = \frac{\hat{\gamma}(h)}{\hat{\gamma}(0)}, \ -n < h < n \tag{6}$$

When the series is incomplete, we cannot obtain *sufficient statistics*, which leads to misspecification problems of posterior models. In fact, the autocorrelation function defined in (6) is one of the most important measures of the behavior of the series, so the imputation of all missed observations is a key step before computing $\hat{\rho}(h)$.

In the next sections, we show all methodological aspects for imputing those missing observations through a GA. The main reason to use GAs is its flexibility and speed for finding solutions to nonlinear and complex problems.

3 The Proposed Genetic Algorithm (GA)

GAs are simple structures (For further information about GAs see Kim-Fung Man, Kit-Sang Tang & Sam Kwong [41]). An individual in a population can be seen as a set of missed data, so it should be imputed in the incomplete series. Figueroa-García, Kalenatic & Lopez [19, 20] use this principle to find an adequate solution to missing data problems, and this approach improves its fitness function based on some key properties of available data.

Our approach is based on six steps which guarantee an adequate solution: 1) a statistical preprocessing of the original series to obtain a stationary process 2) define a fitness function for comparing all individuals 3) generate of a population of individuals where each one is a solution of all missed observations 4) apply evolutionary operators for exploring the search space 6) evaluate the fitness function to select the best solution.

3.1 Why GA for Imputing Missing Data in Time Series?

An incomplete time series is a complex problem in the sense that classical imputation algorithms depend on covariates or additional information, and in many cases we have no this information. On the other hand, this is a multicriteria problem, which has no easy solutions. In this way, GAs are an interesting option for imputing multiple missing data in time series by the following reasons:

- Multicriteria capability.
- Nonlinear capability.
- Flexibility and computational simplicity.
- Efficiency of the solutions.

As shown in the Introduction, some learning-based techniques like neural networks and hill-climbing methods are commonly used to solve this problem. We propose an alternative method which learns from statistical properties of available data to cases where the series has *multiple missed observations* and/or *no covariates*. Those cases are particularly complex because optimal estimation techniques do not produce results when either multiple observations are lost or no covariates exist. Thus, evolutionary optimization arises as a flexible tool for finding solutions to complex cases, as the proposed one.

The following sections describe some general aspects of genetic algorithms applied to imputation in time series.

3.2 Preprocessing of Available Data

Some computational aspects should be kept in mind before applying any genetic operator, among them we have: Linear transformations, lag operators, seasonal and trend decompositions.

Firstly, we standardize data by using a linear transformation, removing the effect of units, and then we apply a lag operator to remove the effect of the mean of the process, obtaining a stationary series. These transformations reduce the complexity of the mean, the variance and the autocorrelation function of the series by removing its units, so its interpretation is easier and its search space is reduced, improving the performance of the algorithm. In this way, the following transformation is applied to available data $\{x_i^a\}$ in order to obtain a new standardized series $\{z_i^a\}$:

$$z_i^a = \frac{x_i^a - \bar{x}^a}{\sqrt{Var(x^a)}} \tag{7}$$

where $\{x_i^a\}$ is a vector of size $(n-m)$ of available data of the series.

Here, the mean $\bar{x}^a$ and variance $Var(x^a)$ of available data $\{x_i^a\}$ are obtained by removing the missing observations from its original one, as follows:

$$\bar{x}^a = \sum_{i=1}^{n-m} \frac{x_i^a}{n-m} \tag{8}$$

$$Var(x^a) = \sum_{i=1}^{n-m} \frac{(x_i^a - \bar{x}^a)^2}{n-m-1} \tag{9}$$

Now, we compute a lag operator of order d, ∇_d, defined as the difference between z_t and itself at period d,

$$\nabla_d(z_t) = z_t - z_{t-d} \tag{10}$$

The main idea of this transformation is to obtain a stationary series with no effect of the units of the series, so $\nabla_d(z_t)$ should be used as the target of the genetic algorithm. To do so, the following definition is given,

Definition 3.1 (Target series). *Hereinafter, we will refer to* $\{z_t\}$ *as a standardized series after applying* (7) *and* (10) *until reach a stationary series with zero mean and if possible, constant variance.*

The autocorrelation function of Z_t, $\hat{\rho}(h)$ cannot be computed when the series is incomplete, so we use a subset of Z_t, $\{z_t^l\}$ defined as the largest and most recent subset from available data, that is:

$$\hat{\gamma}(h)^l = \sum_{t=n_1}^{n_2-|h|} \frac{(z_{t+|h|}^l - \bar{z}^l)(z_t^l - \bar{z}^l)}{n_2 - n_1 + 1}, (n_1 - n_2) \leqslant h \leqslant (n_2 - n_1) \tag{11}$$

where n_1 and n_2 are lower and upper bounds of t, and $\hat{\gamma}(h)^l$ is the autocovariance of the largest and most recent subset of X_t, denoted by l.

$$\hat{\rho}(h)^l = \frac{\hat{\gamma}(h)^l}{\hat{\gamma}(0)^l} \tag{12}$$

In this way, $\hat{\rho}(h)^l$ is an important statistical measure obtained from available data, so its use as a part of the fitness function of the genetic algorithm is essential for time series analysis.

Remark 3.1 (Index sets i and t). *The index set i is related to available data $\{x_i\}$, instead of index t which is related to the series with missing data $\{x_t\}$, so we have $i \in t \in T$.*

An elite-based strategy is combined with a multicriteria fitness function to compose the basic structure of a genetic algorithm for imputing missing data. Its methodological aspects are discussed in the following subsections.

