https://doi.org/ 10.33472/AFJBS.6.Si2.2024.1314-1324

AFJBS

African Journal of Biological Sciences

Journal homepage: http://www.afjbs.com

Research Paper Open Access

REDEFINING MICROARRAY DATA ANALYSIS: MAXIMIZING CLASSIFICATION ACCURACY WITH ENHANCED PREPROCESSING STRATEGIES FOR MISSING VALUE HANDLING

D Saravanakumar 1 , S K Mahendran 2

1 Research Scholar (Computer Science), Bharathiar University, Coimbatore,
Tamil Nadu, India. https://orcid.org/0009-0005-9239-4362 , E-mail: saranji@gmail.com
2 Assistant Professor, Department of Computer Science, Government Arts College
Coimbatore, Tamil Nadu. India. E-mail: sk.mahendran@yahoo.co.in

Article History

Volume 6,Issue Si2, 2024

Received:27 Mar 2024

Accepted : 30 Apr 2024

doi: 10.33472/AFJBS.6.Si2.2024.1314-1324

ABSTRACT

A dataset with no missing values is said to be a complete dataset. A complete dataset is important during the analysis of microarray data and its classification. The goal of missing value handling algorithm is to address the issue of missing data in the microarray dataset and provide a reliable and accurate estimates of these values. In this work, an enhanced imputation method that combines single and multiple methods is proposed. The proposed method handles the missing values works using the information regarding the missing percentage in the dataset. The proposed method works in two stages. The first stage decides on the imputation method based on its missing rate and the second stage uses the decided method to impute the missing values. Experimental results, using six cancer datasets, proved that the proposed method is highly efficient and was able to improve the classification performance in terms of accuracy and speed.

1. INTRODUCTION

Cancer is a phrase used to describe the uncontrolled proliferation of cells that can damage the function of other organs by forming additional tissues known as masses. More than 9.6 million people die from cancer each year, making it the second most lethal disease in the world (WHO, 2022). By 2040, this number is projected to increase to 27.5 million (Cancer Research UK, 2022), making the early and effective diagnosis and treatment of this disease an urgent global necessity. Given that there are more than 100 different forms of cancer, the work of identifying and classifying them is quite difficult. In these circumstances, analysis of microarray gene data can offer the greatest advantages. Microarray data analysis has been widely utilized for cancer classification, enabling the identification of distinct cancer subtypes and aiding in accurate diagnosis.

The usage of machine learning classifiers to identify cancer from microarray data is the most frequently used solution used by several automatic cancer identification systems (Osama *et al.*, 2023). The automatic cancer identification systems treat cancer identification as a binary classification model, where the input microarray data is placed as 'normal' or 'cancerous'. Raw microarray datasets have several issues that include abnormal values, measurement variability, batch effects and missing values (Ramasamy *et al.*, 2008). Careful consideration of these limitations and the application of suitable data analysis techniques are necessary to ensure accurate and reliable classification of microarray data. It is important to note that many of these disadvantages can be mitigated or minimized through appropriate preprocessing methods. Preprocessing is a crucial step in the analysis of gene expression profiles obtained from microarray experiments. It is a step that converts the raw microarray data into a form that is more suitable for the computational environment. In this paper, a preprocessing algorithm to handle missing values in microarray data, is considered.

According to Khan *et al.* (2021), a feature with no corresponding data value is considered to have a missing value. Microarray experiments may have missing values due to technical or biological reasons. It was found by several scientists that microarray datasets with missing data, affect the performance of various data mining tasks, like clustering, classification and identification of differential expressions (Emmanuel *et al.*, 2021; Gond *et al.*, 2021). Presence of missing values also result in performance loss, difficulty while analyzing data and biased results because of the discrepancies between missing and available complete data. Thus, handling missing values is considered very important and is the focal point of this work.

The most straightforward missing value handling technique is to exclude instances with missing values. However, these can result in misclassification and are not very efficient. Alternately, advanced techniques that can deal with missing values in an effective manner can be applied. Several researchers use imputation algorithms, which substitutes a missing attribute value, with a value estimated by an algorithm. The imputation algorithms can be either Single Imputation (SI) or Multiple Imputation (MI) algorithms. SI algorithms replace a missing value with a single value defined by certain rule (Enders, 2010), while MI algorithms impute several values (Graham and Hofer, 2000). SI algorithms are efficient and simple but might fail when the missing rate in a dataset is high. Rezvan *et al.* (2015) and Umar and Gray, 2023 found that MI outperforms SI in terms of uncertainty representation and variance brought on by the missing value. However, they might be unproductive when the missing rate is low, as they have high time complexity. It is important to note that the performance of missing value handling algorithm vary depending on the dataset and the amount of missing in it. Vigorous testing and experiments are required to select the best algorithm that can replace missing values with reliable values in order to ensure the reliability and clinical applicability of gene expression-based cancer classification approaches. In this research work, in order to efficiently handle the missingness in the huge microarray datasets, a simple rule-based selection method based on missing rates is combined with SI and MI algorithms, along with clustering and bootstraping algorithms.

The rest of the paper is organized as follows. Section 2 presents the methodology of the proposed missing value handling algorithm. The experimental results evaluating the proposed algorithm is presented and discussed in Section 3, while Section 4 concludes the work with future research directions.

2. METHODOLOGY

The proposed missing value handling algorithm is referred to as Enhanced Imputation Method Combining Single and Multiple Methods or EMI_SMImpute, in this work. This method handles missing values in microarray datasets in two stages.

- Stage 1 : Decide on imputation algorithm using rule-based selection method
- Stage 2 : Estimate missing values using the decided algorithm

The rule-based selection method uses the missing rates to select an appropriate algorithm to handle the missing values in the microarray dataset. The missing rate of a dataset is estimated using Equation (1).

$$\text{MV\%(D)}= \left(\sum_{k} \text{NaN}(k) / \text{N} \right) * 100 \tag{1}$$

Here, MV% is missing value percentage in a microarray dataset D, NaN represents the missing value, k represents the columns (genes) in D and N is the total number of rows (samples) in D. The above equation starts by counting the number of missing values in each column, which are then summed to obtain the total number of missing values. This value is then divided by the length of the column (or number of rows), to obtain MV% in D. The rule-based selection method (Equation 2) uses MV% then to decide on the imputation method to use. In this equation, column 1 represents the MV% and column 2 represents the imputation method selected.

$$\begin{cases} < 1\% & \text{No Imputation Algorithm} \\ 1\% - 5\% & \text{Median Imputation Algorithm} \\ 5\% - 15\% & \text{Single Imputation Algorithm} \\ > 15\% & \text{Multiple Imputation Algorithm} \end{cases} \qquad (2)$$

After identifying the algorithm to use, the second stage uses this selected algorithm to construct the complete dataset. When the missingness is less than one percent, then it is considered trivial as it does not affect the classification performance and therefore, is ignored. When MV% is between one percent and five percent, the scenario is considered manageable and a simple median imputation algorithm is used. The scenario when MV% is between 5% and 15%, requires sophisticated method, and a SI method based on enhanced weighted KNN imputation method, is proposed. The enhancement operations include a filtering algorithm-based on K-Means clustering algorithm, a weighting scheme and an automatic K value estimation procedure. When the missing rate is higher than 15%, then the missing value handling needs to be performed very carefully and therefore, a multiple imputation algorithm is proposed. The multiple imputation algorithm proposed uses the enhanced SI algorithm with bootstrapping to estimate missing values. This algorithm also uses the filtering algorithm proposed with SI algorithm. The imputation method selected are described in the following subsections.

2.1. Median Imputation Algorithm

Median imputation (MedImpute) is a common algorithm used to estimate missing values in microarray data (Hameed *et al.*, 2022). It involves replacing missing values with the median value of the corresponding feature (gene) across all samples. The MedImpute works using three major steps. The algorithm begins by identifying the missing values in the microarray dataset. The next step, for each feature, computes the median value based on the available non-missing values (or known values), in the class where the instance with missing attribute belongs. The median is defined as the middle value in a sorted list. In case, there are even number of non-missing values, then the algorithm estimates the average of the two middle values, as the median value. If the value x_{ij} of the k-th class, C_k, is missing, the missing value is estimated using Equation (3).

$$x_{ij} = \text{median}_{\{i:xij \in C\}} \{x_{ij}\} \qquad (3)$$

The final step, then replaces the missing values with the calculated median value for the respective feature. This method is applied on each feature independently. The MedImpute algorithm is simple and has less computational complexity and moreover the imputed values less affected by extreme values (outliers) compared to mean imputation.

2.2. Enhanced KNN Imputation Method

When the missing values amount to 5%-15%, then an enhanced KNN imputation algorithm is used. The KNN imputation algorithm (KNNImpute) (Zheng and Huang, 2023) is a widely used algorithm, where the missing values are handled by considering the number of complete instances that are most similar to the instance that has missing values. The similarity between the complete and incomplete instances are estimated using Euclidean distance function. The algorithm requires the input incomplete microarray dataset to be divided into two sets, namely, Complete Set (CS) that has instances without missing values and InComplete Set (ICS) having missing values to be imputed. Let MD denote the incomplete dataset. The first step calculates the distance between ICS and all instances in CS. Let K denote the number of neighbours to be considered during imputation. The K neighbours are identified based on their proximity to the observation with missing values. The identification of the neigbours begins by selecting a gene with missing value and then estimating the distance between this gene and all other genes in the dataset using Euclidean distance. The result is arranged in ascending order and the top K genes are selected as neighbours to the missing gene. The KNNImpute algorithm, uses the mean method, on these identified K neighbours, to estimate the missing value. These steps are repeated for all missing values in ICS. This conventional algorithms has two main issues, which when handled properly, can further enhance the algorithm's performance. The algorithm has high search time and its performance degrades when the missing rate is high. This work proposes methods to solve both these issues, thus enhancing the conventional KNNImpute algorithm. The proposed algorithm is referred to as EWKNNImpute (Enhanced Weighted KNNImputation) algorithm in this work.

The EWKNNImpute algorithm modifies the KNNImpute algorithm in two manners. The first is to use a Clustering-based Filtering (CF) algorithm that groups instances in the microarray dataset into two classes, namely, significant and insignificant, and removes the insignificant features. The second manner of improvement is to use a weighted KNNImpute algorithm incorporated with automatic K estimation method and an enhanced inverse distance weighting algorithm. The steps involved are shown in Figure 1.

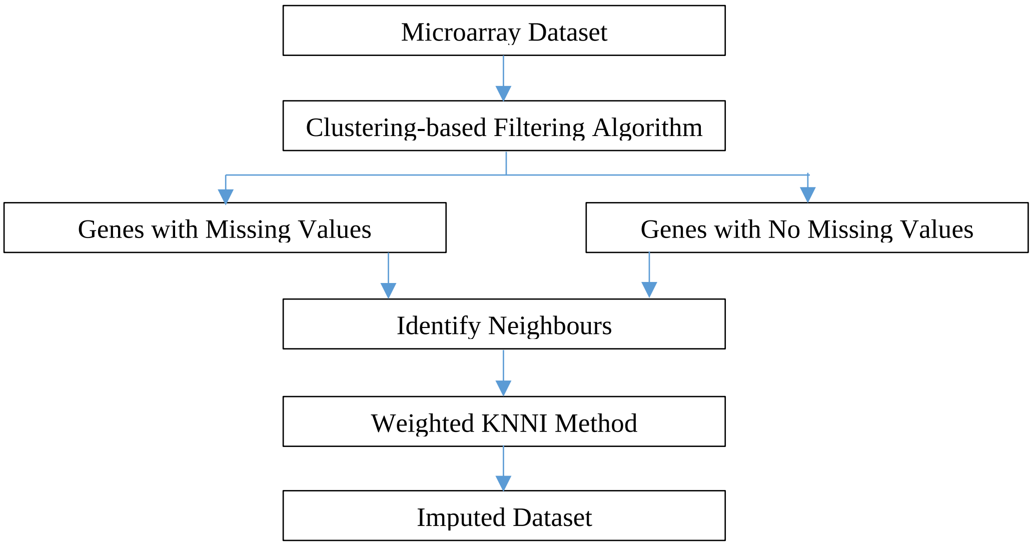
Microarray Dataset
Clustering-based Filtering Algorithm
Genes with Missing Values
Genes with No Missing Values
Identify Neighbours
Weighted KNNI Method
Imputed Dataset

Figure 1 : EWKNNImputation Algorithm

2.3. Clustering-based Filtering (CF) Algorithm

The CF algorithm, starts with the construction of a feature-feature similarity matrix, which uses the Euclidean distance similarity metric to estimate the similarity between each pair of genes in the microarray dataset. In the similarity matrix thus constructed, each entry denotes how similar two genes are. Using the similarity matrix as input, the CF algorithm, then performs KMeans clustering to group similar genes together. The KMeans clustering algorithm requires two important user-defined input, namely, K (number of clusters) and initial centroids. To avoid confusion with the K parameter of KMeans clustering and KNNI method, the K with KMeans algorithm is referred to as K_C in this work. To determine K_c and initial seeds, a pre-clustering approach is used. The pre-clustering algorithm used is a simple single-pass clustering algorithm (https://en.wikipedia.org/wiki/One-pass_algorithm), which is applied to the whole dataset. This algorithm was selected for pre-clustering, as it can form clusters in a fast manner (as it scans the dataset only once), without requiring buffering and has low time complexity. The number of clusters produced by this algorithm is taken as K_c, while its centroids are used as initial seeds. Using these two parameters, the whole dataset is again clustered using KMeans algorithm. The resulting clusters are analyzed, to identify clusters which do not have any impact during imputation.

During this analysis, all small clusters (whose width is less <0.02) are first removed, as they do not contribute during imputation. Next, cluster relevancy is estimated for each large clusters. The relevancy is estimated using a cluster validity index measure (José-García and Gómez-Flores, 2023). In this work, the cluster validity index measure used is the silhouette measure, which measures how well each feature fits into its assigned cluster compared to other clusters. To measure the overall quality, the average silhouette coefficient for all data points in the dataset is estimated and is used as the cluster relevancy indicator of a cluster. A high average silhouette coefficient indicates that the cluster is highly relevant, more distinct and well-separated. The clusters with low average silhouette coefficient indicates that the results, may be ambiguous or overlapping and may hinder with the correct handling of missing values and therefore, are removed.

2.4. Weighted KNNImputation Algorithm

The Weighted KNNImpute (WKNNI) algorithm, an enhanced variant of KNNImpute, assigns different weights to the nearest neighbours according to how close or similar they are to the target feature that is to be imputed. The weights thus estimated are used to represent the relative importance of each of these nearest neighbours during the process of imputation (Khan, 2020; Ling and Dong-Mei, 2009). The core idea behind a WKNNI algorithm is to give more weights to features that are closer together and less weight to features that are farther apart. The most important part of WKNNI method, thus, is the weighting scheme used. In this research work, the most frequently used Inverse Distance Weighting (IDW) (https://en.wikipedia.org/wiki/Inverse_distance_weighting) scheme is enhanced and used.

In conventional IDW, the weight assigned to each neighbor is proportional to the inverse of its distance to the observation with the missing value. The weight of each neighbor is calculated using Equation (4).

$$w_i = 1/d_i \quad (4)$$

In the above equation, w_i is the weight assigned to the i^{th} neighbor, and d_i is the Euclidean distance between the feature with the missing value and the ith neighbor. The intuition behind IDW is that closer neighbors are more similar to the observation with the missing value, and therefore should be given more weight in the imputation.

The IDW method is enhanced through the use of a robust weighting technique. This method instead of using the conventional Euclidean distance, estimates robust distances that are less affected by extreme gene values. The robust distance measure used is Median Absolute Distance (MAD) (https://en.wikipedia.org/wiki/Median_absolute_deviation), which can measure the variability or spread of the dataset and include them during weighting.

2.5. Multiple Imputation Method

Multiple Imputation (MI), is a concept that was originally designed for missing value handling in public-use datasets (Rubin, 2004), which later was extended to large sized datasets like microarray datasets (Kim *et al.*, 2004). MI algorithm is a statistical technique used to handle missing data in a dataset by creating multiple plausible imputed datasets, each with a different set of imputed values for the missing data. The basic idea behind MI is to estimate the uncertainty associated with the missing data by simulating multiple possible values for the missing data, and then combining the results from each imputed dataset to obtain a single set of estimates (Suyundikov *et al.*, 2015). In this work, when the missing values percentage is more than 15%, then a MI method is used. This method begins with the incomplete microarray dataset, D. Next a bootstrapping with replacement is performed N times to obtain N variants of the dataset (N = 3 in this work). Let this be denoted as D_1, D_2 and D_3. Each complete dataset is analyzed using imputed algorithm from previous section (EWKNNI algorithm). This results in N analyzed results (R_1, R_2, R_3), which are pooled into one final results (R) that adequately reflects the amount of uncertainity in the estimates. Thus, a MI algorithm uses a three-step procedure, imputation, analysis and pooling, to handle the missing values in the dataset. This is illustrated in Figure 2.

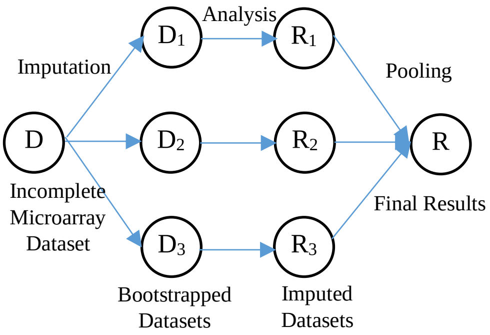

Figure 2 : Multiple Imputation Algorithm

The above described algorithm is enhanced through the use of the CF algorithm and using the EWKNNI method to impute the multiple datasets. The proposed MI algorithm is termed as Enhanced Imputation Method Combining Single and Multiple Methods (EMI_SMImpute) in this work. The first step of EMI_SMImpute algorithm is to use the CF algorithm to obtain a refined microarray dataset (D*). The next step of EMI_SMImpute algorithm then identifies the genes with missing values. The third step performs bootstrapping to create multiple copies of D*. Bootstrapping is a resampling technique used to estimate the sampling distribution of a statistic or to assess the variability of an estimate. It involves creating multiple bootstrap samples by resampling from D* with replacement. Each bootstrap sample is treated as a surrogate population, allowing for inference and estimation based on the resampled data. In this work, a basic bootstrapping method, as described below, is used.

- Step 1 : Start with the original dataset of size N.
- Step 2 : Randomly select N data points from the original dataset with replacement to create a bootstrap sample.
- Step 3 : Repeat Step 2 multiple number of times to generate multiple bootstrap samples.

The EMI_SMImpute algorithm then proceeds with the imputation process, which imputes missing values in the multiple datasets obtained using bootstrapping. This step results with multiple imputed datasets. For each imputed dataset, the EMI_SMImpute algorithm performs the following steps.

- Step 1 : Use MedImpute algorithm on every missing value in the dataset, D. These imputed values are considered as 'Place Holders or PHs'.
- Step 2 : Set back the PHs of one feature (G1) to missing.
- Step 3 : Impute missing values in G1 using EWKNNI algorithm.
- Step 4 : Replace PHs by imputed values
- Step 5 : Consider the next feature
- Step 6 : Repeat Steps 2 to 4 for all features with missing values
- Step 7 : Assess the convergence of the imputed datasets, which is performed by examining the stability of the imputed values across the iterations.

Finally, the EMI_SMImpute algorithm combines the imputed results by calculating the average of the estimated values across the imputed datasets. The steps of EMI_SMImpute algorithm are summarized in Figure 3.

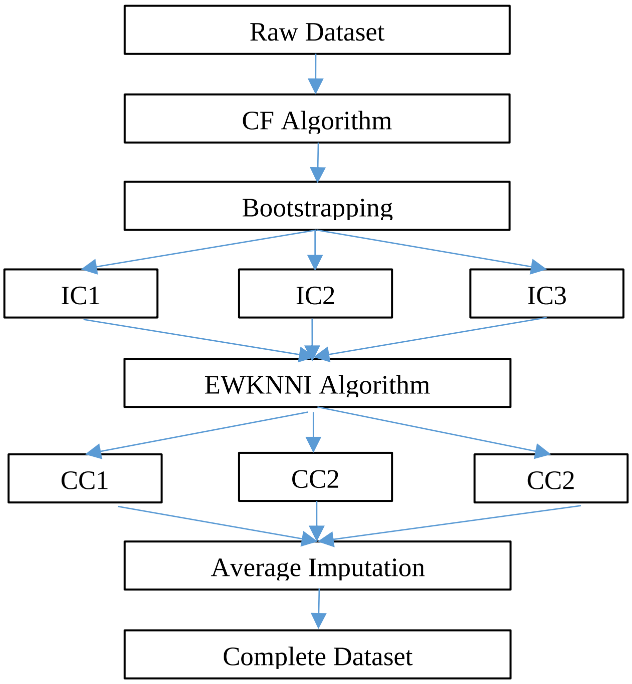

Figure 3 : Steps in EMI_SMImpute Algorithm

3. EXPERIMENTAL RESULTS

The performance of the proposed algorithms was evaluated using several experiments that used six microarray datasets and various performance metrics. The six cancer datasets selected are breast (West *et al.*, 2001), lung (Gordon *et al.*, 2002), lymphoma (Shipp *et al.*, 2002), leukEMI_SMImputea (Golub *et al.*, 1999), colon (Alon *et al.*, 1999) and prostate (Singh *et al.*, 2002) datasets. Each of the proposed algorithm was compared with the conventional and existing methods. Two performance metrics, namely, Normalized Root Mean Square Error (NRMSE) and Execution Speed (Seconds), were used during performance evaluation. The effect of using the missing value algorithms on tumour classification was analyzed using two performance metrics, namely, Accuracy (%) and Classification Time (Seconds). This work assumes that the type of missing data in these datasets is MAR type and thus, the missing data can be deduced from other available data. The coding scheme used during discussion is presented in Table 1.

TABLE 1 : CODING SCHEME

NoImpute	No Imputation Method Used
MedImpute	Median Imputation Method
KNNImpute	KNN Imputation Method
EWKNNImpute	Enhanced Weighted KNN Imputation Method
EMI_SMImpute	Enhanced Multiple Imputation Method Combining Single and Multiple Methods

Figures 4 to 7 shows the performance of MedImpute, KNNImpute, EWKNNImpute and EMI_SMImpute methods while varying the missing rates between 5% and 25% in steps of 5% with respect to NRMSE performance metric, while varying the input dataset.
From the figures, it is evident that the MedImpute works well when the missing rate is very less (<5%), while the KNNImpute and EWKNNImpute produce best performance, when the missing rate is between 6%-15%. The proposed EMI_SMImpute handles missing values in a better manner when the missing rate is greater than 15%. Table 2 shows the comparison of the imputation algorithms while considering average NRMSE, for the six selected microarray datasets.

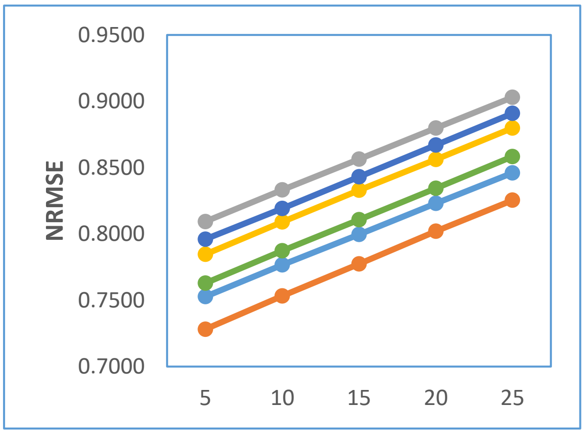

Figure 4 : MedImpute Method

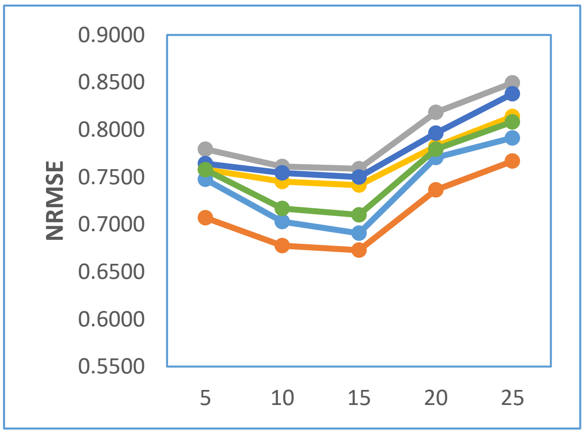

Figure 5 : KNNImpute Method

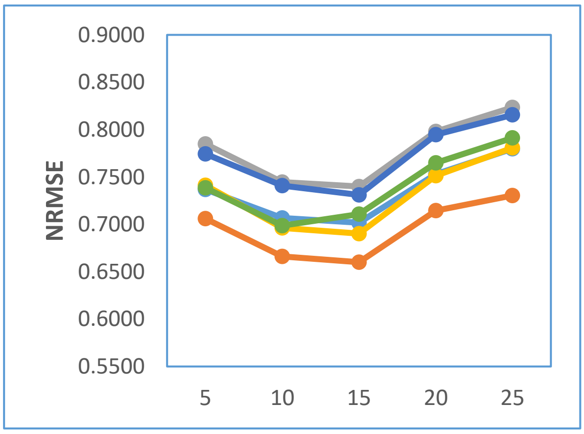

Figure 6 : EWKNNImpute Method

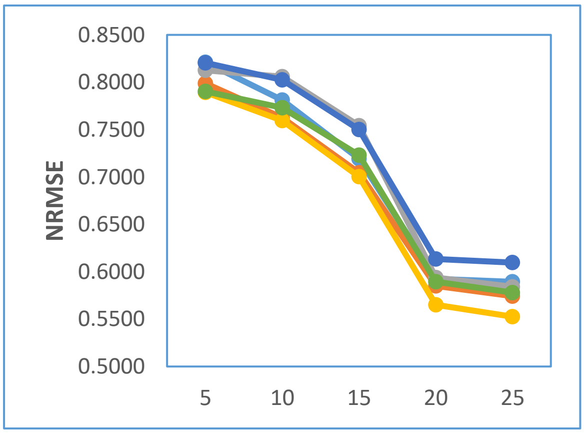

Figure 7 : EMI_SMImpute Method

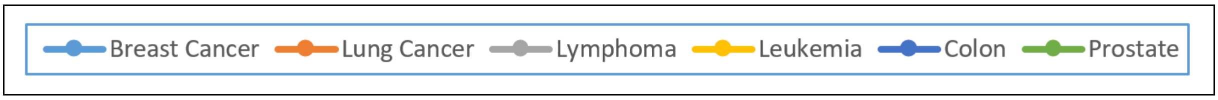

TABLE 2 : COMPARISON OF THE IMPUTATION ALGORITHMS - NRMSE

Datasets	MedImpute	KNNImpute	EWKNNImpute	EMI_SMImpute
Breast Cancer	0.7996	0.7407	0.7356	0.7007
Lung Cancer	0.7772	0.7121	0.6955	0.6850
Lymphoma	0.8564	0.7935	0.7782	0.7102
Leukemia	0.8325	0.7682	0.7320	0.6734
Colon	0.8432	0.7805	0.7713	0.7193
Prostate	0.8108	0.7544	0.7408	0.6907

The EWKNNImpute algorithm shows increase in NRMSE performance and produces NRMSE values between 0.7121 and 0.7935, when compared with the NRMSE values of KNNImpute algorithm, which was in the range of 0.6955-0.7782. This proves that the enhancement operations included, in the conventional KNNImpute algorithm, are highly successful. However, the performance of the proposed algorithm is high when compared to the other algorithms (0.6734-0.7193). This proves that the proposed algorithm works best and can be used to handle the missing values in the microarray dataset.

Table 3 shows the execution time (speed) of the imputation algorithms when tested with the six selected microarray datasets.

TABLE 3 : COMPARISON OF THE IMPUTATION ALGORITHMS (SPEED IN SECONDS)

Datasets	MedImpute	KNNImpute	EWKNNImpute	EMI_SMImpute
Breast Cancer	0.56	1.44	1.32	1.58
Lung Cancer	0.64	1.53	1.41	1.66
Lymphoma	0.54	1.45	1.35	1.56
Leukemia	0.53	1.45	1.37	1.55
Colon	0.44	1.33	1.22	1.46
Prostate	0.65	1.55	1.44	1.66

The speed analysis shows that median imputation method is the fastest. The speed complexity of the proposed algorithm is slightly high when compared to other KNN-based imputation methods. This result was expected as the proposed algorithm needs to execute multiple algorithms during imputation. However, as the proposed algorithm has produced highly efficient NRMSE and as medical applications require high accuracy, this increase in speed complexity is ignored.

The second stage of evaluation conducted experiments to analyze the effect of the imputation algorithm on classification. Figure 8 shows the performance of the classifier in terms of accuracy performance metrics, while using various selected datasets.

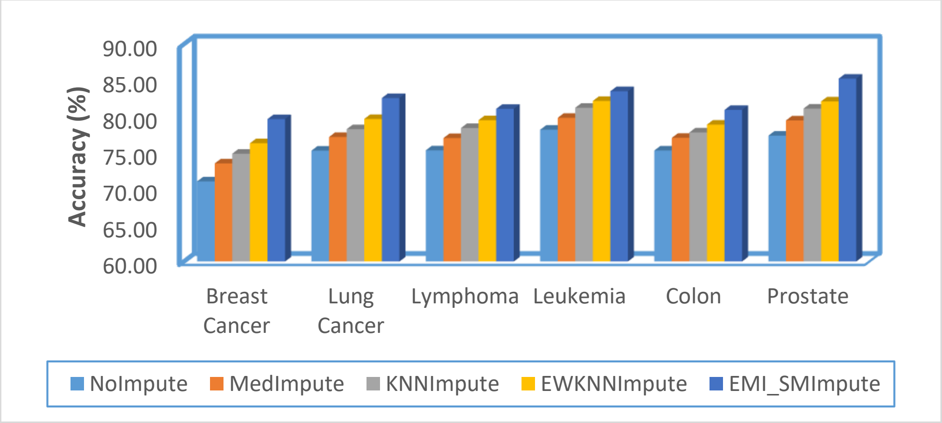

Figure 8 : Effect of Imputation Algorithms on Classification Accuracy

From the results it is evident that the usage of missing value algorithm greatly improves the classification performance. On average, the MedImpute, KNNImpute, EWKNNImpute and EMI_SMImpute algorithms respectively improved the performance by 2.48%, 4.05%, 5.45% and 5.85% respectively. However, comparing the imputation algorithms showed that the proposed algorithm is more efficient in improving the classification performance as it produces a more accurate complete dataset. On average, the EMI_SMImpute algorithm outperformed the MedImpute, KNNImpute, and EWKNNImpute algorithms in terms of accuracy by 5.85%, 4.31%, and 2.90%, respectively.

Figure 9 shows the time taken by the classifier when tested with a single test gene from the selected datasets.

Speed analysis revealed that the MedImpute is the fastest. Comparing KNNImpute and its enhanced variants, the enhanced variants have more speed during classification. Maximum speed efficiency was produced by the proposed EMI_SMImpute algorithm. This demonstrates the optimal algorithm for handling the missing values in the microarray dataset is the one that has been proposed.

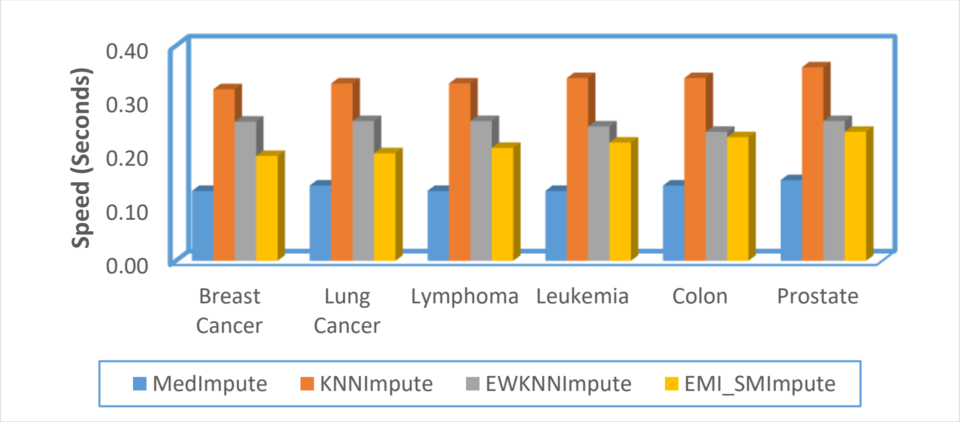

Figure 9 : Classification Speed

4. CONCLUSION

One of the main application of microarray data is the categorization of gene data into normal and malignant. In order to obtain maximum disease detection accuracy, several tasks, like preprocessing and gene selection, are performed, prior to classification. Preprocessing consists of various tasks that can be used to fine-tune the input microarray dataset. One of the most important and challenging task is missing value handling, which is the focal point of this research work. An algorithm based on K nearest neighbour imputation enhanced through the use of filtering algorithm, weights, combined single and multiple algorithms, is proposed. Experimental results proved that the performance of the proposed algorithm is high and help to improve the classification process. Apart from missing values, several other preprocessing tasks, like outlier detection and normalization, also help to improve the underlying classification process. Future work is planned to enhance the working of two preprocessing tasks and study their impact on classification performance.

REFERENCES

- Alon, U., Barkai, N., Notterman, D.A., Gish, K., Ybarra, S., Mack, D. and Levine, A.J. (1999) Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays, Proc. Nat. Acad. Sci. USA, Vol. 96, Pp. 6745–6750.
- Cancer Research UK (2022) Worldwide cancer incidence statistics, https://www.cancerresearchuk.org/ health-professional/cancer-statistics/ worldwide-cancer/incidence, Last Accessed During July, 2023.
- Emmanuel, T., Maupong, T., Mpoeleng, D., Semong, T., Mphago, B. and Tabona, O. (2021) A survey on missing data in machine learning, Journal of big data, Vol. 8, Issue 1, Article ID 140, Pp. 1-37.
- Enders, C.K. (2010) Applied Missing Data Analysis, 1st ed.; Guilford Press.
- Golub, T.R., Slonim, D.K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J.P., Coller, H., Loh, M.L., Downing, J.R., Caligiuri, M.A., Bloomfield, C.D. and Lander, E.S. (1999) Molecular classification of cancer: class discovery and class prediction by gene expression monitoring, Science, Vol. 286, Pp. 531-537.
- Gond, V.K., Dubey, A. and Rasool, A. (2021) A Survey of Machine Learning-Based Approaches for Missing Value Imputation, Third International Conference on Inventive Research in Computing Applications, Pp. 1-8.
- Gordon, G.J., Jensen, R.V., Hsiao, L.L., Gullans, S.R., Blumenstock, J.E., Ramaswamy, S., Richards, W.G., Sugarbaker, D.J. and Bueno, R. (2002) Translation of Microarray Data into Clinically Relevant Cancer Diagnostic Tests Using Gene Expression Ratios in Lung Cancer and Mesothelioma, Cancer Research, Vol. 62, No.17, Pp.4963-4967.
- Graham, J.W. and Hofer, S.M. (2000) Multiple imputation in multivariate research, Modeling Longitudinal and Multilevel Data: Practical Issues, Applied Approaches, and Specific Examples; Little, T.D., Schnabel, K.U., Baumert, J., Eds.; Lawrence Erlbaum Associates Publishers, Pp. 201-218.
- Hameed, W.M. and Ali, N.A. (2022) Missing value imputation techniques : A survey, UHD Journal of Science and Technology, Vol. 6, Issue 1, Pp. 72-81.
- Navatha S et al. 2023. Multitask Learning Architecture For Vehicle Over Speed As Traffic Violations Detection And Automated Safety Violation Fine Ticketing Using Convolution Neural Network And Yolo V4 Techniques. *Chinese Journal of Computational Mechanics*. 5 (Oct. 2023), 431–435.
- Kumar, E. Boopathi, and V. Thiagarasu. "Segmentation using Fuzzy Membership Functions: An Approach." *IJCSE, ISSN* (2017): 2347-2693.
- Reddy, C. S., Yookesh, T. L., & Kumar, E. B. (2022). A Study On Convergence Analysis Of Runge-Kutta Fehlberg Method To Solve Fuzzy Delay Differential Equations. *Journal of Algebraic Statistics*, *13*(2), 2832-2838.
- José-García, A. and Gómez-Flores, W. (2023) CVIK: A Matlab-based cluster validity index toolbox for automatic data clustering, SoftwareX, Vol. 22, Article ID 101359, Pp. 1-8.
- Khan, H., Wang, X. and Liu, H. (2021) Missing value imputation through shorter interval selection driven by Fuzzy C-Means clustering, Computers & Electrical Engineering, Vol. 93, Article ID 107230, Pp. 1-16.
- Khan, M.S. (2020) What is weighted KNN and how does it work, https://medium.com/@mohdsaeed.khan25/what-is-weighted-knn-and-how-does-it-work-aa8e461fd5d7, Last Accessed During July 2023.
- Kim, K.Y., Kim, B.J. and Yi, G.S. (2004) Reuse of imputed data in microarray analysis increases imputation efficiency, BMC Bioinformatics, Vol. 5, Article ID 160, Pp. 1-9.
- Ling, W. and Dong-Mei, F. (2009) Estimation of missing values using a Weighted K-Nearest Neighbors Algorithm, Proceedings of International Conference on Environmental Science and Information Application Technology, Vol. 3, Pp. 660-663.
- Mardia, K.V., Bibby, J.M. and Kent, J.T. (1979) Multivariate analysis, AcadEMI_SMImputec Press, New York.
- Osama, S., Shaban, H. and Ali, A.A. (2023) Gene reduction and machine learning algorithms for cancer classification based on microarray gene expression data: A comprehensive review, Expert Systems with Applications, Vol. 213, Part A, Article ID 118946, Pp. 1-12.
- Ramasamy, A., Mondry, A., Holmes, C. C. and Altman, D.G. (2008) Key issues in conducting a meta-analysis of gene expression microarray datasets, PLoS medicine, Vol. 5, Issue 9, Article ID e184, Pp. 1320-1332.
- Rezvan, P.H., Lee, K.J. and Simpson, J.A. (2015) The rise of multiple imputation: a review of the reporting and implementation of the method in medical research, BMC Medical Research Methodology, Vol. 15, Issue 1, Article ID 30, Pp. 1-14.

- Rubin, D.B. (2004) Multiple imputation for nonresponse in surveys, John Wiley & Sons.
- Shipp, M.A., Ross, K.N., Tamayo, P., Weng, A.P., Kutok, J.L., Aguiar, R.C., Gaasenbeek, M., Angelo, M., Reich, M., Pinkus, G.S., Ray, T.S., Koval, M.A., Last, K.W., Norton, A., Lister, T.A., Mesirov, J., Neuberg, D.S., Lander, E.S., Aster, J.C. and Golub T.R. (2002) Diffuse large B-cell lymphoma outcome prediction by gene-expression profiling and supervised machine learning, