Information Sciences 619 (2023) 947–967

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

A genetic algorithm for multivariate missing data imputation

Juan Carlos Figueroa-García [a,*,1], Roman Neruda [b,2], German Hernandez–Pérez [c,3]

[a] Universidad Distrital Francisco José de Caldas, Bogotá, Colombia
[b] Institute of Computer Science, Czech Academy of Sciences, Prague, Czech Republic
[c] Universidad Nacional de Colombia, Bogotá, Colombia

ARTICLE INFO

Article history:
Received 6 August 2022
Received in revised form 7 October 2022
Accepted 12 November 2022
Available online 17 November 2022

Keywords:
Missing data
Genetic algorithms
Multivariate missing data
Data imputation

ABSTRACT

Some data mining, AI and data processing tasks might have data loss whose estimation/imputation is an important problem to be solved. Genetic algorithms are efficient and flexible global optimization methods able to deal with both multiple missing observations and multiple features such as continuous/discrete/binary data which are often found in multivariate databases unlike classical missing data estimation methods which only deal with univariate–continuous data. This paper presents a genetic algorithm to impute multiple missing observations in multivariate data which minimizes a new multi–objective (fitness) function based on the Minkowski distance of the means, variances, covariances and skewness between available/completed data. To do so, two sets of examples were tested: a continuous/discrete dataset which is compared to both the EM algorithm and auxiliary regressions, and a comparison over seven benchmark datasets.
© 2022 The Author(s). Published by Elsevier Inc. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

1. Introduction and motivation

In the recent years, the missing data problem has gained popularity as a research area in data mining, machine learning, and other data processing tasks since it adds uncertainty to decision making in different disciplines, so there is a need for estimation/imputation methods able to reduce the effects of information loss and biased analysis. The most frequent method used in practice and implemented in data mining software libraries is to simply discard the observations with missing values which is often undesirable because this can cause a too harsh data reduction with important statistical properties can be changed.

Most of the available statistical missing data methods/algorithms were designed for single-feature continuous data (a.k.a. univariate data) under strong assumptions like known populations, normality, estimability, kernels, etc. Also some AI techniques such as neural networks, fuzzy systems, k-nearest neighbors, Genetic Algorithms (GA) etc. have been applied to univariate problems such as time series/regression, classification/pattern recognition, real-world financial, biomedical, etc. with interesting results.

Missing data in multiple-feature datasets (a.k.a. multivariate data) is a more complex problem due to its underlying properties like covariance/correlation structures, non-Gaussian/binary/discrete variables, unbalancing, sphericity/circularity etc.

* Corresponding author.
E-mail addresses: jcfigueroag@udistrital.edu.co (J.C. Figueroa-García), roman@cs.cas.cz (R. Neruda), gjhernandezp@unal.edu.co (G. Hernandez–Pérez).
[1] Faculty of Engineering, Universidad Distrital Francisco José de Caldas, Bogotá, Colombia.
[2] Institute of Computer Science, Czech Academy of Sciences, Prague, Czech Republic
[3] Department of Systems Engineering and Computing, Universidad Nacional de Colombia, Bogotá, Colombia

https://doi.org/10.1016/j.ins.2022.11.037
0020-0255/© 2022 The Author(s). Published by Elsevier Inc.
This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

which together with multiple missing observations makes classical estimation methods show a poor performance. This led us to explore GAs to handle multi-feature data while dealing with multiple missing observations at once since GAs are efficient/powerful to cover integer/binary spaces in NP-hard multiobjective problems which require much more flexible solution approaches.

The scope of this work is to impute missing observations in multivariate data by using a *Multiple Imputation Genetic Algorithm* (MIGA) in order to preserve available matrix properties such as the means, covariances and skewness. Two examples: an integer/binary and a benchmark data are solved by the MIGA, compared to other methods and some statistical tests are performed to validate the obtained results.

1.1. State of the art

Statistical missing data estimation theory usually infers the population of samples to then replace missing observations by minimizing squared error functions. Some statistical methods which gained popularity include the Expectation–Maximi zation (EM) algorithm proposed by Dempster, Laird & Rubin [8], the probabilistic EM algorithm proposed by Celeux & Diebolt [6] and Nielsen [25], Levine & Casella [20] who proposed the stochastic versions of the EM algorithm which is continuous and works for univariate data.

This way, computational intelligence methods are the most used for approximation. For instance, Priya & Kuppuswami [28] proposed a Bayesian GA for multivariate missing data including discrete features where they selected six benchmark datasets from the UCI repository to then remove five different percentages of data (10%-50%) using a MCAR strategy where selection of elite individuals is done by using the RMSE as fitness function and a Bayesian conditional probability whose results outperformed other three methods (mode, Bayes MaxPost and PropPost); Liu et al. [21] proposed to use the EM algorithm together with a Gaussian Bayes information criterion for estimating multiple missed observations in a wind multiple time series problem where a MAR strategy was used to remove six different percentages of data (5%-30%) and evaluated with RMSE and MAE which outperformed the average and k-nearest imputation methods; Abdella & Marwala [1] used neuro–GA method in a brewery database where a GA optimizes predictions given by two neural networks: multi-layer perceptron and radial basis function networks for up to five missing observations which were evaluated by its standard error where the radial basis outperformed the multi-layer perceptron; Wójtowic et al. [34] applied 8 interval methods to six diagnostic models (two score and four regression methods) to ovarian tumor diagnosis with 20 features for different removed/missing data levels (0%-50%) whose goal is to improve accuracy of decisiveness of malignant cases and the obtained results outperformed classical diagnosis techniques. The main features of these works is that they solve supervised problems i.e. missing data are removed data indeed and they use a single-goal (usually RSME).

Multivariate missing data problems were addressed by Sovilj et al. [31] who applied extreme machine learning combined to a regression method initialized in two ways: conditional mean and multiple imputation, both based in mixture Gaussian distributions evaluated with its squared error over six different percentages of data (5%-30%) in six different UCI and LIACC repositories (no comparison to other approaches were done), Mesquita et al. [24] used a mix of expected Euclidean distance and the minimal machine learning method (which implicitly tries to preserve the variance of every variable via Gaussian kernels) which outperformed the conditional mean imputation and expected squared distance methods evaluated with RMSE and relative success rate measures over five benchmark datasets taken from the UCI repository for seven different percentages of removed data (10%–70%), Mesquita et al. [23] proposed to use Gaussian kernel neural network to 11 UCI databases for six missing data percentages (10, 30, 50, 60 and 70%) evaluated with three different measures: RMSE, ARMSE, standard deviation and compared to four methods: Incomplete case k-nearest-neighbors imputation algorithm, singular value thresholding, conditional mean imputation and expected square distance method with mixed results in its performance, Lai et al. [18] used an auto–encoder multi–task learning model (a Gaussian mixture neural network model with a single hidden layer) to classify incomplete datasets having interdependencies among features with six different percentages of removed data (5%-30%) over 7 datasets from the UCI and KEEL repositories which outperformed other six methods: Mean imputation, hot deck imputation, K-nearest neighbors, self-organizing map, multi-layer perceptron and MLP-based multi-task learning classification methods evaluated using RMSE, MAPE and accuracy measures, and Huang et al. [13] used decision trees with a mutual information measure over six different benchmark datasets taken from the UCI repository where five different percentages of data (10%-50%) were removed, evaluated using precision and recall measures and compared to four algorithms: Multi-granulation ensemble classification, mean imputation, k-nearest neighbors and a class center based approach which were outperformed by the proposed method; all these approaches assume a distribution/kernel, solve a single goal (usually RSME) over supervised datasets and they cannot impute/estimate discrete/binary data so there is a need for solving unsupervised problems with mixed discrete/binary/continuous data while preserving its multivariate properties. In contrast to the above-mentioned works, our approach deals with unsupervised continuous/discrete/binary multivariate data while minimizing a new multiobjective fitness function intended to preserve the original means, variances, covariances and skewness.

2. Basic concepts and notations

First, some definitions and notations are provided (we prefer vector/matrix representations in order to simplify the analysis and encoding). A data matrix $\boldsymbol{X}$ is composed by $i \in \mathbb{N}_n$ individuals and $j \in \mathbb{N}_p$ variables, so every element x_{ij} is called an

observation/sample of the variable i for the individual j. The means vector of $\boldsymbol{X}$ is denoted as $\bar{\boldsymbol{x}}$, the sample covariance matrix is denoted as $\boldsymbol{S}$, the sample correlation matrix is denoted as $\boldsymbol{R}$ and the sample skewness vector is denoted as $\boldsymbol{b}$.

2.1. Similarity between two matrices

The concept of similarity between two matrices $\boldsymbol{A},\boldsymbol{B}$ has different points of view, so we focus on a distance based approach to detect differences between $\boldsymbol{A},\boldsymbol{B}$. To do so, we first define the Minkowski r–norm (or distance) between two matrices$\boldsymbol{A},\boldsymbol{B}$ as follows:

Definition 1. Let $\boldsymbol{A}$ and $\boldsymbol{B}$ two $n \times p$ matrices and $r \in \mathbb{N}$, the Minkowski r–norm between $\boldsymbol{A},\boldsymbol{B}$ namely $\mathscr{D}_r(\boldsymbol{A},\boldsymbol{B})$ (or $\|\boldsymbol{A},\boldsymbol{B}\|_r$) is:

$$\mathscr{D}_r(\boldsymbol{A},\boldsymbol{B}) = \left(\sum_i \sum_j |a_{ij} - b_{ij}|^r\right)^{1/r} \tag{1}$$

for any $r \geqslant 1$.

If $p = 1$ then $\mathscr{D}_1(\boldsymbol{A},\boldsymbol{B})$is equivalent to the Frobenius distance between $\boldsymbol{A},\boldsymbol{B}$ namely $\mathscr{D}_{\mathscr{F}}(\boldsymbol{A},\boldsymbol{B})$. From a statistical point of view, it is important to ensure that a solution is to be similar to available data i.e. similar means/covariances/skewness which can be summarized as follows:

Definition 2. Let $\boldsymbol{X},\boldsymbol{Y}$ be two multivariate matrices. It is said that $\boldsymbol{X},\boldsymbol{Y}$ are similar if the distance between their means/covariances/correlations is minimal i.e.

$$\mathscr{D}_r\left(\bar{\boldsymbol{x}},\bar{\boldsymbol{y}}\right) \to 0 \tag{2}$$

$$\mathscr{D}_r(\boldsymbol{S_X},\boldsymbol{S_Y}) \to 0 \tag{3}$$

$$\mathscr{D}_r(\boldsymbol{b_X},\boldsymbol{b_Y}) \to 0 \tag{4}$$

for any $r \geqslant 1$.

This similarity concept is based on the idea that two matrices $\boldsymbol{X},\boldsymbol{Y}$ are similar if they are statistically similar i.e. means/-covariances/skewness. Its deterministic counterpart states that two matrices $\boldsymbol{X},\boldsymbol{Y}$ are similar if $\exists\left(\boldsymbol{Q}\,|\,\boldsymbol{X} = \boldsymbol{Q}^{-1}\boldsymbol{Y}\boldsymbol{Q}\right)$ which is an eigenvector–based definition (hard to find in random variables).

3. A genetic algorithm to impute missing data in multivariate matrices

The multiple-multivariate missing data problem is defined as the *absence of multiple observations in different variables and individuals*. It leads to estimation and reliability problems since most of statistical methods assume that *samples do not contain missing observations*, so $\boldsymbol{S}$ and $\boldsymbol{R}$ are affected by missing values since both require a complete dataset $\boldsymbol{X}$. Two important aspects to be considered in multivariate missing data are: *i) how to preserve matrix properties i.e. means, covariances and correlations* and *ii) how to handle discrete/binary variables* since most of available methods based on continuous models/methods do not preserve such properties.

3.1. Complexity of the problem

A *solution* of the problem is a combination of values that being replaced into the missed positions of the original matrix $\boldsymbol{X}$ shall complete it, so in this paper an optimal solution is such solution which preserves available means/covariances/skewness of $\boldsymbol{X}$ the best. The complexity of finding an optimal solution depends on three main features: *a) the nature of variables (continuous/discrete/binary), b) matrix properties to be preserved*, and *c) sample size and the amount of missing values*. It is well known that handling discrete data leads to NP-Hard computations, so the more missing values the wider the search space to explore which in practice means that the set of possible solutions exponentially increases with the amount of missed values, so we can say that at least it is NP-hard.