3.3 The Fitness Function

A time series $\{z_t\}$ is said to be incomplete if there exist m missing observations located by an index vector v, where $1 \leqslant v \leqslant n$. A vector of imputations of all missed observations is called $\{y_t\}$, where $y_t = 0$ when $t \notin v$ and $y_t = z_j$ when $t \in v$, z_j is the j_{th} element of y_t, $1 \leqslant j \leqslant m$ located in the v_{th} position. A new series where all missed observations are replaced is defined as $\{\hat{z}_t\}$:

$$z_t^a + y_t = \hat{z}_t \tag{13}$$

Now, the main goal is to find a vector y_t which does not change the properties of available data. For our purposes, the autocorrelation function, mean and variance of the available data are the goal of the genetic algorithm.

A genetic solution to the m missing observations should not change its $\hat{\gamma}(h)^l$, $\bar{z}^a$ and $Var(z^a)$ measures. To do so, we define the fitness function of the algorithm as a multicriteria function namely $\mathcal{F}$, regarding a set of H lags used for computing autocorrelations, as follows:

$$\mathcal{F} = \sum_{h \in H} \mid \hat{\rho}(h)^l - \hat{\rho}(h) \mid + \mid \bar{z}^a - \bar{\hat{z}} \mid + \mid Var(z^a) - Var(\hat{z}) \mid \qquad (14)$$

Thus, the main goal of the algorithm is to minimize $\mathcal{F}$ and if possible, reach zero as optimal solution. Note that our proposal is based on the design of a fitness function that minimize the differences among the statistical measures of the available data and the series after imputation $\hat{z}_t$ in three ways:

- Significative autocorrelations, $h \in H$.
- Sample mean.
- Sample variance.

As shown in Definition 2.4, a time series can be described through its mean, variance and covariances, so (14) tries to characterize the time series after imputation of missing data. In this proposal, we aggregated different units in a single function without problems, since $\{z_t\}$ is a standardized variable.

Remark 3.2 (Magnitude of $\mathcal{F}$). *It is important to note that the use of* (7) *and* (10) *leads to obtain measures of* $\{z_t\}$ *with no effect of the mean and units of the original series, so* $\hat{\rho}(h)^l, \bar{z}^a$ *and* $\mathrm{Var}(z^a)$ *are standardized measures that can be added in* (14) *without loss of generality.*

3.4 Individuals

An individual is defined as a vector of a population indexed on a matrix where each one is a solution itself. As always, a genetic structure contains many individuals forming a population.

Each individual represents a complete vector of missed observations, which will be located in z_t by using y_t. Thus, each individual has as much elements (genes) as missed observations exist, indexed by v. A graphical explanation of the genes and individuals of the algorithm is shown in Figure 1

In Figure 1, v is the index vector of the lost observations of z_t. Note that z_j has the same elements than v, but v only has the t_{th} position of the missed observations while z_j is a vector of solutions located by v, which is computed through GAs. Finally, $\hat{z}_t$ is defined in (13) where $y_t = z_j$ located by v.

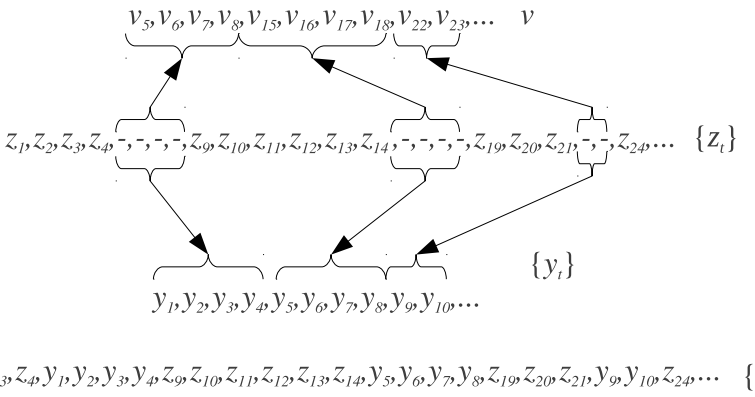

Fig. 1 Individuals, genes, v, z_t and $\hat{z}_t$

3.5 Populations and Number of Generations

An important part of a GA is how to generate a population (collection of individuals), and how many populations (generations) will be used for exploring the search space. First, the population size is defined by the m missing observations and a pre-selected $k \in K$ number of individuals, creating a matrix of size $k \times m$ labelled as $P^g_{k,m}$ where g denotes the *generation index*.

Different population sizes can be selected for exploring the search space. According to Burke et. al. [13], Goldberg [23], Bäck [8], Bagchi [9], and Fonseca and Fleming [21], the selection of a higher population sizes together with fitness-based operators may reduce the performance of the algorithm, even when the search space would be better covered.

In this way, Figueroa-García, Kalenatic & Lopez [19, 20] used three population sizes: $k \in [100, 500, 1000]$, so based in their experimental evidence, we recommend to use a size of $k = 100$ in order to increase the speed of the algorithm, with no loss of ability of exploration of the solution space.

Another important parameter of the algorithm is the number of generations G, which is commonly used as stopping criterion. In this approach, this parameter operates as a controller of the iterations of the algorithm, so we set $\max_g = G$. This parameter depends on the complexity of the problem, the size and the nature of the missed observations, so the analyst should select G experimentally and using knowledge of the problem.

3.6 Population Random Generator

Definition 3.1 establishes that $\{z_t\}$ should be a standardized series, so this condition reduces the complexity of the algorithm, allowing us to use a standard uniform generator, which is computationally simpler than other generators e.g. Normal, exponential or mixed methods.

The uniform random generator is called R_j. It is defined as $R_j(a,b) = a + r_j(b-a)I_{[0,1]}(r_j)$ where a is the minimun value, b is the maximum value and r_j is a random number defined by the Index Function $I_{[0,1]}$.

