MIDA: a Web Tool for MIssing DAta Imputation based on a Boosted and Incremental Learning Algorithm

Giovanni Acampora and Autilia Vitiello
Department of Physics "Ettore Pancini"
University of Naples Federico II
Naples, Italy
{giovanni.acampora, autilia.vitiello}@unina.it

Roberta Siciliano
Department of Industrial Engineering
University of Naples Federico II
Naples, Italy
roberta.siciliano@unina.it

***Abstract*—One of the main issues in machine learning is related to the quality of data used to efficiently train statistical models for classification/regression tasks. Among these issues, the presence of missing values in data sets is particularly prone in affecting the accuracy performance of learning methods. As a consequence there is a strong emergence of software tools aimed at supporting machine learning users in "filling-in" their data sets before inputting them to training algorithms. This paper bridges this gap by introducing a web-based tool for MIssing DAta imputation (MIDA) based on a novel supervised learning method, namely Generalized Boosted Incremental Non Parametric Imputation algorithm (G-BINPI), able to address the missing values issue in scenarios where a "missing at random" assumption occurs. The proposed approach enables machine learning users to remotely imputing their data sets by means of an intuitive graphical user interface. As highlighted in the experimental section, the proposed approach yields better performance than conventional approaches for missing data imputation on different benchmark data sets.**

I. Introduction

Machine learning is the most used set of artificial intelligence techniques aimed at inferring new knowledge from historical data. The rise of machine learning as a key set of methodologies for data analysis is mainly related to the improvement of technology for data storage. As an example, cloud computing architecture enables the efficient storage of huge amount of data, which is easily accessible both by human users and by algorithms dedicated to the extraction of knowledge, enabling the exploitation of these techniques in a plethora of different application scenarios.

However, in spite of its large use in different domains, machine learning techniques could be negatively affected by the quality of datasets used to train the computational models for classification/regression in a given scenario. As a consequence, methods for data pre-processing become of crucial importance in enhancing the performance of machine learning algorithms, by improving the quality of training/testing data sets. In this scenario, approaches for data editing implement suitable methods for performing different improvement operations on data, such as editing (i.e. clean-up), imputing (fill-in) missing or contradictory data, and merging (fusion) data from different sources. Specifically, missing data is one of the most frequent issue to address to enhance the quality of data sets, due to a multitude of reasons why they occur, ranging from human errors during data entry, incorrect sensor readings, to software bugs in the data processing pipeline. As a consequence, there is a strong emergence for introducing innovative and usable software tools allowing machine learning users to efficiently fill-in their datasets.

In this paper a web-based tool, called MIDA (MIssing DAta Imputation), is introduced to address the missing data imputation problem in scenarios where a "missing at random" assumption occurs [1]. The proposed web-tool MIDA embeds a generalised version of the supervised learning approach, introduced by D'Ambrosio, Aria and Siciliano [2], to efficiently manage missing data. The original method, named Boosted Incremental Non Parametric Imputation Algorithm (BINPI), is located within the framework of Vapnik's theory on Statistical Learning [3] [4], and it represents an alternative to the Rubin's probabilistic approach to missing data imputation [5]. The generalised nature of MIDA rises from the providing the possibility to use different types of estimators as weak learners and not only decision trees.

MIDA is accessible to everyone regardless from their computer skills thanks to a user-friendly web interface. Thanks to this, any end-user is able to launch a missing data imputation in remote and get back the result. The results of the imputation data procedure will be displayed on the web page and sent via mail. In literature, few software tools exist for implementing missing data imputation procedures. These include R packages[1] and generalized tools for machine learning such as KEEL [7] [8] [9]. However, differently from all literature software tools, MIDA makes available a missing data imputation procedure to a wide and heterogeneous scientific community thanks to the fact that no programming skills and software installation are required. On the other side, the goodness of the imputation procedure is shown in an experimental session where MIDA outperforms two conventional approaches in