Another way to see complexity is by analyzing the interaction among variables of the problem. Replacing few missed observations with the average of available data in a single feature vector i.e. univariate data can be reasonable since it does not change its mean and slightly modifies its variance after replacing them while multivariate problems not only have to satisfy individual averages/variances but covariances/correlations where the difference of units of every variable implies that slight changes in a single value could significantly change covariance/correlation matrices after replacing missed observations.

This way, imputation/estimation methods for univariate problems have no guarantee to keep multivariate data properties, continuous optimization methods cannot handle discrete/binary variables and the available methods hardly work with second/higher order statistics like covariances/skewness etc., so a suitable way to approximate an optimal solution is by using GAs due to its flexibility/efficiency to solve multiobjective problems with discrete/continuous data and easiness to implement.

3.2. The proposed genetic algorithm - MIGA

A GA is a general heuristic population-based search approach for solving both constrained and unconstrained optimization problems where a population of candidate solutions is modified and selected in iterative search for optimal fitness function values by mimicking biological parents/offspring evolution based on three main features: *selection* via a fitness/goal function, *mutation* and *crossover* in order to evolve solutions to get an appropriate solution. The multivariate multiple missing data problem is *multiobjective* since its means, covariances and skewness have non additive units, so we propose to use dimensionless transformations for its means, covariances and skewness in order to make them additive into a single goal/-fitness function instead. All measures/notations required to operate the proposed MIGA are shown next.

Definition 3. Let us denote y_{ij} as a missed observation indexed by (i,j) into $\boldsymbol{X}$ and let $\boldsymbol{M}$ be the index vector of all j–ordered missed observations of $\boldsymbol{X}$. The matrix $\boldsymbol{X_A}$ contains the complete individuals of $\boldsymbol{X}$ and the matrix $\boldsymbol{X_C}$ contains the individuals of $\boldsymbol{X}$ with at least one missing observation.

Example 1. Let us consider an incomplete matrix $\boldsymbol{X}$ where the matrices $\boldsymbol{X_A}, \boldsymbol{X_C}$ are as follows:

$$\boldsymbol{X} = \begin{bmatrix} 38.09 & -- & 3 \\ 36.38 & 47 & -- \\ -- & 50 & -- \\ 40.5 & 50 & 0 \\ 35.32 & 47 & 4 \\ 36 & -- & -- \end{bmatrix}; \boldsymbol{X_A} = \begin{bmatrix} 40.5 & 50 & 0 \\ 35.32 & 47 & 4 \end{bmatrix}; \boldsymbol{X_C} = \begin{bmatrix} 38.09 & -- & 3 \\ 36.38 & 47 & -- \\ -- & 50 & -- \\ 36 & -- & -- \end{bmatrix}$$

and the index vector $\boldsymbol{M}$ of $\boldsymbol{X_C}$ in $\boldsymbol{X}$ is $\boldsymbol{M} = \{(3,1),(1,2),(5,2),(2,3),(3,3),(5,3)\}$.

Then $\boldsymbol{X_C}$ is composed by the existent values $\{x_{11}, x_{13}, x_{21}, x_{22}, x_{32}, x_{51}\}$ and a set of missed observations $\{y_{12}, y_{23}, y_{31}, y_{33}, y_{52}, y_{53}\}$ indexed by $\boldsymbol{M}$. For the sake of understanding, the matrix $\boldsymbol{X_A}$ contains the available complete individuals i.e. the original matrix $\boldsymbol{X}$ from which the matrix $\boldsymbol{X_C}$ of missing individuals is removed; $\boldsymbol{X_C}$ contains the y_{ij} missed observations.

3.3. Fitness function

Unlike most of the referred works (see Section 1.1) our main goal is to find a set of k values to replace the missing observations without modifying three statistical matrix properties: $\bar{\boldsymbol{x}}, \boldsymbol{S}$ and $\boldsymbol{b}$ (computed from $\boldsymbol{X_A}$ and $\boldsymbol{X_C}$), so it is basically a *multiobjective problem* of preserving matrix properties after replacing missing values which in our case is the problem of solving three goals: similar means, covariances (correlations as consequence) and skewness between available and completed data. To do so, it is convenient *to comprise all goals into a single goal/function in order to reduce the complexity of finding a solution and to avoid local optima* so we need to find an appropriate representation for $\bar{\boldsymbol{x}}, \boldsymbol{S}$ and $\boldsymbol{b}$ since $\bar{\boldsymbol{x}}$ has the units of $\boldsymbol{X}, \boldsymbol{S}$ has squared units and $\boldsymbol{b}$ is a third power measure. This way, we standardize $\bar{\boldsymbol{x}}, \boldsymbol{S}$ and $\boldsymbol{b}$ to then represent the multiobjective problem using a single dimensionless fitness function, as shown as follows.

Definition 4. Let $\bar{\boldsymbol{x}}_A$ be the vector of means of the available data matrix $\boldsymbol{X_A}$, $\bar{\boldsymbol{x}}_C$ be the vector of means of the completed available data matrix $\boldsymbol{X_C}$, and $\boldsymbol{S_A}$, $\boldsymbol{S_C}$ be the covariance matrices of available/completed data, n_A, n_C be the amount of individuals of $\boldsymbol{X_A}$, $\boldsymbol{X_C}$ and $\nu_A = n_A - 1,\ \nu_C = n_C - 1,\ \nu_T = n_A + n_C - 2$ be their degrees of freedom. Under the hypothesis of equality on means $H_0 : \bar{\boldsymbol{x}}_A = \bar{\boldsymbol{x}}_C$ we define The standardized vector of means namely $\tilde{\boldsymbol{t}}_A$ and $\tilde{\boldsymbol{t}}_C$ are defined as follows:

$$\tilde{x}_A = \frac{\bar{x}_A}{S_p}, \tag{5}$$

$$\tilde{x}_C = \frac{\bar{x}_C}{S_p}, \tag{6}$$

$$S_p^2 = \frac{dg(S_A)\cdot \nu_A + dg(S_C)\cdot \nu_C}{\nu_T}, \tag{7}$$

where $dg(S_A)$, $dg(S_C)$ are the diagonal matrices of S_A, S_C and S_p^2 is a diagonal matrix composed by the pooled variances of X_A, X_C and $\tilde{x} = 0$ only iff $\bar{x}_A = \bar{x}_C$. Under the hypothesis of equality on covariances $H_0 : S_A = S_C$ we have that $S_A^{-1/2} S_C S_A^{-1/2} = I$, so the matrix of pooled covariances $\widetilde{S} : S_A \times S_C \to \mathbb{R}^{m\times n}$ is defined as follows:

$$\widetilde{S} = S_A^{-1/2} S_C S_A^{-1/2} \tag{8}$$

where $\widetilde{S} - I = 0$ only iff $S_A = S_C$. Under the hypothesis of equality on skewness $H_0 : b_A = b_C$ we can define a vector of difference between skewness $\tilde{b} : b_A \times b_C \to \mathbb{R}^n$ as follows:

$$\tilde{b} = b_A - b_C \tag{9}$$

where $\tilde{b} = 0$ only iff $b_A = b_C$.

It is clear that $\widetilde{x}_A,\ \widetilde{x}_C,\ \widetilde{S},\ I,\ b_A,\ b_C$ are dimensionless which helps to compute distances among them. To obtain a matrix X_C similar to X_A is a combinatorial multi–objective problem and it leads to bigger challenges than classical missing data estimation/imputation. To do so, we define the goals of the GA using the distances shown in Eqs. (2)–(4) (see Definition 2):

$$G_1: \quad \min \mathscr{D}_r\left(\bar{x}_A, \bar{x}_C\right) = \min\left(\sum_j |\bar{x}_{j_A} - \bar{x}_{j_C}|^r\right)^{1/r}$$

$$G_2: \quad \min \mathscr{D}_r\left(\widetilde{S}, I\right) = \min\left(\sum_i \sum_j |\widetilde{s}_{ij} - \delta_{ij}|^r\right)^{1/r}$$

$$G_3: \quad \min \mathscr{D}_r(b_A, b_C) = \min\left(\sum_j |b_{jA} - b_{jC}|^r\right)^{1/r}$$

In order to make G_1, G_2 and G_3 additive, we use $\widetilde{x}_A,\ \widetilde{x}_C,\ \widetilde{S},\ b_A$ and b_C (see Eqs. (5), (6), (8) and (9)))) to obtain a dimensionless measure (there is no need to compute $R_A,\ R_C$ since they come from S_A and S_C). Then, we propose the following single fitness function:

Definition 5. Let $\tilde{x}$ be the relative means vector, $\widetilde{S}$ be the relative covariance matrix and b be the skewness vector of a matrix X. A **fitness/goal function** $\mathscr{F}_r$ that minimizes the Minkowsky r–norm between the relative means/covariances and skewness of available/completed data is defined as follows:

$$\mathscr{F}_r := \min\left(\mathscr{D}_r(\tilde{x}_A, \tilde{x}_C) + \mathscr{D}_r\left(\widetilde{S}, I\right) + \mathscr{D}_r(b_A, b_C)\right) \tag{10}$$

where I is the identity.

To understand $\mathscr{F}_r$, we provide the following example.

Example 2. Let us define $\mathscr{F}_r$ for $p = 2$ variables and $r = \infty$. $\bar{x}_A,\ \bar{x}_C,\ S_A,\ S_C,\ b_A$ and b_C are as follows:

$$\bar{x}_A = \begin{bmatrix} \widetilde{x}_{1A} & \widetilde{x}_{2A} \end{bmatrix};\ \bar{x}_C = \begin{bmatrix} \widetilde{x}_{1C} & \widetilde{x}_{2C} \end{bmatrix}$$

$$\widetilde{S} = S_A^{-1/2} S_C S_A^{-1/2} = \begin{bmatrix} \widetilde{s}_{11} & \widetilde{s}_{12} \\ \widetilde{s}_{21} & \widetilde{s}_{22} \end{bmatrix};\ I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$b_A = [\, b_{1A} \quad b_{2A} \,];\ b_C = [\, b_{1C} \quad b_{2C} \,]$$

then $\mathscr{F}_\infty$ is as follows:

$$\mathscr{F}_\infty := \min \max\left\{\left|\widetilde{x}_{1A} - \widetilde{x}_{1C}\right|, \left|\widetilde{x}_{2A} - \widetilde{x}_{2C}\right|\right\} + \max\left\{\left|\widetilde{s}_{11} - 1\right|, \left|\widetilde{s}_{12} - 0\right|, \left|\widetilde{s}_{21} - 0\right|, \left|\widetilde{s}_{22} - 1\right|\right\} + \max\left\{|b_{1A} - b_{1C}|, |b_{2A} - b_{2C}|\right\} \tag{11}$$

All elements of $\mathscr{F}_r$ are dimensionless which is convenient for optimization. For instance, the first element of Eq. (11) $|\widetilde{x}_{1A} - \widetilde{x}_{1C}|$ is intended to minimize $\bar{x}_{1A} - \bar{x}_{1C}$ so if $\bar{x}_{1C} \to \bar{x}_{1A}$ then $|\widetilde{x}_{1A} - \widetilde{x}_{1C}| \to 0$ (same idea behind the second third terms). Note that $\mathscr{F}_r$ has a theoretical minimum at zero only if there is a perfect match among $\bar{x}_A,\ S_A,\ b_A$ and $\bar{x}_C,\ S_C,\ b_C$ i.e.

$$\bar{x}_A = \bar{x}_C,\ S_A = S_C,\ b_A = b_C \iff \mathscr{F}_r = 0$$

Roughly speaking, $\mathscr{F}_r$ is designed to preserve the structure of available data as follows:

- $\mathscr{D}_r(\tilde{x}_A, \tilde{x}_C)$ to keep equal means (equivalent to preserve the central trend of each variable)
- $\mathscr{D}_r\left(\widetilde{S}, I\right)$ to preserve the original covariances/correlations of X_A
- $\mathscr{D}_r(b_A, b_C)$ to preserve the original skewness of X_A

3.4. Population, operators and stopping criteria

An **individual** is defined as a vector p of missing observations indexed by M where every position (i,j) is a **gene** that represents the missing observation y_{ij}. A set of multiple individuals conforms a **population** namely P which is a $l \times k$ matrix where $k = n(M)$. Then, the i_{th} vector of size k of P is called to be the i_{th} individual p_i of a population whose genes are indexed by M i.e. the j-ordered location of the missing observations y_{ij}. The elements of an individual are not ordered by individuals but by variable which actually helps to subdivide p_i into consecutive genes belonging to the same variable.