An important analysis of random number generation has been made by Devroye [17] and Law & Kelton [33]. They concluded that the uniform number generator is an adequate method for covering the search space, so we recommend to use a uniform generator instead of the sample distribution.

3.7 Mutation and Crossover Operators

In Figure 1 we have explained how an individual and a gene are defined. This allows us to easily compose a population through R_j which is ranked using an elite-based method. After that, a mutation and crossover strategy can be applied to get a better exploration of the search space, as proposed below:

Mutation strategy:

1. Select a random position for each orderly individual in $P^g_{k,m}$ by its fitness function.
2. Replace the selected position with a new individual obtained by using a random generator $R_j(a,b)$.
3. Repeat (2) for the c_1 better individuals orderly for each population $P^g_{k,m}$ at the generation g.

Crossover strategy:

1. Select the c_2 first individuals in the orderly population $P^g_{k,m}$ by its fitness function.
2. Generate a new individual by replacing all even genes with their respective even gene located in the next individual.
3. Generate a new individual by replacing all odd genes with their respective odd gene located in the next individual for each one.
4. Repeat (3) for the c_2 better individuals orderly for each population $P^g_{k,m}$ at the generation g.

Remark 3.3 (Ranking of the solutions). *Figueroa-García, Kalenatic and López [19, 20] used an elite-based method for ranking the individuals of the population. Although it is a classical method which shows a good behavior, we encourage the reader to implement other ranking methods for the sake of new developments and improvements.*

3.7.1 Completing the Population

A classical strategy for exploring the space of solutions is by replacing the worst individuals by new ones, preserving the best ones at each population $P^g_{k,m}$. As usual, the number of best individuals is a free parameter, and in some cases it is involved as a random part of the algorithm.

Now, $P_{k,m}^{g+1}$ is updated by a set of random individuals, which is generated by replacing the worst individuals with new ones, in order to find better solutions. In short, the best k^1 individuals are preserved for the next generation and later it is completed by $\{k - k^1 - c_1 - c_2\}^m$ new individuals.

3.8 Stopping Strategy

There are different criteria for stopping a GA. Two of the most used methods are: A first one which uses a predefined maximum number of iterations called G, that is $g \to G$, and a second one which stops a GA when its fitness function $\mathcal{F}$ has no a significant improvement after a specific number of iterations.

Aytug and Koehler [6, 7] proposed an alternative stopping criterion for GAs based on a function of its mutation rate, the size of the population strings and the population size. Bhattacharrya and Koehler [5] generalized their results to non-binary alphabets. Pendharkar and Koehler [40] proposed a stopping criterion based on the markovian properties of a GA, and Safe et.al. [43] proposed entropy measures for constructing stopping operators.

The number of generations G is another degree of freedom of a GA. Usually, as $\mathcal{F}$ has no improvements, then G should be reduced. Finally, the best individual is selected by ranking $\mathcal{F}$ through all runs and generations, so the best individual will be imputed in the original series to complete the series.

An elite-based approach usually gets better solutions because this ensures the improvement of the solution through all generations. On the other hand, different stopping criteria can be used as long as the solutions are improved.

A brief description of the algorithm is presented in the Algorithm 1.

Algorithm 1. Genetic algorithm

Require: $v, n, n_1, n_2, m, H, c_1, c_2, k^1, \hat{\rho}(h)^l, \bar{z}^a, Var(z)^a$
  Generate an initial population of size k by using R_j
  for $g = 1 \to G$ **do**
    return $\mathcal{F}$ For each k_{th} individual
    Index $P_{k,m}^g$ by $\mathcal{F}$
    Apply the mutation operator to the c_1 better individuals
    Apply the crossover operator to the c_2 better individuals
    return $\mathcal{F}$ For each k_{th} individual
    Index $P_{k,m}^g$ by $\mathcal{F}$
    Preserve the best k^1 individuals, indexed by $\mathcal{F}$
    Complete the population with a vector of size $\{k - k^1 - c_1 - c_2\}^m$
  end for
  return $\mathcal{F}$ For each k_{th} individual
  Index $P_{k,m}^G$ by $\mathcal{F}$
  return $\min \mathcal{F}$
  Replace $P_{1,m}^G$ in the original series, indexing it by using v

Another common strategy for exploring the space of solutions is by running the algorithm several times, *Runs*. The number of runs should be a function of k, m, G and the computing time of each run, so we recommend to initialize with small attempts before running an adequate experiment. A graphical display of the imputation strategy is shown in Figure 2.

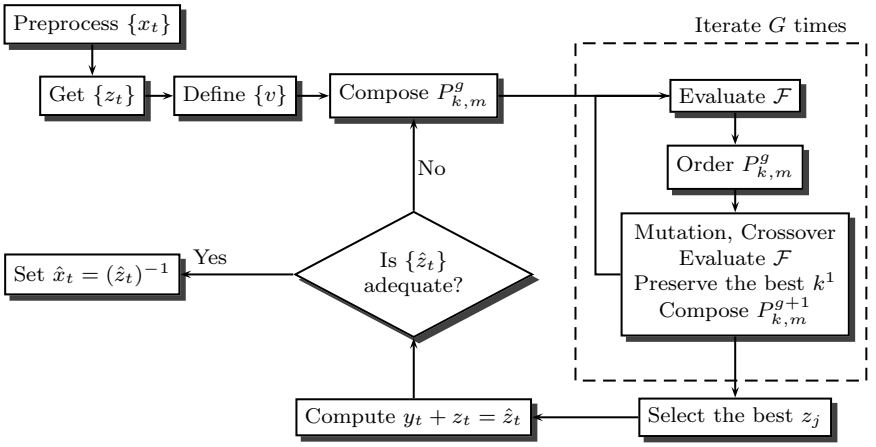

Fig. 2 Flowchart of the proposed GA

In the following section, an application of the algorithm is presented and compared to a classical imputation algorithm.

