COMPUTER METHODS AND PROGRAMS IN BIOMEDICINE 108 (2012) 1247–1254

journal homepage: www.intl.elsevierhealth.com/journals/cmpb

WIMP: Web server tool for missing data imputation☆

D. Urda[a,*], *J.L. Subirats*[a], *P.J. García-Laencina*[b], *L. Franco*[a], *J.L. Sancho-Gómez*[c], *J.M. Jerez*[a]

[a] *Departamento de Lenguajes y Ciencias de la Computación, ETSI Informática, University of Málaga, Spain*
[b] *Centro Universitario de la Defensa de San Javier, MDE-UPCT, Spain*
[c] *Departamento de Tecnologías de la Información y las Comunicaciones, Universidad Politécnica de Cartagena, Spain*

ARTICLE INFO

Article history:
Received 11 November 2011
Received in revised form
30 July 2012
Accepted 13 August 2012

Keywords:
Imputation
Missing data
Machine learning
Web application

ABSTRACT

The imputation of unknown or missing data is a crucial task on the analysis of biomedical datasets. There are several situations where it is necessary to classify or identify instances given incomplete vectors, and the existence of missing values can much degrade the performance of the algorithms used for the classification/recognition. The task of learning accurately from incomplete data raises a number of issues some of which have not been completely solved in machine learning applications. In this sense, effective missing value estimation methods are required. Different methods for missing data imputations exist but most of the times the selection of the appropriate technique involves testing several methods, comparing them and choosing the right one. Furthermore, applying these methods, in most cases, is not straightforward, as they involve several technical details, and in particular in cases such as when dealing with microarray datasets, the application of the methods requires huge computational resources. As far as we know, there is not a public software application that can provide the computing capabilities required for carrying the task of data imputation. This paper presents a new public tool for missing data imputation that is attached to a computer cluster in order to execute high computational tasks. The software WIMP (Web IMPutation) is a public available web site where registered users can create, execute, analyze and store their simulations related to missing data imputation.

1. Introduction

In the fields of pattern recognition and machine learning, it is generally accepted that important factors affecting the obtained results regarding the prediction accuracy or percentage of correct classification are the quality of the dataset and the appropriate selection of a learning algorithm. Related to the quality of the data, it is a very common situation to find incomplete datasets containing unknown or missing data. Several strategies or methods can be applied to deal with incomplete datasets. A simple and common strategy is to ignore missing values, reducing the size of the useful dataset. Other valid approaches to deal with incomplete datasets tend to use supervised learning or statistical analysis to impute the missing data so as to use the total number of samples available in the dataset [1,9,16,18,25,27,23,22,21]. In fact, most of the biomedical studies have focussed on developing missing value estimation methods for incomplete biomedical or microarray datasets [35,38,42,20,2,12,19,11,30,5,39,3,36,41,7,10].

☆ http://www.icb.uma.es/wimp.

* *Corresponding author*. Tel.: +34 952132847.
E-mail addresses: durda@lcc.uma.es (D. Urda), jlsubirats@lcc.uma.es (J.L. Subirats), pedroj.garcia@cud.upct.es (P.J. García-Laencina), lfranco@lcc.uma.es (L. Franco), josel.sancho@upct.es (J.L. Sancho-Gómez), jja@lcc.uma.es (J.M. Jerez).
URLs: http://www.cud.upct.es/ (P.J. García-Laencina), http://www.lcc.uma.es (J.M. Jerez).

0169-2607/$ – see front matter © 2012 Elsevier Ireland Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.cmpb.2012.08.006

In concrete, in the area of bioinformatics, there are several research groups working on the application of information contained in microarray datasets to classify either the diagnosis or prognosis of certain diseases according to gene expression signatures corresponding to patients' samples. Unfortunately, a common problem is the quality of microarray datasets and especially those studies using microarray gene expression data. Most of these datasets can be found and downloaded from public websites (Gene Expression Omnibus[1] or Standford Microarray Databases[2]) and, in most cases, they contain incomplete samples due to unknown or missing data. Incomplete microarray data could be caused by administrative errors, defective techniques or punctual technology failures.

A common approach to make the prognosis of a certain disease is the use of machine learning algorithms, and as mentioned before, the performance of these algorithms strongly depends on the quality of the data. In addition to this inconvenient, the research personnel face another disadvantage when working with microarray datasets: the few number of available instances, approximately around 40–80 samples. Thus, the strategy of just ignoring missing values samples in microarray datasets is not recommended because of the small number of samples available and because this simple technique can introduce substantial biases in the study, especially when missing data is not distributed randomly.

Therefore, the problematic of dealing with microarray datasets makes the imputation of unknown or missing data an important task to be considered when applying a machine learning algorithm. Quinlan [24] shows that missing values in either the training data or test data affect prediction accuracy of learning classifiers. Moreover, Luque et al. [17] show the benefits of having a good quality dataset instead of a good learning algorithm. Therefore, the imputation of unknown or missing data on the area of bioinformatics arises as a problem to be dealt especially on this kind of datasets, as it is described in [37,13,14,33,26,4,32,15,31,28].

