Multiple Imputation Using the Average Code from Autoencoders

Edwar Macias*, Javier Serrano, Jose Lopez Vicario and Antoni Morell

Wireless Information Networking (WIN) Group - Universitat Autònoma de Barcelona (UAB), Bellaterra 08193, Spain

ARTICLE INFO

Keywords:
average code
deep learning
autoencoder
multiple imputation

Abstract

Background: Missing information is a constant issue in the clinical setting. The presence of missing values (MV) is triggered by the wrong acquisition of data or sudden events in the patient's health condition. Imputation arises to replace the non-existent information with the twofold purpose of benefiting from existing information and reducing bias in clinical settings. Mechanism based on deep learning and multiple imputation (MI) are leading alternatives to impute MVs because of their capacity to extract complex relationships and the consideration of uncertainty that MI adds.

Objective: This study aims to improve the reconstruction of missing information through a novel imputation alternative that integrates a MI paradigm into deep learning models.

Methods: The proposed method integrates the MI paradigm into the latent representations of an autoencoder, the so-called codes. The average code is then computed, boosting a better latent representation of data. Finally, the average code is decoded to reconstruct MVs.

Results: The proposed method is tested in 6 datasets with different mechanisms of appearance of MVs. It is compared with solutions based on autoencoders and generative adversarial networks. For the random appearance of MVs, the proposed method outperforms 95% of the scenarios with a reconstruction gain that ranges 4-27%. For the other MVs mechanisms, the proposed method improves the reconstruction in at least 69% of the experiments.

Conclusion: The findings of the proposed approach showed that the reconstructive capacity of the average code outperforms in most of the scenarios its competitors. The integration of the MI paradigm into latent representations of data and the computation of average codes allow a more robust representation of the data and enable the enhancement of current state-of-the-art methods for high MVs rates.

1. Introduction

Missing information is inherent to data used in pattern recognition, data mining, and machine learning applications. The presence of missing values (MV) affects the quality of such applications and may add biases to experiments [1]. In the clinical domain, this is a recurrent problem. Electronic health records such as laboratory tests, clinical observations and bedside monitoring usually present MVs in their registers. The integration of such data, the omission of information by patients, data acquisition equipment errors and measurements with different sampling periods are the most representative sources of MVs in the medical domain [2, 3].

The alternatives to handle this issue include removing records with MVs and applying methods to estimate them. The first option is known as complete case analysis and is commonly used in clinical studies [2]. This approach has the drawback of considerably reducing the amount of data, adding bias to the experiments since it analyses only the complete samples in the dataset [4]. In contrast, imputation attempts to replace missing information with the twofold purpose of extracting knowledge from incomplete samples and reducing bias in clinical studies [4].

Imputation methods replace the MVs considering either one value or multiple estimates for an MV. Relying on imputing by one value underestimates the variance and does not consider data uncertainty [5, 4]. Multiple Imputation (MI) [5] was designed to address this concern by considering several estimates for a single MV. However, the challenge in MI lies in the choice of the estimative model [6]. Statistical models for MI are based on estimates that consider only

*Corresponding author

edwar.macias@uab.cat (E. Macias); javier.serrano@uab.cat (J. Serrano); jose.vicario@uab.cat (J.L. Vicario); antoni.morell@uab.cat (A. Morell)

ORCID(s): 0000-0001-5568-9237 (E. Macias); 0000-0003-1235-2145 (J. Serrano); 0000-0002-3574-4697 (J.L. Vicario); 0000-0003-2249-8594 (A. Morell)

linear relationships in the data. In several scenarios, they cannot handle large datasets and are limited when there are different types of data and MVs patterns. In contrast, such limitations may be exploited by methods based on deep learning (DL).

DL techniques have shown an exceptional ability to exploit complex relationships in large datasets [7]. Such relationships are extracted in latent representations in the hidden layers of artificial neural networks (ANN). Promising methods in imputation in the literature are based on generative adversarial networks (GAN) [8, 9] and autoencoders (AE) [10, 11, 6]. However, since GANs are based on two competing neural networks, they are difficult to tune and often present convergence problems [12]. In contrast, AEs learn in an unsupervised way, functions that encode the input into a latent representation of data and then use a decoder function that reconstructs such representation to match the input. With this mechanism, it is possible to extract the most relevant information from the data, even from those samples that present MVs. To include such samples, it is necessary to carry out an initial imputation. Most authors use a constant value to perform this initial imputation without considering that the learning models can memorize the data with which they were fitted. Thus, choosing adequate values to perform this initial imputation is a challenge to minimize the bias they add.

On the other hand, the application of AEs, jointly with the MI approach, has shown promising imputation [6]. In the method proposed in [6], several copies of the data are estimated by AEs. The MI mechanism is applied once the reconstructions are combined. Thus, motivated by the ability to extract knowledge that AEs have and the inclusion of uncertainty that MI provides, in this manuscript it is presented a novel alternative to impute MVs based on the adoption of the MI paradigm into latent representations of data through AEs. This integration allows the combination of several latent representations of the data into the so-called average code (AVG code). This combination generates a more robust representation of data. Once the AVG code is decoded and missing information is better imputed, the complete information may be used to support learning tasks in specific domains.

The rest of this manuscript is organized as follows. Section 2 presents the background on imputation and the learning mechanism of AEs. Details of the proposed method are presented in Section 2.5. Section 3.2 shows the empirical evaluation of the proposed method in real-life datasets, followed by the analysis of the results in Section 4. The conclusion of this manuscript and the insights are presented in Section 5.

The main contributions of this work are as follows:

- Provide a novel alternative to compute a more robust representation of latent spaces through the computation of the average latent representation.
- Integrate MI paradigm into latent spaces of deep neural networks.
- Improve the imputation of missing values through the computation of the average code.

2. Methods

In this section, all the necessary components to carry out the proposed method are presented. Initially, the MVs mechanisms are described. Then, the imputation problem is formally introduced, followed by the description of MI and the learning models on which the proposed method is based. Finally, the proposed method is presented.

2.1. Missing data mechanisms

Missing information can be characterized by the relationship that exists between MVs and data attributes. MVs are classified based on three mechanisms [13]: missing completely at random (MCAR), missing at random (MAR) and missing not at random (MNAR). MCAR refers to the fact that the appearance of the MVs does not depend on the data itself; missingness appears as a random process. Forgetting to include information on a patient attribute is considered an MCAR since it does not rely on the patient or the missing attribute. MAR refers to the fact that the appearance of MVs depends partially on the observed data but not on the information that is missing. The abrupt stop in the measurement of temperature in a patient whose health condition worsens in an intensive care unit (ICU) is an example of this type of MVs. In this case, the appearance MVs in temperature depends directly on a variable that indicates the state of the patient in the ICU. Finally, MVs are classified as MNAR when they are directly related to their values. For instance, a depression survey is more likely not to be answered by patients with depression.

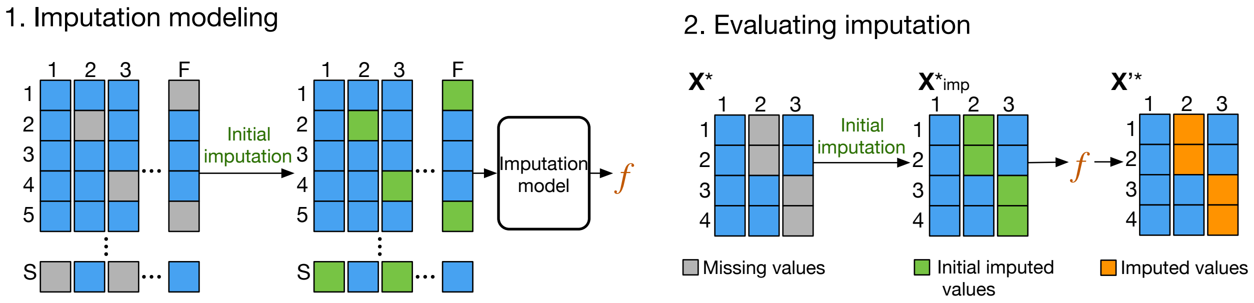

Figure 1: Imputation scheme. The initial imputation is necessary to fit the model that reconstructs missing information.

2.2. Imputation problem

Let $\mathbf{X} \in \mathbb{R}^{s,f}$ be a dataset, with $s = 1, \dots, S$ samples and $f = 1, \dots, F$ features. The elements of $\mathbf{X}$ are denoted by $x_{s,f}$. Each sample is denoted by $\mathbf{x}_s = [x_{s,1}, x_{s,2}, \dots, x_{s,f}, \dots, x_{s,F}]$. Then, a MV indicator $\mathbf{m}_s = [m_{s,1}, m_{s,2}, \dots, m_{s,f}, \dots, m_{s,F}]$ is associated to the samples $\mathbf{x}_s$ and tracks the registers that are missing. A matrix $\mathbf{M}$ can be constructed and all the MVs in a dataset can be tracked. Its content can be define as follows,

$$\mathbf{M} = \begin{cases} m_{s,f} = 1, \text{if } x_{s,f} \text{ is missing} \\ m_{s,f} = 0, \text{if } x_{s,f} \text{ is observed.} \end{cases} \tag{1}$$

Thus, $\mathbf{X}$ can be divided into two components, observed and missing data, $\mathbf{X}_{obs}$ and $\mathbf{X}_{miss}$, respectively. $\mathbf{X}_{obs}$ contains samples without MVs in their features, while those samples with MVs are stored in $\mathbf{X}_{miss}$. Imputation aims to find a function, $f(\cdot)$, that best estimates MVs in $\mathbf{X}_{miss}$, which in turn minimizes bias added by the inclusion of information that did not exist before. This function can be generated based only on the $\mathbf{X}_{obs}$ or by including also $\mathbf{X}_{miss}$ in its estimation. The first case considers the distribution of the observed data, and the MVs of $\mathbf{X}_{miss}$ are replaced by values that best fit such distributions. When $\mathbf{X}_{miss}$ is included for the computation of $f(\cdot)$, it is necessary to perform an initial imputation. This imputation replaces MVs and works as seed values that change iteratively in the training process for the imputation model.

Including samples with MVs adds robustness to the models in the sense that it is possible to extract knowledge that is directly linked to the appearance and generation of MVs in the dataset. Fig. 1 illustrates the imputation problem in two stages. The first one is the generation of the function $f(\cdot)$. An initial imputation is performed to compute the function that best fit the data. In the second stage, $f(\cdot)$ is applied in new data $\mathbf{X}^*$ and imputes their MVs. In summary, a robust imputation should include information from the samples with MVs and an adequate imputation model that minimizes bias in the experiments.

2.3. Multiple imputation

MI creates several versions of a data set that are used to improve a single imputation. Each version may contain different estimates for the MVs. This imputation paradigm addresses the problem of uncertainty that exists when imputing an MV with a single estimation. The mechanisms to find one or more imputation functions follow the same imputation process mentioned in the imputation problem. In this case, N versions of the data are generated, and the initial imputation replaces MVs with slightly different values. Then, N imputation models generate estimates that are finally grouped and estimate the MVs of the dataset.

2.4. Autoencoders

An AE is a type of ANN whose purpose is to replicate the input $\mathbf{x}_s$ to the output $\mathbf{x'}_s$, with the minimum possible error. This mechanism forces an AE to learn the data representation in a latent space. Structurally, an AE has three main components, the encoder, the code and the decoder. Fig. 2 illustrates its components. The encoder corresponds to the first layers in the ANN. Typically, the input is compressed in a latent space of a smaller dimension than the input data. The activations at the last layer in the encoder, are interpreted as a code that represents the latent space. The decoder is the portion of the network in charge of reconstructing the information of the code, and this information is compared with the original input using a cost function. Both the learning mechanism and the AE structure allow encapsulating the abstract relationships that the data may have in their codes.

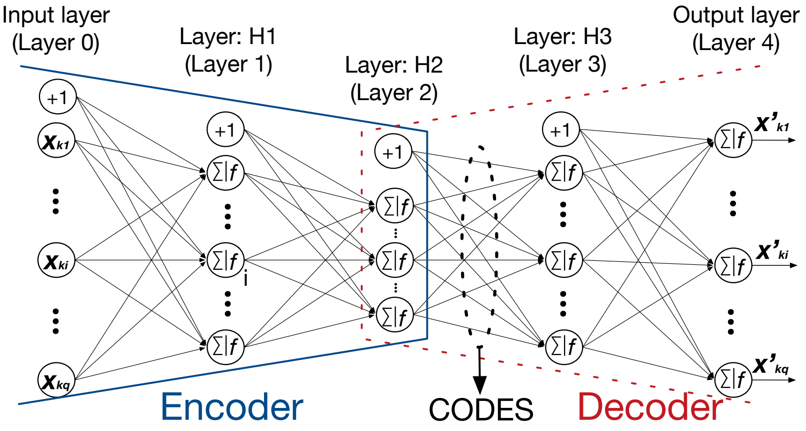

Figure 2: Autoencoder with three hidden layers.

The training of an AE is carried out through the back-propagation algorithm [14]. It aims to minimize a cost function that measures the error between the input and the output of the network. In this work, the root of the mean squared error (RMSE) as cost function is used,

$$RMSE = \sqrt{\frac{1}{S}\sum_{i=1}^{S}\left\|\mathbf{x}_s^{(i)} - \mathbf{x'}_s^{(i)}\right\|^2}. \tag{2}$$

2.5. Proposed method

Motivated by the ability to represent complex relationships that AEs have and the solution that MI presents to handle the uncertainty problem, we propose to compute the AVG code of an AE as a mechanism for enhancing imputation. In our approach, we reinterpret the solution proposed in the work entitled Multiple Imputation using Denoising Autoencoder (MIDA) [6], by integrating MI in the latent spaces of an AE instead of at the output layer of the AE, as in MIDA. The proposed method consists of two stages, the training of an AE and the imputation mechanism based on the AVG code.

2.5.1. Learning model

For the knowledge extraction process, an AE is trained. The dimension of the latent representations is smaller than the dimension of the input data. This stage differs considerably compared to MIDA. N AEs are trained in MIDA, and in our approach, just one is used to perform the learning task. To carry out the initial imputation, MIDA uses the average value to impute MVs. Imputing with constant values may generate conflicts in terms of generalization for the learning model. The model may memorize the values with which the MVs were initially imputed. To solve this issue, we carry out the initial imputation based on random values that follow the distribution of the attributes of the observed values in the training data. Additionally, as a regularization mechanism, the imputing values are changed in every epoch of the training process. For categorical variables, the most representative categories are used as imputers and these change in every epoch. An epoch is when the entire training sample is passed forward and backwards through the AE only once. This iterative variation is carried out with the twofold purpose of preventing the models from memorizing the imputing data and guaranteeing a more robust representation of the data in the codes. Finally, the encoder and decoder are extracted from the trained AE. Fig. 3 illustrates the initial stage of the proposed method.

2.5.2. Imputation

At this stage, the MI approach is integrated into the latent space of the trained AE. The proposed method differs from the usual MI because the combination of information is not performed at the end of the estimation of the MVs,

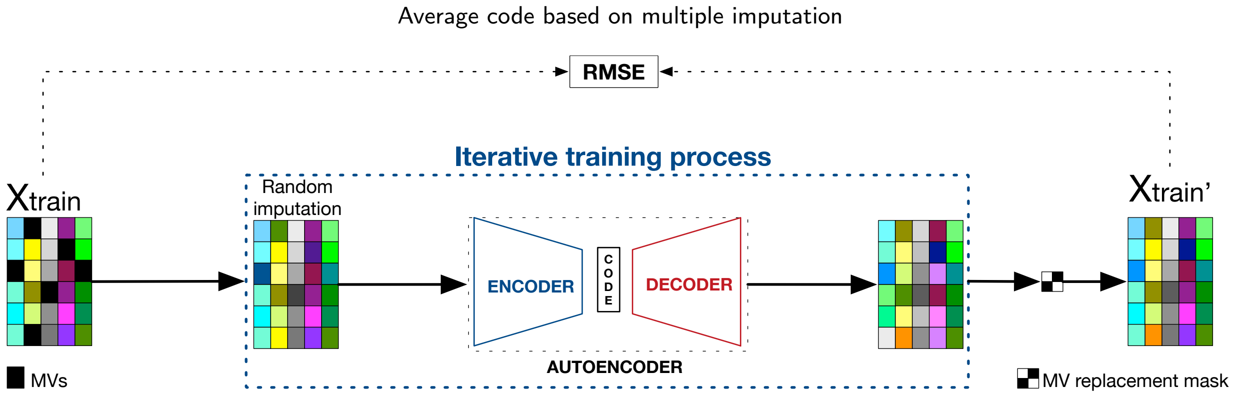

Figure 3: Learning process of the proposed method. Missing values (MV) replacement mask contains the location of MVs in $\mathbf{X}_{train}$ and it is used to preserve the observed values of $\mathbf{X}_{train}$ and replace the MVs with values estimated by the AE.

but in the latent space of the AE, as illustrated in Fig. 4. To perform the imputation on new data, $\mathbf{X}_{test}$, it is necessary to generate different copies of randomly imputed data. This imputation follows the same mechanism as the initial imputation described for the previous learning process. Then, the encoder function of the trained AE generates the codes. These codes are combined into an AVG code as follows,

$$\bar{c} = \frac{1}{N}\sum_{i=1}^{N} c_i. \tag{3}$$

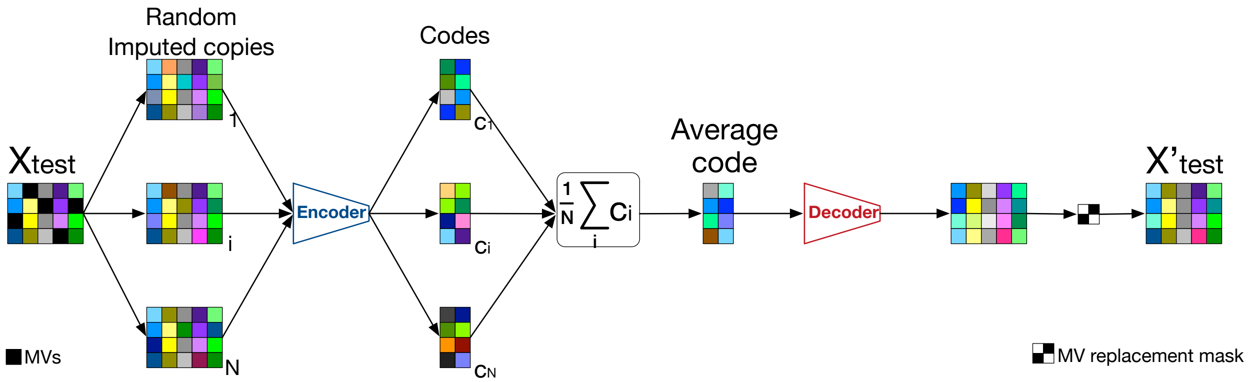

Figure 4: Second stage of the proposed method. Encoder and decoder functions are extracted from a trained autoencoder. $\mathbf{X}_{test}$ are different data from training data and $\mathbf{X'}_{test}$ is the reconstruction considering a missing values (MV) replacement mask that contains the location of MVs in $\mathbf{X}_{test}$ and it is used to preserve the observed values of $\mathbf{X}_{test}$ and replace the MVs with values estimated by the decoder function.

The decoding function is applied to the AVG code. The final reconstruction, $\mathbf{X}'_{test}$, is the mixture of the observed data and the data that are tracked by the MV indicator.

3. Results

3.1. Datasets

Six datasets have been used to test the imputation capacity of the AVG code. Diabetes, breast cancer and liver datasets have been extracted from the UCI repository [15]. Spam and letter datasets have been included as they are widely used as benchmark datasets to evaluate imputation methods. Finally, in the datasets, we have included data extracted from the MIMIC-III database [16]. From this massive database, those patients with acute kidney injury (AKI) were filtered based on the kidney disease improving global outcomes (KDIGO) clinical practice guideline [17].

All mentioned datasets have a mixture of categorical and continuous attributes. Table. 1 shows the dimension and amount of attributes of the datasets.

Table 1
Dataset used to compare the performance of the imputation mechanisms.

Dataset	Observations	Attributes
Diabetes	442	10
Breast cancer	569	30
Liver	579	9
Spam	4601	57
Letter	20000	16
AKI	56274	8

3.2. Experiments

The performance of the AVG code was compared with two imputation methods from the state-of-the-art, imputing MVs with MIDA and using an imputation alternative based on GANs [9]. The imputation capacity of each model is measured by computing the RMSE in the imputation stage for each model. The comparison of the three methods is carried out using the same data with 5-fold cross-validation. Having the same data for training and testing for the three methods ensures that the results are fair and generalizable to different data subsets.

The architecture and the hyperparameters used to train the state-of-the-art imputation models are the ones proposed in such works. For the AVG code, AEs have been trained with two hidden layers of $F/2$ and $F/4$ units, where F is the number of attributes of a specific data set. The inputs are standardized between 0 and 1 to facilitate the convergence of the models. To speed up the training, ADAM [18] optimizer was used with an $LR = 0.001$. A dropout [19] of 0.1 was applied as a regularizer to the hidden layers in the AEs. In addition to using conventional regularizers, early stopping [20] was used to prevent the learning model from being overtrained and stop the training process when a model stopped learning.

The imputation capacity of the AVG code is evaluated in scenarios where the missingness mechanisms and the percentage of MVs vary. MVs are synthetically generated since the datasets do not contain MVs. To provide a wide range of comparisons, in the experiments part of the information is eliminated to generate MVs with ratios ranging from 10-60%. Ten copies of the data have been used and imputed with random values to generate the AVG codes of the experiments. Next, the mechanisms used to generate the synthetic MVs are covered.

3.2.1. *MCAR*

In the first experiment, it is evaluated how the random appearance of MVs affects imputation. For this case, MVs are synthetically generated varying the percentage of MVs from 10-60%. This is the most typical scenario for real-life datasets. In Fig. 5 it can be appreciated that the AVG code has a lower reconstruction error than state-of-the-art solutions. Additionally, it can be seen that the models based on GANs are very volatile to the change in the MV rate with MCAR.

3.2.2. *MAR*

For the appearance of MVs following the MAR mechanism, the proposed method is evaluated in a scenario more closely to a clinical environment. The appearance of MVs is not a random process. There is a dependency between attributes and MVs. According to the value of one or several attributes, MVs appear in other ones. The recommendations in [21] have been followed to emulate the generation of this type of MVs. In this case, the registers in the selected features, where MVs are synthetically generated, are deleted based on lower values of an observed feature. For this experiment, rates of MVs vary from 10-60%, corresponding to a number of variables from 10-60% of the total of attributes for each dataset. To illustrate the experiment, let's consider the Spam dataset. The experiment starts with 10% of MVs in 6 attributes, then 20% of MVs in 12 attributes and so on.

In Fig. 6 it can be appreciated that the solution based on GANs has a volatile behaviour in most of the datasets. However, it showed a better performance dealing with attributes with 10% of MVs. The proposed approach for 10% of MVs shows a weak performance for the datasets with few samples. For the rest of them, the AVG code solution has an overall better performance than MIDA and GANs. For Cancer dataset, the solution based on MIDA has a

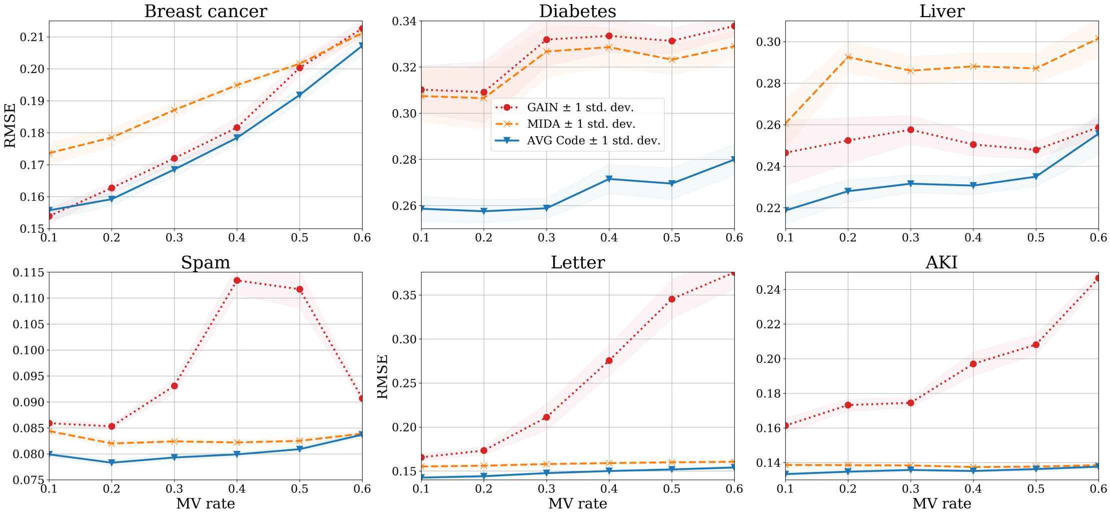

Figure 5: Comparison of imputation mechanisms with missing complete at random missing values, varying the missing value rate for the available datasets.

better performance than the proposed approach. For the rest of the evaluations, the proposed approach showed better performance than GAN and compared with MIDA, for a range of MVs from 20-50% the proposed approach presented a better performance. Just for the datasets with more samples and 60% of MVs MIDA is competitive with the AVG code.

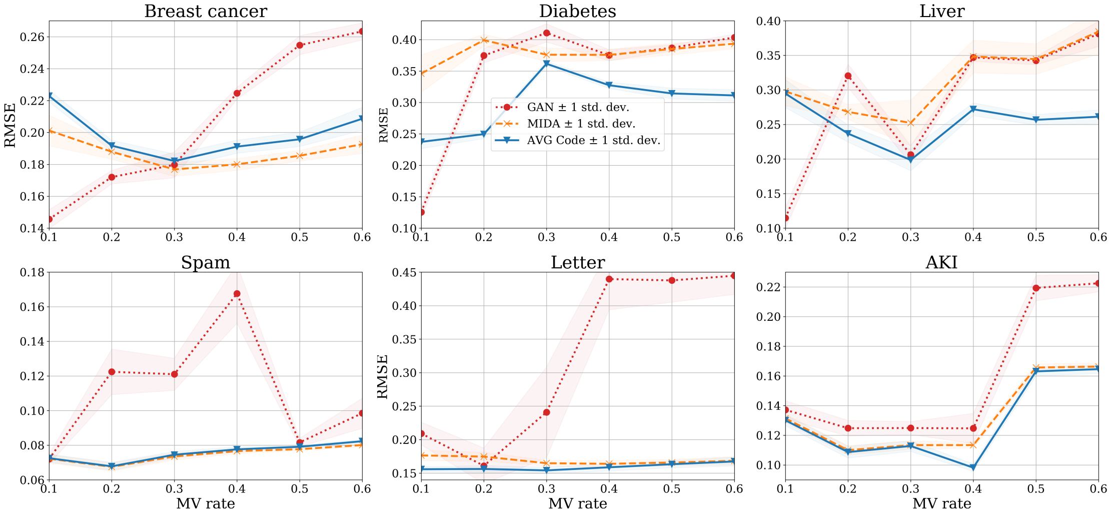

Figure 6: Comparison of the imputation methods in missing at random, varying the missing value rate.

3.3. MNAR

To generate MVs following an MNAR mechanism, it was followed the recommendations of [21]. In this case, the generation of the synthetic MVs is based on the variable itself. The recommendations suggest that the lower values of the variables are deleted. The rate of MVs vary from 10-60% and these MVs appears in 10-60% of the chosen attributes, as in MAR.

In Fig. 7 it can be appreciated that the AVG code shows a competitive performance with GANs for datasets with few samples and MV rates between 20-40%. For the rest of the datasets, the AVG code presents an overall better performance, standing out among those datasets with more samples.

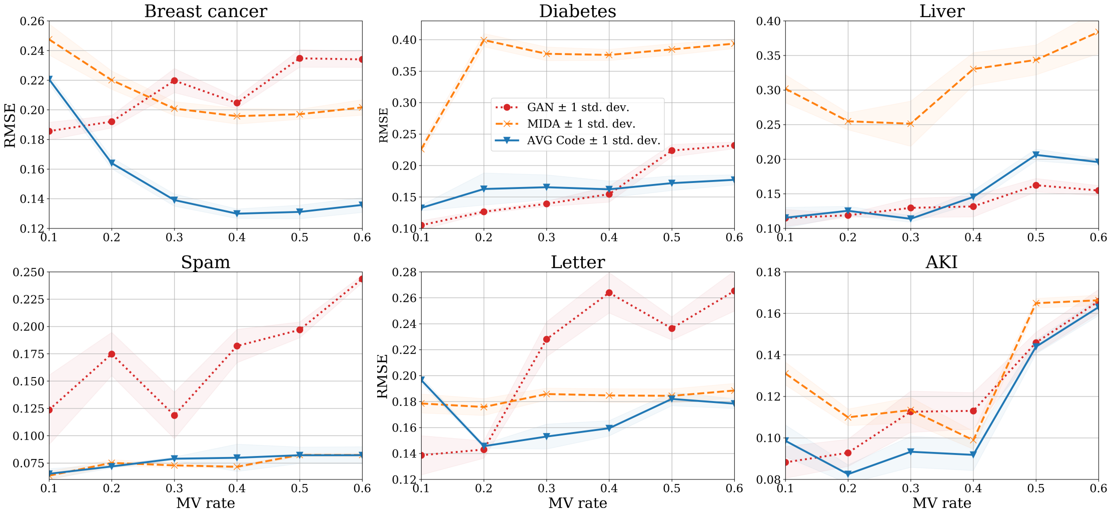

Figure 7: Comparison of the imputation methods in missing not at random, varying the missing value rate.

4. Discussion

In this work, a new alternative to impute MVs based on the integration of a MI in the latent spaces of an AE and the computation of the AVG code was proposed. The AVG code combined the information from complex representations of data in latent spaces. The comparative results can be separated into two groups based on the dimension of the datasets. The first collection groups the breast cancer, diabetes and liver datasets, and the other three datasets are grouped because they contain much more samples than the first group. Table. 2 shows the reconstructive percentage of gain that the AVG code has over its competitors in all the performed experiments. A negative gain means that the competitor outperforms the proposed method.

For the experiment with MCAR values, the AVG code has an outstanding performance compared to its competitors. For the first collection of datasets, the AVG code offered a better reconstructive capacity in 95% of the evaluated cases. Compared with MIDA, the reconstructive error of the proposed method exceeds the performance in 14±6%. Compared to GAN, the AVG code had an improvement of 9 ± 7%. For the second group of datasets, the AVG code outperformed in all the evaluated cases compared with its competitors. With MIDA, the improvement in RMSE was 4 ± 2%. For GAN, the proposed method outperforms in 27 ± 16% in RMSE.

For the experiments with MAR values, for both groups of datasets, the proposed method outperformed in 69% and 86% of the experiments, respectively. The gain in reconstruction concerning GAN was $-3 \pm 49\%$, while for MIDA, it was 11 ± 16%. The proposed method is sensitive to low percentages of MVs ratios with datasets with few samples. This concern may be due to an under-fitting issue in the training of the AEs in both MIDA and AVG code. In the second group of datasets, the AVG code outperformed GAN 28 ± 22% and 2 ± 5% compared to MIDA. Although GAN presents better reconstruction when there are few MVs in 10% of the variables for datasets with few samples, the AVG code has a better reconstruction error when increasing the MV rate. For the second group of datasets at most MV rates, the AVG code is robust enough to improve the performance of MIDA and, in a few cases, have similar performance.

In the last experiment, it was appreciated that the AVG code had similar behaviour to MAR. In this case, the proposed method outperform 75% and 78% of the experiments for the first and second groups of datasets, respectively. With this type of MVs, it was possible to improve the reconstructive capacity in 4 ± 26% and 45 ± 14% for GAN and MIDA, respectively. In the second group of datasets, it shown a improvement of 25 ± 28% and 6 ± 13% for GAN and MIDA, respectively. This MV mechanism showed to be more suitable for GANs than MIDA and competitive with the proposed method for the first group of datasets. For the second one, the AVG code outperformed for most of the cases, being more representative for Spam and AKI.

Table 2
Reconstructive gain using AVG codes compared to MIDA and GANs.

		Cancer		Diabetes		Liver		Spam		Letter		AKI	
	MV rate	GAN	MIDA	GAN	MIDA	GAN	MIDA	GAN	MIDA	GAN	MIDA	GAN	MIDA
MCAR	0,1	-1	10	17	16	11	16	7	5	14	8	17	4
	0,2	2	11	17	16	10	22	8	5	17	8	22	3
	0,3	2	10	22	21	10	19	15	4	30	7	22	2
	0,4	2	8	19	17	8	20	30	3	46	6	31	2
	0,5	4	5	19	17	5	18	28	2	56	5	35	1
	0,6	3	2	17	15	1	15	8	0	59	4	44	1
MAR	0,1	-53	-11	-89	31	-157	1	-1	0	26	12	5	1
	0,2	-11	-2	33	37	26	12	45	0	3	11	13	1
	0,3	-1	-3	12	4	4	21	38	-1	36	6	10	1
	0,4	15	-6	13	13	22	22	54	-1	64	3	21	13
	0,5	23	-6	19	18	25	25	3	-2	63	2	26	2
	0,6	21	-8	23	21	31	32	16	-3	62	1	26	1
MNAR	0,1	-19		-26	41	0	62	47	-3	-42	-10	-12	25
	0,2	15	25	-29	59	-5	51	59	4	-2	17	11	25
	0,3	37	31	-19	56	12	55	33	-8	33	18	17	18
	0,4	37	34	-5	57	-10	56	56	-12	40	14	19	7
	0,5	44	33	23	55	-27	40	58	0	23	1	1	13
	0,6	42	33	24	55	-27	49	66	0	33	5	2	2

5. Conclusion

In this work, the capacity to reconstruct missing information based on the computation of the AVG code from an AE was presented. It was evaluated the imputation capacity of a novel mechanism that integrated a MI paradigm into the latent representation of information extracted by deep ANNs. The AVG code has a sufficiently robust imputation capacity to replace MVs to different MV rates and under various MV appearance mechanisms, such as MCAR, MAR and MNAR. The AVG code demonstrated to be robust enough to maintain a low reconstruction error with different percentages of MVs. The variation used in the proposed approach, based on the training of an AE and the integration of MI in the latent space, revealed that the AVG code contains a robust representation of the information reflected in a considerable improvement in the reconstruction error. In conclusion, the work presented in this manuscript shows that the integration of classical mechanisms such as MI to the latent spaces of a DL-based solution adds performance and robustness benefits in comparison to other methods in the literature based on deep learning.

Acknowledgements

This work is supported by the Spanish Government under Project TEC2017-84321-C4-4-R co-funded with European Union ERDF funds and also by the Catalan Government under Project 2017 SGR 1670.

References

[1] Milena A. Gianfrancesco, Suzanne Tamang, Jinoos Yazdany, and Gabriela Schmajuk. Potential Biases in Machine Learning Algorithms Using Electronic Health Record Data, 2018.

[2] Federico Cismondi, André S Fialho, Susana M Vieira, Shane R Reti, João M C Sousa, and Stan N Finkelstein. Missing data in medical databases: Impute, delete or classify? Artificial Intelligence in Medicine, 58(1):63–72, 2013.

[3] E. M. Mirkes, T. J. Coats, J. Levesley, and A. N. Gorban. Handling missing data in large healthcare dataset: A case study of unknown trauma outcomes. Computers in Biology and Medicine, 2016.

[4] Jonathan A.C. Sterne, Ian R. White, John B. Carlin, Michael Spratt, Patrick Royston, Michael G. Kenward, Angela M. Wood, and James R. Carpenter. Multiple imputation for missing data in epidemiological and clinical research: Potential and pitfalls, 2009.

[5] William M. Campion and Donald B. Rubin. Multiple Imputation for Nonresponse in Surveys. Journal of Marketing Research, 1989.

[6] Lovedeep Gondara and Ke Wang. MIDA: Multiple imputation using denoising autoencoders. In Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2018.

[7] Travers Ching, Daniel S. Himmelstein, Brett K. Beaulieu-Jones, Alexandr A. Kalinin, Brian T. Do, Gregory P. Way, Enrico Ferrero, Paul Michael Agapow, Michael Zietz, Michael M. Hoffman, Wei Xie, Gail L. Rosen, Benjamin J. Lengerich, Johnny Israeli, Jack Lanchantin, Stephen Woloszynek, Anne E. Carpenter, Avanti Shrikumar, Jinbo Xu, Evan M. Cofer, Christopher A. Lavender, Srinivas C. Turaga, Amr M.

Alexandari, Zhiyong Lu, David J. Harris, Dave Decaprio, Yanjun Qi, Anshul Kundaje, Yifan Peng, Laura K. Wiley, Marwin H.S. Segler, Simina M. Boca, S. Joshua Swamidass, Austin Huang, Anthony Gitter, and Casey S. Greene. Opportunities and obstacles for deep learning in biology and medicine. Journal of the Royal Society Interface, 2018.

[8] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Advances in Neural Information Processing Systems, 2014.

[9] Jinsung Yoon, James Jordon, and Mihaela Van Der Schaar. GAIN: Missing data imputation using generative adversarial nets. In 35th International Conference on Machine Learning, ICML 2018, 2018.

[10] Brett K. Beaulieu-Jones and Jason H. Moore. Missing data imputation in the electronic health record using deeply learned autoencoders. In Pacific Symposium on Biocomputing, 2017.

[11] Edwar Macias Toro, Guillem Boquet, Javier Serrano, Jose Lopez Vicario, Jose Ibeas, and Antoni Morell. Novel Imputing Method and Deep Learning Techniques for Early Prediction of Sepsis in Intensive Care Units. In 2019 Computing in Cardiology Conference (CinC), 2019.

[12] Hamed Alqahtani, Manolya Kavakli-Thorne, and Gulshan Kumar. Applications of generative adversarial networks (gans): An updated review. Archives of Computational Methods in Engineering, pages 1–28, 2019.

[13] Donald B. Rubin. Inference and missing data. Biometrika, 1976.

[14] David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. Learning representations by back-propagating errors. Nature, 1986.

[15] Dheeru Dua and Casey Graff. Uci machine learning repository, 2017.

[16] Alistair E.W. Johnson, Tom J. Pollard, Lu Shen, Li Wei H. Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G. Mark. MIMIC-III, a freely accessible critical care database. Scientific Data, 2016.

[17] Lesley A. Inker, Brad C. Astor, Chester H. Fox, Tamara Isakova, James P. Lash, Carmen A. Peralta, Manjula Kurella Tamura, and Harold I. Feldman. KDOQI US commentary on the 2012 KDIGO clinical practice guideline for the evaluation and management of CKD. American Journal of Kidney Diseases, 2014.

[18] Diederik P. Kingma and Jimmy Lei Ba. Adam: A method for stochastic optimization. In 3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings, 2015.

[19] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. Journal of Machine Learning Research, 2014.

[20] Lutz Prechelt. Early stopping - But when? Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2012.

[21] Miriam Seoane Santos, Ricardo Cardoso Pereira, Adriana Fonseca Costa, Jastin Pompeu Soares, Joao Santos, and Pedro Henriques Abreu. Generating synthetic missing data: A review by missing mechanism. IEEE Access, 2019.