4 Application Example

The selected study case is a weather time series that has multiple missed observations produced by a failure in the measurement device. In this case, we use the minimum temperature *(MT)* recorded at the town of Chía - Colombia during 1368 days between 10 p.m. and 5 a.m. when maximum and minimum levels are registered, each one measured every half an hour throughout the night. All missed observations are displayed by discontinuities. Figure 3 shows the original series and the series after preprocessing by applying (7) and (10), where, a) is the original series and b) is the series after preprocessing.

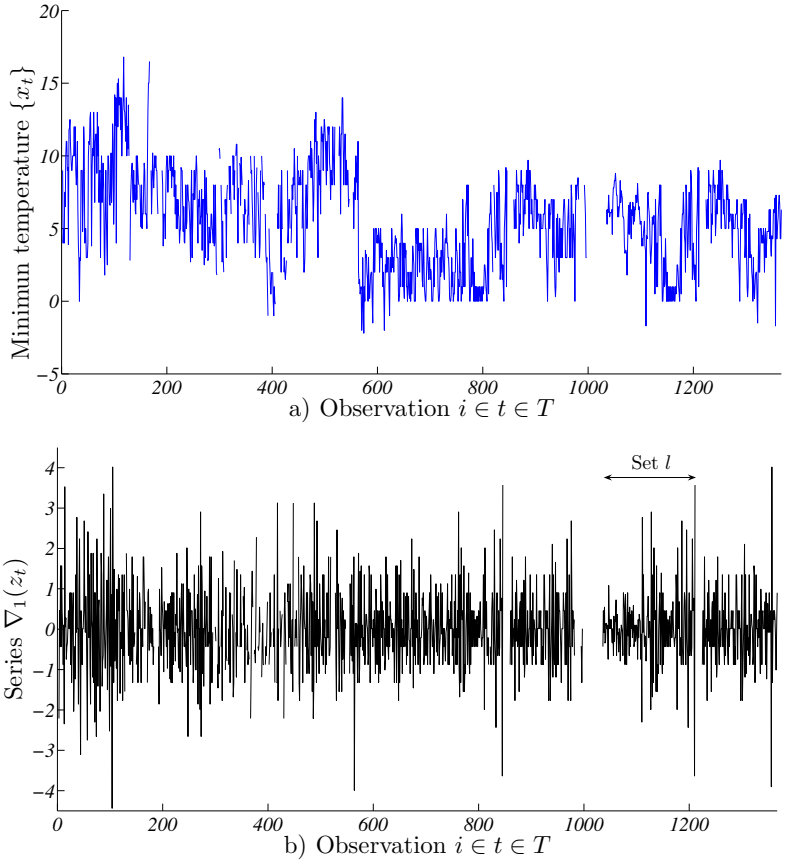

Fig. 3 Study Case measured in Chía - Colombia

4.1 Statistical Analysis

Some basic statistics obtained from available data are shown in Table 1. Its mean and variance are used as estimations of (8), (9). We obtain $\hat{\rho}(h)^l$ by using (12) for $H = \{1, \cdots, 6\}$ in order to define the fitness function $\mathcal{F}$ for each genetic structure.

Table 1 Observed statistics

Measure	$\hat{\rho}(1)$	$\hat{\rho}(2)$	$\hat{\rho}(3)$	$\hat{\rho}(4)$	$\hat{\rho}(5)$	$\hat{\rho}(6)$	$\bar{z}^a$	$Var(z^a)$	min	max
Value	-0.339	-0.052	0.004	-0.104	0.129	-0.051	-0.027	5.053	-4.436	4.015

In this case, we have 1367 observations ($N = 1367$) of the minimum temperature at the town of Chía - Colombia, where 159 are lost ($m = 159, n = 1208$). Some randomness tests done over $\{z_i^a\}$ are shown in Table 2.

Table 2 Tests of randomness

Tests on normality				
Test.	*p-value*	*p-value*	*p-value*	*p-value*
Shapiro-Wilks	0.0023	≈ 0	≈ 0	≈ 0
K-S.	≈ 0	≈ 0	0.0014	≈ 0
Tests on randomness				
Test.	*p-value*	*p-value*	*p-value*	*p-value*
Runs Test	0.001	≈ 0	≈ 0	≈ 0
Turning Points	≈ 0	≈ 0	≈ 0	≈ 0
Ljung-Box [b]	≈ 0	≈ 0	≈ 0	≈ 0
ARCH [b]	≈ 0	≈ 0	≈ 0	0.0025

b This test is made by using the first lag of the series

All tests conclude that the series is not a random variable. The Ljung-Box and ARCH tests reject the hypothesis that the series has no serial correlation, this means that it presents autocorrelation at least on its first lag. Both Shapiro-Wilks and Kolmogorov-Smirnov (K-S) tests reject the hypothesis that each series is normally distributed, which is an important constraint for some imputation methods that are based on strong normality assumptions.

4.2 Classical Estimation Methods

One of the most popular imputation algorithms is the expectation maximization *(EM)* algorithm, which is based on conditional expectations of a random variable, obtained from a set of auxiliary variables which give an estimate of the behavior of the missing data. Its principal objective is to maximize the *Likelihood or Log-Likelihood Function* of the *pdf* sample, obtaining an optimal estimation of the missing observations. This algorithm was proposed by Dempster [16], and Gaetana & Yao [22] proposed a variation of the EM algorithm based in a simulated annealing approach to improve its efficiency for the multivariate case. Celeux & Diebolt [14], Levine & Casella [34], Nielsen [38] have reported some modifications for a stochastic scenario, and Arnold [4] estimates the parameters of a state-dependant AR model by using the EM algorithm with no prior knowledge about state equations.