We next describe some of the standard problems that arise at the time of imputing unknown or missing data on a research study: (i) a good knowledge of the different imputation methods that best fits the type of dataset used in the study should be acquired; (ii) a valid implementation of these methods should be available; (iii) the application of the selected imputation methods may need heavy computational resources, a requirement that may be difficult to be satisfied for small research groups.

Our proposal in this paper is to try to help to overcome these points above described by providing a public website tool, named WIMP (Web IMPutation), that includes several imputation methods and that offers to the scientific community the possibility of applying them to the dataset involved in their study. Further, WIMP incorporates on its backend a powerful computational cluster enabling users to execute the selected imputation methods onto a previously loaded dataset in a reasonable amount of time.

Of course, widely used statistical software, as SPSS[3] or SAS,[4] also incorporate some imputation methods. Nevertheless, SPSS and SAS are commercial software with license keys that are not affordable to every clinicians and researchers, and, overall, these tools only provide imputation methods based on statistical models [40,34,8], but not based on machine learning techniques.

The present paper is structured as follows. Section 2 describes the system architecture of WIMP, including descriptions of the database used, the computational cluster that resides on the backend of WIMP and a brief description of the imputation methods that are currently available. Section 3 presents a case of study where missing data imputation is needed and shows how it can be solved by utilizing the available resources of WIMP. Finally, in Section 4 we provide some conclusions and further improvements that could be done in relationship to the present work.

2. WIMP application

The WIMP application software has been developed using the latest.NET technology [29]. WIMP is a public website application that enables users to log in after being registered in order to impute missing data using several available imputation methods. The main goal of WIMP is to help potential users so that they can focus their efforts on carrying their experiments instead of dedicating resources for searching, understanding and applying the different imputation methods.

Also, in order to speed up the simulations involved in the imputation process, on the backend of the WIMP application (and thanks to the Computational Intelligence in Biomedicine research group of the University of Malaga), a computer cluster is available so the different imputation methods can be executed relatively fast. Basically, whenever a user planifies and launches a new simulation through the website environment, it is actually executed on the computational cluster on the background as a separate process. Finally, and once the process ends, the user is informed via email, obtaining the results of the imputation method applied to the dataset previously loaded.

The complete system consists of:

- A website application developed with the latest.NET technology providing users a friendly and usable environment in order to interact with the system.
- An implementation of an SQL Server 2008 database including all the necessary information concerning users, projects, simulations, files, etc.
- A computational cluster composed by 27 quad-cores nodes PC's with 4 GB of RAM each one, all of them running under the Linux operating system.
- A web service implementation that enables the communication between either the website application or the computationtal cluster and the system database to grant them access to the information stored in it.

[1] http://www.ncbi.nlm.nih.gov/geo/.
[2] http://smd.stanford.edu/.
[3] http://www-01.ibm.com/software/es/analytics/spss/.
[4] http://www.sas.com/.

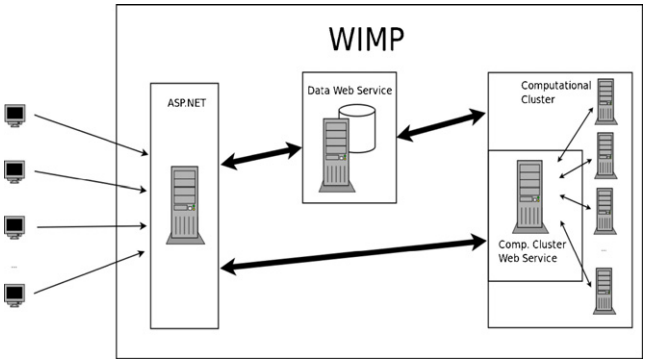

Fig. 1 – System architecture of the WIMP application consisting of three layers (see the text for details).

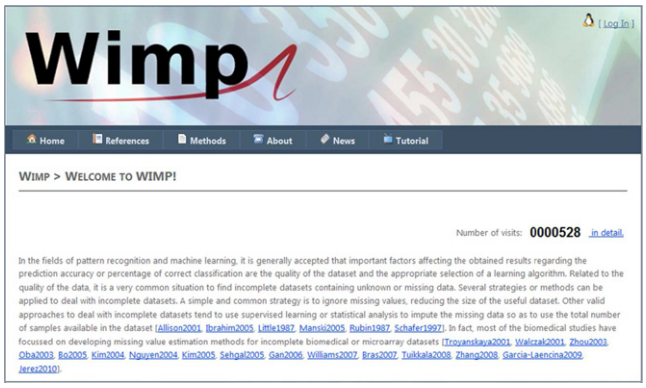

Fig. 2 – Screenshot of the home page of the website application WIMP.

- Another web service implementation that enables the communication between the website application and the computational cluster in order to launch the processes corresponding to every simulation on the computational cluster.