This way, there are at most l subsets of i consecutive individuals of the same variable j. Every gene of the j_{th} subset of p_l belongs to the same variable j which is generated with a **random variable generator** R_j, so it is desirable R_j to keep the statistical properties of each variable via goodness of fit analysis to select the best generator per variable j (see Law & Kelton [19] and Devroye [9]).

The **mutation** operator selects a random individual among the best c_1 individuals of the population P^g for the g generation then choose a random gene and replace it with a new one, then repeat it c_3 times per individual. The **crossover** operator selects the best c_2 individuals of the population P^g for the g generation then choose a random feature $j\prime \leqslant p$ among the p variables and switch the genes corresponding to $y_{ij\prime}$ from the best individual to the next individual then repeat it for the $c_2 - 1$ remaining individuals. Example 3 shows the main idea behind the crossover operator.

Example 3. Let us go back to Example 1. Consider $p_1,\ p_2$ as the best individuals of a given population, suppose $j' = 2$ and $c_2 = 2$, then the crossover operator for this configuration is as follows (Fig. 1):

Note that $j\prime = 2$ corresponds to all missing observations of the variable $j = 2$ in the original matrix X i.e. $y_{12},\ y_{52}$. This crossover strategy provides $2(c_2 - 1)$ new individuals to the next population P^{g+1}.

It is not recommended to use large values for mutation/crossover parameters $c_1, c_2 > 5$ and $c_3 > 10$ since they increase computing time and a large amount of elite individuals $c > 5$ could lead to local optima.

The **evolution** strategy is *elitism* in which the best $c \in \mathbb{Z}$ individuals $p_1, \cdots, p_c$ (ordered via $\mathscr{F}_r$) of every generation g are preserved for the next generation P^{g+1}. Now, the overall solution of the problem is the best individual p_q^* of the last generation P^G for the q_{th} run i.e. $p^* := \left\{p_q^* \,|\, \min_q \mathscr{F}_r\left(p_q^*\right)\right\}$.

The **diversity** strategy is to replace the $(l - c - c_1c_3 - 2(c_2 - 1))$ worst individuals of P^g with new random genes R_j in the next population P^{g+1}. Any $P^g, g > 1$ must be larger than the individuals generated by elitism, mutation and crossover i.e. $l > (c + c_1c_3 + 2(c_2 - 1))$. A predefined amount of **runs** $Q \in \mathbb{Z}$ also helps diversity, so we recommend to perform $Q \geqslant 6$ runs (see Law & Kelton [19] and Devroye [9]).

The **stopping criteria** selected this paper is to select a maximum number of iterations $\max_g = G$ where P^g is the population of the g_{th} generation. Figueroa et al. [11] pointed out that small sizes of l do not cover well the search space, so it is recommended to set $100 \leqslant l \leqslant 1000$ and $G \geqslant 2l$.

$$\boldsymbol{p}_1 = [\, p_{11}, \overbrace{p_{12}, p_{13},}^{j'=2} p_{14}, p_{15}, p_{16} \,]$$

$$\boldsymbol{p}_2 = [\, p_{21}, \overbrace{p_{22}, p_{23},}^{j'=2} p_{24}, p_{25}, p_{26} \,]$$

$$\boldsymbol{p}_3 = [\, p_{11}, \overbrace{\boxed{p_{22}, p_{23},}}^{\text{swap}} \, p_{14}, p_{15}, p_{16} \,]$$

$$\boldsymbol{p}_4 = [\, p_{21}, \overbrace{\boxed{p_{12}, p_{13},}}^{\text{swap}} \, p_{24}, p_{25}, p_{26} \,]$$

Fig. 1. Crossover operator.

3.5. Pseudocode and flowchart of MIGA

The pseudocode of MIGA is summarized in Algorithm 1.

Algorithm 1: Multiple imputation genetic algorithm - MIGA

Require: $X, X_A, M, l, G, c, c_1, c_2, c_3, q, r$
  COMPUTE $\bar{x}_A$, S_A, b_A, $\tilde{x}_A$ from the original dataset
  for $q : 1 \rightarrow Q$ **do**
    for $g : 1 \rightarrow G$ **do**
      INITIALIZE a population P^g
      EVALUATE the fitness function $\mathscr{F}_r$ for every individual l
      SELECT the c best individuals in ascending order from $\mathscr{F}_r$
      COMPUTE MUTATION, CROSSOVER and DIVERSITY
      COMPUTE $\mathscr{F}_r$ for the new individuals
      PRESERVE the best estimation per generation $p_g^* := \{p_l \,|\, \min_l \mathscr{F}_r(p_l)\}$
    end for
    return the best estimation per run $p_q^* := \left\{p_g \,|\, \min_g \mathscr{F}_r\left(p_g\right)\right\}$
  end for
  return the best overall estimation $p^* := \left\{p_q \,|\, \min_q \mathscr{F}_r\left(p_q\right)\right\}$

MIGA can also be explained as the flowchart displayed in Fig. 2.

3.6. Complexity of the algorithm

The time computational complexity of the MIGA depends on three components: fitness function, population size and operators. The most expensive part of the GA is to compute the fitness function which is composed of three parts: differences among means $O(np)$, differences among skewness $O(np)$ and complexity of difference between covariance matrices $O(2np \cdot \min\{n, p\})$ which comes from multiplying $S_A \times S_A$ and $S_C \times S_C$ respectively (some algorithms implemented in MatLab and Python have complexity of $O(np^2)$ per matrix), all of them per generation $g \in G$. The complexity of ordering elitist GAs is $O(l \cdot \log l)$ in average, and the complexity of mutation, crossover and diversity depends on population size $O(lk)$. Computations are done per generation G, so the time complexity T of the GA per run is polynomial i.e.

$$T = O(2npG) + O(2(np \cdot \min\{n, p\})G) + O(G(l \cdot \log l)) + O(Glk)$$

since either $2(n^2p)G \leqslant 2(np^2)G$ or $2(np^2)G \leqslant 2(n^2p)G$ and $Gl < G(l \cdot \log l) < Gl^2$.

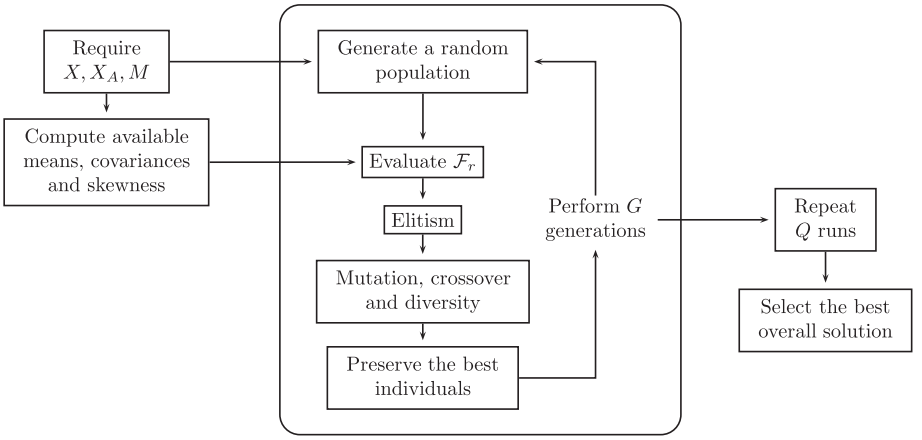

Fig. 2. Flowchart of the proposed MIGA.

4. Application example

This example is a multivariate problem of 311 individuals and $p = 6$ variables for a total of 1866 observations (the full dataset is available at https://comunidad.udistrital.edu.co/lamic/tools/ labeled as Dataset 1). A total of $k = 199$ missing observations ($\sim 10\%$) were found on 131 individuals in different variables, so only 180 out of 311 individuals are complete ($\sim 58\%$). Three out of six variables (2,3,6) are discrete which increases the complexity of the problem. The available means/-covariance/correlation matrices $\bar{x}_A, S_A, b_A$ are shown next, and other useful statistics are shown next:

$$\bar{x}_A = [36.002 \quad 43.928 \quad 6.672 \quad 35.438 \quad 2.401 \quad 1.506]$$

$$S_A = \begin{bmatrix} 5.260 & -1.463 & -7.800 & 7.314 & 0.338 & -0.018 \\ -1.463 & 104.012 & 15.602 & -5.894 & -1.232 & -3.662 \\ -7.800 & 15.602 & 24.389 & -12.024 & -1.302 & -0.593 \\ 7.314 & -5.894 & -12.024 & 19.208 & 0.661 & -0.666 \\ 0.338 & -1.232 & -1.302 & 0.661 & 3.465 & 0.214 \\ -0.018 & -3.662 & -0.593 & -0.666 & 0.214 & 1.693 \end{bmatrix}$$

$$b_A = [0.772 \quad -0.280 \quad 0.672 \quad -0.431 \quad 1.385 \quad -0.066]$$

The determinants $|S_A| = 268650, |R_A| = 0.1787$ give an idea of the size of each matrix and will be used to test the obtained results. Now, three methods were selected for missing data imputation/estimation: the MIGA, the Expectation Maximization (EM) algorithm and Auxiliary Regressions (AR) which are well known statistical methods. the obtained results are shown as follows.

4.1. Results of the GA

The parameters of the GA that obtained the best results are:

- $r = \infty, p = 6, k = 199, n_A = 180, n_C = 131, v_A = 179, v_C = 130, v_T = 329, l = 1000, G = 2000,$ $c = 3, c_1 = 3, c_2 = 3, c_3 = 10, Q = 12$
- Random variable generators R_j (obtained from samples):

R_1: Normal dist. with $\mu = 36.002,\ \delta = 5.260$
R_2: Discrete uniform dist. with $a = 14,\ b = 64$
R_3: Discrete uniform dist. with $a = 0,\ b = 25$
R_4: Normal dist. with $\mu = 35.438,\ \delta = 4.382$
R_5: Exponential dist. with $\theta = 2.401$
R_6: Poisson dist. with $\lambda = 1.506$

A value $r = \infty$ is selected because it shows the most consistent results i.e. the GA selects the individual with the minimum of the biggest differences among available and imputed means, covariances and asymmetries (other values of r tend to compensate biggest differences with smaller ones), and $Q = 6$ runs were done to improve the search. The obtained results for $r = \infty$, $\mathscr{D}_\infty$ are as follows.

$$\mathscr{F}_\infty = 0.2110;\ \mathscr{D}_\infty\left(\tilde{\bar{x}}_A, \tilde{\bar{x}}_C\right) = 0.0163;\ \mathscr{D}_\infty\left(\tilde{S}, I\right) = 0.0796;\ \mathscr{D}_\infty(b_A, b_C) = 0.1151$$

Skewness is also incorporated into the analysis in order to obtain a better fit between original and imputed distributions since it is also desirable to keep original cumulative distributions (see Fig. B.4 in Appendix B). The obtained $\bar{x}_C$, $\bar{S}_C$ and $\bar{b}_C$ of the GA are shown next:

$$\bar{x}_C = [36.088 \quad 44.230 \quad 6.603 \quad 35.337 \quad 2.359 \quad 1.527]$$

$$S_C = \begin{bmatrix} 5.326 & -1.232 & -7.645 & 7.178 & 0.236 & 0.023 \\ -1.232 & 104.763 & 15.261 & -5.682 & -1.255 & -3.691 \\ -7.645 & 15.261 & 24.487 & -11.692 & -1.167 & -0.543 \\ 7.178 & -5.682 & -11.692 & 19.353 & 0.804 & -0.624 \\ 0.236 & -1.255 & -1.167 & 0.804 & 3.568 & 0.257 \\ 0.023 & -3.691 & -0.543 & -0.624 & 0.257 & 1.790 \end{bmatrix}$$

$$b_C = [0.657 \quad -0.170 \quad 0.780 \quad -0.546 \quad 1.355 \quad 0.0394]$$

Its determinants are $|S_C| = 340409.65, |R_C| = 0.2016$ and the best solution p^* indexed by $(i,j) \in M$ is shown in Table A.8 (see Appendix A). Fig. (3) shows $\mathscr{F}_\infty$, $\mathscr{D}_\infty\left(\tilde{x}_A, \tilde{x}_C\right)$, $\mathscr{D}_\infty\left(\tilde{S}, I\right)$ and $\mathscr{D}_\infty(b_A, b_C)$ for $G = 2000$ generations.