By using the EM algorithm, a maximum likelihood estimator is obtained by replacing all v positions of $\{x_i\}$ by its expected value $E(x)$, so we have

that $\{x_t\} = E(x)$; and the regression method replaces all missed observations by random residuals of available data; its results are displayed in Figure 4.

Another method is based on auxiliary regressions, which consists on estimate the mean and variance of available data, and then each missed observation is replaced by the estimated mean plus a random residual obtained from a regression of available data against auxiliary variables. In the case of univariate series, this method is a simple estimation of its mean and variance.

In Figure 4, *a*) shows the results of the EM algorithm and *b*) presents the results of the regression method. According to Table 3, we can conclude that these approaches does not show desirable properties for univariate time series. Their statistical properties are presented in Table 3.

Table 3 Results of classical estimation methods

Measure	$\hat{\rho}(1)$	$\hat{\rho}(2)$	$\hat{\rho}(3)$	$\hat{\rho}(4)$	$\hat{\rho}(5)$	$\hat{\rho}(6)$	$\hat{z}$	$Var(\hat{z})$	*It.*	$\mathcal{F}$
EM algorithm	-0.302	-0.071	-0.046	-0.027	0.042	-0.0003	-0.0002	0.883	4	4.516
Regression	-0.266	-0.071	-0.051	-0.018	0.044	0.005	0.0128	0.997	N.A.	4.470

By evaluating the Fitness Function $\mathcal{F}$, both classical algorithms provide great differences among the obtained mean, variance, $\hat{\rho}^l(h)$ and their available values. With these evidences it is clear that the EM algorithm is not the best option to estimate missing data on a univariate time series context.

4.3 Genetic Approach

Figure 2 describes the methodology proposed in this chapter. First, we apply (7) and (10) to standardize data, then we compute $\rho(h)^l$ using (12) for the first $H = 6$ lags, and later we compute $\bar{z}^a$ and $Var(z^a)$ of available data, as shown in Table 5.

Figueroa-García, Kalenatic and Lopez [19, 20] have found better results with $k = 100$ individuals, outperforming computing time and improving the quality of solutions. The crossover, mutation and remaining parameters used in the GA for each series are shown in Table 4, where a and b are obtained from the observed series as its potential maximum and minimum values. c_1, c_2 are free parameters which modify the mutation and crossover rates of the GA, and G is selected by trial and error based on the behavior of $\mathcal{F}$. n_1 and n_2 are initial and end points of the *largest and most recent* complete dataset (See *"Set l"* in Figure 3-b), which are needed by(11) to obtain $\hat{\rho}(h)^l$ through the use of (12) for $H = \{1, \cdots, 6\}$.

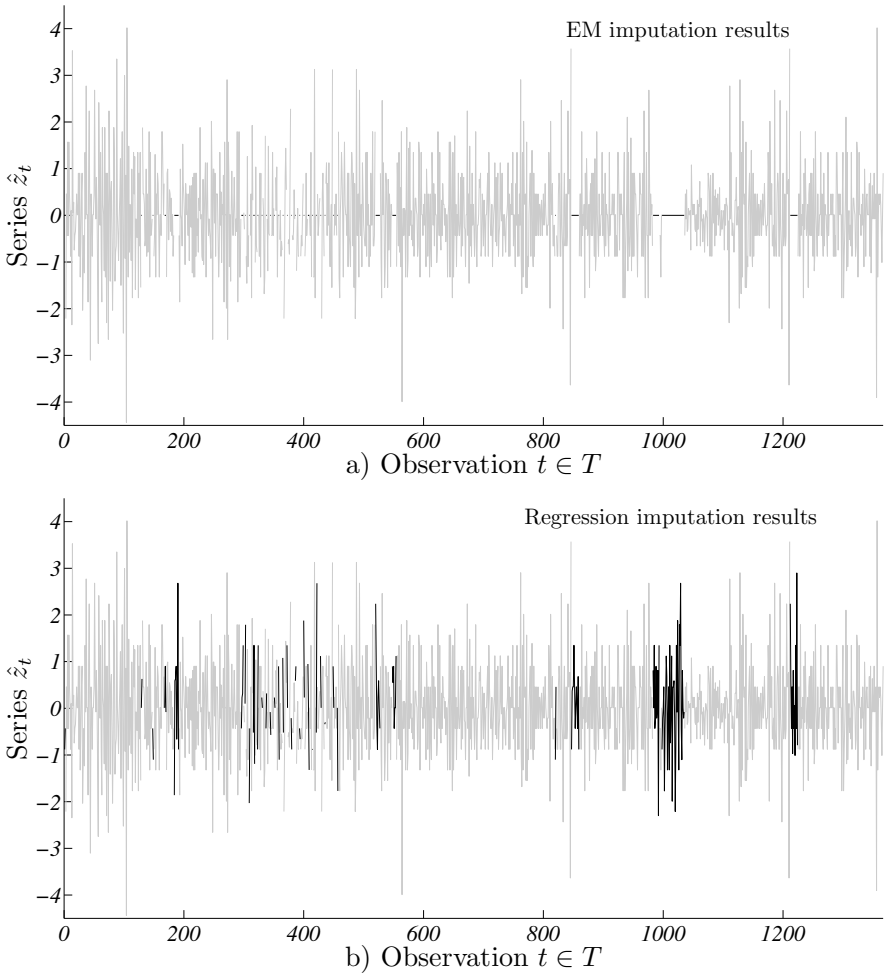

Fig. 4 Results of classical imputation methods

Table 4 Genetic algorithm parameters

Parameter	k	a	b	c_1	c_2	n_1	n_2	m	G	*Av. time (sec)*
Value	100	-5.5	5	4	4	1036	1211	159	5000	448.3

In this Table, the average time (in sec.) was obtained from 25 runs of the algorithm. The maximum processing time was 497.3 sec and the lower processing time was 421.1 sec. After the total 125.000 generations of the algorithm divided into 25 runs, the best solution (minimum $\mathcal{F}$), was selected. The obtained solution is displayed in black in Figure 5 and the obtained results for all imputed data are presented in Table 5.