2.1. *System architecture*

Fig. 1 shows the architecture of the whole system. The system is divided into three different layers that interacts between them by the services provided on the web services developed in WIMP. On the left side of Fig. 1, several users may connect to WIMP through the website application. The development of this layer has been done to achieve a great experience to the user's view and several browsers has been taken into account for this purpose. Therefore, users of WIMP are able to navigate with their preferred browser. On the right hand of Fig. 1, we have the computational cluster that is constantly attending to requests send by the website application. Every time that a user planifies and decides to launch a certain imputation method over a dataset, on the background the website application is sending this simulation to the computational cluster to get the final results. Finally, the system database is displayed on the middle of the figure, that is in charge of storing all the information concerning the users, their projects, simulations and files, which is necessary to execute a task on the computational cluster.

2.1.1. *Website application structure*

Fig. 2 shows the main screen of the website application of WIMP. The website application is structured in several tabs or options, as described next:

- LOGIN: A webpage allowing users to log in and log out from the system, or to create a new user account.
- HOME: It contains a brief introduction to the WIMP system, and also information about the problematic of imputing missing or unknown data.
- REFERENCES: List of references used to develop WIMP.
- METHODS: It contains the different imputation methods included in WIMP as well as a brief description of each of them.
- ABOUT: It shows some information about the developers of the system.
- NEWS: This webpage shows relevant and current news related with the application or imputation methods. This tab can only be edited by the administrators.
- TUTORIAL: It includes several tutorials about using the different imputation methods and the application to a concrete dataset.

Once a user is logged into the system, the website application automatically enables the following options:

- Expandable option MY WIMP, where the user can choose among three options:
  1. PROJECTS: Lists all the projects created by the current user, providing options to add, modify or delete projects.
  2. SIMULATIONS: For a given project, this webpage lists all the simulations related to it. It provides different options to check the simulations run by the user, in concrete (i) WAITING: it shows all the simulations of the user that are pending to be executed on the computational cluster for any reason (available resources), (ii) RUNNING: it contains a list of all the simulations of the user that are currently running on the computational cluster, (iii) FINISHED OK: it lists all the simulations of the user that have been satisfactorily completed (in this sense, the website application shows a link that redirects to the simulation webpage in case the user wants to collect the results), and (iv) FINISHED WITH ERRORS: it shows a list of the simulations of the user that have ended with some kind of error.
- MY FOLDER: Offers the users the different options available for file manipulation, allowing them to add, modify or delete data files.

2.1.2. *Database model*

The database model of WIMP is shown in Fig. 3 as an entity-relation diagram. Currently, the system database is an SQL Server 2008 database. WIMP recognizes two different types of account: administrator and generic user profiles. The administrator role can manage the information of all accounts and also add or modify news, settings, etc., while the generic user

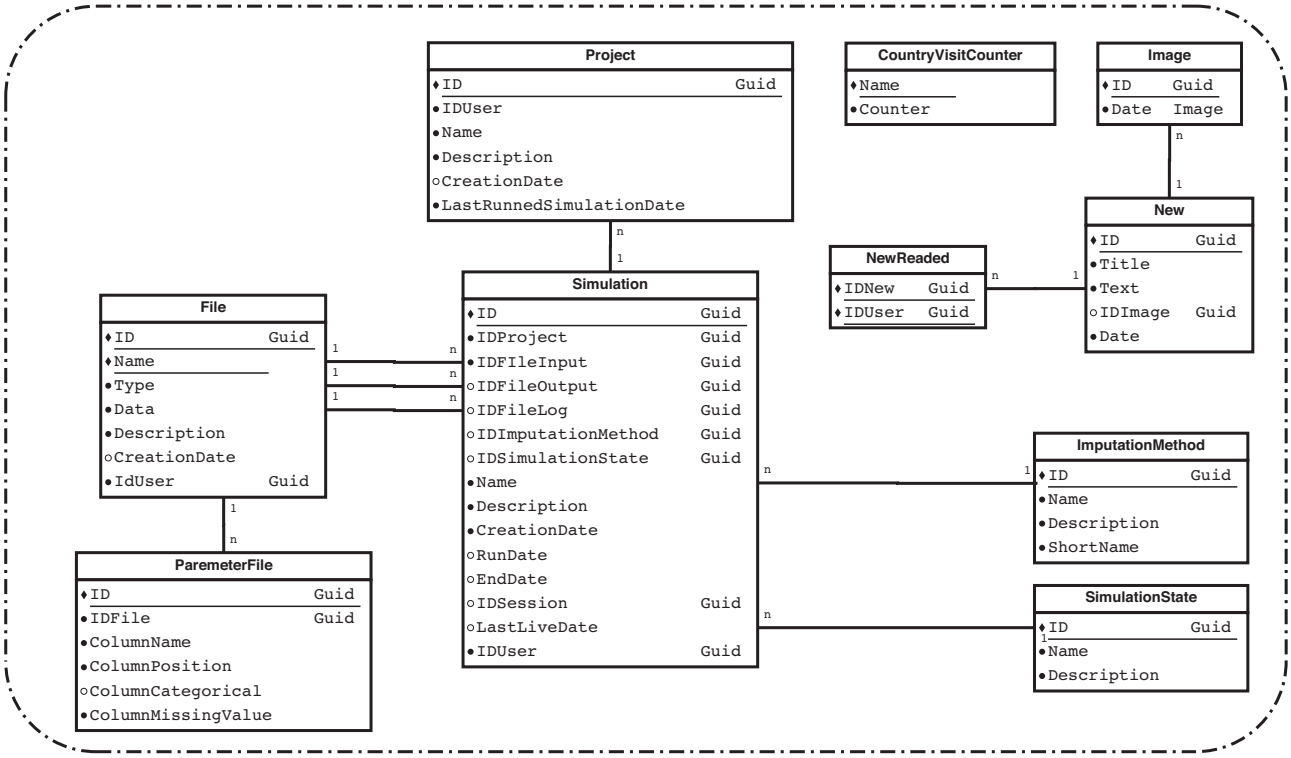

Fig. 3 – Entity-relation diagram corresponding to the system database of WIMP.