4.2. Results of the EM algorithm

The EM algorithm uses the conditional expectations of a set of auxiliary variables to provide an estimate of missing observations whose main goal is to maximize a Likelihood or Log-Likelihood function of the sample distribution, obtaining an optimal estimation of the missing observations (see Dempster et al. [8]) (it is a popular method since the likelihood function is monotonically increasing). A major drawback of the EM algorithm is that it must need at least one complete auxiliary variable in order to compute conditional expectations. If this vector is either not available or complete, the EM algorithm will replace all missing values with $S(\theta, \theta_{k-1}) = E_{\theta_{k-1}}\{\log f(X;\theta)|X = y\} = E_{\theta_{k-1}}(Y)$. It is a strong restriction to multiple missing data because many cases have multiple missing observations in different variables. Another major drawback is that it is sensible to deviations from normality (which is our case since we have discrete variables). The main results of the EM algorithm are shown next:

$$\bar{x}_C = [36.112 \quad 44.312 \quad 6.112 \quad 35.685 \quad 2.348 \quad 1.643]$$

$$S_C = \begin{bmatrix} 3.155 & -0.773 & -5.212 & 5.075 & 0.282 & -0.278 \\ -0.773 & 72.060 & 8.442 & -3.750 & -0.238 & -4.302 \\ -5.211 & 8.442 & 14.818 & -8.090 & -1.034 & 0.091 \\ 5.075 & -3.750 & -8.090 & 15.388 & 0.503 & -1.042 \\ 0.282 & -0.238 & -1.034 & 0.503 & 2.479 & 0.358 \\ -0.278 & -4.302 & 0.091 & -1.042 & 0.358 & 1.322 \end{bmatrix}$$

$$b_C = [0.332 \quad -0.001 \quad 0.821 \quad -1.207 \quad 1.693 \quad -0.202]$$

The obtained determinants are $|S_C| = 18762,\ |R_C| = 0.1105$. The results of evaluating the distances for $r = \infty$, $\mathscr{D}_\infty$ are:

$$\mathscr{F}_\infty = 1.3862;\ \mathscr{D}_\infty\left(\tilde{x}_A, \tilde{x}_C\right) = 0.0883;\ \mathscr{D}_\infty\left(\tilde{S}, I\right) = 0.5228;\ \mathscr{D}_\infty(b_A, b_C) = 0.7751$$

4.3. Results of auxiliary regressions

Another popular approach is the auxiliary regressions method which uses a set of complete covariates to estimate missing data using a linear regression model. The main drawback of this technique is that there are no complete covariates to estimate missing data, in fact, all variables have missing observations. To do so, we use available data X_A to obtain a linear model that estimates missing observations, so only the mean and variance of available data are preserved (in theory there is no guarantee of preserving the covariance structure). The main results of auxiliary regressions are shown next:

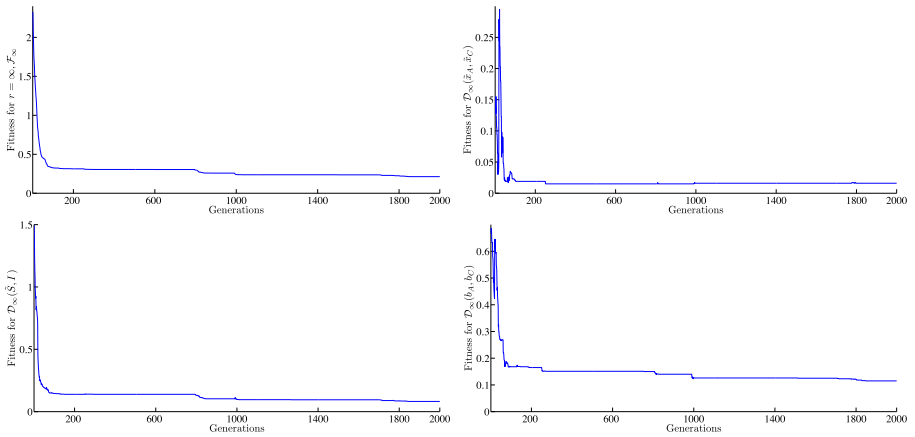

Fig. 3. Fitness function values for $\mathscr{F}_\infty$, $D_\infty\left(\tilde{x}_A, \tilde{x}_C\right)$, $\mathscr{D}_\infty\left(\tilde{S}, I\right)$ and $\mathscr{D}_\infty(b_A, b_C)$.

$$\bar{x}_C = [36.216 \quad 44.525 \quad 5.949 \quad 35.820 \quad 2.465 \quad 1.596]$$

$$S_C = \begin{bmatrix} 3.796 & 0.099 & -5.804 & 5.594 & 0.230 & -0.491 \\ 0.099 & 86.209 & 5.817 & -1.851 & -2.129 & -4.339 \\ -5.804 & 5.817 & 18.459 & -8.224 & -0.152 & 0.552 \\ 5.594 & -1.851 & -8.224 & 18.075 & 0.461 & -1.432 \\ 0.230 & -2.129 & -0.152 & 0.461 & 3.688 & 0.511 \\ -0.491 & -4.339 & 0.552 & -1.432 & 0.511 & 1.754 \end{bmatrix}$$

$$b_C = [0.622 \quad 0.171 \quad 0.639 \quad 0.129 \quad 1.595 \quad -0.371]$$

The obtained determinants are $|S_C| = 142898.86,\ |R_C| = 0.2023$. The results of evaluating the distances for $r = \infty,\ \mathscr{D}_\infty$ are:

$$\mathscr{F}_\infty = 0.9117;\ \mathscr{D}_\infty\left(\tilde{\bar{x}}_A, \tilde{\bar{x}}_C\right) = 0.1545;\ \mathscr{D}_\infty\left(\tilde{S}, I\right) = 0.2711;\ \mathscr{D}_\infty(b_A, b_C) = 0.4861$$

4.4. Output Analysis: Multivariate Tests on Means and Variances

To detect differences between available X_A and completed X_C data we perform statistical tests on means (based in the t–student test), variances (based on the f test), covariances (based on a χ^2 and the f distributions), correlations (based on the χ^2 distribution) and goodness of fit tests for two samples (based on the Kolmogorov–Smirnov and Cucconi tests). The main idea is to statistically contrast if the imputed data preserves the statistical properties of the available complete dataset X_A.

Test on means The null/alternate hypotheses to test differences on means are:

$$H_0: \ \bar{x}_A = \bar{x}_C$$
$$H_1: \ \bar{x}_A \neq \bar{x}_C$$

which can be checked using any test like the Welch, Brown-Forsythe or t–student test for means. To perform the t–student test over individual means (from the same population), we use the test statistic $\tilde{t}_0$.

Definition 6. Let $\bar{x}_A$ be the vector of means of the available data matrix X_A, $\bar{x}_C$ be the vector of means of the completed available data matrix X_C, S_A, S_C be the covariance matrices of available/completed data, $n_A,\ n_C$ be the amount of individuals of $X_A,\ X_C$ and $\nu_A = n_A - 1,\ \nu_C = n_C - 1,\ \nu_T = n_A + n_C - 2$ be their degrees of freedom. The standardized vector of means namely $\tilde{t}_A$ and $\tilde{t}_C$ are defined as follows:

$$\tilde{t}_0 := \frac{\bar{x}_A - \bar{x}_C}{S_p\left(\frac{1}{\nu_A} + \frac{1}{\nu_C}\right)^{1/2}} \tag{12}$$

$$S_p^2 := \frac{(\nu_A)dg(S_A) + (\nu_C)dg(S_C)}{\nu_T}, \tag{13}$$

where $dg(S_A)$, $dg(S_C)$ are the diagonal matrices of $S_A,\ S_C$ and S_p^2 is a diagonal matrix composed by the *pooled* variances of $X_A,\ X_C$.

Then we can use (12) and (13) to test the hypothesis using a t–student distribution:

$$t_0 = \tilde{t}_{j_A} - \tilde{t}_{j_C} \to t_{\nu_A;\nu_C}$$

Test on individual variances For individual variances, the classical Fisher f test uses the following hypothesis:

$$H_0: \ s^2_{ij_A} = s^2_C$$
$$H_1: \ s^2_{ij_C} \neq s^2_C$$

The test statistic f_0 is:

$$f_0 = \frac{s^2_{ij_A}}{s^2_{ij_A}} \to f_{\nu_A;\nu_C}$$

Test on covariance matrices based on the χ^2 distribution To test equality of covariance matrices $S_A,\ S_C$, a Likelihood-ratio based test (see Timm [32], Box [4] and Anderson [3]) is used to contrast the equality on variance:

$$H_0: \ S_A = S_C$$
$$H_1: \ S_A \neq S_C$$

The test statistics ρ, η are based in the following variables:

$$S_t = \frac{v_A S_A + v_C S_A}{v} \tag{14}$$

$$\eta = v_T \ln|S_t| - (v_A \ln|S_A| + v_C \ln|S_C|) \tag{15}$$

where $v_A = n_A - 1,\ v_C = n_C - 1,\ v_T = n_A + n_C - 2$. Thus, ρ is defined as:

$$\rho = 1 - \frac{2p^2 + 3p - 1}{6(p+1)}\left(\frac{1}{v_A} + \frac{1}{v_C} - \frac{1}{v_T}\right) \tag{16}$$

where p is the amount of variables of X. Box [4,5] proposed the following approximate distribution for the likelihood ratio λ_1^*:

$$\rho\eta = -2\rho \ln \lambda_1^* \to \chi_w^2 \tag{17}$$

$$w = p(p+1)/2 \tag{18}$$

Test on covariance matrices based on the f distribution. A comparison test based on the Fisher f distribution is as follows:

$$\rho_0 = \frac{(p-1)(p+2)}{6}\left(\frac{1}{v_A^2} + \frac{1}{v_C^2} - \frac{1}{v_T^2}\right) \tag{19}$$

$$w_0 = \frac{(w+2)}{|\rho_0 - (1-\rho)^2|} \tag{20}$$

so if $\rho_0 - (1-\rho)^2 > 0$ then the test statistic f_1 is:

$$f_1 = \eta/a \to f_{w,w_0} \tag{21}$$

where $a = w/[\rho - (w/w_0)]$. If $\rho_0 - (1-\rho)^2 < 0$, then

$$f_1 = (w_0\eta)/(w(b-\eta)) \to f_{w,w_0} \tag{22}$$

where $b = w_0/(\rho + 2/w_0)$. For further information see Krishnaia & Lee [16].

Test on correlation matrices To test equality on correlations, Kullback [17] and Aitkin [2] proposed the following test:

$$H_0: \quad R_A = R_C$$
$$H_1: \quad R_A \neq R_C$$

The test statistic t_2 is based in the following variables:

$$t_2 = -2\log\frac{\left(|R_A|^{n_A/2}|R_C|^{n_C/2}\right)}{|\overline{R}|^{n/2}} \tag{23}$$

where $n = n_A + n_C,\ n\overline{R} = n_A R_A + n_C R_C$. Kullback [17] provided asymptotic values for $t_2 \to \chi^2_{p(p-1)}$. Other tests such as sphericity, linear structure and circularity are not tested since they are used to test linear models instead (see Timm [32] and Anderson [3]).