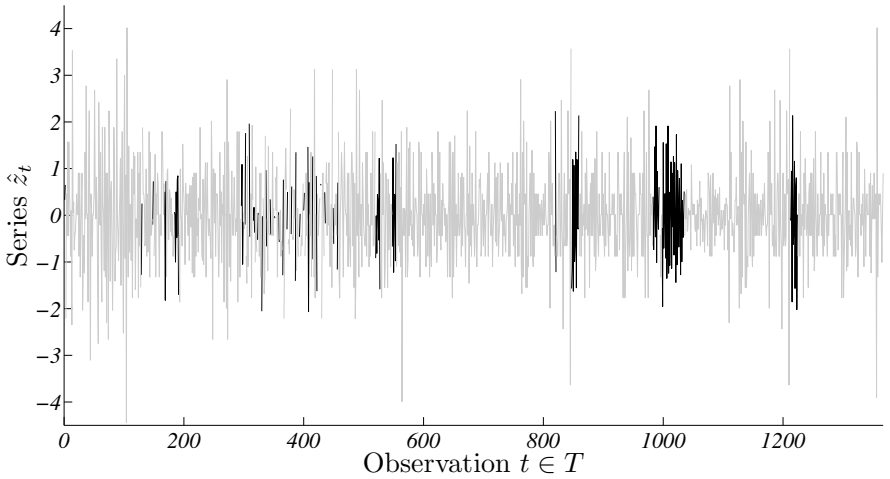

Fig. 5 Complete dataset with imputed missing data

Table 5 Results of evolutionary optimization

Measure	$\hat{\rho}(1)$	$\hat{\rho}(2)$	$\hat{\rho}(3)$	$\hat{\rho}(4)$	$\hat{\rho}(5)$	$\hat{\rho}(6)$	$\bar{\hat{z}}$	$Var(\hat{z})$	$\mathcal{F}$
Value	-0.340	-0.052	0.004	-0.105	0.119	-0.052	-0.027	5.053	0.0112

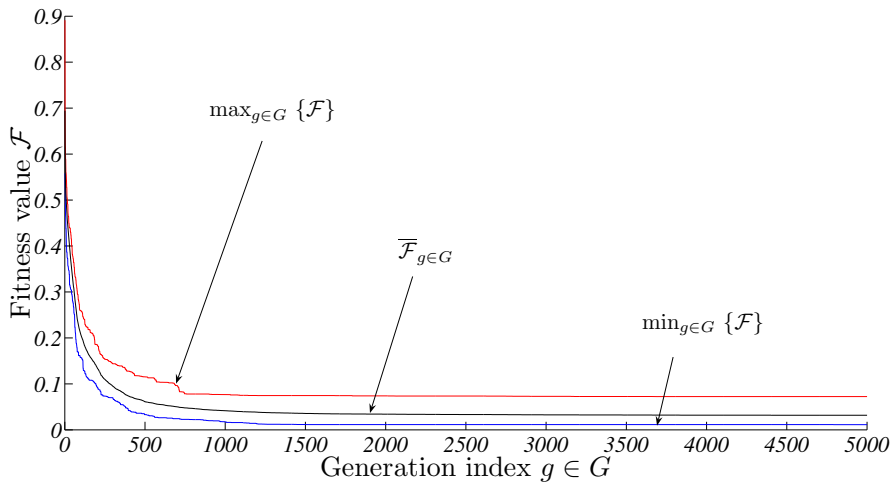

Fig. 6 Behavior of the GA for 25 runs

In Figure 6, the behavior of the proposed GA is measured by the minimum, average and maximum values of $\mathcal{F}$ over 25 runs, called $\max_{g\in G}\{\mathcal{F}\}, \overline{\mathcal{F}}_{g\in G}$ and $\min_{g\in G}\{\mathcal{F}\}$ respectively. Note that the GA always goes to stable values of $\mathcal{F}$, and they have no a high improvement after about $g = 2000$ iterations.

In general, the GA solution has no great differences to original data and it does not change its statistical properties. In this way, the proposed method seems to be a better method for imputing multiple missing observations in time series than other classical algorithms.

5 Output Analysis

This section focuses on analyzing the original series vs. genetic imputation. The output analysis is based on comparisons of some interesting statistical measures, among them we have: Tests on means, variances, autocorrelations and experimental design. Figure 7 shows the way all of them are connected.

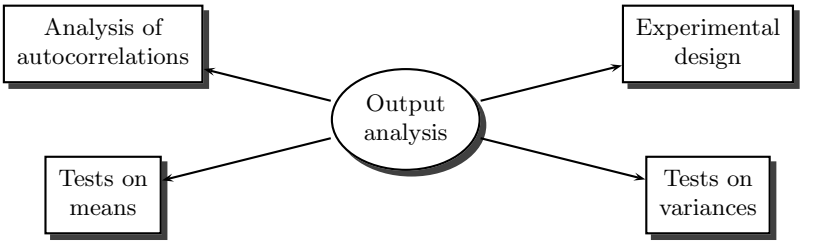

Fig. 7 Output analysis

Some descriptive statistics, randomness, differences on means, variances and autocorrelations tests were performed, as shown in Tables 6, 8 and 9.

Table 6 Tests of normality and randomness (significance)

Test	*Runs*	*Turning-point*	*S-W*	*K-S*	*Ljung-Box*[b]	*ARCH*[b]
Original series	≈ 0	≈ 0	≈ 0	≈ 0	0.0001	≈ 0
Imputed series	≈ 0	≈ 0	≈ 0	≈ 0	≈ 0	≈ 0

b This test is made by using the first lag of the series.

5.1 Tests on Means and Variances

The general hypotheses used for finding differences between original and imputed series, are as follows

Table 7 Hypothesis on means and variances

	Test on means	*Test on variances*
H_0 :	$\bar{\hat{z}} = \bar{z}^a$	$Var(\hat{z}) = Var(z^a)$
H_a :	$\bar{\hat{z}} \neq \bar{z}^a$	$Var(\hat{z}) \neq Var(z^a)$

The obtained results of the Tests on means are presented in Table 8.

Table 8 Tests on means (significance)

Test	*ANOVA*	*Welch*	*Brown-Forsythe*	*K-S*	*Mann-Whitney*
Original vs. Imputed	0.964	0.964	0.964	≈1	0.6404

With these statistical evidences, the hypothesis on means defined in Table 7 with a 95% confidence level, is accepted. We implement the Levene's test for contrasting their variances. Its results are shown in Table 9.

Table 9 Levene test

Test	*Levene stat*	*Significance*
Original vs. Imputed	0.00414	0.948

The ANOVA, Welch, Brown-Forsythe, K-S, Mann-Whitney and Levene tests conclude that there are no differences between $\bar{\hat{z}} \rightarrow \bar{z}^a$, and $Var(\hat{z}) \rightarrow Var(z^a)$ respectively, this means that the genetic solution has no statistical differences to available data. With these statistical evidences, the hypothesis defined in Table 7 are accepted with a 95% confidence level.

Remark 5.1 (Additional analysis). *Although we recommend the use of experimental design and autocorrelation analysis (See Figure 7), we did not perform those analysis due to the high similarity among autocorrelations and the absence of differences between means and variances. In case where means and/or variances have differences, it is recommended to perform an experimental design for finding the causes of differences among each run of the GA and the original series.*

Roughly speaking, the GA outperforms the solution provided by classical algorithms, in terms of the statistical properties of the series. The obtained results have no any statistical evidence to reject H_0, so we can accept them.

6 Concluding Remarks

The following concluding remarks can be made

1. The proposed genetic algorithm outperforms classical algorithms, providing better solutions without modifying their available properties.
2. The flexibility of evolutionary methods allows us to design efficient algorithms for finding missing observations in a time series context; its nonlinear capability becomes a powerful tool for exploring the search space.
3. The use of multi-criteria fitness operators are alternative tools in front to classical imputation methods as the EM algorithm and its modifications.
4. Most of optimization techniques need additional variables to be consistent. The presented approach finds successful solutions with no additional information, which is a common issue in univariate time series.
5. Some emerging applications as multivariate data analysis, signal and image processing problems are proposed for future applications.

Finally, we encourage the reader to improve the presented results by modifying our proposal. The fitness function ($\mathcal{F}$), population size, c_1, c_2, k^1 and g can be modified, so other strategies can be used for getting better results in other missing data cases.

References

1. Abdella, M., Marwala, T.: Treatment of missing data using neural networks and genetic algorithms. In: IEEE (ed.) Proceedings of International Joint Conference on Neural Networks, pp. 598–603. IEEE (2005)
2. Abdella, M., Marwala, T.: The use of genetic algorithms and neural networks to approximate missing data in database. In: IEEE (ed.) IEEE 3rd International Conference on Computational Cybernetics, ICCC 2005, vol. 3, pp. 207–212. IEEE (April 2005)
3. Anderson, T.W.: The Statistical Analysis of Time Series. John Wiley and Sons (1994)
4. Arnold, M.: Reasoning about non-linear AR models using expectation maximization. Journal of Forecasting 22(6), 479–490 (2003)
5. Aytug, H., Bhattacharrya, S., Koehler, G.J.: A markov chain analysis of genetic algorithms with power of 2 cardinality alphabets. ORSA Journal on Computing 96(6), 195–201 (1997)
6. Aytug, H., Koehler, G.J.: Stopping criteria for finite length genetic algorithms. ORSA Journal on Computing 8(2), 183–191 (1996)
7. Aytug, H., Koehler, G.J.: New stopping criterion for genetic algorithms. European Journal of Operational Research 126(1), 662–674 (2000)
8. Bäck, T.: Evolutionary Algorithms in Theory and Practice: Evolution Strategies, Evolutionary Programming, Genetic Algorithms. Oxford University Press (1996)
9. Bagchi, T.: Multiobjective Scheduling by Genetic Algorithms. Kluwer Academic Publishers (1999)
10. Brockwell, P., Davis, R.: Time Series: Theory and Methods. Springer (1998)