profile is restricted to create new projects, simulations and run them on the computational cluster. As it can be observed in Fig. 3, the entity "Simulation" is the nexus between the website application and the computational cluster, where all the information needed to run a simulation on the cluster is stored. Once a simulation correctly finishes, the final results obtained are also stored in the database.

2.1.3. *Computational cluster*

Basically, a computational cluster could be described as a group of linked computers, working together closely so that in many respects they form a single computer. The computational cluster attached to WIMP is composed by 27 Quad-cores nodes and 4 GB of RAM, interconnected on a 1 Gb/s LAN and is property of the ICB research group of the University of Málaga. The server of the cluster hosts an implementation of the web service that is constantly attending requests that may be sent by the website application in order to launch a new simulation on the cluster, whenever there are available resources on it. Considering a properly ended simulation, the computational cluster is also in charge of storing the results of the simulation on the system database through the web service.

2.2. *Imputation methods*

2.2.1. *Mean imputation*

Mean imputation [16] is one of the simplest existing approaches for imputing missing data. It is accomplished simply by averaging the corresponding variables belonging to complete row data cases. Thus, the missing values are systematically substituted by the mean value averaged across all cases containing no such missing variable. For categorical variables the mode is used instead of the mean.

2.2.2. *Hot-deck imputation*

The hot-deck imputation method [16] is implemented as follows: when a missing variable is to be imputed, the right set of candidate donors consisting of row data cases with complete data belonging to the same group is selected. From this set of candidates, the case closest to the receptor one is selected as the donor from which the missing values are taken. For simplicity, the squared Euclidean norm is used as the similarity measure between pairs of patient cases.

2.2.3. *Multiple imputation*

There are several algorithms and techniques for applying multiple imputation [13,25,27]. In concrete, WIMP incorporates the implementation, known as Multivariate Imputation by Chained Equations (MICE) that imputes incomplete multivariate data by Fully Conditional Specification (FCS). This software is freely available on the internet as an R software package.[5]

2.2.4. *SOM imputation*

A Self-Organizing Map (SOM) or Self-Organizing Feature Map (SOFM) is a type of artificial neural network trained unsupervisely to produce a low-dimensional (typically two-dimensional), discretized representation of the input space of the training samples, called a map. SOMs are different from other artificial neural networks in the sense that they use a

[5] http://www.r-project.org/.

neighborhood function to preserve the topological properties of the input space.

As it is explained in detail in [6], the SOM has been adapted to handle and impute missing values by changing the treatment of the input data. In particular, when an observation with missing features is given as input to the map, the missing variables are simply ignored when the distances between the input vector and the nodes are computed. This principle is also applied both for selecting the image-node and for updating weights. Once the SOM can handle missing input data, this model is used for imputation. First, when an incomplete pattern is presented to the SOM, its image-node is chosen ignoring the distances in the missing variables; secondly, an activation group composed of image-node's neighbours is selected; and finally, each imputed value is computed based on the weights of the activation group of nodes in the missing dimensions.

2.2.5. *MLP imputation*

A Multi-Layer Perceptron (MLP) is a feed-forward artificial neural network model that maps sets of input data onto a set of appropriate outputs. An MLP consists of multiple layers of nodes in a directed graph, with each layer fully connected to the next one. Except for the input nodes, each node is a neuron (or processing element) with a nonlinear activation function. MLP utilizes a supervised learning technique called back propagation for training the network.

MLP imputation [6] consists of training an MLP architecture using only the complete cases as a regression model: given D input features, each incomplete attribute is learned (it is used as output) by means of the $(D-1)$ other attributes given as inputs. The MLP outputs are used to impute unknown values given the observed ones. When missing items appear in several attributes, several MLP schemes have to be designed, one per missing variables combination.

2.3. *Workflow in WIMP*

The workflow diagram that follows every process planified and executed in WIMP is shown in Fig. 4. Whenever a user planifies a simulation, WIMP internally stores in the database all the information concerning this simulation (files, datasets, etc.). In this sense, the first state assigned to a simulation that has been just planified by a certain user is "Wait". This means that the task is pending for available resources on the computational cluster in order to be executed.

Next, the website application sends a request to the computational cluster with the ID of the simulation that the user wants to execute on it. Therefore, the web service that resides on the server of the computational cluster attends this request and extracts all the information related to this simulation launching it on an available node of the cluster. At this moment, the status of the simulation is changed to "Run".