[1] https://cran.r-project.org/web/views/MissingData.html

978-1-7281-6932-3/20/$31.00 ©2020 IEEE

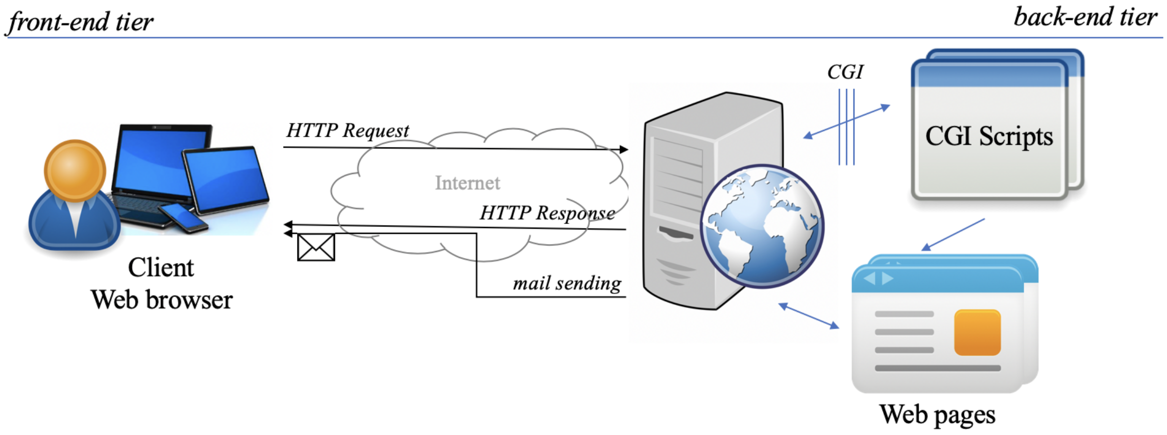


Fig. 1. MIDA architecture [6]

managing missing data of well-known datasets.

This paper is structured in the following way. In Section II, missing data imputation problem is presented. The core of the paper is the Section III, where the architecture of MIDA and the implemented generalized missing data imputation algorithm are described. Before concluding in Section V, the details about the experimental settings and results are reported in Section IV.

II. Missing data imputation problem

The missing data problem is arguably the most common issue encountered by machine learning practitioners when analyzing real-world data [10]. In detail, missing data occur when no data value is stored for one or more measurements of a given variable. There are several reasons why an attribute value is missing. For example, in the emerging Internet of Things architectures, missing values for a variable can occur because of connectivity problems of the corresponding sensor.

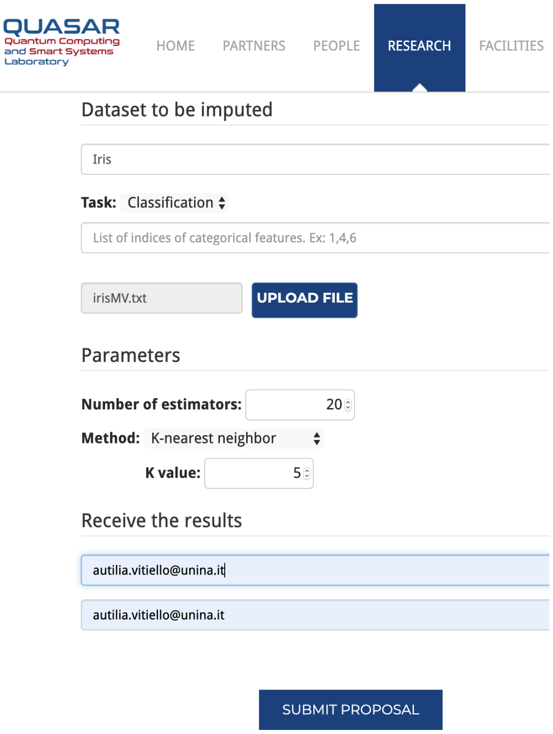
The goodness of data analysis does depend on the way missing data are treated in data pre-processing. Imputation is the replacement of a missing or incorrectly reported item using logical edits or statistical procedures [1] [5]. In other words, imputation replaces a missing or incorrect data item with an *educated guess*. Aim of this paper is that a learning machine automatically replaces the *inconsistent* item with a consistent value. Missing data mechanisms are categorized in three types: (i) Missing Completely At Random (MCAR) where the probability that an observation is missing is unrelated to the value of the variable or to the value of any other variables (in this case missing values cannot predicted any better with all information observed or not); (ii) Missing At Random (MAR) where data meet the requirement that missing-ness does not depend on the value of the variable after controlling for another variable; (iii) Not Missing At Random (NMAR) where the reason for observations being missing still depends on the unseen observations themselves and, as a consequence, cannot be imputed.

