Artificial Intelligence in Medicine 50 (2010) 105–115

Contents lists available at ScienceDirect

Artificial Intelligence in Medicine

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

Missing data imputation using statistical and machine learning methods in a real breast cancer problem

José M. Jerez [a,*], Ignacio Molina [b], Pedro J. García-Laencina [c], Emilio Alba [d], Nuria Ribelles [d], Miguel Martín [e], Leonardo Franco [a]

[a] Departamento de Lenguajes y Ciencias de la Computación, Universidad de Málaga, E.T.S.I. Informática, Campus de Teatinos s/n, 29071 Málaga, Spain
[b] Departamento de Tecnología Electrónica, Universidad de Málaga, Campus de Teatinos s/n, 29071 Málaga, Spain
[c] Departamento de Tecnologías de la Información y las Comunicaciones, Universidad Politécnica de Cartagena, Plaza del Hospital 1, 30202 Cartagena (Murcia), Spain
[d] Servicio de Oncología Médica, Hospital Clínico Universitario Virgen de la Victoria, Campus de Teatinos s/n, 29010 Málaga, Spain
[e] Servicio de Oncología Médica, Hospital Clínico San Carlos, Profesor Martín Lagos s/n, 28040 Madrid, Spain

ARTICLE INFO

Article history:
Received 29 March 2009
Received in revised form 27 April 2010
Accepted 12 May 2010

Keywords:
Missing data
Statistical imputation techniques
Machine learning imputation methods
Survival analysis
Breast cancer prognosis
Early breast cancer

ABSTRACT

Objectives: Missing data imputation is an important task in cases where it is crucial to use all available data and not discard records with missing values. This work evaluates the performance of several statistical and machine learning imputation methods that were used to predict recurrence in patients in an extensive real breast cancer data set.

Materials and methods: Imputation methods based on statistical techniques, e.g., mean, hot-deck and multiple imputation, and machine learning techniques, e.g., multi-layer perceptron (MLP), self-organisation maps (SOM) and k-nearest neighbour (KNN), were applied to data collected through the "El Álamo-I" project, and the results were then compared to those obtained from the listwise deletion (LD) imputation method. The database includes demographic, therapeutic and recurrence-survival information from 3679 women with operable invasive breast cancer diagnosed in 32 different hospitals belonging to the Spanish Breast Cancer Research Group (GEICAM). The accuracies of predictions on early cancer relapse were measured using artificial neural networks (ANNs), in which different ANNs were estimated using the data sets with imputed missing values.

Results: The imputation methods based on machine learning algorithms outperformed imputation statistical methods in the prediction of patient outcome. Friedman's test revealed a significant difference ($p = 0.0091$) in the observed area under the ROC curve (AUC) values, and the pairwise comparison test showed that the AUCs for MLP, KNN and SOM were significantly higher ($p = 0.0053$, $p = 0.0048$ and $p = 0.0071$, respectively) than the AUC from the LD-based prognosis model.

Conclusion: The methods based on machine learning techniques were the most suited for the imputation of missing values and led to a significant enhancement of prognosis accuracy compared to imputation methods based on statistical procedures.

1. Introduction

Decisions about how to treat breast cancer patients after surgery have been contingent on the accuracy of estimating the behaviour and outcome of the disease. Different techniques have been used to aid clinicians in the estimation of the prognosis of different diseases. Standard statistical tools, like the Cox regression model or logistic regression [1], are normally used. However, more sophisticated models, based on machine learning methods, have been applied in recent years. Within these machine learning methods, artificial neural networks (ANNs) have been widely studied, and many works have demonstrated their utility in prognosis prediction. In many cases, they perform better than standard statistical tools [2–9]. One of the advantages of the use of machine learning models is that they are usually much more flexible than the standard statistical models and can capture higher-order interactions between the data, which results in better predictions. On the other hand, the predictions are made based on complex relationships between the data, and as a result the interpretability of the results is sometimes more difficult, even if there are tools to extract the knowledge acquired by these models. Thus, these alternative models are often criticised [10].

Prognosis models in breast cancer survival analysis are usually constructed from records that include clinical and histopathological information. However, clinical information databases commonly contain missing values or incomplete data that reduce the

* Corresponding author. Tel.: +34 95 213 2895; fax: +34 95 213 1397.
E-mail address: jja@lcc.uma.es (J.M. Jerez).

0933-3657/$ – see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.artmed.2010.05.002

number of available cases for analysis or might distort the analysis by introducing a bias into the estimation and/or prediction process. This drawback is particularly important in survival analysis, where an inadequate treatment can lead to adverse secondary effects or even death in the patient. Nevertheless, in many cases the simple and common strategy to deal with absent values continues to involve directly ignoring them. Several works [11,12] have demonstrated the dangers of simply removing cases using the listwise deletion method (LD) on the original data set. Such deletion can introduce substantial biases in the study, especially when missing data are not randomly distributed. In this sense, missing data imputation is an area of statistics that has attracted much attention in recent decades [11,13–17]. Several strategies inspired in statistics and machine learning have been developed to address this problem. A review of the literature reveals that the efficacy of the proposed methods depends strongly on the problem domain (e.g., number of cases, number of variables, missingness patterns), and thus there is no clear indication that favours one method over the others. Within statistics-based imputation methods, Pérez et al. [18] presented single, hot-deck and multiple imputation (MI) methods to impute missing data in the construction of a scoring system for predicting death in ICU patients. Results showed that differences in areas under the ROC curve (AUC) were statistically significant but not clinically relevant. The hot-deck method (with distance-based donor selection) was also used in an MI scheme [19], and it was found that inferences from depression treatment trial data were not sensitive to most definitions of distance. In another study [20], six popular imputation procedures (i.e., LD, item mean substitution, person mean substitution at two levels, regression imputation and hot-deck imputation) were compared in different data sets, with better performance observed for the hot-deck imputation technique. Software packages that implement MI (SOLAS, SAS, S-Plus and MICE) were compared in another study [21], and no advantages in performance were presented by any of them. Several imputation methods (hot-deck and MI) were also recently used in still another study [22] to handle missing predictor values in a risk model, and the best results in terms of prognosis accuracy and biased estimates were obtained by the MI algorithm using MICE software [23]. Finally, Kenward and Carpenter [24] provide an overview of MI and current perspectives on its use in medical research. The authors explore the problem of obtaining proper imputations and raise questions that have emerged from the increasing use of MI, which point to future research directions.

Imputation methods inspired by machine learning are based on the construction of a predictive model to estimate absent values from the information available in the data set. Well-known learning algorithms such as multi-layer perceptron (MLP), k-nearest neighbours (KNN), self-organising maps (SOM) and decision tree (DT) construction algorithms have been commonly used as imputation methods in different problem domains and in emerging disciplines such as bioinformatics [25–27]. MLP outperformed imputation induction methods in a thyroid disease database collected in a clinical situation [28]. MLP was also useful for imputing individual values for survey attributes utilising available administrative census data [29]. Experimental results [30] also show that MLP consistently outperforms other methods for reconstructing missing values in multivariate analysis. DT algorithms were successfully used to impute data in industrial databases [31]. Additionally, missing data imputation based on the KNN algorithm outperformed internal methods used by *C*4.5 and *CN*2 to treat missing data [32]. A clustering approach based on KNN was proposed for dealing with incomplete data [33]. KNN has been also used [34] to process missing information from DNA microarrays. Finally, SOM-based imputation techniques [35–37] were also successfully used in a number of problems, including modelling of intelligent tutoring systems and a real-size transport survey database.

In this work, six well-known methods, i.e., mean, hot-deck, MI, MLP, SOM and KNN, are used to impute absent values in the "El Álamo I" breast cancer data set, which contains 3679 records; we then use ANN models to predict early breast cancer relapse, and compare the performance of predictions from the different methods. We used neural network-based models to predict early breast cancer recurrence because these models are currently being used in the clinical environment, and several works have demonstrated their utility [2–4,6,5,8,7,9]. Moreover, ANN behaviour has been shown to be robust with data that include missing values, and this is also part of the scope of this work. In this paper, we analyse the prediction of early breast cancer relapse, as there is growing evidence that early and late breast cancer relapse can be caused by different factors. Studies using data sets from different countries have shown the existence of a two-peak hazard function with an early peak related to early breast cancer recurrence. This peak is centred at about 14–24 months after surgery and might be related to the tumour dormancy hypothesis. A second peak appears a couple of years later, for which the dynamic and time of appearance are less clear.

The paper is structured as follows: we give details of the breast cancer data set in Section 2, followed by a detailed review of the methods used for the imputation of missing values in Section 3. Section 3 also includes a description of the ANN model used for obtaining the prognosis prediction and the statistical tests used for analysing the significance of the results. The results are presented in Section 4 and discussed in Section 5.

2. The "El Álamo-I" breast cancer dataset

Data were collected from the "El Álamo-I" project [38,39], one of the largest databases on breast cancer in Spain. The data set analysed in this study includes demographic, therapeutic and recurrence-survival information from 3679 women with operable

Table 1
Characterization of the "El Álamo I" breast cancer data set containing records from 3679 patients. Range, mean or mode, and missingness percentage are shown for the eight covariates considered relevant in the prognosis (age, tumour size, number of axillary lymph nodes, histological grade, histological type, hormonal receptors, type of treatment and survival time). The survival status is the variable to be predicted by the system.

Prognostic factors	Range	Mean or mode	Type/scale	Missingness (%)
Age (AG)	20–90	56.22	Quantitative/ratio	0.19
Tumour size (TS)	0.1–16	2.87	Quantitative/ratio	4.02
Axillary lymph nodes (AN)	0–35	2.49	Quantitative/ratio	2.83
Histological grade (HG)	1, 2, 3	2	Qualitative/ordinal	42.84
Histological type (HT)	1, 2, 3, 4	1	Qualitative/nominal	0.60
Hormonal receptors (HR)	0, 1	1	Qualitative/nominal	0.0
Type of treatment (TT)	1, 2, 3, 4, 5, 6, 7, 8, 9	3	Qualitative/nominal	0.0
Survival time (ST)	1.12–128.88	69.74	Quantitative/ratio	0.0
Survival status (SS)	0, 1	0	Qualitative/nominal	0.0

invasive breast cancer diagnosed in 32 hospitals belonging to the Spanish Breast Cancer Research Group (GEICAM) between the years 1990 and 1993. This study uses a set of eight clinical and pathological covariates selected by clinicians as more significant prognostic factors in the prediction of patient outcomes: age, tumour size, axillary lymph nodes, tumour histological grade, histological type, hormonal receptors, type of treatment and survival time. Table 1 shows ranges, statistics (mean or mode, depending upon the type of the variable) and missing data distribution for the set of covariates. Of these, four quantitative variables are all ratio-scaled. The remaining covariates are qualitative (or categorical) variables that use numeric values to represent each category. It should be noted that the variable histological grade can be considered ordinal because there is a positive correlation between the numeric value of the covariate and the real histological grade of the tumour malignancy. Nevertheless, in this work, this covariate is treated as a pure nominal variable because not all imputation methods used have models for this type of data.

The missing data (considering every attribute for every patient case as data) represent 5.61% of the overall data set. If we count the number of cases with at least one missing value, as considered by the listwise procedure that discards all instances containing missing values, then 1678 cases contain missing data, representing 45.61% of the 3679 patients. Of these, there are 1511 patient cases where one value is missing (41.07% of the total), 155 with two values missing (4.21%) and only 12 cases (0.33%) with three values missing. There are no cases with more than three values missing. The last column in Table 1 (under the heading of missingness) shows that missingness mainly affects the variable histological grade (42.84%). The remaining covariates exhibit much lower percentages, with no missing data in the covariates hormonal receptors, type of treatment and survival time.

3. Methods

In this section, we introduce and describe the methods applied to impute the original incomplete data set and describe the prediction method used based on ANN. The subsequent subsections are organised as follows. First, several general considerations are made to explain how the imputation methods have been implemented. Then, the six imputation techniques applied are described:

- the statistical methods, which include mean, hot-deck, and MI,
- the machine learning based methods, which include MLP, SOM and KNN.

Finally, the ANN model to predict survival probabilities is described together with statistical methods commonly used in model accuracy evaluation.

3.1. Missing data imputation of the incomplete data set

Prior to describing the methods used for data imputation, several key remarks concerning all the imputation methods should be clearly presented:

- Missing data pattern: Our data set can be considered missing at random (MAR), i.e., the probability that an observation is missing can depend on the observed set but not on the missing set. This assumption is satisfied because the probability that a record is missing depends exclusively on the availability of the clinical data, which were recorded from 32 hospitals in the considered data set.
- Treatment of categorical variables using dummies: A coding scheme based on dummy variables was utilised for categorical inputs, which was necessary to implement some of the imputation methods. For example, in our data set, each of the three categories of the histological grade, i.e., 1, 2 and 3, were coded as 00, 01 and 10, respectively. In addition, another consideration that was addressed when using dummies was rounding of the resulting imputed values. If a binary dummy variable with values 0 and 1 is treated as an individual normal variate, imputation methods will then impute a continuous value to it. Different approaches can be used to produce the final dummy code from the imputed dummies, but following the conclusions of a previous study [40], we did not round the imputed dummy variable.
- Dependent variable: The binary variable 'survival status' (SS) is completely observed and represents the value that should be predicted for test cases after the imputation stage; thus it is not accounted for by imputation. The resulting imputed variables are independent of the dependent variable, making it possible to study the relationship between SS and the remaining variables in posterior stages.

3.2. Statistical methods

The statistical imputation methods used in this work include mean imputation, hot-deck, and MI methods based on regression and the expectation maximisation (EM) algorithm.

Mean and hot-decking imputations are simple imputation methods in which missing data are replaced by plausible estimates (one estimate per missing value) before applying standard complete-data methods to the filled-in data. Mean imputation, which can be considered the simplest approach [13], imputes the mean values of each variable on the respective missing variables as an estimate of the missing value. Because of its simplicity, mean imputation is commonly used in the social sciences as a fast alternative to LD. In our work, this approach is analysed because it can be used as a simple reference method. On the other hand, hot-deck imputation is an intuitively simple method for accommodating incomplete data. For this reason, it has been successfully applied to missing values in large data sets [11,41]. In hot-deck imputation, a missing value of a receptor instance is generally taken from a similar donor case that has complete data, although other alternatives exist. Hot-deck imputation produces unbiased estimates of population means [42], and in fact, it is asymptotically equivalent to the mean-score method for the estimation of a regression model parameter.

Up to now, we have introduced simple imputation approaches, where a single missing value is replaced by a unique value, obtained from the complete data portion. Conversely, MI replaces an unknown value with a set of plausible data and uses an appropriate model that incorporates random variation. MI has several desirable features: (1) an appropriate random error is introduced into the imputation process to obtain approximately unbiased estimates of all parameters; (2) good estimates of standard errors are obtained from repeated imputation; and (3) MI can be used for any kind of data and any kind of analysis without specialised software.

According to a previous report [43], three steps in MI are carried out to estimate incomplete data regression models. First, various plausible values (typically 5–10 imputations) for missing values are obtained that reflect uncertainty about the non-response model. These multiple missing values result in the creation of a number of complete data sets. Second, each of these data sets is analysed using complete-data methods. Finally, the results are combined, allowing the uncertainty regarding the imputation to be taken into account.

3.2.1. Mean imputation

In the general approach to mean imputation, which can be viewed as a simple application of regression imputation, the mean value of each non-missing variable is used to fill in missing values for all observations. In our case, the two categorical incomplete variables, histological grade (treated as pure ordinal as explained in previous sections) and histological type, are imputed using the mode instead of the mean, as this ensures that no values different from those within the range are generated. In Table 1, the third column gives the value of the mean or mode for each of the covariates.

3.2.2. Hot-deck imputation

We apply nearest neighbour hot-deck imputation, where a non-respondent is assigned the value of the nearest neighbour record according to a similarity criterion. As in the case of mean imputation, this approach does not make use of dummies to code categorical variables. Instead, each category is named using a natural number according to Table 1.

Obviously, one key aspect of hot-decking is the selection of an adequate dissimilarity measurement (distance function) between pairs of patient cases. A simple way to account for both quantitative and qualitative variables is to use a heterogeneous distance function with different distance metrics that depend upon the nature of the variable. In Ref. [44], a heterogeneous Euclidean-overlap metric (HEOM) distance function is presented and analysed, which uses the so-called overlap metric for categorical attributes and a normalised city-block distance (Manhattan, L1, or Minkowski function) for linear numeric quantitative attributes. The overlap metric is a normalised Hamming distance given as the percentage of coordinates that differ. The HEOM distance is intended to remove the effects of the arbitrary ordering of categorical values, and it constitutes an overly simplistic approach to handling these kinds of attributes.

Consider that a patient case is represented by an n-dimensional input vector[1], $\mathbf{x} = [x_1, x_2, \ldots, x_n]^T$. Moreover, in our notation, $\mathbf{m}$ is a vector of binary variables such that $m_j = 1$ if x_j is unknown and $m_j = 0$ if x_j is present. Given a pair of patient cases, represented by $\mathbf{x}_a$ and $\mathbf{x}_b$, the HEOM distance between them is:

$$d(\mathbf{x}_a, \mathbf{x}_b) = \sqrt{\sum_{j=1}^{n} d_j(x_{aj}, x_{bj})^2} \tag{1}$$

where $d_j(x_{aj}, x_{bj})$ is the distance between $\mathbf{x}_a$ and $\mathbf{x}_b$ on its jth attribute:

$$d_j(x_{aj}, x_{bj}) = \begin{cases} 1 & \text{if } (1 - m_{aj})(1 - m_{bj}) = 0, \\ d_O(x_{aj}, x_{bj}) & \text{if } x_j \text{is a categorical attribute}, \\ d_N(x_{aj}, x_{bj}) & \text{if } x_j \text{is a quantitative attribute}. \end{cases} \tag{2}$$

Unknown data are handled by returning a distance value of 1 (i.e., maximal distance) if either of the input values is unknown. The *overlap* distance function d_O assigns a value of 0 if the discrete attributes are the same; otherwise, the value is 1. The *range normalised difference* distance function d_N is given by:

$$d_N(x_{aj}, x_{bj}) = \frac{|x_{aj} - x_{bj}|}{max(x_j) - min(x_j)} \tag{3}$$

where $max(x_j)$ and $min(x_j)$ are the maximum and minimum values, respectively, observed in the training set for the numerical attribute x_j; thus, the normalisation attempts to scale the attribute down to the point where differences are almost always less than one, and the resulting distance matrix is set to range between 0 and 1. The difference $|x_{aj} - x_{bj}|$ is the city-block distance.

[1] In the following, the terms case, observation, instance, and example are used as synonyms.

3.2.3. Multiple imputation

There are several programs and routines available that implement different algorithms and techniques for MI. For instance, the stand-alone Windows program NORM or the S-PLUS libraries NORM, CAT, MIX, PAN, and MICE are all data augmentation algorithms. MICE and IVEware (a free SAS-callable application) are flexible tools for generating multivariate imputations for different kinds of variables using chained equations. Moreover, the commercial SAS procedures MI and *MIANALYZE* provide a parametric and a nonparametric regression imputation approach, as well as the multivariate normal model. The free Windows software packages Amelia and Amelia II, which implement EM-based imputation, are also available. Finally, it should be mentioned LogXact by Cytel implements logistic regression analysis.

In this work, we consider three different implementations based on SAS, Amelia II and WinMICE to generate multiple imputed values. In all cases, 5 imputations for each missing value were obtained, and dummy variables were used for categorical variables as described in Section 3.1. The details of each of the three implementations are given next:

- *Amelia II*: This software implements a bootstrapping-based EM algorithm [45]. It includes features for imputing cross-sectional surveys, time series data, and time series cross-sectional data. The package allows users to set priors on individual missing values in the observations when that knowledge is available. Amelia II performs the imputation step, while separate analyses and combination of results can be undertaken in R or in a separate statistics package (i.e., SAS or Stata). In our study, the five imputations are carried out in Amelia II with no options for individual variables and no prior beliefs about the data. Categorical variables were replaced by the corresponding dummies, as described in Section 3.1. After these considerations are settled, running Amelia II is straightforward.
- *WinMICE*: The package MICE (Multiple Imputation by Chained Equations) is available from the Comprehensive R Archive Network and is a library for S-Plus and R. WinMICE is a stand-alone program under Windows that implements imputation on a linear mixed model. In practice, a variety of imputation models are supported, including forms of predictive mean matching and regression methods, logistic and polytomous regression, and discriminant analysis. In addition, WinMICE allows users to program their own imputation functions. In theory, this could facilitate sensitivity analyses of different missingness models. For our data set, we chose regression as the method for imputation, and thus, dummies (as needed for Amelia II) were treated as pure numeric quantitative variables. Five imputations were obtained, and the results were combined by averaging regression coefficients, as proposed in a previous report [12].
- *SAS*: This software also provides an efficient implementation of the MI procedure that is used in our work to fill in missing values and create five complete data sets. Procedure *MI* assumes that data have a multivariate normal distribution. However, simulation studies show that *MI* is moderately robust to violations of this assumption [16]. Our pattern corresponds to an arbitrary pattern of missingness, as opposed to monotone patterns. Regarding categorical data, the use of dummies removes the need to use *CLASS* statements, which can be used to impute categories via logistic (or ordinal logistic) regression. This method for imputing *CLASS* variables was introduced into the procedure *MI* in SAS version 9.0 on an experimental basis. However, it is only available for monotonic missing data patterns and is implemented with the *logistic* option on the *monotone* statement. Each covariate has been dummy-coded according to the nomenclature and variable types in Table 1. The necessary

steps to carry out the imputation stage of the data matrix are accomplished by first computing the covariance matrix using the *CORR* procedure. The resulting covariance matrix, *covin*, is then used to set a prior parameter on the Markov Chain Monte Carlo (MCMC) method. The procedure *MI* uses an MCMC approach that is similar to EM in its iterative implementation. However, unlike EM, the results of *MI* are not a set of maximum likelihood estimates (MLEs) for parameters; instead, they are a distribution for the missing data that incorporates the uncertainty about the parameters. The *MCMC* statement makes use of some other parameters, but no description is needed when the default values are taken. The main input to the *MI* procedure is the data set to be imputed. We also set the *round*, *minimum* and *maximum* parameter options to restrict the imputed values. The number of imputations was set to the default value of 5. The EM can produce MLEs of the means and covariances for an incomplete data set, and these MLEs are used by default in the *MCMC* statement in the *MI* procedure.

3.3. Machine learning methods

Imputation methods based on machine learning are sophisticated procedures that generally consist of creating a predictive model to estimate values that will substitute the missing items. These approaches model the missing data estimation based on information available in the data set. If the observed data contain useful information for predicting the missing values, an imputation procedure can make use of this information and maintain high precision. This section describes three well-known imputation techniques using machine learning approaches: MLP, SOM and KNN.

3.3.1. Multi-layer perceptron

An MLP [46–48] consists of multiple layers of computational units interconnected in a feed-forward way. Each unit in one layer is directly connected to the neurons of the subsequent layer. The standard MLP architecture is composed of two layers of weights. Weights of the first layer connect the input data variables to the H hidden units (neurons), and the second layer weights connect these hidden neurons to the output units. First, given an n-dimensional input vector $\mathbf{x}$, the H hidden neuron outputs are computed in the form:

$$z_h = f\left(\sum_{j=1}^{n} w_{hj}^{(1)} x_j + w_{h0}^{(1)}\right) \tag{4}$$

where $h = 1, \ldots, H$, and the superscript (1) indicates that the corresponding parameters are in the first layer of the MLP. In our notation, the parameters z_h denote the hidden neuron outputs, $w_{hj}^{(1)}$ the first layer weights and $w_{h0}^{(1)}$ the corresponding biases parameters. The activation function $f(\cdot)$ is generally chosen to be sigmoidal, e.g., the logistic sigmoid or the hyperbolic tangent function. Subsequently, z_h are linearly combined and transformed using the appropriate output activation function, $g(\cdot)$, to give a set of outputs y_k:

$$y_k = g\left(\sum_{h=1}^{H} w_{kh}^{(2)} z_h + w_{k0}^{(2)}\right) \tag{5}$$

where $k = 1, \ldots, K$, and K is the total number of network outputs. This transformation corresponds to the second layer of the MLP, $w_{kh}^{(2)}$ and $w_{k0}^{(2)}$. The choice of $g(\cdot)$ is determined by the nature of the data and the target variables [46]. Thus, if the target variables are continuous data, the activation function is the identity, such that $g(a) = a$. Similarly, for binary classification problems, the neuron outputs are obtained using the logistic sigmoid, i.e., $g(a) = 1/(1 + \exp(-a))$. For multi-class problems with a 1-of-c coding scheme, a softmax function is used, i.e., $g(a_k) = (\exp a_k)/(\sum_{k'} \exp a_{k'})$. Finally, we can combine the two processing stages to give the overall network function:

$$y_k = g\left(\sum_{h=1}^{H} w_{kh}^{(2)} f\left(\sum_{j=1}^{n} w_{hj}^{(1)} x_j + w_{h0}^{(1)}\right) + w_{k0}^{(2)}\right). \tag{6}$$

A fully connected, two-layered MLP architecture was used in this study. This architecture was used because any continuous functional mapping can be represented by a network having two layers of weights and sigmoidal activation functions [46,47]. Once the MLP architecture has been chosen, the following two issues must be addressed: the optimal network weights and the optimal number of hidden neurons.

First, we consider the training process of an MLP with a fixed architecture (number of hidden neurons). During the training stage, the network weights are usually obtained by a gradient optimisation method to minimise the selected error function. In particular, the scaled conjugate gradient method is used as the optimisation technique for training the MLP network [46]. Depending on the type of target variables (continuous, binary or multi-class), different error functions should be considered [46,47] within a maximum likelihood approach. For continuous target data, the sum of squares error (SSE) is chosen; meanwhile, the cross-entropy error (CEE) function is suitable for binary and multi-class target data.

Second, the number of hidden units in an MLP can have a significant impact on its performance. Various techniques have been developed for optimising the architecture [49]. This paper uses a combined growing and pruning algorithm, which uses the performance results on a validation set as the criterion for adding (growing) or deleting (pruning) a hidden neuron during the learning process [50–52]. For a complete description of MLP properties, training procedures and applications, see Refs. [46–48].

MLP networks can be used to estimate missing values by training an MLP to learn the incomplete features (used as outputs), using the remaining complete features as inputs [28–30]. The MLP imputation scheme developed in this study can be described as follows:

(1) Given an incomplete data set $\mathbf{X}$, separate the input vectors that do not contain any missing data (complete set, $\mathbf{X}^C$) from the ones that have missing values (incomplete set, $\mathbf{X}^I$).
(2) For each possible combination of incomplete attributes in $\mathbf{X}^I$, construct an MLP scheme using $\mathbf{X}^C$. The target variables are the attributes with missing data, and the input variables are the other remaining attributes [28]. In this approach, there is one MLP model per combination of missing variables. Depending on the nature of the attributes to be imputed (numerical or discrete), different error functions (SSE or CEE) are minimised during the training process.
(3) After the optimal MLP architectures are chosen, for each incomplete pattern in $\mathbf{X}^I$, unknown values are predicted using the corresponding MLP model (according to the attributes to be imputed).

Given that weight initialisation is critical in the MLP training, the previous MLP imputation process is repeated 30 times to obtain a realistic missing data estimation. Imputed values are obtained by averaging the missing data estimation provided by each MLP model.

The MLP approach can be a useful tool for reconstructing missing values [28–30,53]. However, its main disadvantage is that when missing items appears in several combinations of attributes

in a high-dimensional problem, many MLP models have to be constructed.

3.3.2. Self-organisation maps

An SOM is a neural network model made out of a set of nodes (or neurons) that are organised on a 2D grid and fully connected to the input layer [54]. Each jth node has a specific topological position in the grid ($\mathbf{r}_j \in \mathbb{R}^2$), as well as a vector of weights ($\mathbf{w}_j \in \mathbb{R}^n$) of the same dimension used for the input vectors. Fig. 1 shows an SOM network of 4×4 nodes connected to the input layer.

The training of a basic SOM is performed using an iterative process. After the weight vectors are initialised, they are updated using all input training vectors. In the iteration step τ and for the input vector $\mathbf{x}$, the best matching unit (BMU) $\mathbf{w}_c$ is obtained as the closest node to $\mathbf{x}$ according to a distance metric (usually the Euclidean distance) [54]. The BMU is defined by the condition, $d(\mathbf{x}, \mathbf{w}_c) \leq d(\mathbf{x}, \mathbf{w}_j), \forall j$. Thus, each weight vector is updated by

$$\mathbf{w}_j^{\tau+1} = \mathbf{w}_j^{\tau+1} + h_{c,j}(\mathbf{x} - \mathbf{w}_j) \tag{7}$$

where $h_{c,j}$ is called the *neighbourhood function* [54], and is a time-variable decreasing function of the distance between the jth and c th nodes on the map grid. The neighbourhood function is often taken to be the Gaussian:

$$h_{c,j} = \alpha(\tau)\exp\left(-\frac{\|\mathbf{r}_j - \mathbf{r}_c\|^2}{2\sigma^2(\tau)}\right) \tag{8}$$

where $0 < \alpha(\tau) < 1$ is the learning-rate factor, which decreases monotonically with the training steps, and $\sigma(\tau)$ corresponds to the width of the neighbourhood function, which also decreases monotonically with the training process [54].

The self-organising process consists of slightly moving the nodes in the data definition space according to the data distribution. The weight adjustment is performed while taking into account the neighbouring relations between nodes in the map. The mapping between input vectors and nodes is then said to preserve topological relations insofar as observations that are close in the original input space will be associated with nodes that are close on the map. For a complete description of SOM properties and applications, see Ref. [54].

When an incomplete input vector is used as input to an SOM [35–37], the missing observations are simply ignored when calculating distances between observations and nodes: $\sum_{j=1}^{n}(1 - m_j)(x_j - w_j)^2$. This principle is applied for selecting the image node and also for updating weights [54]. Because the same variables are ignored in each distance calculation (over which the minimum is taken for obtaining the BMU), it is a valid solution.

After the SOM model has been trained, it can be used to estimate missing values [35–37]. When an incomplete observation is presented to the SOM, the missing input variables are ignored during the selection of the BMU. The incomplete data are imputed by the feature values of the BMU in the missing dimensions. The imputation process can be described as follows:

(1) Presentation of an incomplete observation in the input layer.
(2) Selection of the BMU by minimising the distance between the observation and nodes. Missing components are handled by simply excluding them from the distance calculation.
(3) The replacement value for a missing item in the input vector is taken as the value for that item in the corresponding BMU.

The SOM imputation approach is implemented in this study using the SOM toolbox available at http://www.cis.hut.fi/projects/somtoolbox/.

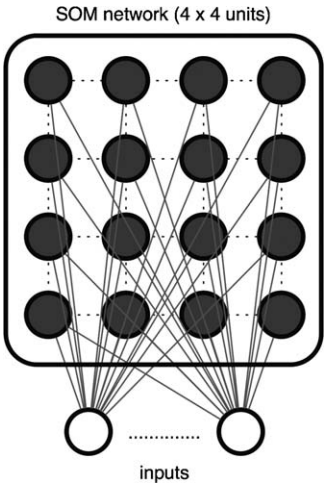

Fig. 1. Architecture of an SOM network of 4×4 units. Weights are represented by links between nodes and input layer units.

3.3.3. K-nearest neighbours

Rather than using all available instances in the data, the KNN imputation algorithm uses only similar cases with the incomplete pattern [32–34,55]. Given an incomplete pattern x, this method selects the K closest cases that are not missing values in the attributes to be imputed (i.e., features with missing values in $\mathbf{x}$), such that they minimise some distance measure [32]. In our notation, $\mathcal{V} = \{v_k\}_{k=1}^{K}$ represents the set of K nearest neighbours of $\mathbf{x}$ arranged in increasing order of distance. Although the K nearest neighbours can be selected for instances without any missing values, it is also recommended for cases with an incomplete pattern [32,34]. The optimal value of K is usually chosen by cross-validation.

Once the K nearest neighbours have been found, a replacement value for the missing attribute value must be estimated [32]. How the replacement value is calculated depends on the type of data; for example, the mode is frequently selected for discrete data, while the mean is used for numerical data.

As in hot-deck, the KNN method is based on the use of a distance metric to compute the nearest neighbours. For this purpose, we make use of the HEOM distance given by Eqs. (1) and (2).

To estimate missing values with KNN, consider that $\mathbf{x}$ represents a missing value on the jth input feature (i.e., $m_j = 1$). Once its K nearest neighbours have been chosen, $\mathcal{V} = \{v_k\}_{k=1}^{K}$, the unknown value is estimated using the corresponding jth feature values of $\mathcal{V}$.

If the jth input feature is a continuous variable, different estimation procedures can be implemented in the KNN approach:

- *Mean estimation*: The imputed value ($\tilde{x}_j$) is obtained by the mean value of its K nearest neighbours, i.e.,

$$\tilde{x}_j = \frac{1}{K}\sum_{k=1}^{K} v_{kj} \tag{9}$$

- *Weighted mean estimation*: One obvious refinement is to weight the contribution of each v_k according to its distance to $\mathbf{x}$, i.e., $d(\mathbf{x}, v_k)$, giving greater weight to closer neighbours:

$$\tilde{x}_j = \frac{1}{K}\sum_{k=1}^{K} w_k v_{kj} \tag{10}$$

where w_k denotes the weight associated to the k th neighbour. An appropriate choice for w_k is the inverse square of the distance:

$$w_k = \frac{1}{d(\mathbf{x}, v_k)^2} \tag{11}$$

When the jth input feature is a discrete variable, the most popular choice is to use the mode of $\{v_{kj}\}_{k=1}^{K}$. Another option is to consider a weighted decision scheme, where a weight λ_k is assigned to each v_k. One suitable way to obtain λ_k is

$$\lambda_k = \begin{cases} \dfrac{d(v_K, \mathbf{x}) - d(v_k, \mathbf{x})}{d(v_K, \mathbf{x}) - d(v_1, \mathbf{x})}, & \text{if } d(v_K, \mathbf{x}) \neq d(v_1, \mathbf{x}) \\ 1, & \text{otherwise.} \end{cases} \tag{12}$$

Thus, $\tilde{x}_j$ is imputed by the category (each possible discrete value of the jth input feature) for which the weights of the representatives among the K nearest neighbours sum to the largest value. In this paper, we use this weighted approach to estimate missing values in numerical and discrete variables. It has been shown that this method provides a robust procedure for missing data estimation [32–34,55]. Its major drawback is that whenever the KNN method looks for the most similar instances, the algorithm has to search the entire the data set. This limitation is especially problematic for large databases.

3.4. The ANN prognosis model

We performed numerical simulations on the data imputed by the methods described above on neural networks comprising a single hidden layer with the number of neurons between 2 and 50. For each size of the neural networks and set of parameters, a 10-fold complete cross-validation scheme was used. Each 10-fold cross-validation was set up by iteratively choosing a test fold, then a random validation fold, and then using the remaining 8 folds for training. The cross-validation process was repeated 50 times with a different random seed value. To avoid over-fitting, an early stopping procedure was implemented in which the generalisation error was monitored on a validation set, and the generalisation value on a different test set was measured at the minimum of the validation error.

The simulations were conducted using Matlab code under the Linux operating system on a cluster of 25 Pentium IV 2.0 Mhz PCs interconnected by Openmosix. The scaled conjugate gradient back propagation algorithm was used for the learning algorithm [56], which combines the model-trust region approach (used in the Levenberg-Marquardt algorithm) with the conjugated gradient approach. This algorithm has shown superlinear convergence properties on most problems and works well with the early stopping procedure applied to avoid over-fitting. A brief description of the parameters and their values used for simulations with the neural network toolbox in Matlab [57] are given below:

- Starting learning rate: 0.001. Regulates the change in synaptic weights (the rate is later adjusted automatically during the training process).
- Maximum number of epochs for training: 15,000. Stop the training after this maximum number of iterations.
- Minimum performance gradient: $1e-6$. Stop the training if the magnitude of the gradient drops below this value.
- Maximum validation failures: 50. Parameter related to the early stopping procedure.
- Sigma parameter: $5.0e-5$. Determines the change in weight for the calculation of the approximate Hessian matrix in the scaled conjugate gradient algorithm.
- Lambda: $5.0e-7$. Parameter for regulating the indefiniteness of the Hessian.
- Neuron transfer function: Sigmoid.
- Weight values initialisation: Random values from the range $[0, 0.05]$.

3.5. Model evaluation

The accuracy of the prognosis models is evaluated by testing two main properties: discrimination and calibration. Discrimination is the ability to separate patients with and without a relapse event, and it is commonly assessed using the AUC value [58]. Calibration is the ability to correctly estimate the risk or probability of a future event. It measures how well the predicted probabilities from the model agree with the observed rate of later relapse. Calibration was assessed by the Hosmer-Lemeshow (HL) goodness-of-fit test [59]. Under the null hypothesis, the HL statistic suggests evidence of a lack of the model fitting (p-value much greater than 0.05 would indicate very good model calibration, while $p < 0.05$ reveals poor model calibration).

To evaluate more precisely the difference in prognosis accuracy among the missing data imputation methods, Demsar [60] suggests using the Friedman test [61] on the averaged results when 10-fold cross-validation is the sampling method. The Friedman test is a nonparametric test (similar to the ANOVA parametric test) that compares the average ranks of N algorithms ($N > 2$). Under the null hypothesis, the Friedman test states that the N algorithms perform equivalently, and the observed difference is merely random. If the null hypothesis is rejected (means are significantly different from each other), the Wilcoxon signed-rank test [62] is used to study exactly which means differ from the control model (i.e., the LD imputation method).

4. Results

Missing data imputation techniques based on both statistical and machine learning methods were applied to impute absent values in data from patients with breast cancer. The goal was to analyse the improvements in prognosis accuracy when different algorithms were applied to impute missing data values. A neural network model was used to predict whether a patient would suffer an early cancer relapse, and we analysed how the different imputation techniques affected this prediction.

To compare and study the convenience of imputing data, the reference model was first estimated by simply removing missing values from the original data set; this process is usually described as listwise or case deletion. Then, the methods described in Section 3 were applied to impute absent values, and an ANN-based model was used to predict early relapse in breast cancer patients

Table 2
Mean, standard deviation and MSE values for the AUC values computed for the control model and for each of the eight imputation methods considered.

AUC	LD	Mean	Hot-deck	SAS	Amelia	Mice	MLP	KNN	SOM
Mean	0.7151	0.7226	0.7111	0.7216	0.7169	0.7250	0.7340	0.7345	0.7331
Std. dev.	0.0387	0.0399	0.0456	0.0296	0.0297	0.0301	0.0305	0.0289	0.0296
MSE	0.0358	0.0235	0.0324	0.0254	0.1119	0.1119	0.0240	0.0195	0.0204

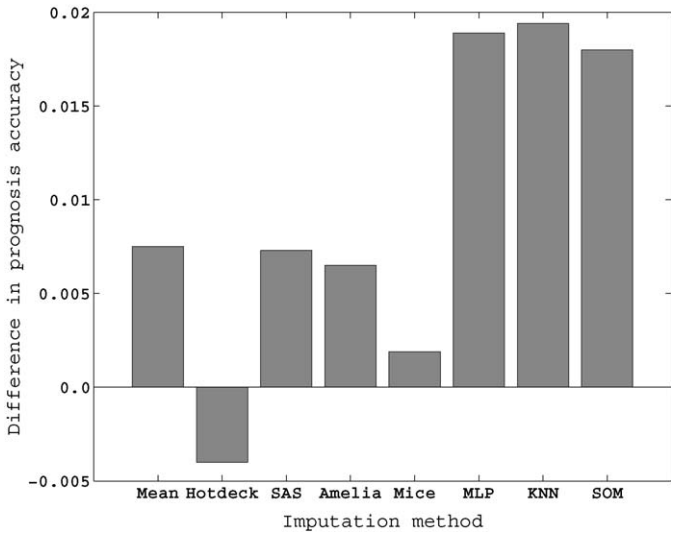

Fig. 2. Difference in AUC means between the reference model and each imputed data ANN model.

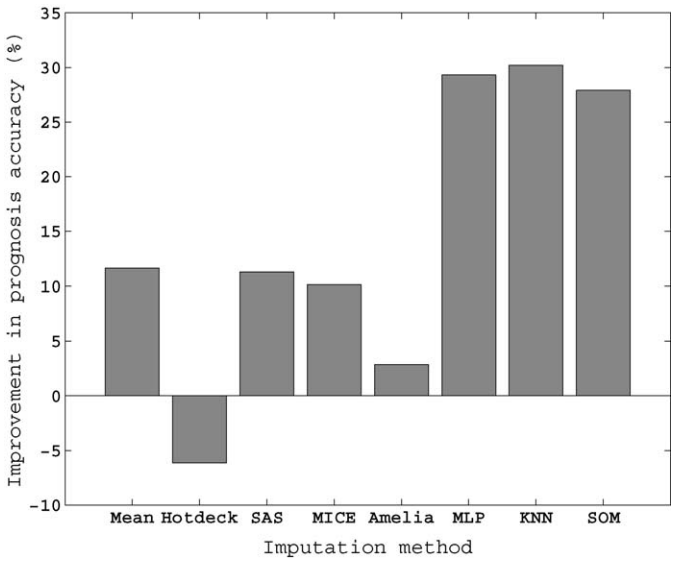

Fig. 3. Improvement in prognosis accuracy obtained as a percentage of the optimum (AUC_{reg}), computed using Eq. (13) (see text for more details).

belonging to a separate test set. An ROC analysis was performed to compare the results of each of the imputation methods. Table 2 shows means and standard deviations for the AUC values that were computed using a 10-fold cross-validation procedure.

As seen in Table 2, the prognosis models based on imputation methods described in Section 3 (except for the hot-deck method) outperformed the reference model in terms of AUC averaged over the cross-validation procedure. Fig. 2 depicts the difference in AUC means between the reference model and each imputed data ANN model, where the maximum difference (0.0194) was obtained for the KNN method.

The two-way nonparametric ANOVA Friedman's test was then used to test the overall effects of the algorithms on mean AUC values. The reference ANN model estimated by removing records with missing values was selected as the control model. Friedman's test revealed that there was a significant algorithm effect ($p = 0.0091$) on the observed AUC values. The Wilcoxon signed-rank test was then employed to determine the statistical significance of the difference in AUC means across ANN models based on imputed data and the control model. This pairwise comparison test showed that the AUCs for MLP, KNN and SOM were significantly higher ($p = 0.0053$, $p = 0.0048$ and $p = 0.0071$, respectively) than the AUC from the LD prognosis model. Differences in AUCs were not statistically significant for mean, hot-deck and MI methods ($p > 0.01$). The calibration of the models was assessed using the HL statistic, and the LD, SOM, MLP and KNN methods had good calibration indices ($p = 0.47$, $p = 0.54$, $p = 0.63$ and $p = 0.71$ for the HL statistic, respectively).

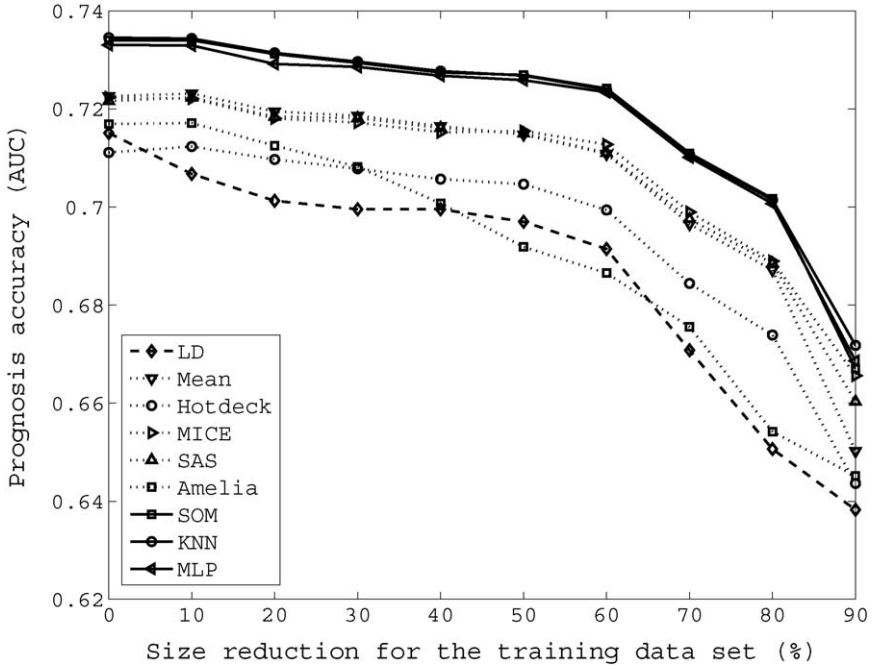

Fig. 4. Prognosis accuracies for the LD model and all considered imputation methods as the training data set size is reduced. A progressive decrease in the prognosis accuracy is clearly observed when fewer data were used.

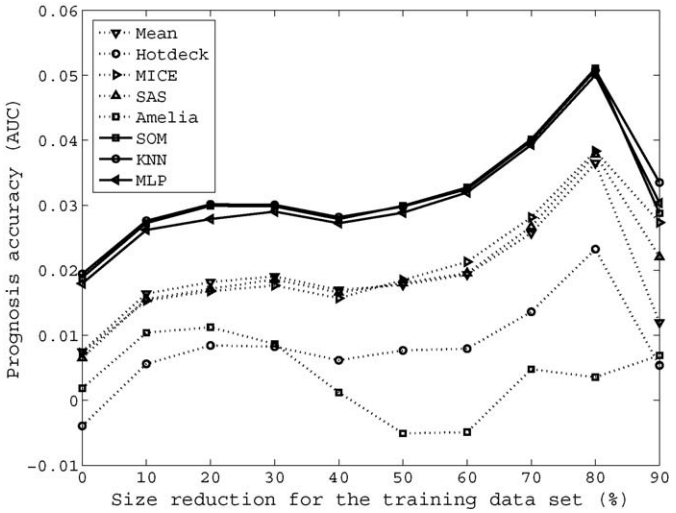

Fig. 5. Differences in performance (prognosis accuracy) for each imputation method with respect to the LD model as the training data set size is reduced.

Fig. 2 depicts absolute values of the difference in prognosis accuracies, which does not seem relevant in terms of model performance, e.g., $AUC_{KNN} - AUC_{LD}$ is just 0.0194. In clinical applications of pattern classification methods, the following question is the most important: what percentage of improvement does the generalisation ability present over the value obtained using the optimal model? In this experiment, an upper bound for prognosis accuracy (possible optimal model) was obtained using logistic regression with the test data set. The prognosis accuracy for the logistic regression model was 0.7795; based on this figure, it is possible to compute the difference in generalisation ability, as a percentage, relative to the maximum that could be obtained. Fig. 3 shows the improvement, I, as a function of the prognosis accuracy obtained from the LD model over that obtained from imputation methods. This figure is expressed as a percentage of the optimum (AUC_{reg}) and computed using the following equation:

$$I = \frac{AUC_{im} - AUC_{LD}}{AUC_{reg} - AUC_{LD}} \times 100 \tag{13}$$

where AUC_{im} and AUC_{LD} are the prognosis accuracies for the imputation method and the LD model, respectively. As seen in Fig. 3, in spite of low values for the absolute difference in prognosis accuracies, the percentage in improvement reaches the 30% when machine learning methods were used to impute missing data.

Up to this point, the results indicate that imputation methods based on machine learning algorithms (MLP, SOM and KNN) outperformed statistical imputation methods in the prediction of patient outcome in breast cancer early relapse. However, as noted in Section 1, many works in the literature have demonstrated that the benefits of imputation methods depend mainly on the problem domain, the size of the available data set and missingness patterns. We explored the dependency between the size of the available data set and the performance of the imputation methods. To accomplish this, an experiment was carried out to analyse the performance of the imputation methods when the size of the training data set was reduced in the split-sample validation scheme. The size of the training set was successively reduced in steps; in each step, 10% of the original data was eliminated. The prognosis accuracies were then newly tested for the LD model and imputation methods over the same original test data set. Fig. 4 depicts progressive decreases in the prognosis accuracy, where prognosis models performed more poorly when fewer data were used. The effect became more pronounced when the training set was reduced by 60%.

However, in general, higher differences in prognosis accuracy occur in Fig. 4 as the percentage of reduction is increased. Fig. 5 more specifically shows the differences in performance for each imputation method with respect to the LD model. The prognosis accuracy increased for almost every imputation method as the training set size was gradually reduced. This general improvement was even higher when the size of the training set was further reduced, and the differences in prognosis accuracies became statistically significant for mean, SAS and MICE imputation methods when the size decreased from 30% through 90%.

5. Conclusions and discussion

We applied six imputation methods to treat the problem of missing data in 3679 records from breast cancer patients in the "El Álamo-I" data set. First, we reviewed and provided technical details of the different methods used, which comprised three methods based on statistical analysis and three methods based on machine learning. The statistical methods included mean imputation, hot-deck and three different MI implementations applied with the software packages SAS, MICE and Amelia, while the machine learning-based methods included MLP, KNN and SOM. These approaches to missing data imputation were then applied to the breast cancer data set, in which 45.61% of observations contained missing data. Details of the data set are given in Table 1, where it can be observed that most of the missing values were present in the covariate histological grade. Once the unknown data were imputed, a prognostic model was created based on ANN to predict early breast cancer relapse, and the effectiveness of the different imputation techniques was compared. Breast cancer prognosis is important for determining the treatments for different patients; thus, the key issue in the present analysis was whether imputation techniques could improve the prognosis accuracy. As depicted in Table 2 and Fig. 2, all imputation methods except for the hot-deck method led to an improvement in prediction accuracy, as measured by the AUC. Statistical tests were conducted to determine whether the differences observed were significant; only the results obtained using the three machine learning-based techniques, MLP, KNN and SOM, were significantly different from those in which records containing missing values are eliminated (LD method). The calibration of these 3 methods was further assessed, and there was no significant difference from the LD case, for which the obtained model was already calibrated. Thus, there were no significant differences between the observed prognosis and the calculated one when the data were split in deciles with increasing risk. The best predictions were obtained using the KNN method, in which the AUC was 0.7345 ± 0.0289 (mean plus or minus the standard deviation computed using 10-fold cross-validation); this represents an improvement of 2.71% over the LD case. To analyse how far these results are from the optimal prediction, the optimal prediction value was estimated as the value obtained using a logistic regression model computed over the observed results, which led to a prognosis accuracy of 0.7795. Then, we compared the obtained results to this value and to the LD case using Eq. (13), which measures the percentage improvement in accuracy from the optimal with the respect to the LD procedure. The results are shown in Fig. 3. With respect to the statistically based methods, the MI methods implemented with SAS and WinMICE led to values higher than the comparison case (LD), but the difference was not statistically significant. The same occurred with the very simple method of the mean. In contrast, MI implementation through Amelia and hot-deck imputation led to relatively poor prognosis predictions.

Because the size of the data set is known to influence the effect of the imputation method, we conducted further tests in which the size of the available training data was reduced while the test set

remained unaltered. Fig. 4 shows the AUC values obtained as the training set was reduced by 0–90%, where the performance of the prediction decreased monotonically as the size of the training data set was reduced. The results plotted in Fig. 5, which shows the difference in the observed AUC between the obtained values and the LD approach, are particularly interesting; from this graph, the improvement increased until the training data were reduced by 80%. The increase in improvement highlights the importance of imputation techniques when small data sets are available; this finding suggests that almost any of the imputation techniques used in the present work could provide significant improvements when data sets are small.

We conclude that machine learning techniques may be the best approach to imputing missing values, as they led to statistically significant improvements in prediction accuracy. Imputation techniques depend on the available data and the prediction model used; thus, the present results might not generalise to different data sets.

Acknowledgements

The authors acknowledge support from CICYT (Spain) through grants TIN2005-02984 and grant number TIN2008-04985 (including FEDER funds) and from Junta de Andalucía through grant P06-TIC-01615 and P08-TIC-04026. Support from grants A/6030/06 and A/12805/07 from AECI is also acknowledged. Leonardo Franco acknowledges support from the Spanish Ministry of Science and Innovation (MICIIN) through a Ramón y Cajal fellowship. Fruitful discussions with Dr S.A. Cannas and Dr. F. Tamarit are acknowledged.

References

[1] Cox DR. Regression models and life-tables. Journal of the Royal Statistical Society Series B (Methodological) 1972;34(2):187–220.
[2] Biganzoli E, Boracchi P, Mariani L, Marubini E. Feed-forward neural networks for the analysis of censored survival data: a partial logistic regression approach. Statistics in Medicine 1998;17(10):1169–86.
[3] Burke HB, Goodman PH, Rosen DB, Henson DE, Weinstein JN, Harrell FE, et al. Artificial neural networks improve the accuracy of cancer survival prediction. Cancer 1997;79(4):857–62.
[4] Fujikawa K, Matsui Y, Kobayashi T, Miura K, Oka H, Fukuzawa S, et al. Predicting disease outcome of non-invasive transitional cell carcinoma of the urinary bladder using an artificial neural network model: results of patient follow-up for 15 years or longer. International Journal of Urology 2003;10(3):149–52.
[5] Jerez-Aragonés JM, Gómez-Ruiz JA, Ramos-Jiménez G, Muoz-Pérez J, Alba-Conejo E. A combined neural network and decision trees model for prognosis of breast cancer relapse. Artificial Intelligence in Medicine 2003;27(1):45–63.
[6] Jerez JM, Franco L, Alba E, Llombart-Cussac A, Lluch A, Ribelles N, et al. Improvement of breast cancer relapse prediction in high risk intervals using artificial neural networks. Breast Cancer Research and Treatment 2005;94(3):265–72.
[7] Lisboa PJG, Wong H, Harris P, Swindell R. A bayesian neural network approach for modelling censored data with an application to prognosis after surgery for breast cancer. Artificial Intelligence in Medicine 2003;28(1):1–25.
[8] Lisboa PJG, Taktak AFG. The use of artificial neural networks in decision support in cancer: a systematic review. Neural Networks 2006;19(4):408–15.
[9] Ripley BD, Ripley RM. Neural networks as statistical methods in survival analysis. In: Clinical application of artificial neural networks. Cambridge University Press; 2001. p. 237–55.
[10] Elizondo DA, Góngora MA. Current trends on knowledge extraction and neural networks. In: Duch W, Kacprzyk J, Oja E, Zadrozny S, editors. Artificial neural networks: formal models and their applications—ICANN 2005, vol. 3697, 15th international conference. 2005.p. 485-90.
[11] Little RJA, Rubin DB. Statistical analysis with missing data, 2nd ed., Hoboken, New Jersey: John Wiley & Sons, Inc.; 2002.
[12] Rubin DB. Multiple imputation for nonresponse in surveys. Hoboken, New Jersey: John Wiley & Sons, Inc.; 1987.
[13] Allison PD. Missing data. Thousand Oaks, CA: Sage Publications, Inc.; 2001.
[14] Ibrahim JG, Chen MH, Lipsitz SR, Herring AH. Missing-data methods for generalized linear models: a comparative review. Journal of the American Statistical Association 2005;100(469):332–46.
[15] Manski CF. Partial identification with missing data: concepts and findings. International Journal of Approximate Reasoning 2005;39(2–3):151–65.
[16] Schafer J. Analysis of incomplete multivariate data. London: Chapman & Hall; 1997.
[17] Schafer JL, Graham JW. Missing data: our view of the state of the art. Psychological Methods 2002;7(2):147–77.
[18] Pérez A, Dennis RJ, Gil JFA, Rondón MA, López A. Use of the mean, hot deck and multiple imputation techniques to predict outcome in intensive care unit patients in Colombia. Statistics in Medicine 2002;21(24):3885–96.
[19] Siddique J, Belin TR. Multiple imputation using an iterative hot-deck with distance-based donor selection. Statistics in Medicine 2008;27(1):83–102.
[20] Hawthorne G, Elliott P. Imputing cross-sectional missing data: comparison of common techniques. Australian and New Zealand Journal of Psychiatry 2005;39(7):583–90.
[21] Horton NJ, Lipsitz SR. Multiple imputation in practice: comparison of software packages for regression models with missing variables. The American Statistician 2001;55(3):244–54.
[22] Ambler G, Omar RZ, Royston P. A comparison of imputation techniques for handling missing predictor values in a risk model with a binary outcome. Statistical Methods in Medical Research 2007;16(3):277–98.
[23] Van Buuren S, Brand JPL, Groothuis-Oudshoorn CGM, Rubin DB. Fully conditional specification in multivariate imputation. Journal of Statistical Computation and Simulation 2006;76(12):1049–64.
[24] Kenward MG, Carpenter J. Multiple imputation: current perspectives. Statistical Methods in Medical Research 2007;16(3):199–218.
[25] Brock GN, Shaffer JR, Blakesley RE, Lotz MJ, Tseng GC. Which missing value imputation method to use in expression profiles: a comparative study and two selection schemes. BMC Bioinformatics 2008;9:12.
[26] Nguyen DV, Wang N, Carrol RJ. Evaluation of missing value estimation for microarray data. Journal of Data Science 2004;2:347–70.
[27] Scheel I, Aldrin M, Glad IK, Shrum R, Lyng H, Frigessi A. The influence of missing value imputation on detection of differentially expressed genes from microarray data. Bioinformatics 2005;21(23):4272–9.
[28] Sharpe PK, Solly RJ. Dealing with missing values in neural-network-based diagnostic systems. Neural Computing & Applications 1995;3(2):73–7.
[29] Nordbotten S. Neural network imputation applied to the Norwegian 1990 census data. Journal of Official Statistics 1996;12(4):385–401.
[30] Gupta A, Lam MS. Estimating missing values using neural networks. Journal of the Operational Research Society 1996;47:229–38.
[31] Lakshminarayan K, Harp SA, Samad T. Imputation of missing data in industrial databases. Applied Intelligence 1999;11(3):259–75.
[32] Batista GEAPA, Monard MC. A study of k-nearest neighbour as an imputation method. In: Abraham A, del Solar JR, Köppen M, editors. HIS, vol. 87 of frontiers in artificial intelligence and applications. Santiago, Chile: IOS Press; 2002. p. 251–60.
[33] Hruschka ER, Hruschka ER, Ebecken NFF. Towards efficient imputation by nearest-neighbors: a clustering-based approach. In: AI 2004: advances in artificial intelligence, vol. 3339 of lecture notes in computer science. Springer Berlin/Heidelberg; 2005. p. 513–25
[34] Troyanskaya O, Cantor M, Sherlock G, Brown P, Hastie T, Tibshirani R, et al. Missing value estimation methods for DNA microarrays. Bioinformatics 2001;17(6):520–5.
[35] Fessant F, Midenet S. Self-organising map for data imputation and correction in surveys. Neural Computing & Applications 2002;10(4):300–10.
[36] Piela P. Introduction to self-organizing maps modelling for imputation—techniques and technology. Research in Official Statistics 2002;2:5–19.
[37] Samad T, Harp SA. Self-organization with partial data. Network Computation in Neural Systems 1992;3(2):205–12.
[38] Martín M, Llombart-Cussac A, Lluch A, Alba E, Munarriz B, Tusquets I, et al. Epidemiological study of the geicam group about breast cancer in Spain, El Álamo project. Medicina Clínica 2004;122(1):12–7.
[39] Ruíz A, Lluch A, Martín M, Munárriz B, Antón A, Alba E, et al. Spanish breast cancer research group (GEICAM) population-based study on breast cancer outcomes: El Álamo project. Journal of Clinical Oncology 2005;23(16S):585.
[40] Ake C. Rounding after multiple imputation with non-binary categorical covariates. In: Proceedings of the thirtieth annual SAS Users Group international conference; 2005.
[41] Kalton G, Kasprzyk D. Imputing for missing survey responses. In: Proceedings of the section on survey research methods, annual meeting of the American Statistical Association; 1982. p. 22–31.
[42] Kaiser J. The effectiveness of hot-deck procedures in small samples. In: Proceedings of the section on survey research methods; 1983. p. 523–8.
[43] Rubin DB. Multiple imputation after 18+ years. Journal of the American Statistical Association 1996;91(434):473–89.
[44] Wilson DR, Martinez TR. Improved heterogeneous distance functions. Journal of Artificial Intelligence Research 1997;6:1–34.
[45] Honaker J, King G, Blackwell M. Amelia software website, http://gking.harvard.edu/amelia; 2008 [accessed 15.12.08].
[46] Bishop CM. Pattern recognition and machine learning, information science and statistics. Springer Science+Business Media, LLC; 2007.
[47] Duda RO, Hart PE, Stork DG. Pattern classification, 2nd ed., Wiley-Interscience; 2000.
[48] Mitchell TM. Machine learning. McGraw-Hill; 1997.
[49] Campbell C. Constructive learning techniques for designing neural network systems. In: Neural network systems, techniques and applications. San Diego: Academic Press; 1997. p. 1–54.
[50] Bartlett EB. A dynamic node architecture scheme for layered neural networks. Journal of Artificial Neural Network 1994;1:229–45.

[51] Hirose Y, Yamashita K, Hijiya S. Back-propagation algorithm which varies the number of hidden units. Neural Networks 1991;4(1):61–6.
[52] Setiono R. Feedforward neural network construction using cross validation. Neural Computation 2001;13(12):2865–77.
[53] Sancho-Gómez JL, García-Laencina PJ, Figueiras-Vidal AR. Combining missing data imputation and pattern classification in a multi-layer perceptron. Intelligent Automation and Soft Computing 2009;15(4):539–53.
[54] Kohonen T. Self-organizing maps, 3rd ed., Berlin: Springer Series in Information Sciences, Springer-Verlag; 2001.
[55] García-Laencina PJ, Sancho-Gómez JL, Figueiras-Vidal AR, Verleysen M. K nearest neighbours with mutual information for simultaneous classification and missing data imputation. Neurocomputing 2009;72(7–9):1483–93.
[56] Moller MF. A scaled conjugate-gradient algorithm for fast supervised learning. Neural Networks 1993;6(4):525–33.
[57] Demuth H, Beale M. Neural network toolbox: for use with MATLAB: user's guide. Cochituate Place, 24 Prime Park Way, Natick, MA, USA: The Mathworks; 1993.
[58] Hanley JA, McNeil BJ. The meaning and use of the area under a receiver operating characteristic (roc) curve. Radiology 1982;143(1):29–36.
[59] Hosmer DW, Lemeshow S. Applied logistic regression, Wiley series in probability and statistics. Wiley-Interscience Publication; 2000.
[60] Demsar J. Statistical comparisons of classifiers over multiple data sets. Journal of Machine Learning Research 2006;7:1–30.
[61] Friedman M. The use of ranks to avoid the assumption of normality implicit in the analysis of variance. Journal of the American Statistical Association 1937;32(200):675–701.
[62] Sidak Z, Sen PK, Hajek J. Theory of rank tests, probability and mathematical statistics, 2nd ed., San Diego, CA: Academic Press; 1999.