Once the simulation finishes, the computational cluster is in charge of collecting the final results, storing them into the system database and finally sending an e-mail to the user that launched this simulation informing that the results are available to be downloaded from the website application. All these processes are obviously achieved through the web service that resides on the server of the cluster.

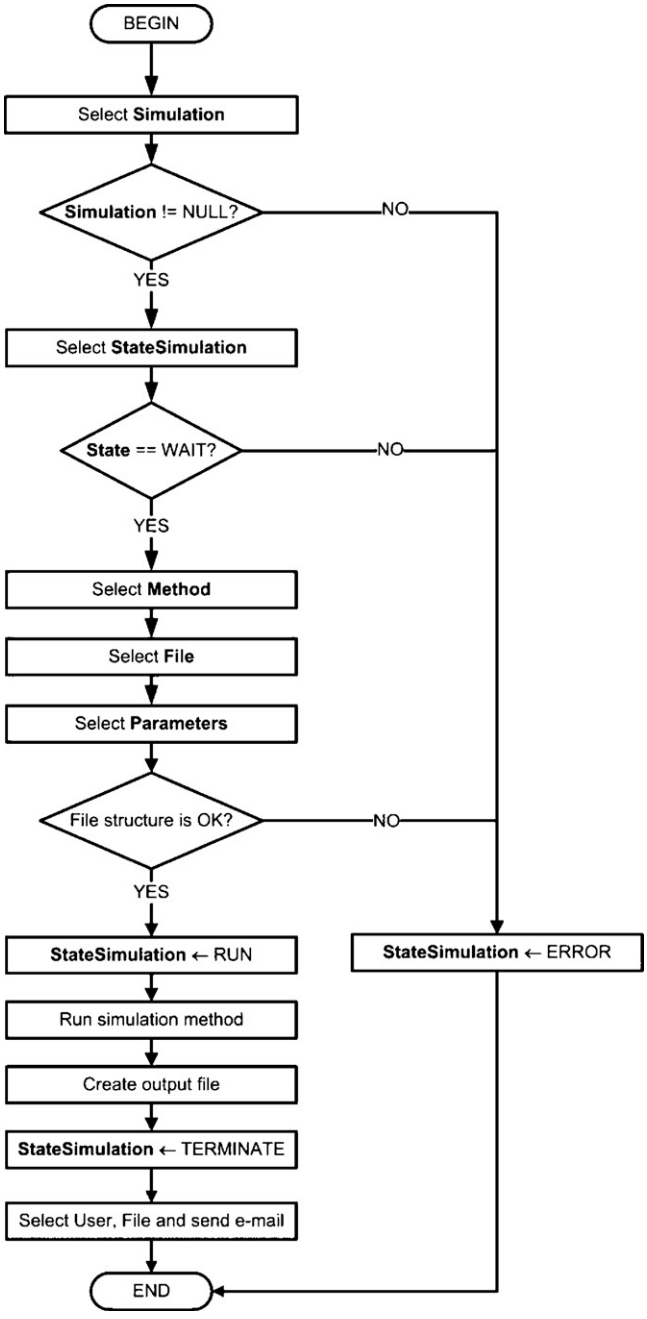

Fig. 4 – Flow diagram of the process executed in the computational cluster.

In case any error occurs, i.e., datasets with incorrect format or mistakes on values of some parameters of the imputation method selected, then the state of the simulation is immediately changed to "Error". Moreover, it is stored into the system database a log file and an e-mail is sent to the user that launched this simulation in order to report this situation. Thus, the user could access to the log file through the website application and try to get over, if desired, the encountered problems.

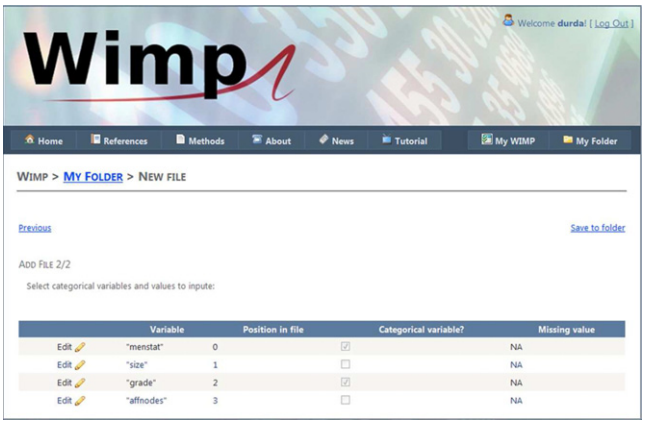

Fig. 5 – Screenshot of the loading step for processing a file, in which the type of each variable has to be selected together with the symbol used in the file for marking missing data.