11. Brockwell, P., Davis, R.: Introduction to Time Series and Forecasting. Springer (2000)
12. Broersen, P., de Waele, S., Bos, R.: Application of autoregressive spectral analysis to missing data problems. IEEE Transactions on Instrumentation and Measurement 53(4), 981–986 (2004)
13. Burke, E.K., Gustafson, S., Kendall, G.: Diversity in genetic programming: An analysis of measures and correlation with fitness. IEEE Transactions on Evolutionary Computation 8(1), 47–62 (2004)
14. Celeux, G., Diebolt, J.: The SEM algorithm: a probabilistic teacher algorithm derived from the EM algorithm for the mixture problem. Computational Statistics Quarterly 2(1), 73–82 (1993)
15. Chambers, R.L., Skinner, C.J.: Analysis of Survey Data. John Wiley and Sons (2003)
16. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum-likelihood from incomplete data via the EM algorithm. Journal of Royal Statistical Society 39(1), 1–38 (1977)
17. Devroye, L.: Non-Uniform Random Variate Generation. Springer, New York (1986)
18. Eklund, N.: Using genetic algorithms to estimate confidence intervals for missing spatial data. IEEE Transactions on Systems, Man and Cybernetics, Part C: Applications and Reviews 36(4), 519–523 (2006)
19. Figueroa García, J.C., Kalenatic, D., Lopez Bello, C.A.: Missing Data Imputation in Time Series by Evolutionary Algorithms. In: Huang, D.-S., Wunsch II, D.C., Levine, D.S., Jo, K.-H. (eds.) ICIC 2008. LNCS (LNAI), vol. 5227, pp. 275–283. Springer, Heidelberg (2008)
20. Figueroa García, J.C., Kalenatic, D., López, C.A.: An evolutionary approach for imputing missing data in time series. Journal on Systems, Circuits and Computers 19(1), 107–121 (2010)
21. Fonseca, C.M., Fleming, P.J.: Genetic algorithms for multiobjective optimization: Formulation, discussion and generalization. Evolutionary Computation 3(1), 1–16 (2004)
22. Gaetan, C., Yao, J.F.: A multiple-imputation metropolis version of the EM algorithm. Biometrika 90(3), 643–654 (2003)
23. Goldberg, D.E.: Genetic Algorithms in Search, Optimization, and Machine Learning. Adisson-Wesley (1989)
24. González, S., Rueda, M., Arcos, A.: An improved estimator to analyse missing data. Statistical Papers 49(4), 791–796 (2008)
25. Grimmet, G., Stirzaker, D.: Probability and Random Processes. Oxford University Press (2001)
26. Hair, J.F., Black, W.C., Babin, B.J., Anderson, R.E.: Multivariate Data Analysis, 7th edn. Prentice-Hall (2009)
27. Hamilton, J.D.: Time Series Analysis. Princeton University (1994)
28. Harville, D.A.: Matrix Algebra from a Statician's Perspective. Springer-Verlag Inc. (1997)
29. Huber, P.: Robust Statistics. John Wiley and Sons, New York (1981)
30. Ibrahim, J., Molenberghs, G.: Missing data methods in longitudinal studies: a review. TEST 18(1), 1–43 (2009)
31. JiaWei, L., Yang, T., Wang, Y.: Missing value estimation for microarray data based on fuzzy c-means clustering. In: IEEE (ed.) Proceedings of High-Performance Computing in Asia-Pacific Region, 2005 Conference, pp. 616–623. IEEE (2005)

32. Kalra, R., Deo, M.: Genetic programming for retrieving missing information in wave records along the west coast of india. Applied Ocean Research 29(3), 99–111 (2007)
33. Law, A., Kelton, D.: Simulation System and Analysis. McGraw Hill International (2000)
34. Levine, L.A., Casella, G.: Implementations of the monte-carlo EM algorithm. Journal of Computational Graphic Statistics 10(1), 422–439 (2000)
35. Londhe, S.: Soft computing approach for real-time estimation of missing wave heights. Ocean Engineering 35(11), 1080–1089 (2008)
36. Mood, A.M., Graybill, F.A., Boes, D.C.: Introduction to the Theory of Statistics. Mc Graw Hill Book Company (1974)
37. Nelwamondo, F.V., Golding, D., Marwala, T.: A dynamic programming approach to missing data estimation using neural networks. Information Sciences (in press 2012)
38. Nielsen, S.F.: The stochastic EM algorithm: Estimation and asymptotic results. Bernoulli 6(1), 457–489 (2000)
39. Parveen, S., Green, P.: Speech enhancement with missing data techniques using recurrent neural networks. In: IEEE (ed.) Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2004), vol. 1, pp. 733–738. IEEE (2004)
40. Pendharkar, P.C., Koehler, G.J.: A general steady state distribution based stopping criteria for finite length genetic algorithms. European Journal of Operational Research 176(3), 1436–1451 (2007)
41. Qin, Y., Zhang, S., Zhu, X., Zhang, J., Zhang, C.: Semi-parametric optimization for missing data imputation. Applied Intelligence 27(1), 79–88 (2007)
42. Ross, S.M.: Stochastic Processes. John Wiley and Sons (1996)
43. Safe, M., Carballido, J., Ponzoni, I., Brignole, N.: On Stopping Criteria for Genetic Algorithms. In: Bazzan, A.L.C., Labidi, S. (eds.) SBIA 2004. LNCS (LNAI), vol. 3171, pp. 405–413. Springer, Heidelberg (2004)
44. Siripitayananon, P., Hui-Chuan, C., Kang-Ren, J.: Estimating missing data of wind speeds using neural network. In: IEEE (ed.) Proceedings of the 2002 IEEE Southeast Conference, vol. 1, pp. 343–348. IEEE (2002)
45. Ssali, G., Marwala, T.: Computational intelligence and decision trees for missing data estimation. In: IEEE (ed.) IJCNN 2008 (IEEE World Congress on Computational Intelligence), pp. 201–207. IEEE (2008)
46. Tsiatis, A.A.: Semiparametric Theory and Missing Data. Springer Series in Statistics (2006)
47. Wilks, A.: Mathematical Statistics. John Wiley and Sons, New York (1962)
48. Zhong, M., Lingras, P., Sharma, S.: Estimation of missing traffic counts using factor, genetic, neural, and regression techniques. Transportation Research Part C: Emerging Technologies 12(2), 139–166 (2004)