Starting from this analysis, the proposed missing data imputation aimed at addressing MAR scenario where machine learning techniques can catch on thanks to the fact the missingness only depends on the complete part of the dataset and not on the missing values themselves. In other words, missing at random condition reflects a pattern of missing data related to the observed data in the data set. Thus, in this scenario, missing data imputation can be treated as a *supervised learning process*. In particular, the lower is the generalization error provided by the learning machine in imputing missing data, the better is the method used to solve the problem and the better is the solution achieved. In detail, as it will describe in the next section, MIDA implements a boosting algorithm where any supervised methods can be used as weak learners.

III. Our proposal for missing data imputation

MIDA is a web-based tool that permits 1) to perform a missing data imputation procedure on a given dataset by using a set of user's inputs, 2) to obtain the corresponding imputed dataset; 3) to display the quality of the imputed dataset by considering different supervised learning methods; 4) to compare the obtained quality with two conventional approaches. MIDA implements the Generalized BINPI algorithm (G-BINPI) as imputation data mechanism. The variables of the data set to be imputed can be both numerical and categorical and the task to be accomplished can be both regression and classification according to the kind of the target variable.

MIDA is accessible by means of any web browser and it manages the requests of users thanks to the architecture displayed in Fig. 1 similar to that proposed in [6]. In detail, the web browser contacts the web server using the HTTP connection over Internet. The web server receives the request and calls a Common Gateway Interface (CGI) script. In turn, the CGI script performs the imputation data mechanism on the given dataset by using all user's input parameters. Then, it builds a HTML page that will be displayed to the user with results dynamically produced. Moreover, the CGI script uses other hardware resources of the web server such as the server email SMTP in order to send results to user's email address, too. Hereafter, a description of the main architecture tiers of MIDA will be given.



Dataset to be imputed

Iris

Task: Classification

List of indices of categorical features. Ex: 1,4,6

irisMV.txt

Parameters

Number of estimators: 20

Method: K-nearest neighbor

K value: 5

Receive the results

autilia.vitiello@unina.it

autilia.vitiello@unina.it

Fig. 2. MIDA interface

A. Front-end tier

The web tool is accessible at the following link: http://quasar.unina.it/missingDataImputation.html. Fig. 2 shows the graphical web interface that permits users to introduce all information useful to run the proposed data imputation procedure implemented on server-side. In detail, this information includes:

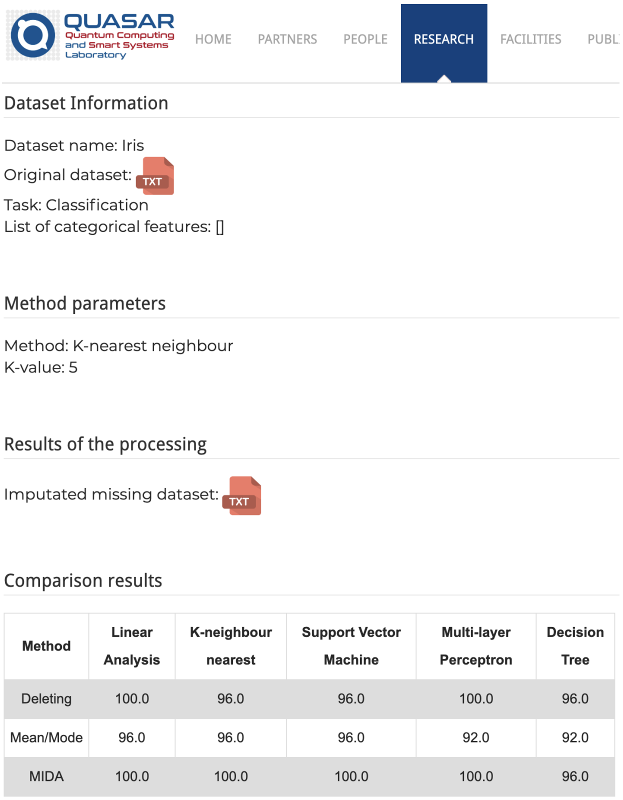
- the path of the file containing the dataset to be imputed. Thanks to this information, the file is uploaded and processed on server-side. The file should have an extension .txt or .csv. The value used for representing missing data can be one of the most typical ones, namely ”null”, ”?”, ”NaN”;
- the choice of the task between classification and regression. Indeed, MIDA manages datasets whose the target variable can be both numerical (in the case of regression) and categorical (in the case of classification). This information is necessary to run different supervised methods according to the task and show preliminary results on the imputed datasets;
- the list of categorical variables. This information is useful to select between classifiers or regressor methods when missing values of the variable are to be imputed. If this list is empty, all variables will be considered as numerical ones;
- the number of the estimators to be used in the boosting procedure implemented in the proposed missing data imputation mechanism;
- the estimator to be used in the boosting procedure of the implemented missing data imputation mechanism. It is possible to select among five methods (Linear approach, K-nearest neighbour, Support Vector Machine, Multi-layer Perceptron and Decision Tree). MIDA runs the version of these algorithms for classification or regression according to the nature of the variable to be imputed;
- hyper-parameters of the selected estimator. In detail, for each estimator, it is possible to insert information as follows:
  - Linear approach: Linear Discriminant Analysis will be executed for categorical variables and Linear regression for numerical ones without setting specific parameters;
  - K-nearest neighbour: the K value (from 1 to 11) both in the classification and regression case;
  - Multi-layer perceptron: the *number of hidden layers* (from 1 to 5), the *number of neurons* for the hidden layers (from 1 to 100), the *tolerance error* (from 10^{-7} to 0.1), the *maximum number of iterations* (from 1 to 10^4) and the *learning rate* (from 10^{-6} to 10^4) both in the classification and regression case;
  - Support Vector Machine: the C value (from 10^{-6} to 10^5), the *tolerance error* (from 10^{-7} to 0.1), the *maximum number of iterations* (from 1 to 10^4) and information about the kernel both in the classification and regression case. Among kernels, it is possible to select the *linear*, the *polynomial* or the *gaussian* one. Then, for the polynomial kernel it is possible to set the *degree* (ranging from 1 to 10), whereas, for the gaussian one it is possible to set the *gamma* value (ranging from 10^{-6} to 10^5);
  - Decision Tree: the *maximum depth* (from 1 to 500) and the *criterion* to measure the quality of a split (*gini* and *entropy* in the classification case and the *Mean Squared Error* (MSE) and the *Mean Absolute Error* (MAE) in the regression case).
- the user email that permits MIDA to send results to the user in an asynchronous way.

Once submitted the form in Fig. 2, MIDA web server handles the request and computes a response in the form of a web page. The displayed web page contain a report of the imputation data mechanism procedure. Fig. 3 shows this web page useful to download the imputed dataset and to visualize some preliminary results related to its quality. Moreover, since the imputation data imputation procedure could work also for several seconds, MIDA exploits the content of the email field of the submitted form to send users the results of the computation via email. So, it is not necessary that users wait the results appear on web page.

B. back-end tier

The web server receives the request by the web client and handles it by calling a CGI script written in Python [2]. The CGI script computes the imputed dataset by applying the G-BINPI

[2] https://www.python.org/



Dataset Information

Dataset name: Iris
Original dataset: TXT
Task: Classification
List of categorical features: []

Method parameters

Method: K-nearest neighbour
K-value: 5

Results of the processing

Imputated missing dataset: TXT

Comparison results

Method	Linear Analysis	K-neighbour nearest	Support Vector Machine	Multi-layer Perceptron	Decision Tree
Deleting	100.0	96.0	96.0	100.0	96.0
Mean/Mode	96.0	96.0	96.0	92.0	92.0
MIDA	100.0	100.0	100.0	100.0	96.0

Fig. 3. MIDA report web page

algorithm, a variant of the work proposed in [2]. This work is based on two main concepts: an incremental approach based on a lexicographic order and a boosting algorithm. In detail, the incremental approach consists of using a lexicographic ordering where each column presenting missing values, at turn, plays the role of target variable to be imputed by the complete set of variables playing the role of predictors. After imputation it concurs to form the complete set of predictors used for the subsequent imputation [11] [12] [13]. The incremental imputation of each variable at time (instead of each single data at time) allows a more efficient algorithm, thus reducing the computational cost of the overall incremental procedure. Moreover, the boosting algorithm is used to perform the supervised learning process described in Section II. As estimator, FAST tree partitioning [14] is used. The choice fell on this method because boosting works better than other ensembles (i.e. bagging) since, while reducing the variance it does not increase the complexity of the space of the learning process. MIDA generalizes this algorithm to include any supervised machine learning algorithm as estimator. In particular, MIDA makes available five supervised machine learning algorithms: Support Vector Machine, K-nearest neighbour, Multi-layer perceptron, Decision Tree and linear approaches. To conclude, Table I reports the pseudo-code of the G-BINPI.