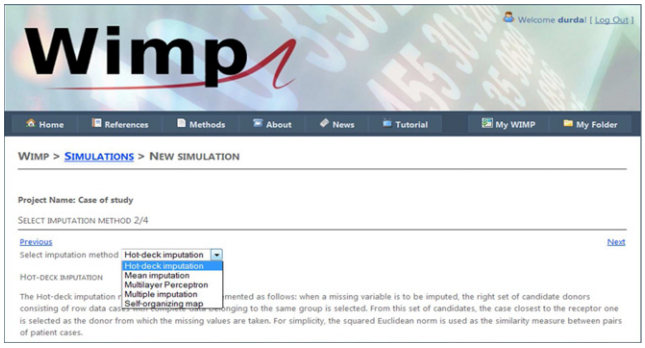

Fig. 6 – Screenshot of the WIMP application when one of the available imputation methods is selected.

3. A case of study

In order to provide an example of missing data imputation using our WIMP application, a real biomedical dataset has been selected consisting of 935 oncologic patient samples and each of them describing four features related to the disease of the study. These features are: Menopause state (V1), categorical feature with two possible values ("premenopausal" or "postmenopausal"); tumor size (V2), quantitative feature that measures the size of the tumor; grade (V3), categorical feature that indicates the grade of the tumor; and number of affected nodes (V4), quantitative feature that measures the number of nodes involved in the disease.

Within the samples of the dataset, missing values are located on every feature with the symbol "NA". Notice that actually WIMP allows users to indicate a symbol for the missing value of each feature of the dataset but, in order to simplify this example, the same symbol "NA" is used for the missing value of every feature of the dataset.

Once the user has the dataset ready to apply an imputation method, it must be loaded as a first step of the imputation process. This step, shown in Fig. 5, stores the dataset on the system database along some extra information like the type of every feature (quantitative or categorical) or the symbol of the missing value of each feature.

After this first step is finished, the user must choose one of the five imputation methods that are available in WIMP at the moment (see Fig. 6). Depending on the selected method, some extra parameters should be configured in order to launch the simulations.

Original dataset

V1	V2	V3	V4
PREMENOPAUSAL	2,5	G2	4
POSTMENOPAUSAL	5	NA	0
PREMENOPAUSAL	1,5	G1	0
POSTMENOPAUSAL	1,5	G2	1
POSTMENOPAUSAL	2,5	G2	3
POSTMENOPAUSAL	2,5	G2	7
POSTMENOPAUSAL	5	G3	8
POSTMENOPAUSAL	3,8	G2	7
POSTMENOPAUSAL	2,5	G2	1
PREMENOPAUSAL	0,6	G2	0
POSTMENOPAUSAL	1,2	G2	0
POSTMENOPAUSAL	2,5	NA	0
POSTMENOPAUSAL	2,5	G2	7
PREMENOPAUSAL	0,5	NA	0
NA	1	NA	0
POSTMENOPAUSAL	NA	NA	12

Mean imputation

V1	V2	V3	V4
PREMENOPAUSAL	2,5	G2	4
POSTMENOPAUSAL	5	G1	0
PREMENOPAUSAL	1,5	G1	0
POSTMENOPAUSAL	1,5	G2	1
POSTMENOPAUSAL	2,5	G2	3
POSTMENOPAUSAL	2,5	G2	7
POSTMENOPAUSAL	5	G3	8
POSTMENOPAUSAL	3,8	G2	7
POSTMENOPAUSAL	2,5	G2	1
PREMENOPAUSAL	0,6	G2	0
POSTMENOPAUSAL	1,2	G2	0
POSTMENOPAUSAL	2,5	G1	0
POSTMENOPAUSAL	2,5	G2	7
PREMENOPAUSAL	0,5	G1	0
POSTMENOPAUSAL	1	G1	0
POSTMENOPAUSAL	2,5	G1	12

Hot-deck imputation

V1	V2	V3	V4
PREMENOPAUSAL	2,5	G2	4
POSTMENOPAUSAL	5	G3	0
PREMENOPAUSAL	1,5	G1	0
POSTMENOPAUSAL	1,5	G2	1
POSTMENOPAUSAL	2,5	G2	3
POSTMENOPAUSAL	2,5	G2	7
POSTMENOPAUSAL	5	G3	8
POSTMENOPAUSAL	3,8	G2	7
POSTMENOPAUSAL	2,5	G2	1
PREMENOPAUSAL	0,6	G2	0
POSTMENOPAUSAL	1,2	G2	0
POSTMENOPAUSAL	2,5	G3	0
POSTMENOPAUSAL	2,5	G2	7
PREMENOPAUSAL	0,5	G2	0
PREMENOPAUSAL	1	G2	0
POSTMENOPAUSAL	6,8	G3	12

Fig. 7 – Missing data imputation in some incomplete vectors from the biomedical dataset by using the "Mean imputation" and "hot-deck imputation" methods implemented on WIMP.

When the user launches a new simulation applying the chosen imputation method to a certain dataset, the simulation is automatically queued until there are resources available on the computational cluster. This process is transparent to the user who can continue loading, configuring and launching new simulations using WIMP. Through the options offered, the user can check at any time processes that are queued, running or have already finished.

Finally, WIMP sends an e-mail to the user when a simulation ends, reporting this fact with some useful related information. The user could get the resulting file dataset with the imputed data through the ended simulations option in WIMP. In Fig. 7, it can be appreciated part of the dataset with some missing data to be imputed (left side) and the resulting part of the dataset after applying the mean imputation method (center) and hot-deck imputation method (right side) with the missing values filled in.

