# A flexible ensemble regression model of extreme learning machine for missing value imputation of DNA microarray data

*Xiuwei* Pan[1], *Wenlu* Dong[2], and *Hualong* Yu[2,*]

[1]Beijing Huanjia Communication Technology Co., Ltd, Beijing 100192, China
[2]School of Computer, Jiangsu University of Science and Technology, Zhenjiang 212003, China

**Abstract.** Missing value imputation (MVI) is important for DNA microarray data analysis because microarray data with missing values would significantly degrade the performance of the downstream analysis. Although there have been lots of MVI algorithms for dealing with the missing DNA microarray data, we note that most of them have a lack of robustness owing to only adopting the single model. In this paper, a flexible and robust MVI algorithm named EELMimpute is proposed to address missing DNA microarray data imputation problem. First, the algorithm constructs a relevant feature space for the missing target gene, where the relevant feature space only includes those co-expression genes of the target gene based on calculating their Pearson's correlation coefficients. Then, some fix-sized feature subspaces are randomly extracted from the relevant feature space to construct extreme learning machine (ELM) regression models between the extracted genes and the target gene. Furthermore, selecting those models without missing input gene values to construct the ensemble framework, and then imputing the missing gene by calculating the average output of all models included in the ensemble framework. Experimental results show that the EELMimpute algorithm is able to reduce the estimated errors in comparison with several previous imputation algorithms.

## 1 Introduction

In the past two decades, DNA microarray has been gradually developed to be one of the most important molecular biology techniques in the post-genomic era, and the microarray data has also been widely applied by biologists and medical experts to analyze various behaviors of life [1-2].

However, there may exist some missing gene expression values in microarray data, owing to various reasons, e.g., experimental errors, hybridization failures, the existence of dust or scratch on the chip surface, etc. [3]. The existence of missing values might significantly degrade the performance of some downstream analysis, thus, providing the complete gene expression data is necessary for the further downstream analysis.

* Corresponding author: yuhualong@just.edu.cn

To solve the problem above, a missing value imputation (MVI) strategy is required. Indeed, some complex MVI strategies have been developed for imputing the missing microarray data. The methods include K-nearest neighbors imputation (KNNimpute) [4], singular value decomposition imputation (SVDimpute) [4], Bayesian principal component analysis (BPCAimpute) [5], local least squares imputation (LLSimpute) [6], shrinkage local least squares imputation (SLLSimpute) [7], iterative locally auto-weighted least squares imputation (ILAW-LSimpute) [8], etc. In general, these methods can provide more accurate estimations for the missing values existing in the microarray data. However, all these methods adopt the single model, meanwhile we note the microarray data is generally small-sized, thereby the robustness of the imputing results by these methods can't be guaranteed.

In this paper, we propose a novel missing data imputation algorithm named ensemble regression model of extreme learning machine imputation (EELMimpute) for microarray data. Our EELMimpute involves three key phases. In the first phase, some co-expression genes whose expressions are strongly associated with that of the target gene are extracted from the original data to construct the relevant feature space. In fact, these co-expression genes could reflect the real values of the missing position of the target gene to some extent. Then, in the second phase, some fix-sized feature subspaces are randomly extracted from the relevant feature space to train some extreme learning machine (ELM) regression models. That means a lot of different mappings between a target gene and its several co-expression genes can be created. Here, the reason of selecting ELM as the regression modeling tool is that ELM is not only robust, but also time-saving [9-11]. Finally, the third phase takes charge of integrating all regression models into an ensemble framework, and further providing the estimated missing values by averaging the results from all models containing in the ensemble framework. Actually, this phase takes advantage of ensemble learning to further promote the robustness and accuracy of the imputing results. Experimental results on eight different DNA microarray data sets indicate that the proposed EELMimpute algorithm is able to reduce the estimated errors in comparison with several popular state-of-the-art imputation algorithms.

The contributions of this study can be concluded as follows,

A fast and robust regression model-ELM is adopted to address missing DNA microarray data imputation problem;

A robust ensemble learning framework based on the idea of feature subspace strategy is proposed to improve the imputation quality of missing DNA microarray data.

## 2 Method

### 2.1 Construction of relevant feature space

As we know, there exists a large number of noisy and irrelevant genes when specifying a target gene in DNA microarray data, which means these genes have no or very weak co-expression with the target gene. Obviously, it is unhelpful to use these genes for accurately impute the missing value of target gene. Therefore, a relevant feature space should be firstly extracted from the original gene space for reconstructing the missing target gene values. For this purpose, we adopt the Pearson correlation coefficient as the similarity metric between two genes, because of its accuracy and robustness on measuring the linear relationship between two equilong vectors. Without loss of generality, suppose the target gene is $g_i$, then the Pearson correlation coefficient $r_{ij}$ that indicates the similarity between the target gene and the gene $g_j$ is calculated as follows,

$$r_{ij}=\frac{\sum_{k=1}^{h}(e_{ik}-\overline{g_i})(e_{jk}-\overline{g_j})}{(\sqrt{\sum_{k=1}^{h}(e_{ik}-\overline{g_i})^2})(\sqrt{\sum_{k=1}^{h}(e_{jk}-\overline{g_j})^2})} \tag{1}$$

where $r_{ij}\in[-1, 1]$, and when it approximates 1 or -1 indicating there is a strong positive or negative linear relationship between two measured genes. As for $e_{ik}$ and $e_{jk}$, they respectively denote the expression value of gene $i$ and gene $j$ on the $k$th instance, while $\overline{g_i}$ and $\overline{g_j}$ respectively represents the average expression of gene $i$ and gene $j$ across all $h$ instances which don't include missing values on either of these two genes. Specifically, $h$ is decided by the missing positions existing in both genes. For example, the expression pattern of two genes across 6 instances is given as follows,

$$\begin{bmatrix} 0.8 & ? & 1.0 & 0.7 & ? & 0.9 \\ ? & 1.2 & 0.8 & 0.5 & ? & 0.8 \end{bmatrix} \tag{2}$$

Then $h$ equals 3 as the other three positions (1st, 2nd and 5th) hold missing values either on one gene or on both.

For a given target gene, $m$-1 Pearson correlation coefficients can be calculated. Next, the absolute values of these coefficients are ranked in descending sort to indicate which genes are strongly correlated with the target gene. The reason of adopting absolute values to sort the correlation coefficients lies in that no matter the strong positive correlation ($r\approx1$) or strong negative correlation ($r\approx-1$) indicates there is a strong linear relationship between the target gene and the investigated gene, which is very helpful for estimating the missing values.

Finally, we extract the genes corresponding to the first $K$ top-ranked coefficients to construct the relevant feature space, which will be further used to recovery the missing target gene values. In general, $K$ should be much smaller than $m$.

### 2.2 Construction of relevant feature space

Extreme Learning Machine (ELM) that was proposed by Huang *et al.*, [9-11], is a specific learning algorithm for training single-hidden layer feedforward neural networks (SLFN). The main characteristic of ELM that distinguishes it from those conventional learning algorithms of SLFN is the random generation of hidden nodes. Therefore, ELM does not need to iteratively adjust parameters to make them approach the optimal values, thus it has faster learning speed and better generalization ability. In this work, we consider ELM as a regression model.

Let us consider a regression problem with $N$ training instances to approximating $m$ continuous variables, and then the $i$th training instance can be represented as $(x_i, t_i)$, where $x_i$ is an $n\times1$ input vector, and $t_i$ is the corresponding $m\times1$ continuous output vector. Suppose there are $L$ hidden nodes in ELM, and that all weights and biases on these nodes are generated randomly. Then, for the instance $x_i$, its hidden layer output can be represented as a row vector $\mathrm{h}(x_i)=[\mathrm{h}_1(x_i), \mathrm{h}_2(x_i),..., \mathrm{h}_L(x_i)]$ by mapping with an activation function (the most popular sigmoid function is used throughout this paper). The mathematical model of ELM could be described as:

$$\mathrm{H}\beta=\mathrm{T} \tag{3}$$

where $\mathrm{H}=[\mathrm{h}(x_1), \mathrm{h}(x_2),..., \mathrm{h}(x_N)]^{\mathrm{T}}$ is the hidden layer output matrix over all training instances, $\beta$ is the weight matrix of the output layer, and $\mathrm{T}=[t_1, t_2,..., t_N]$ denotes the regression target

matrix. Obviously, in Eq. (3), only $\beta$ is unknown, so we can adopt the least square algorithm to acquire its solution, which can be described as follows.

$$\beta = \mathrm{H}^{\dagger}\mathrm{T}=\begin{cases}\mathrm{H}^{\mathrm{T}}(\mathrm{H}\mathrm{H}^{\mathrm{T}})^{-1}\mathrm{T}, \text{ when } N \le L\\ (\mathrm{H}\mathrm{H}^{\mathrm{T}})^{-1}\mathrm{H}^{\mathrm{T}}\mathrm{T}, \text{ when } N > L\end{cases} \tag{4}$$

where $\mathrm{H}^{\dagger}$ denotes the Moore-Penrose generalized inverse of the hidden layer output matrix H, which can guarantee the solution is the least-norm least-squares solution of Eq. (5).

In our work, the instance $x_i$ is a $q\times1$ vector, where $q$ denotes the selected relevant genes, and then it is used as the input in ELM to regress the value of target gene on the same instance. In other words, ELM models an approximate $q$-to-1 map between the selected relevant genes and the target gene. However, the map is non-linear but not linear as modeling by KNNimpute [4] or LSimpute family [7-8] algorithms.

### 2.3 Construction of relevant feature space

To further improve the robustness and accuracy of imputation results, we also introduce the ensemble learning to fill the missing gene expression values. Specifically, considering in general, DNA microarray data is small-sized but high-dimensional, it is easier to disturb the feature space but not sample space for generating diverse base learners, hence in this work, we adopt feature subspace but not bagging as the ensemble learning framework.

In the relevant feature space containing $K$ co-expression genes, we randomly and independently extracted $q$ genes for $M$ times, where $q<<K$ and $M\ge1$. Then combining each subspace with the target gene, and on each $(q+1)$-dimensional subset, we train an ELM regression model that using $q$ relevant genes as input and the target gene as output. Specifically, the training data is required to be complete. Next, on each regression model, we input the expression values of the corresponding $q$ relevant genes on the target instance $s_j$, and then run the model to acquire the corresponding output $\tau_{ij}$ which can be seen as an approximation for the missing gene expression value $e_{ij}$. Finally, the imputation value of $e_{ij}$ can be calculated by,

$$\psi_{ij} = \sum_{k=1}^{K} \tau_{ij}^{k} \Big/ K \tag{5}$$

where $\tau_{ij}^{k}$ denotes the approximation value provided by the $k$th model, and $\psi_{ij}$ represents the ensemble imputed value for the position $e_{ij}$.

It is clear that the ensemble learning model adopts the average value of outputs from all sub-models as the final imputation value. In comparison with the single regression model, this ensemble model is expected to provide an more accurate and robust result.

Combining the idea of relevant feature space construction, ELM regression modeling and feature subspace ensemble, a novel imputation algorithm named EELMimpute is proposed. Its procedure can be simply described as follows.

**Algorithm**. Pseudo-code description of EELMimpute

**Input**: A microarray data set $G\in\Re^{m\times n}$ with missing values, the number of hidden nodes $L$ in ELM, the dimension of relevant space $K$, the dimension of feature subspace $q$, and the number of base regression models $M$.

**Output**: A complete microarray data set $G'$ without missing values.

**Procedure**:

1. **For** $i$=1:$m$

2. **If** $g_i$ does not contain the missing value in sample space
3. Put the copy of $g_i$ into $G'$ and return;
4. **Else**
5. Find $K$ relevant co-expression genes for $g_i$ by Eq.(1);
6. **For** $j$=1:$M$
7. Select $q$ genes from $K$ relevant genes randomly;
8. Train an ELM regression model using the instances with the complete $q$ selected gene values and the target gene (i.e., $g_i$) values by Eq.(3) and Eq.(4);
9. **End for**
10. Find all missing positions for the target gene $g_i$;
11. Use the trained $M$ regression models to approximate the value of each missing position, and impute them by using Eq.(5);
12. Put the imputed completely $g_i$ into $G'$;
13. **End if**
14. **End for**
15. Output the imputation microarray data set $G'$.

# 3 Experiments

## 3.1 Datasets

In our experiments, eight different DNA microarray data sets were used to evaluate our proposed algorithm and to compare with the other imputation algorithms. Table 1 provides the basic description for the used DNA microarray data sets.

**Table 1.** Datasets descriptions.

| Dataset | Number of instances | Number of genes | Data Type |
|---|---|---|---|
| Colon | 62 | 2000 | Non-time series |
| SRBCT | 83 | 2308 | Non-time series |
| Leukemia | 72 | 7129 | Non-time series |
| Hepatocellular | 22 | 10523 | Non-time series |
| Drosophila | 48 | 5070 | Non-time series |
| Candida | 9 | 16052 | Non-time series |
| Saccharomyces | 16 | 5282 | Time series |
| Yeast | 28 | 5716 | Time series |

## 3.2 Experimental settings

In the experiments, we compared the proposed EELMimpute algorithm with several popular and state-of-the-art imputation algorithms, including SVDimpute [4], KNNimpute [4], BPCAimpute [5], LLSimpute [6], SLLSimpute [7] and ILAW-LSimpute [8]. For each imputation algorithm, the parameters use the best default ones recommended in the corresponding references.

As for the proposed EELMimpute, the parameters are empirically designated as that $K$ equals to be 200, $q$ equals to be 5, $M$ equals to be 100 according to the feedback of the a huge of experimental results. Considering $q$, which corresponds to the input dimension in ELM, is small enough, the number of hidden-layer nodes $L$ is empirically set to be 10 as a default setting, because of some previous work have indicated that for low-dimension data, it is inappropriate for assigning an excess of hidden-layer nodes in ELM with two

considerations as follows: 1) it would add unnecessary training time consumption; 2) the model tends to be overfitting.

We compared all imputation algorithms on eight data sets with different missing rates, 1%, 5%, 10%, 15% and 20%, respectively. That means 1%, 5%, 10%, 15% and 20% entries in the original complete data sets are randomly missed. In order to evaluate the imputation quality and compare the performance of various imputation algorithms, the normalized root mean square error (NRMSE), which is presented as follows, is used as the performance metric,

$$NRMSE=\frac{\sqrt{\text{mean}[(e_{\text{real}}-e_{\text{impute}})^2]}}{\text{std}(e_{\text{real}})} \tag{6}$$

where $e_{\text{real}}$ and $e_{\text{impute}}$ respectively indicates the real value and the estimated value of a missing position in the DNA microarray data, the mean function denote the mean square error of all missing positions imputed by an imputation algorithm, and the std function denotes the calculation of standard deviation.

In this paper, each experiment is randomly conducted ten times, then the average NRMSE is given to evaluate the quality of each imputation algorithm, which is also a popular practice used in the experimental settings of most previous work [5-8].

### 3.3 Results and analysis

Fig. 1 presents the NRMSE performance of each imputation algorithm at different missing rates on eight DNA microarray data sets. Overall, the proposed EELMimpute algorithm has better performance on nearly all data sets at most missing rates. Specifically, EELMimpute averagely reduces 26.26%, 6.95%, 2.96%, 9.44%, 7.70% and 3.95% NRMSE in comparison with SVDimpute, KNNimpute, BPCAimpute, LLSimpute, SLLSimpute and ILAW-LSimpute algorithms, respectively. Meanwhile, the proposed algorithm has also acquired 28 best results in 40 independent testing conditions (8 data sets × 5 different missing rates). The reasons for the success of EELMimpute algorithm may be attributed to the use of ELM regression model which provides a powerful nonlinear mapping ability, and the adoption of ensemble learning framework which provides a stronger generalization. Actually, EELMimpute can be seen as a multiple imputation algorithm for estimating the missing values.

Another interesting phenomenon that can be observed in Fig. 1 is that the proposed EELMimpute algorithm is more stable than the other imputation algorithms, i.e., its performance can be less impacted by the missing rate. We believe that its stability can mainly attribute to its feature space construction mechanism. As we know, no matter for KNNimpute or LSimpute family algorithms, its neighboring genes only come from those complete genes without missing values, thus with the increase of missing rate, the calculation of relevant genes will become less and less accurate, further degrade the quality of imputation. However, our proposed EELMimpute algorithm always uses the whole gene space to extract the relevant feature space, and when there exists missing values for one specific gene, only the positions without missing value participate in the calculation of Pearson correlation coefficient, thereby the useful information can be fully reserved. Another reason lies in that the use of ensemble learning framework, which has been testified to be effective for improving the stability of learning results.

Next, we tested the actual difference between the EELMimpute algorithm and the other imputation algorithms in statistics. Specifically, the critical difference (CD) metric is used to show the difference of various algorithms. Fig. 2 shows the CD diagram at a standard level of significance $\alpha$=0.05, where the average ranking of each imputation algorithm is

marked along the axis (higher rankings to the left). In CD diagram, if a group of algorithms are not significantly different under Nemenyi test [12], these algorithms will be connected by a thick line.

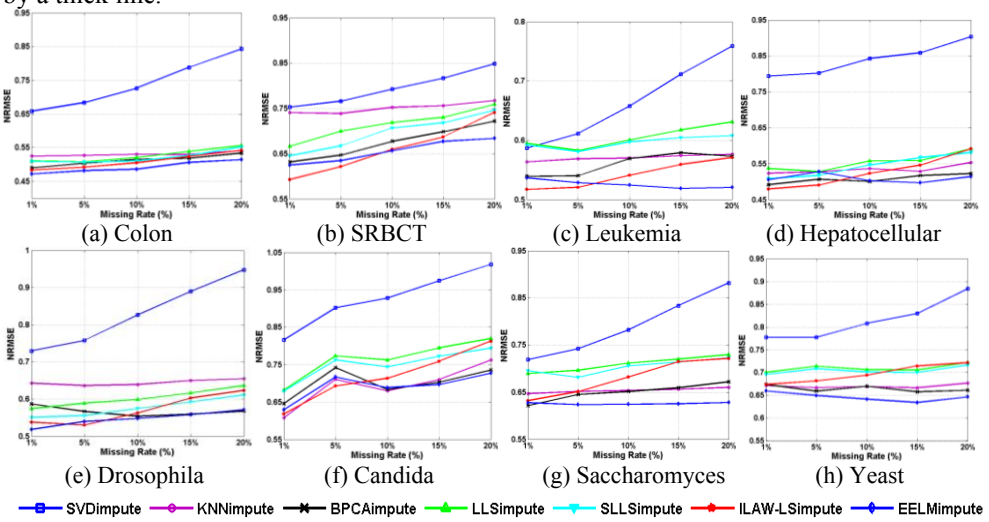


**Fig. 1.** Average NRMSE of the SVDimpute, KNNimpute, BPCAimpute, LLSimpute, SLLSimpute, ILAW-LSimpute and the proposed EELMimpute algorithms at different missing rates on eight DNA microarray data sets (a) Colon, (b) SRBCT, (c) Leukemia, (d) Hepatocellular, (e) Drosophila, (f) Candida, (g) Saccharomyces and (h) Yeast.

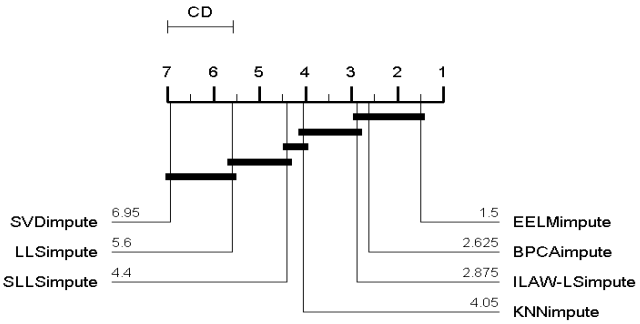


**Fig. 2.** CD diagram of various imputation algorithms at a standard level of significance $\alpha$=0.05.

From the results shown in Fig. 2, we observe that the EELMimpute algorithm achieves the statistically superior imputation performance than SVDimpute, LLSimpute, SLLSimpute and KNNimpute algorithms, and although we can't say it has significant difference with BPCAimpute and ILAW-LSimpute algorithm, it has a lower average rank than those two algorithms. To summarize, the proposed EELMimpute algorithm is a better choice than those popular imputation algorithms when there exists missing values in DNA microarray data.

## 4 Concluding remarks

In this paper, we presented EELMimpute, which is a flexible and robust ensemble regression model based on extreme learning machine for imputing the missing values in DNA microarray data. For a target gene, EELMimpute first constructs a relevant feature space for it by utilizing Pearson correlation analysis method, then randomly divides the

space into multiple subspaces and trains ELM regression model on each subspace, finally integrates all models to calculate the imputation results for the missing values belonging to the target gene. The experimental results on eight different DNA microarray data sets have verified the effectiveness of the proposed EELMimpute algorithm. In comparison with several popular imputation algorithms, it has acquired significantly superior results.

## References

1. S.D. Krämer, J. Wöhrle, P.A. Meyer, Sci. Rep., **9**, 13940 (2019).
2. A. Alonso, V. Larraga, P.J. Alcolea, Acta Trop., **187**, 129-139 (2018).
3. S. Draghici, P. Khatri, A.C. Eklund, Trends Genet., **22**, 101-109 (2006).
4. O. Troyanskayaet, M. Cantor, G. Sherlock, Bioinformatics, **17**, 520-525 (2001).
5. S. Oba, M.A. Sato, I. Takemasa, Bioinformatics, **19**, 2088-2096 (2003).
6. H. Kim, G.H. Golub, H. Park, Bioinformatics, **21**,187-198 (2005).
7. H. Wang, C.C. Chiu, Y.C. Wu, BMC Syst. Biol., **7**, S11 (2013).
8. Z. Yu, T. Li, S.J. Horng, IEEE T Nanobiosci., **16**, 21-33 (2017).
9. G.B. Huang, Q.Y. Zhu, C.K. Siew, Neurocomputing, **70**, 489-501 (2006).
10. G.B. Huang, H. Zhou, X. Ding, IEEE Trans. Syst. Man Cybern. B, **42**, 513-529 (2012).
11. G. Huang, G.B. Huang, S. Song, Neural Netw., **61**, 32-48 (2015).
12. J. Demsar, J Mach. Learn. Res., **7**, 1-30 (2006).