Kolmogorov–Smirnov goodness of fit test. The two sample Kolmogorov–Smirnov (KS) test (see Kolmogorov [15] and Smirnov [30]) verifies if X_A and X_C come from the same population:

$$H_0: \quad F(X_A) = F(X_C)$$
$$H_1: \quad F(X_A) \neq F(X_C)$$

The test statistic D_0 is computed as follows:

$$D_0 = \max_{\beta}|F_\beta(X_A) - F_\beta(X_C)| \to D_\alpha$$

where $\beta \in [0,1]$ is a percentile of $\hat{X} = [X_C, Y]$, $F_\beta(X_A)$, $F_\beta(X_C)$ are the cumulative values of X_A, X_C for β and D_α is the KS distribution value for an α confidence level which can be computed as follows.

$$D_\alpha = c(\alpha)\sqrt{\frac{n_A + n_C}{n_A \cdot n_C}}, c(\alpha) = \sqrt{-\ln(\alpha/2)\cdot 1/2}.$$

Cucconi goodness of fit test. The Cucconi goodness of fit (see Cucconi [7] and Nishino & Murakami [26]) is as follows:

$$H_0: \quad F(X_A) = F(X_C)$$
$$H_1: \quad F(X_A) \neq F(X_C)$$

The test statistic t_c is computed as follows:

$$t_c = \frac{Q_1^2 + Q_2^2 - 2\rho Q_1 Q_2}{2(1-\rho^2)}; Q_1 = \frac{\sum_j^{n_C} R_{1j}^2 - \frac{n_C(N+1)(2N+1)}{6}}{\sqrt{\frac{n_A n_C (N+1)(2N+1)(8N+11)}{180}}}; Q_2 = \frac{\sum_j^{n_C} (N+1-R_{1j})^2 - \frac{n_C(N+1)(2N+1)}{6}}{\sqrt{\frac{n_A n_C (N+1)(2N+1)(8N+11)}{180}}};$$

$$\rho = \frac{2(N^2-4)}{(2N+1)(8N+11)} - 1$$

where R_{1j} is the rank of the j_{th} smallest observation in the i_{th} sample in the pooled sample $[X_A, X_C]$ with $N = n_A + n_C$. If $t_c > -\ln(\alpha)$ then we reject the null hypotheses at an α significance level.

4.5. Tests over the genetic approach

We compute the values of all tests of the solution provided by the GA. The obtained results for the test on means are summarized in the vector t_0:

$$t_0 = [-0.326 \quad -0.258 \quad 0.122 \quad 0.201 \quad 0.198 \quad -0.134]$$

where the confidence interval for $\alpha = 0.05$ is $t_{0.05,179,130} = [-1.649, 1.649]$ which means that H_0 is statistically significant at a 0.05 confidence level. The obtained results for the individual variances test are summarized in the vector f_0

$$f_0 = [0.988 \quad 0.993 \quad 0.996 \quad 0.993 \quad 0.971 \quad 0.946]$$

where the confidence interval for $\alpha = 0.05$ is $f_{0.05,179,130} = [0.81, 1.25]$ which means that H_0 is statistically significant at a 0.05 confidence level. The obtained results for the χ^2 test over covariance matrices are summarized as follows:

$$S_t = \begin{bmatrix} 5.288 & -1.366 & -7.735 & 7.257 & 0.295 & -0.001 \\ -1.366 & 104.327 & 15.458 & -5.805 & -1.242 & -3.674 \\ -7.735 & 15.458 & 24.430 & -11.884 & -1.245 & -0.572 \\ 7.257 & -5.805 & -11.884 & 19.269 & 0.721 & -0.648 \\ 0.295 & -1.242 & -1.245 & 0.721 & 3.508 & 0.232 \\ -0.001 & -3.674 & -0.572 & -0.648 & 0.232 & 1.733 \end{bmatrix}$$

where $|S_t| = 270605.56$, $\rho = 0.9787$, $\eta = 0.9515$, $w = 21$ and the test statistic for $1-\alpha = 0.95$ is $\chi^2_{0.95,21} = 32.7$ which means that $\rho\eta = 0.9312 < \chi^2_{0.95,21}$ and H_0 is statistically significant at a 0.05 confidence level. The obtained results for the f test over covariance matrices are $\rho_0 = 0.0005$, $w_0 = 288050.66$, $a = 21.458$ and the test statistic for $1-\alpha = 0.95$ is $f_{0.95,21,288051} = 1.56$ which means that $f_1 = \eta/a = 0.0443 < f_{0.95,21,288051}$ and H_0 is statistically significant at a 0.05 confidence level, and the obtained results for the correlation matrices test are summarized as follows:

$$\overline{R} = \begin{bmatrix} 1 & -0.058 & -0.681 & 0.719 & 0.069 & 0.000 \\ -0.058 & 1 & 0.306 & -0.129 & -0.065 & -0.273 \\ -0.681 & 0.306 & 1 & -0.548 & -0.135 & -0.088 \\ 0.719 & -0.129 & -0.548 & 1 & 0.088 & -0.112 \\ 0.069 & -0.065 & -0.135 & 0.088 & 1 & 0.094 \\ 0.000 & -0.273 & -0.088 & -0.112 & 0.094 & 1 \end{bmatrix}$$

where $|\overline{R}| = 0.1811, t_2 = 0.7071$ and the test statistic for $1-\alpha = 0.95$ ii $\chi^2_{0.95,30} = 25$ which means H_0 is statistically significant at a 0.05 confidence level. Goodness of fit using the KS test for $\beta = \{0, 0.01, \cdots, 1\}$, $\alpha = 0.05$ (in vector form D_0) obtains the following results:

$$D_0 = [0.0966 \quad 0.0816 \quad 0.0772 \quad 0.0836 \quad 0.0419 \quad 0.0271], D_\alpha = 0.1406$$

which means that H_0 is statistically significant at a 0.1 confidence level. Goodness of fit using the Cucconi test for $\alpha = 0.05$ (in vector form T_c) obtains the following results:

$$t_c = [1.006 \quad 0.478 \quad 0.140 \quad 0.030 \quad 0.252 \quad 0.610], -\ln(\alpha) = 2.9957$$

which means that H_0 is statistically significant at a 0.05 confidence level. The cumulative densities for the original data against the GA imputation are shown in Fig. B.4 (see Appendix B).

4.6. Tests over the EM algorithm

We compute the values of all tests of the solution provided by the EM algorithm. The obtained results of the test on means are summarized in the vector t_0:

$$t_0 = [\,-0.573 \quad -0.165 \quad 0.769 \quad -0.354 \quad 0.087 \quad -0.553\,]$$

which means that H_0 is statistically significant at a 0.05 confidence level. The obtained results for the individual variances test are summarized in the vector f_0

$$f_0 = [\,1.667 \quad 1.443 \quad 1.646 \quad 1.248 \quad 1.397 \quad 1.281\,]$$

which means that H_0 is statistically significant for $\{4,6\}$ and H_1 is statistically significant for $\{1,2,3,5\}$ at a 0.05 confidence level. The obtained results of the χ^2 test over covariance matrices are as follows:

$$S_t = \begin{bmatrix} 4.374 & -1.173 & -6.711 & 6.372 & 0.315 & -0.128 \\ -1.173 & 90.569 & 12.590 & -4.992 & -0.814 & -3.931 \\ -6.711 & 12.590 & 20.362 & -10.369 & -1.189 & -0.305 \\ 6.372 & -4.992 & -10.369 & 17.601 & 0.594 & -0.824 \\ 0.315 & -0.814 & -1.189 & 0.594 & 3.050 & 0.275 \\ -0.128 & -3.931 & -0.305 & -0.824 & 0.275 & 1.537 \end{bmatrix}$$

where $|S_t| = 106607.23,\ \rho = 0.9787,\ \eta = 60.41,\ w = 21$ which means that $\rho\eta = 59.12 > \chi^2_{0.95,21}$ and H_1 is statistically significant at a 0.05 confidence level. The obtained results of the f test over covariances are $\rho_0 = 0.0005,\ w_0 = 288050.66,\ a = 21.46$ which means that $f_1 = \eta/a = 2.815 > f_{0.95,21,288051}$ and H_1 is statistically significant at a 0.05 confidence level, and the obtained results of the correlation matrix test are summarized as follows:

$$\overline{R} = \begin{bmatrix} 1.000 & -0.058 & -0.720 & 0.728 & 0.088 & -0.061 \\ -0.058 & 1.000 & 0.288 & -0.124 & -0.045 & -0.345 \\ -0.720 & 0.288 & 1.000 & -0.547 & -0.154 & -0.045 \\ 0.728 & -0.124 & -0.547 & 1.000 & 0.081 & -0.165 \\ 0.088 & -0.045 & -0.154 & 0.081 & 1.000 & 0.135 \\ -0.061 & -0.345 & -0.045 & -0.165 & 0.135 & 1.000 \end{bmatrix}$$

where $|\bar{R}| = 0.1534,\ t_2 = 15.443$ i.e. H_0 is significant at a 0.05 confidence level. Goodness of fit using the KS test for $\beta = \{0, 0.01, \cdots, 1\},\ \alpha = 0.05$ (in vector form D_0) obtains the following results:

$$D_0 = [\,0.1272 \quad 0.1036 \quad 0.1536 \quad 0.1189 \quad 0.1474 \quad 0.1370\,],\ D_\alpha = 0.1406$$

which means that H_0 is significant for $\{1,2,4,6\}$ and H_1 is significant for $\{3,5\}$ at a 0.1 confidence level. Goodness of fit using the Cucconi test for $\alpha = 0.05$ (in vector form T_c) obtains the following results:

$$T_c = [\,3.381 \quad 2.451 \quad 8.552 \quad 2.932 \quad 6.812 \quad 17.598\,],\ -\ln(\alpha) = 2.9957$$

which means that H_0 is significant for $\{2,4\}$ and H_1 is significant for $\{1,3,5,6\}$ at a 0.05 confidence level.

4.7. Tests over the auxiliary regressions

We compute the values of all tests of the solution provided by auxiliary regressions. The obtained results for the test on means are summarized in the vector t_0:

$$t_0 = [\,-1.008 \quad -0.360 \quad 1.084 \quad -0.634 \quad -0.499 \quad -0.222\,]$$

which means that H_0 is significant at a 0.05 confidence level. The obtained results for the individual variances test are summarized in the vector f_0

$$f_0 = [\,1.386 \quad 1.207 \quad 1.321 \quad 1.063 \quad 0.939 \quad 0.965\,]$$

which means that H_0 is significant for $\{2,4,5,6\}$ and H_1 is significant for $\{1,3\}$ at a 0.05 confidence level. The obtained results for the χ^2 test over covariances are as follows:

$$S_t = \begin{bmatrix} 4.644 & -0.806 & -6.960 & 6.590 & 0.293 & -0.217 \\ -0.806 & 96.522 & 11.485 & -4.193 & -1.609 & -3.947 \\ -6.960 & 11.485 & 21.894 & -10.425 & -0.818 & -0.111 \\ 6.590 & -4.193 & -10.425 & 18.731 & 0.577 & -0.988 \\ 0.293 & -1.609 & -0.818 & 0.577 & 3.558 & 0.339 \\ -0.217 & -3.947 & -0.111 & -0.988 & 0.339 & 1.719 \end{bmatrix}$$

where $|S_t| = 218804.58,\ \rho = 0.9787,\ \eta = 18.65,\ w = 21$ i.e. $\rho\eta = 18.25 < \chi^2_{0.95,21}$ and H_0 is significant at a 0.05 confidence level. The obtained results for the f test over covariance matrices are $\rho_0 = 0.0005,\ w_0 = 288050.66,\ a = 21.46$ which means that $f_1 = \eta/a = 0.869 < f_{0.95,21,288051}$ and H_0 is statistically significant at a 0.05 confidence level, and the obtained results for the correlation matrices test are as follows:

$$\overline{R} = \begin{bmatrix} 1.000 & -0.034 & -0.691 & 0.706 & 0.072 & -0.084 \\ -0.034 & 1.000 & 0.241 & -0.096 & -0.088 & -0.308 \\ -0.691 & 0.241 & 1.000 & -0.511 & -0.090 & -0.013 \\ 0.706 & -0.096 & -0.511 & 1.000 & 0.071 & -0.175 \\ 0.072 & -0.088 & -0.090 & 0.071 & 1.000 & 0.136 \\ -0.084 & -0.308 & -0.013 & -0.175 & 0.136 & 1.000 \end{bmatrix}$$

where $|\overline{R}| = 0.1963, t_2 = 12.93$ which means H_0 is significant at a 0.05 confidence level. Goodness of fit using the KS test for $\beta = \{0, 0.01, \cdots, 1\},\ \alpha = 0.05$ (in vector form D_0) obtains the following results:

$$D_0 = [0.1272 \quad 0.1036 \quad 0.1536 \quad 0.1189 \quad 0.1474 \quad 0.1370],\ D_\alpha = 0.1406$$

which means that H_0 is significant for $\{1, 2, 4, 6\}$ and H_1 is significant for $\{3, 5\}$ at a 0.1 confidence level. Goodness of fit using the Cucconi test for $\alpha = 0.05$ (in vector form T_c) obtains the following results:

$$t_c = [3.381 \quad 0.611 \quad 3.741 \quad 1.229 \quad 0.284 \quad 19.970],\ -\ln(\alpha) = 2.9957$$

which means that H_0 is significant for $\{2, 4, 5\}$ and H_1 is significant for $\{1, 3, 6\}$ at a 0.05 confidence level.