4. Conclusions

In this work, we present a novel web application called WIMP that offers the scientific community the possibility of imputing missing data to a given input dataset in a user-friendly way. The statistical imputation methods included are mean, hot-deck and multiple imputations. Besides other imputation methods based on machine learning approaches are available such as MLP or SOM. We believe that WIMP could be a useful tool for researchers as up to now, in order to impute missing data, they have to develop (or download in some cases) the different imputation methods with the consequent extra effort. Moreover, as some imputation methods need high computational requirements, the availability of WIMP can be a great advantage for individual researchers or small research groups that do not have enough computational resources.

WIMP arises as a free available resource on the internet that integrates the most common imputation methods. The system can help the research community by saving users from developing, downloading or fully understanding the imputation method that they are going to use. In this sense, users will only need to upload the dataset and add some extra information related to the features of the dataset (type and symbol for missing values) in order to launch a simulation and get the results as a file with the imputed data. Furthermore, WIMP incorporates on its backend a computational cluster where these imputation methods are executed, providing results in a reasonable amount of time, even for complex machine learning based approaches.

In the near future, WIMP will incorporate more imputation methods, such as K-Nearest Neighbours (KNN) or Local Least Squares (LLS) techniques, and a series of modifications in terms of usability and user-experience. In this sense, feedback from users will be much appreciated.

Acknowledgements

The authors acknowledge support from MICIIN (Spain) through Grants TIN2008-04985 and TIN2010-16556 (including FEDER funds) and from Junta de Andalucía through Grant P08-TIC-04026.

Appendix A. Glossary of acronyms

FCS	Fully Conditional Specification
KNN	K-Nearest Neighbours
LLS	Local Least Squares
MICE	Multivariate Imputation by Chained Equations
MLP	Multilayer Perceptron
SOM	Self Organizing Maps
WIMP	Web IMPutation

REFERENCES

[1] P.D. Allison, Missing Data, Sage Publications, Thousand Oaks, 2001.
[2] T.H. Bo, B. Dysvik, I. Jonassen, LSimpute: accurate estimation of missing values in microarray data with least squares methods, Nucleic Acids Research 32 (2004) e34+.
[3] L.P. Brás, J.C. Menezes, Improving cluster-based missing value estimation of DNA microarray data, Biomolecular Engineering 24 (2007) 273–282.
[4] M.K. Choong, M. Charbit, H. Yan, Autoregressive-model-based missing value estimation for DNA microarray time series data, IEEE Transactions on Information Technology in Biomedicine 13 (2009) 131–137.
[5] X. Gan, A.W.C. Liew, H. Yan, Microarray missing data imputation based on a set theoretic framework and biological knowledge, Nucleic Acids Research 34 (2006) 1608–1619.
[6] P.J. García-Laencina, J.L. Sancho-Gómez, A.R. Figueiras-Vidal, Pattern classification with missing data: a review, Neural Computing & Applications 19 (2010) 263–282.
[7] P.J. García-Laencina, J.L. Sancho-Gómez, A.R. Figueiras-Vidal, M. Verleysen, K nearest neighbours with mutual information for simultaneous classification and missing data imputation, Neurocomputing 72 (2009) 1483–1493.
[8] IBM, SPSS Missing Values 17.0, 2010. http://www.helsinki.fi/komulain/Tilastokirjat/IBM-SPSS-Missing-Values.pdf.
[9] J.G. Ibrahim, M.H. Chen, S.R. Lipsitz, A.H. Herring, Missing-data methods for generalized linear models: a comparative review, Journal of the American Statistical Association 100 (2005) 332–346.
[10] J.M. Jerez, I. Molina, P.J. García-Laencina, E. Alba, N. Ribelles, M. Martín, L. Franco, Missing data imputation using statistical and machine learning methods in a real breast cancer problem, Artificial Intelligence in Medicine 50 (2010) 105–115.
[11] H. Kim, G.H. Golub, H. Park, Missing value estimation for DNA microarray gene expression data: local least squares imputation, Bioinformatics 21 (2005) 187–198.
[12] K.Y. Kim, B.J. Kim, G.S. Yi, Reuse of imputed data in microarray analysis increases imputation efficiency, BMC Bioinformatics 5 (2004) 160.
[13] K.F. Lam, Y. Xu, T.L. Cheung, A multiple imputation approach for clustered interval-censored survival data, Statistics in Medicine 29 (2010) 680–693.
[14] Y. Li, A. Ngom, L. Rueda, Missing value imputation methods for gene-sample-time microarray data analysis, Computational Intelligence in Bioinformatics and Computational Biology (CIBCB) (2010) 1–7.
[15] J.H. Lin, P.J. Haug, Exploiting missing clinical data in Bayesian network modeling for predicting medical problems, Journal of Biomedical Informatics 41 (2008) 1–14.

[16] R.J.A. Little, D.B. Rubin, Statistical Analysis with Missing Data Wiley Series in Probability and Statistics, 1st ed., Wiley, New York, USA, 1987.
[17] R.M. Luque, D. Elizondo, E. López-Rubio, E. Palomo, GA-based feature selection approach in biometric hand systems, in: International Joint Conference on Neural Networks, 2011, pp. 246–253.
[18] C. Manski, Partial identification with missing data: concepts and findings, International Journal of Approximate Reasoning 39 (2005) 151–165.
[19] D.V. Nguyen, N. Wang, R.J. Carroll, Evaluation of missing value estimation for microarray data, Journal of Data Science 2 (2004) 347–370.
[20] S. Oba, M.A. Sato, I. Takemasa, M. Monden, K. Matsubara, S. Ishii, A Bayesian missing value estimation method for gene expression profile data, Bioinformatics 19 (2003) 2088–2096.
[21] Y. Qin, Q. Lei, On empirical likelihood for linear models with missing responses, Journal of Statistical Planning and Inference 140 (2010) 3399–3408.
[22] Y. Qin, L. Li, Q. Lei, Empirical likelihood for linear regression models with missing responses, Statistics & Probability Letters 79 (2009) 1391–1396.
[23] Y. Qin, J.N.K. Rao, Q. Ren, Confidence intervals for marginal parameters under fractional linear regression imputation for missing data, Journal of Multivariate Analysis 99 (2008) 1232–1259.
[24] J.R. Quinlan, C4.5: Programs for Machine Learning, Morgan Kaufmann Series in Machine Learning, 1st ed., Morgan Kaufmann, Burlington, Massachusetts, 1992.
[25] D.B. Rubin, Multiple Imputation for Nonresponse in Surveys, John Wiley & Sons, New York, USA, 1987.
[26] C. Ryan, D. Greene, G. Cagney, P. Cunningham, Missing value imputation for epistatic MAPs, BMC Bioinformatics 11 (2010) 197.
[27] J.L. Schafer, Analysis of Incomplete Multivariate Data, Chapman and Hall, London, 1997.
[28] I. Scheel, M. Aldrin, I.K. Glad, R. Sørum, H. Lyng, A. Frigessi, The influence of missing value imputation on detection of differentially expressed genes from microarray data, Bioinformatics 21 (2005) 4272–4279.
[29] H. Schildt, C# 4.0: The Complete Reference, McGraw Hill, USA, 2010.
[30] M.S.B. Sehgal, I. Gondal, L.S. Dooley, Collateral missing value imputation: a new robust missing value estimation algorithm for microarray data, Bioinformatics 21 (2005) 2417–2423.
[31] M.S.B. Sehgal, I. Gondal, L.S. Dooley, R. Coppel, Ameliorative missing value imputation for robust biological knowledge inference, Journal of Biomedical Informatics 41 (2008) 499–514.
[32] N.A. Setiawan, P.A. Venkatachalam, A.F.M. Hani, A comparative study of imputation methods to predict missing attribute values in coronary heart disease data set, in: 4th Kuala Lumpur International Conference on Biomedical Engineering 2008, vol. 21, IFMBE Proceedings, 2008, pp. 266–269.
[33] J. Shi, Z. Luo, Missing value estimation for DNA microarray gene expression data with principal curves, in: International Conference on Bioinformatics and Biomedical Technology (ICBBT), 2010, pp. 262–265.
[34] Z. Shuping, L. Jane, Z. Xingshu, A SAS(r) Macro for Single Imputation, 2008, http://www.lexjansen.com/pharmasug/2008/sp/sp10.pdf.
[35] O. Troyanskaya, M. Cantor, G. Sherlock, P. Brown, T. Hastie, R. Tibshirani, D. Botstein, R.B. Altman, Missing value estimation methods for DNA microarrays, Bioinformatics 17 (2001) 520–525.
[36] J. Tuikkala, L.L. Elo, O.S. Nevalainen, T. Aittokallio, Missing value imputation improves clustering and interpretation of gene expression microarray data, BMC Bioinformatics 9 (2008) 202.
[37] B. Twala, M. Phorah, Predicting incomplete gene microarray data with the use of supervised learning algorithms, Pattern Recognition Letters 31 (2010) 2061–2069.
[38] B. Walczak, D. Massart, Dealing with missing data: Part ii, Chemometrics and Intelligent Laboratory Systems 58 (2001) 15–27.
[39] D. Williams, X. Liao, Y. Xue, L. Carin, B. Krishnapuram, On classification with incomplete data, IEEE Transactions on Pattern Analysis and Machine Intelligence 29 (2007) 427–436.
[40] Y.C. Yuan, Multiple Imputation for Missing Data: Concepts and New Development, SAS Institute Inc., Rockville, MD, 2000, http://support.sas.com/rnd/app/papers/multipleimputation.pdf.
[41] X. Zhang, X. Song, H. Wang, H. Zhang, Sequential local least squares imputation estimating missing value of microarray data, Computers in Biology and Medicine 38 (2008) 1112–1120.
[42] X. Zhou, X. Wang, E.R. Dougherty, Missing-value estimation using linear and non-linear regression with Bayesian gene selection, Bioinformatics 19 (2003) 2302–2307.