IV. Experiments and Results

MIDA enables researchers coming from different domain areas to impute a dataset with missing values easily thanks to a web interface. However, this advantage would be not enough if the imputed dataset is not characterized by a good quality. Therefore, this section is devoted to show the good performance of the proposed imputation data mechanism by means of a set of experiments involving a set of well-known datasets and a comparison with two conventional approaches. Hereafter, more details about the experimental configuration and the results are given.

A. Experimental set-up

The experiments involve well-known datasets from the UCI Machine Learning Database Repository[3] where missing values are induced as described in the paper [15]. Table II shows the features in terms of number of attributes, instances and classes, and the amount of missing values in percentage of the used datasets. As it is possible to see, the datasets have been selected to cover a different set of values for the aforementioned features.

The evaluation of the proposed method is carried out by using a test dataset and comparing the performance of MIDA with that obtained by two conventional approaches. The first approach, denoted as *mean/mode*, consists of imputing missing values by using mode value for the categorical variables and mean value for numerical ones. The second conventional method, denoted as *deleting*, consists of imputing the dataset by removing all instances containing missing values. The performance measure is the quality in prediction that a supervised method obtains on the test dataset when trained with the imputed dataset. To build the test dataset, the amount of 25% of the number of complete instances has been selected randomly and extracted by the original dataset. The performance metric is a well-known measure, i.e., the accuracy (being all selected datasets characterized by a categorical target variable). Formally,

$$Accuracy = \frac{c}{t} \tag{1}$$

where t is the number of new instances to be predicted and c is the number of the correctly predicted instances.

To perform a complete experimental session, MIDA has been applied different times, each time by considering a different supervised method as estimator and a different value for the number of estimators selected in the set $\{10, 20, 40, 70, 100\}$. Moreover, all supervised methods made available by MIDA are used as classifiers to predict on the test dataset. Table III shows the experimental configuration for all the supervised methods.

B. Results

Table IV show the results of the proposed imputation data mechanism by considering all datasets, the different supervised learning methods and the different values of the number of estimators. The reported values are the average of the accuracy values obtained by all the classifiers in predicting the test dataset. For sake of readability, in Table IV, the configuration of MIDA with the best average accuracy value

[3] https://archive.ics.uci.edu/ml/index.php

```
Let $\mathbf{Q}$ be the $N \times K$ original matrix with $0 < k_0 < K$ completely observed variables and $0 < n_0 < N$ complete records.
Let $\mathbf{R}$ to be the indicator matrix with $r_{ij}$-th entry 1 if $q_{ij}$ is missing and zero otherwise.
Let the row vector $\mathbf{r}' = \mathbf{1}'\mathbf{R} = [r_1, \dots, r_j, \dots, r_K]'$, where $r_j$ is equal to the number of missing values of variable in the $j$-th column, for any $j = 1, \dots, K$, with $\mathbf{1}$ a column vector of ones.
Let $\tilde{\mathbf{r}}' = [r_{(1)}, \dots, r_{(j)}, \dots, r_{(K)}]'$ be the order statistic such that $0 = r_{(1)} = \dots < r_{(k_0)} \leq r_{(k_0+1)} \leq \dots \leq r_{(K)}$ defines the lexicographic ordering of variables.
  • initialize:
    Perform columns permutation of the raw data matrix according to the lexicographic ordering of variables to provide the column-wise matrix $\mathbf{Z} = [\mathbf{X}|\mathbf{Y}]$, where the $N \times k$ matrix $\mathbf{X}$ is the complete part and the $N \times (K-k)$ matrix $\mathbf{Y}$ includes missing data in the last $(N-n)$ rows, using at the first iteration $k = k_0$ and $n = n_0$.
For $l = k, \dots, K-1$:
  • Define the $n$-dimensional training sample considering as target variable the $(l+1)$-th column of $\mathbf{Z}$ and as predictors the current $l$ columns of $\mathbf{X}$;
  • Run v-fold Boosting iterations to impute the $r_{(l+1)}$ missing data in variable $x_{l+1}$ considering a suitable supervised weak learner for either categorical or numerical variable.
  • Update the matrix $\mathbf{X}$ with the $r_{(l+1)}$ imputed values, adding up a further column and $n = n + r_{(l+1)}$ rows.
Output: all missing data are imputed.
```

TABLE I
PSEUDO-CODE OF GENERALIZED BINPI ALGORITHM

TABLE II
DATASETS INFORMATION.

Dataset name	#instances	#attributes	#classes	% missing values
iris	150	4	3	32.67%
pima	768	8	2	50.65%
wine	178	13	3	70.22%
satimage	6435	36	7	87.80%
magic	1902	10	2	58.20%

TABLE III
EXPERIMENTAL CONFIGURATION OF HYPER-PARAMETERS

Method	Parameters
K-nearest neighbour (KNN)	k-value=5
Support Vector Machine (SVM)	C-value=1.0, Tolerance=0.001, maximum_number_of_iterations=200, kernel=Gaussian, gamma=1/number_of_features
Multi-layer perceptron (MLP)	number_of_hidden_layers=1, number_of_neurons=100, Tolerance=0.0001, maximum_number_of_iterations=200, learning rate=0.001
Decision Tree (DT)	criterion=gini/mse, maximum_depth=200

TABLE IV
PERFORMANCE IN TERMS OF THE AVERAGE ACCURACY (IN PERCENTAGE) ON DIFFERENT CLASSIFIERS OF THE DIFFERENT CONFIGURATIONS OF THE PROPOSED IMPUTATION DATA MECHANISM.

Dataset iris					
#estimators	KNN	**SVM**	MLP	DT	LDA
10	96	96	95.2	96	96.8
20	96.8	96.8	96	96	97.6
40	96	97.6	97.6	96	97.6
70	96	**98.4**	96.8	96	97.6
100	96	97.6	96	96	96.8
Dataset pima					
#estimators	KNN	SVM	MLP	**DT**	LDA
10	73.41	73.40	74.04	**75.75**	72.13
20	74.89	74.04	72.13	74.47	71.70
40	72.98	74.47	72.77	74.89	74.89
70	72.77	75.32	75.11	74.89	72.98
100	73.40	73.62	71.49	74.25	73.40
Dataset wine					
#estimators	**KNN**	SVM	MLP	DT	LDA
10	86.15	86.15	80	81.54	86.15
20	86.15	87.69	81.54	81.54	75.39
40	83.08	87.69	81.54	81.54	86.15
70	**89.23**	87.69	83.08	81.54	76.92
100	84.62	86.15	81.54	81.54	86.15
Dataset satimage					
#estimators	KNN	**SVM**	MLP	DT	LDA
10	85.79	87.11	83.65	86.8	87.01
20	85.68	86.29	86.09	86.40	85.48
40	87.21	86.50	86.19	86.80	86.40
70	86.7	85.18	86.80	86.50	86.50
100	86.19	**87.31**	87.01	87.00	82.44
Dataset magic					
#estimators	KNN	SVM	MLP	**DT**	LDA
10	65.43	64.42	66.23	65.43	64.02
20	64.02	64.52	63.42	**68.24**	64.72
40	64.32	65.12	63.72	67.24	64.42
70	65.23	66.33	63.02	67.44	63.82
100	65.03	65.93	65.12	67.24	65.22

is written in bold. In order to perform a comparison study, in Table V, this best value among the different configurations of MIDA is compared with the accuracy values computed by the considered conventional approaches for each dataset. Also for the conventional approaches, the reported values are the average accuracy on the results of all considered classifiers. By analysing the results of the Table V, MIDA imputation mechanism outperforms mean/mode approach and the deleting approach for 4 out of 5 datasets.

V. CONCLUSIONS

The paper presents a web-based tool, named MIDA, for imputing datasets with missing values. The algorithm for missing data imputation, named G-BINPI, is based on two main concepts: an incremental approach and the boosting algorithm. G-BINPI is a generalized version of BINPI algorithm since it is possible to use any weak learner. In particular, MIDA

TABLE V
COMPARISON STUDY

Dataset name	Our proposal	Mean/mode	Deleting
iris	**98.4**	95.2	96.8
pima	**75.75**	72.98	71.70
wine	**89.23**	**89.23**	78.46
satimage	**87.31**	85.99	84.16
magic	68.24	65.03	**69.85**

makes available five weak learners: Support Vector Machine, K-nearest neighbour, Multi-layer perceptron, Decision Tree and a linear approach. Its web nature makes MIDA a tool usable for every researchers even if they are not equipped with programming skills or other computer capabilities. As shown in the experiments, MIDA produces good imputed data as well as to be user-friendly.

ACKNOWLEDGMENT

Authors acknowledge financial support from the European Union's Horizon 2020 research and innovation programme under Grant Agreement No. 689669 (MAGIC). This work reflects the authors' view only; the funding agencies are not responsible for any use that may be made of the information it contains.

REFERENCES

[1] G. Molenberghs, G. Fitzmaurice, M. G. Kenward, A. Tsiatis, and G. Verbeke, *Handbook of missing data methodology*. Chapman and Hall/CRC, 2014.

[2] A. D'Ambrosio, M. Aria, and R. Siciliano, "Accurate tree-based missing data imputation and data fusion within the statistical learning paradigm," *Journal of classification*, vol. 29, no. 2, pp. 227–258, 2012.

[3] V. N. Vapnik, "An overview of statistical learning theory," *IEEE transactions on neural networks*, vol. 10, no. 5, pp. 988–999, 1999.

[4] V. Vapnik, *The nature of statistical learning theory*. Springer science & business media, 2013.

[5] R. J. Little and D. B. Rubin, *Statistical analysis with missing data*. John Wiley & Sons, 2019, vol. 793.

[6] G. Acampora and A. Vitiello, "TSSweb: a web tool for training set selection," in *2020 IEEE International Conference on Fuzzy Systems, FUZZ-IEEE*. IEEE, 2020, pp. 1–6.

[7] I. Triguero, S. González, J. M. Moyano, S. García López, J. Alcalá Fernández, J. Luengo Martín, A. Fernández Hilario, J. Díaz, L. Sánchez, F. Herrera Triguero *et al.*, "Keel 3.0: an open source software for multi-stage analysis in data mining," 2017.

[8] J. Alcalá-Fdez, L. Sanchez, S. Garcia, M. J. del Jesus, S. Ventura, J. M. Garrell, J. Otero, C. Romero, J. Bacardit, V. M. Rivas *et al.*, "Keel: a software tool to assess evolutionary algorithms for data mining problems," *Soft Computing*, vol. 13, no. 3, pp. 307–318, 2009.

[9] J. Alcalá-Fdez, A. Fernández, J. Luengo, J. Derrac, S. García, L. Sánchez, and F. Herrera, "Keel data-mining software tool: data set repository, integration of algorithms and experimental analysis framework." *Journal of Multiple-Valued Logic & Soft Computing*, vol. 17, 2011.

[10] D. Bertsimas, C. Pawlowski, and Y. D. Zhuo, "From predictive methods to missing data imputation: an optimization approach," *The Journal of Machine Learning Research*, vol. 18, no. 1, pp. 7133–7171, 2017.

[11] C. Conversano and R. Siciliano, "Incremental tree-based missing data imputation with lexicographic ordering," *Journal of classification*, vol. 26, no. 3, pp. 361–379, 2009.

[12] R. Siciliano and C. Conversano, "Tree-based classifiers for conditional missing data incremental imputation," in *Dataclean 2002*, vol. 1. University of Jyväskylä, 2002, pp. 41–45.

[13] A. D'Ambrosio, M. Aria, and R. Siciliano, "Robust tree-based incremental imputation method for data fusion," in *International Symposium on Intelligent Data Analysis*. Springer, 2007, pp. 174–183.

[14] F. Mola and R. Siciliano, "A fast splitting procedure for classification trees," *Statistics and Computing*, vol. 7, no. 3, pp. 209–216, 1997.

[15] J. Luengo, S. García, and F. Herrera, "A study on the use of imputation methods for experimentation with radial basis function network classifiers handling missing attribute values: The good synergy between rbfns and eventcovering method," *Neural Networks*, vol. 23, no. 3, pp. 406–418, 2010.