4.8. Discussion of the results

It is interesting to see that MIGA has obtained the best results without rejecting any of the applied statistical tests. Also $\bar{x}_C,\ S_C,\ b_C$ are very similar to $\bar{x}_A,\ S_A,\ b_A$ and they fit to the distribution of X_A (see Fig. B.4). Fig. 3 shows the conflict of interest between means and covariances/skewness since they increase/decrease differently at initial generations, but they converge to a stable solution.

The EM algorithm rejected some tests on individual variances, covariance matrix and it estimates everything as continuous data which is wrong since variables 2, 3 and 6 are integer with a low performance on skewness. Auxiliary regressions rejected some tests on individual variances but passed means, covariances, correlations, it estimates integer missing values as continuous and it also has a poor performance on skewness. It is clear that regressions preserve correlation structures better than likelihoods since they are optimal estimates of the correlation among variables.

In general, both the EM algorithm and auxiliary regressions passed some tests but they estimate integer observations as continuous data so there are some visible differences between $S_A,\ R_A,\ b_A$ and $S_C,\ R_C,\ b_C$. It is important to note that two very different matrices can have the similar determinants (which leads to errors in statistical testing) while two very similar matrices are rare to have different determinants (it is the main drawback to include eigenvalues and eigenvectors into $\mathscr{F}_r$).

MIGA has obtained the best results i.e. the smallest differences for all measures and we point out that all methods obtained the biggest difference over covariances at the pair $(2, 2)$ while the biggest difference over correlations were obtained at pairs $(5, 1), (6, 2)$ and $(6, 3)$ respectively. Table 1 shows the biggest absolute differences on means, variances, correlations and skewness for all methods.

To summarize, MIGA passed all tests and shows better results (smallest differences) than the EM algorithm and auxiliary regressions with a proper imputation of discrete missing values.

Table 1
Biggest differences between available and complete datasets.

Method	$\lvert\bar{x}_{jA} - \bar{x}_{jC}\rvert$	$\lvert s^2_{ijA} - s^2_{ijC}\rvert$	$\lvert r_{ijA} - r_{ijC}\rvert$	$\lvert b_{jA} - b_{jC}\rvert$
GA	0.301	0.751	0.025	0.116
EM algorithm	0.560	31.952	0.165	0.775
Aux. regression	0.723	17.803	0.189	0.486

5. Comparative example

A comparison between the MIGA and the proposal of Sefidian & Daneshpour [29] is presented in this section. Seven benchmark datasets were taken from the UCI Machine Learning repository (https://archive.ics.uci.edu/ml/index.php) whose description is shown in Table 2.

A total of eight methods are compared to the proposed MIGA: the Correlation Maximization-based Imputation Methods (CMIM), the k-Nearest Neighbors Imputation (k-NNI), the Fuzzy C-Mean Imputation (FCMI), the Decision Tree-based Imputation (DMI), the Incremental Attribute Regression Imputation (IARI), the Auto-encoder Neural Network Imputation (ANNI) and a Genetic Programming k-Nearest Neighbor method (GPNNI). The performance of each algorithm is evaluated by using the same measures used by Sefidian & Daneshpour [29]: Root Mean Squared Error (RMSE), Mean Absolute Error (MAD) and Coefficient of Determination (CoD) which are defined as follows:

$$\text{RMSE} = \left(\sum_i (o_i - p_i)^2/n\right)^{1/2};\ \text{MAD} = \sum_i |o_i - p_i|/n;\ \text{CoD} = 1 - \frac{\sum_i (o_i - p_i)^2}{\sum_i \left(o_i - \bar{o}\right)^2}$$

where o_i is the observed value, p_i is the predicted value, $\bar{o}$ is the average of observed values and n is the amount of missed observations to be imputed.

All missed values were selected according to the Missing At Random (MAR) method in which the probability of selecting a feature is independent of the missed observations in such feature but it depends on how many features were selected to then remove a uniform amount of observations per feature at random positions. The amount of missed values n is determined by four percentages: 30%, 40%, 50%, and 60% which were selected from a maximum of half of the features $(m/2)$ per dataset i.e. a maximum of $m/2$ features of every dataset might have missing values.

Sefidian & Daneshpour [29] standardized data using the z transform $z_i = \left(x_{ij} - \bar{x}_j\right)/s_j$ which cannot estimate integer/binary data while MIGA deals with non standardized and integer/binary data by minimizing Eq. (10) (see Definition 5) per every dataset where the required means, skewness, covariance and correlations were obtained from the original datasets to then compute RMSE, MAD and CoD. The MIGA was ran $Q = 12$ runs for three Minkowski orders $r = \{1, 2, \infty\}$ to then select the best results per dataset. Although the main goal of the comparison is to minimize the RMSE, MAD and CoD for each dataset, the MIGA was implemented in its original form without modifications where the best overall results for each dataset were obtained with the parameters shown in Table 3. The obtained RMSE are shown in Table 4.

Table 2
Benchmark datasets.

Dataset	Amount of records	# features (m)
Iris	150	4
Wine	178	13
Glass	214	10
Haberman	306	3
Wholesale Customers	440	8
Cardiotocography	2126	23
Adult	48842	14

Table 3
Parameters of MIGA.

Dataset	c	c_1	c_2	c_3	l	G	r
Iris	3	3	2	5	100	1000	2
Wine	3	3	2	5	100	1000	1
Glass	3	3	3	5	100	1000	2
Haberman	3	3	3	10	100	1000	∞
Wholesale	3	3	3	10	100	1000	∞
Cardio	5	5	5	10	200	2000	∞
Adult	5	5	5	10	200	2000	∞

Table 4
RMSE of the benchmark datasets.

Dataset	%	MIGA	CMIM	ANNI	GPNNI	IARI	FCMI	DMI	k-NNI	Mean
Iris	30%	**0.0987**	0.1273	0.1319	0.1328	0.1424	0.1912	0.144	0.1621	0.2994
	40%	**0.0998**	0.148	0.1564	0.1572	0.1638	0.2198	0.1654	0.1854	0.3457
	50%	**0.1302**	0.1655	0.1774	0.1795	0.1821	0.2554	0.1872	0.2098	0.3801
	60%	**0.1668**	0.186	0.198	0.2018	0.2029	0.2922	0.2034	0.2355	0.4119
Wine	30%	**0.0971**	0.1004	0.1131	0.1227	0.1242	0.1404	0.1278	0.124	0.1645
	40%	**0.0997**	0.1169	0.1251	0.1401	0.1446	0.1632	0.1517	0.1441	0.1866
	50%	**0.1332**	0.1384	0.147	0.1635	0.1723	0.184	0.1734	0.1644	0.218
	60%	**0.1492**	0.1529	0.1611	0.1751	0.1888	0.2064	0.1893	0.1796	0.2406
Glass	30%	0.0991	**0.0987**	0.1272	0.1486	0.1249	0.1925	0.1473	0.1363	0.2072
	40%	**0.1003**	0.1348	0.1344	0.1572	0.1415	0.2284	0.1561	0.1818	0.2343
	50%	**0.1391**	0.1432	0.1556	0.1749	0.1594	0.2468	0.1728	0.2055	0.263
	60%	0.1461	**0.1449**	0.1618	0.1977	0.1756	0.269	0.1964	0.2195	0.2851
Haberman	30%	**0.2121**	0.3233	0.355	0.3572	0.3634	0.3413	0.3737	0.4116	0.3359
	40%	**0.2567**	0.374	0.4078	0.4095	0.4237	0.4044	0.4267	0.4622	0.3959
	50%	**0.3301**	0.4306	0.4483	0.4549	0.4766	0.4416	0.4852	0.5164	0.4473
	60%	**0.3761**	0.4774	0.5031	0.5067	0.5194	0.495	0.5213	0.5777	0.4992
Wholesale	30%	**0.1176**	0.1368	0.1764	0.186	0.161	0.2098	0.171	0.1662	0.2003
	40%	**0.1231**	0.1603	0.2084	0.2239	0.1918	0.2592	0.2011	0.1979	0.2353
	50%	**0.1671**	0.1918	0.2438	0.2564	0.2258	0.2807	0.248	0.2282	0.2641
	60%	**0.2001**	0.2157	0.2641	0.2732	0.2528	0.3211	0.2645	0.2545	0.2993
Cardio	30%	0.0521	**0.0493**	0.0608	0.0605	0.0507	0.1193	0.0644	0.0657	0.1207
	40%	0.0573	**0.0554**	0.0649	0.0691	0.0588	0.1358	0.0677	0.0753	0.1373
	50%	0.0611	**0.0593**	0.0714	0.0708	0.0638	0.1532	0.0761	0.0878	0.1547
	60%	**0.0643**	0.0693	0.0767	0.0785	0.0728	0.1698	0.0828	0.0981	0.1712
Adult	30%	**0.0982**	0.1424	0.145	0.1538	0.152	0.1641	0.1615	0.155	0.1851
	40%	**0.1087**	0.171	0.1739	0.181	0.1777	0.1928	0.1864	0.1817	0.2128
	50%	**0.1786**	0.1942	0.1981	0.2034	0.1992	0.2164	0.2089	0.2044	0.2374
	60%	**0.1907**	0.2158	0.213	0.2206	0.2148	0.2328	0.2271	0.2211	0.2648

The obtained MAD are presented in Table 5.

Table 5
MAD of the benchmark datasets.

Dataset	%	MIGA	CMIM	ANNI	GPNNI	IARI	FCMI	DMI	k-NNI	Mean
Iris	30%	**0,0187**	0,0274	0,029	0,03	0,0304	0,0468	0,0325	0,0349	0,0732
	40%	**0.0211**	0.0361	0.0385	0.039	0.0397	0.0615	0.0437	0.0452	0.0972
	50%	**0.0327**	0.0456	0.0479	0.0483	0.0499	0.0805	0.0513	0.0562	0.1182
	60%	**0.0464**	0.0566	0.0588	0.0592	0.0604	0.1008	0.0672	0.0704	0.1399
Wine	30%	**0.0091**	0.013	0.0146	0.0147	0.0149	0.0189	0.0149	0.0156	0.0225
	40%	**0.0101**	0.0175	0.0189	0.0233	0.0193	0.0251	0.0215	0.0207	0.0293
	50%	**0.0133**	0.0229	0.0243	0.0273	0.0252	0.0316	0.0272	0.0267	0.0385
	60%	**0.0167**	0.0276	0.0289	0.0309	0.0304	0.0384	0.0321	0.0316	0.0461
Glass	30%	**0.0102**	0.0122	0.0138	0.0179	0.0142	0.0242	0.0176	0.0152	0.0276
	40%	**0.0137**	0.0164	0.0177	0.0205	0.019	0.0329	0.0203	0.0221	0.0368
	50%	**0.0187**	0.021	0.0231	0.0279	0.0243	0.0405	0.0267	0.029	0.0471
	60%	**0.0201**	0.0235	0.0269	0.0322	0.0286	0.0491	0.0304	0.034	0.0555
Haberman	30%	**0.0687**	0.0853	0.0893	0.0926	0.0938	0.0879	0.0988	0.1067	0.0876
	40%	**0.0962**	0.1129	0.121	0.1238	0.1233	0.1166	0.1235	0.1361	0.1174
	50%	**0.1311**	0.1464	0.1575	0.158	0.1584	0.1529	0.1645	0.1715	0.1566
	60%	0.1832	**0.1766**	0.1848	0.1859	0.1879	0.1827	0.1923	0.2087	0.1844
Wholesale	30%	**0.0109**	0.0211	0.0243	0.0253	0.0239	0.0271	0.0249	0.0238	0.0341
	40%	**0.0133**	0.0289	0.0334	0.0357	0.0321	0.0382	0.0342	0.0329	0.0458
	50%	**0.0267**	0.0387	0.0432	0.0463	0.0423	0.047	0.0452	0.0433	0.0595
	60%	**0.0332**	0.0471	0.0528	0.0549	0.0511	0.0572	0.0548	0.0527	0.0723
Cardio	30%	0.0031	**0.002**	0.0037	0.0036	0.0031	0.0107	0.0053	0.0044	0.01
	40%	0.0037	**0.0028**	0.0051	0.0058	0.0043	0.0141	0.0061	0.0063	0.0143
	50%	0.0045	**0.0037**	0.0064	0.0073	0.006	0.0179	0.0077	0.0084	0.0181
	60%	**0.0061**	0.0066	0.0082	0.009	0.0078	0.0219	0.0101	0.0133	0.0222
Adult	30%	**0.0097**	0.0112	0.012	0.0143	0.0129	0.016	0.0151	0.0143	0.0191
	40%	**0.0122**	0.0155	0.0168	0.0187	0.0172	0.0213	0.0205	0.019	0.0232
	50%	**0.0167**	0.0191	0.0205	0.0228	0.0215	0.0267	0.026	0.0239	0.0285
	60%	**0.0211**	0.0263	0.025	0.0279	0.0255	0.0316	0.0309	0.0286	0.0334

The obtained CoD are presented in Table 6.

Table 6
CoD of the benchmark datasets.

Dataset	%	MIGA	CMIM	ANNI	GPNNI	IARI	FCMI	DMI	k-NNI	Mean
Iris	30%	**0.9911**	0.9821	0.9783	0.9779	0.9777	0.963	0.9762	0.9713	0.9094
	40%	**0.9881**	0.9759	0.9721	0.9715	0.9705	0.9514	0.9654	0.9629	0.8793
	50%	**0.9876**	0.9703	0.9675	0.9647	0.964	0.9345	0.9617	0.9531	0.8542
	60%	**0.9677**	0.9618	0.9581	0.9574	0.9543	0.9144	0.95	0.9401	0.8286
Wine	30%	**0.9901**	0.9897	0.987	0.9863	0.9852	0.9801	0.985	0.9831	0.9726
	40%	**0.9853**	0.986	0.9834	0.9815	0.9815	0.973	0.9786	0.9779	0.9648
	50%	**0.9828**	0.9804	0.9774	0.9755	0.9744	0.9658	0.9717	0.9722	0.9521
	60%	**0.9781**	0.976	0.9731	0.9687	0.9685	0.9572	0.9659	0.9669	0.9417
Glass	30%	**0.9891**	**0.9891**	0.9838	0.9734	0.983	0.9578	0.9746	0.9796	0.9537
	40%	**0.9816**	0.9781	0.9794	0.9566	0.9784	0.9419	0.9574	0.9613	0.9418
	50%	**0.9798**	0.9781	0.9767	0.9489	0.9726	0.9321	0.9501	0.9533	0.9276
	60%	**0.9777**	0.9763	0.976	0.9424	0.9676	0.92	0.9436	0.9478	0.9155
Haberman	30%	**0.9391**	0.8959	0.8726	0.8713	0.8682	0.8822	0.8525	0.8314	0.8842
	40%	**0.9217**	0.8611	0.8325	0.8312	0.8208	0.8357	0.8034	0.7879	0.8497
	50%	**0.9002**	0.8156	0.7977	0.7842	0.773	0.8043	0.7637	0.7361	0.7986
	60%	**0.8931**	0.7739	0.759	0.749	0.7488	0.7581	0.7157	0.6693	0.7518
Wholesale	30%	**0.9894**	0.9795	0.9653	0.9651	0.9718	0.9471	0.9647	0.9694	0.9561
	40%	**0.9837**	0.9723	0.9574	0.9555	0.9604	0.9257	0.9547	0.9575	0.9408
	50%	**0.9787**	0.961	0.9383	0.9383	0.9451	0.9157	0.9375	0.9446	0.9263
	60%	**0.9697**	0.9514	0.9253	0.926	0.9318	0.8923	0.9249	0.9315	0.9057
Cardio	30%	**0.9987**	0.9984	0.996	0.9963	0.9973	0.9852	0.9957	0.9955	0.9848
	40%	0.9935	**0.9972**	0.995	0.9948	0.9964	0.9808	0.9946	0.9941	0.9841
	50%	0.9901	**0.996**	0.9938	0.9929	0.9953	0.9755	0.993	0.992	0.982
	60%	0.9878	**0.9952**	0.9926	0.991	0.9945	0.9699	0.99	0.989	0.9801
Adult	30%	**0.9898**	0.9781	0.976	0.9735	0.9769	0.9731	0.9737	0.976	0.9702
	40%	**0.9793**	0.9698	0.967	0.9667	0.9684	0.9628	0.9632	0.967	0.9603
	50%	**0.9701**	0.9632	0.9606	0.9583	0.9603	0.9532	0.9547	0.9582	0.9512
	60%	**0.9667**	0.9516	0.9549	0.9505	0.9539	0.9458	0.9482	0.9511	0.9424

It is interesting to see that the MIGA outperformed all other methods except in the Cardiotocography dataset and small mixed results were obtained for the Glass and Haberman datasets. A possible reason is that the Cardiotocography dataset shows higher correlation structures than other datasets which could help the CMIM methods to get a better estimation (the CMIM are a set of incremental methods based on the idea of selecting a proper regression method between highly correlated complete features/ vectors and missing observations by ranking its features to be estimated with a random-forest procedure), so GAs take more time (and runs) to find a similar or better solution than estimation methods. It is important to point out that the MIGA is a nonlinear imputation method based on random exploration of the set of all possible solutions while estimation methods take advantage from regressions (usually linear) and/or optimal statistical methods (asymptotically normal) to find better results.

The most complex datasets to analyze were Haberman which obtained the poorest imputation results since it is a pure integer/binary dataset which makes the problem harder to solve and the Adult dataset since it is composed by a mix of continuous/integer variables, mixed correlation structures and a bigger sample size which leads to sparse covariances and (by consequence) to a harder search; however MIGA ouperformed all other methods due to its flexibility to deal with integer/binary data. On the other hand, the easiest dataset to solve was the Iris which is composed by four positive continuous features and only one nominal integer feature plus it is the smallest dataset which makes the sampling space smaller and easier to solve.

Table 7 shows the results of the corrected Diebold-Mariano test whose null hypothesis (=) is equality between the Mean Squared Error (MSE) of MIGA and CMIM methods and the alternative hypothesis (×) is difference between MSEs (see Diebold

Table 7
Results of the corrected Diebold-Mariano test.

Dataset	30%	40%	50%	60%
Iris	×	×	×	×
Wine	×	=	=	=
Glass	=	×	×	=
Haberman	×	×	×	×
Wholesale	×	×	×	×
Cardio	=	=	=	=
Adult	×	×	×	×

& Mariano [10] and Harvey et al. [12]). Note that Cardiotocography, Wine and Glass datasets show no difference between performances while MIGA is statistically different from CMIM in the other datasets. Although this statistical test is not conclusive about the overall performance of an algorithm, we can see that the proposed MIGA is at least statistically equal or better than CMIM in terms of its MSE.

6. Concluding Remarks

A multiple imputation genetic algorithm - MIGA is proposed to solve the problem of multiple missing observations in multivariate matrices with continuous and discrete variables. The proposed fitness function $\mathscr{F}_r$ is defined as the Minkowski r–norm of dimensionless measures for the means, covariances and skewness between available and completed data in order to put those three goals (seen as distances) together into a single additive fitness function. The proposal was applied to eight datasets: a first one with $\sim 58\%$ of incomplete individuals which was compared to two well known estimation methods: the EM algorithm and auxiliary regressions, and seven benchmark datasets with different percentages of missed observations which were solved with eight methods (CMIM, ANNI, GPNNI, IARI, FCMI, DMI, k-NNI and Means). MIGA has outperformed all other methods in seven out-of 8 datasets for which their means, covariance/correlations, skewness and asymmetry structures were preserved by the GA while other methods did not preserve some of the means, variances, almost all skewness and could not estimate discrete variables.

MIGA has shown advantages over other considered methods: it can handle binary, integer and multiple variables; it deals with multi–objective problems and it can be extended to other goals (via the Minkowski r–norm) with satisfactory results; it can provide more than a single solution (individuals with good fitness) and it can be adapted to different user requirements and problems. Clearly, MIGA has shown satisfactory results for the provided example which is intended to be comprehensive and opened to be solved using other techniques/methods, and it also outperformed other methods like CMIM for all benchmark datasets with binary/integer features like Haberman and Adult except for the Cardiotocography dataset which exhibits higher correlations than the others.

Finally, the obtained results are satisfactory since they passed all statistical tests (means, individual variances, covariances, correlations and goodness of fit) so there is enough statistical evidence to say that MIGA is an appropriate method to impute missing data in multivariate matrices and the obtained results fit to the original populations (see probability densities shown in Fig. B.4). It means that the proposed method is not restricted to any predefined kernel and/or distribution assumption for sampling and it can fit any kind of data unlike other proposals and methods which are sensitive about distribution assumptions.

7. Future work

Convergence speed and the quality of solutions can be possibly improved by implementing some interesting algorithms for multiobjective evolutionary optimization as proposed by Wang et al. [33] who focus to diversity operators to increase speed of search space, Long et al. [22] who proposed a GA for unconstrained multiobjective problems and Liu et al. who proposed a hybrid Gaussian/regularity/NSGA-II algorithm for handling noisy data. Other interesting applications for MIGA include time series, financial/risk analysis and image processing which exhibit particular properties and patterns. Also the use of alternative genetic operators and other evolutionary strategies such as meta-learning (see Pilát & Neruda [27] and Kazík, Pilát & Neruda [14]) could improve computing time and convergence of the algorithm.

CRediT authorship contribution statement

Juan Carlos Figueroa-García: Writing - original draft, Formal analysis, Conceptualization, Validation. **Roman Neruda:** Writing - review & editing, Methodology, Visualization. **German Hernandez–Pérez:** Writing - review & editing, Methodology, Software.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A. Genetic Solution

This appendix presents the best GA solution p^* indexed by its location $(i,j) \in M$.

Table A.8
Complete genetic results.

(i,j)	$\hat{x}_{ij}$	(i,j)	$\hat{x}_{ij}$	(i,j)	$\hat{x}_{ij}$	(i,j)	$\hat{x}_{ij}$	(i,j)	$\hat{x}_{ij}$
(17,1)	36.787	(173,2)	42	(204,3)	0	(285,4)	34.224	(269,5)	2.414
(18,1)	35.139	(174,2)	31	(223,3)	17	(305,4)	45.249	(270,5)	2.428
(19,1)	35.513	(175,2)	26	(224,3)	14	(306,4)	38.928	(271,5)	1.298
(35,1)	40.414	(189,2)	30	(254,3)	2	(307,4)	34.190	(272,5)	3.255
(36,1)	36.255	(192,2)	53	(285,3)	3	(3,5)	3.424	(273,5)	2.574
(55,1)	39.033	(225,2)	57	(295,3)	0	(4,5)	2.234	(299,5)	1.247
(67,1)	35.533	(237,2)	28	(296,3)	0	(5,5)	3.633	(300,5)	0.513
(101,1)	31.997	(247,2)	49	(27,4)	35.198	(15,5)	1.793	(301,5)	5.539
(127,1)	37.456	(248,2)	58	(32,4)	31.061	(26,5)	0.749	(17,6)	0
(128,1)	28.983	(249,2)	50	(70,4)	31.689	(27,5)	2.102	(18,6)	0
(129,1)	34.822	(250,2)	29	(76,4)	31.891	(28,5)	4.998	(19,6)	0
(130,1)	31.991	(261,2)	46	(99,4)	30.668	(44,5)	4.069	(40,6)	0
(162,1)	34.560	(262,2)	49	(100,4)	32.636	(45,5)	3.038	(41,6)	2
(189,1)	35.432	(304,2)	52	(101,4)	30.456	(46,5)	3.563	(42,6)	4
(190,1)	35.964	(305,2)	26	(102,4)	31.695	(47,5)	4.552	(61,6)	1
(191,1)	34.865	(306,2)	26	(103,4)	30.116	(61,5)	0.789	(62,6)	3
(192,1)	35.253	(307,2)	26	(116,4)	41.074	(75,5)	2.420	(63,6)	3
(193,1)	40.622	(308,2)	36	(143,4)	32.048	(76,5)	0.019	(93,6)	2
(237,1)	37.141	(25,3)	8	(147,4)	31.852	(77,5)	1.948	(97,6)	2
(238,1)	35.118	(26,3)	3	(165,4)	32.506	(102,5)	0.049	(126,6)	1
(239,1)	43.203	(27,3)	0	(166,4)	30.710	(103,5)	2.975	(127,6)	0
(269,1)	43.733	(28,3)	9	(167,4)	31.049	(104,5)	0.988	(128,6)	0
(270,1)	39.169	(45,3)	12	(168,4)	43.333	(126,5)	1.112	(152,6)	0
(271,1)	41.900	(46,3)	7	(169,4)	33.561	(127,5)	5.318	(153,6)	0
(272,1)	41.147	(47,3)	0	(170,4)	30.958	(145,5)	0.832	(154,6)	0
(273,1)	34.704	(48,3)	8	(209,4)	29.854	(148,5)	2.814	(155,6)	0
(308,1)	34.577	(49,3)	0	(215,4)	34.247	(155,5)	3.998	(188,6)	0
(6,2)	57	(126,3)	6	(225,4)	34.135	(180,5)	0.826	(189,6)	0
(7,2)	47	(127,3)	3	(236,4)	31.883	(181,5)	0.016	(199,6)	3
(40,2)	52	(128,3)	19	(248,4)	34.136	(182,5)	3.053	(217,6)	0
(41,2)	54	(129,3)	20	(249,4)	33.712	(193,5)	5.057	(220,6)	0
(42,2)	26	(130,3)	17	(250,4)	33.410	(199,5)	0.838	(244,6)	0
(67,2)	47	(131,3)	19	(251,4)	40.048	(217,5)	2.208	(245,6)	1
(101,2)	39	(132,3)	11	(252,4)	39.963	(218,5)	0.350	(246,6)	2
(102,2)	28	(168,3)	0	(253,4)	34.884	(219,5)	3.420	(247,6)	0
(125,2)	50	(169,3)	4	(271,4)	45.442	(241,5)	3.653	(273,6)	4
(148,2)	62	(170,3)	15	(272,4)	43.222	(242,5)	9.078	(278,6)	0
(152,2)	58	(171,3)	15	(273,4)	31.856	(243,5)	0.189	(283,6)	4
(171,2)	59	(172,3)	15	(283,4)	35.069	(244,5)	2.634	(305,6)	1
(172,2)	51	(203,3)	3	(284,4)	29.867	(251,5)	1.698		

Appendix B. Cumulative densities

This appendix shows a graphical comparison of the cumulative density functions obtained by the best solution provided by the proposed MIGA against the original dataset.

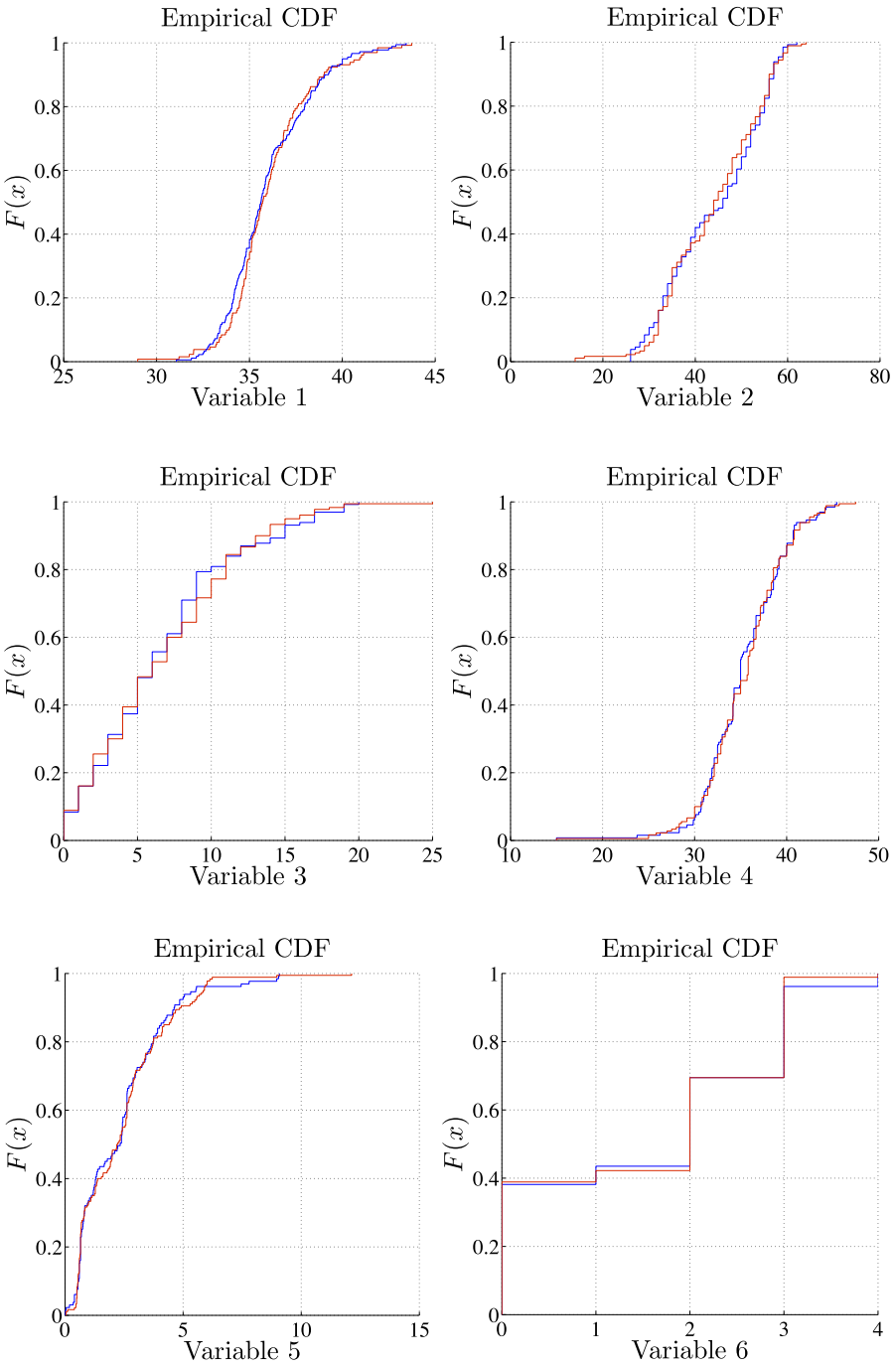

Fig. B.4. Cumulative density functions of the imputed GA (red) vs. available data (blue).

References

[1] M. Abdella, T. Marwala, The use of genetic algorithms and neural networks to approximate missing data in database, in: IEEE (Ed.), IEEE 3rd International Conference on Computational Cybernetics, 2005. ICCC 2005, IEEE. pp. 207–212.
[2] M.A. Aitkin, Some tests for correlation matrices, Biometrika 56 (1969) 443–446, URL:https://www.jstor.org/stable/2334438.
[3] T. Anderson, An Introduction to Multivariate Statistical Analysis, John Wiley and Sons, 1984.
[4] G. Box, A general distribution theory for a class of likelihood criteria, Biometrika 36 (1949) 317–346.
[5] G. Box, Problems in the analysis of growth and wear curves, Biometrics 6 (1950) 362–389.
[6] G. Celeux, J. Diebolt, The SEM algorithm: a probabilistic teacher algorithm derived from the EM algorithm for the mixture problem, Computational Statistics Quarterly 2 (1993) 73–82.
[7] O. Cucconi, Un nuovo test non parametrico per il confront tra due gruppi campionar, Giornale Degli Econmisti Annali di Econmia 27 (1968) 225–248.
[8] A.P. Dempster, N.M. Laird, D.B. Rubin, Maximum-likelihood from incomplete data via the EM algorithm, Journal of Royal Statistical Society 39 (1977) 1–38.
[9] L. Devroye, Non-Uniform Random Variate Generation, Springer-Verlag, New York, 1986.
[10] F. Diebold, R. Mariano, Comparing predictive accuracy, Journal of Business and Economic Statistics 13 (1995) 253–263, https://doi.org/10.1198/073500102753410444.
[11] J.C. Figueroa, D. Kalenatic, C.A. Lopéz, An evolutionary approach for imputing missing data in time series, Journal Of Circuits, Systems And Computers 19 (2010) 107–121.
[12] D. Harvey, S. Leybourne, P. Newbold, Testing the equality of prediction mean squared errors, International Journal of Forecasting 13 (1997) 281–291, https://doi.org/10.1016/S0169-2070(96)00719-4.
[13] H. Huang, H. Wang, M. Sun, Incomplete data classification with view-based decision tree, Applied Soft Computing 77 (2019) 356–365.
[14] O. Kazík, M. Pilát, R. Neruda, Meta learning in multi-agent systems for data mining, in: IEEE (Ed.), IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology, IEEE. pp. 433–434.
[15] A. Kolmogorov, Sulla determinazione empirica di una legge di distribuzione, G. Ist. Ital. Attuari 4 (1933) 83–91.
[16] P. Krishnaia, J. Lee, Likelihood ratio tests for mean vectors and covariance matrices, Handbook of Statistics 1 (1980) 513–570.
[17] S. Kullback, On testing correlation matrices, Applied Statistics 16 (1967) 80–85, URL:ttp://www.jstor.org/stable/2985240.
[18] X. Lai, X. Wua, L. Zhang, Autoencoder-based multi-task learning for imputation and classification of incomplete data, Applied Soft Computing 98 (2021), https://doi.org/10.1016/j.asoc.2020.106838 106838.
[19] A. Law, D. Kelton, Simulation System and Analysis, Mc Graw Hill International, 2000.
[20] L.A. Levine, G. Casella, Implementations of the Monte-Carlo EM algorithm, Journal of Computational Graphic Statistics 10 (2000) 422–439.
[21] T. Liu, H. Wei, K. Zhang, Wind power prediction with missing data using gaussian process regression and multiple imputation, Applied Soft Computing 71 (2018) 905–916.
[22] Q. Long, C. Wu, T. Huang, X. Wang, A genetic algorithm for unconstrained multi-objective optimization, Swarm and Evolutionary Computation 22 (2015) 1–14, https://doi.org/10.1016/j.swevo.2015.01.002.
[23] Mesquita, D.P., ao P.P. Gomes, J., Corona, F., Junior, A.H.S., Nobre, J.S., 2019. Gaussian kernels for incomplete data. Applied Soft Computing 77, 356–365.
[24] Mesquita, D.P., ao P.P. Gomes, J., Junior, A.H.S., Nobre, J.S., 2017. Euclidean distance estimation in incomplete datasets. Neurocomputing 248, 11–18. doi: 10.1016/j.neucom.2016.12.081.
[25] S.F. Nielsen, The stochastic EM algorithm: Estimation and asymptotic results, Bernoulli 6 (2000) 457–489.
[26] T. Nishino, H. Murakami, The generalized cucconi test statistic for the two-sample problem, Journal of the Korean Statistical Society 48 (2019) 593–612.
[27] M. Pilát, R. Neruda, Aggregate meta-models for evolutionary multiobjective and many-objective optimization, Neurocomputing 116 (2013) 392–402, https://doi.org/10.1016/j.neucom.2012.06.043.
[28] R.D. Priya, S. Kuppuswami, A genetic algorithm based approach for imputing missing discrete attribute values in databases, WSEAS Transactions on Information Science and Applications 9 (2012) 169–178.
[29] A.M. Sefidian, N. Daneshpour, Estimating missing data using novel correlation maximization based methods, Applied Soft Computing 91 (2020) 106249.
[30] N. Smirnov, Table for estimating the goodness of fit of empirical distributions, Annals of Mathematical Statistics 19 (1948) 279–281.
[31] D. Sovilj, E. Eirola, Y. Miche, K.M. Björk, R. Nian, A. Akusok, A. Lendasse, Extreme learning machine for missing data using multiple imputations, Neurocomputing 174 (2016) 220–231, https://doi.org/10.1016/j.neucom.2015.03.108.
[32] N.H. Timm, Applied Multivariate Analysis, Springer-Verlag, 2002.
[33] C. Wang, H. Pan, Y. Su, A many-objective evolutionary algorithm with diversity-first based environmental selection, Swarm and Evolutionary Computation 53 (2020), https://doi.org/10.1016/j.swevo.2019.100641 100641.
[34] A. Wójtowicz, P. Zywica, A. Stachowiak, K. Dyczkowski, Solving the problem of incomplete data in medical diagnosis via interval modeling, Applied Soft Computing 47 (2016) 424–437.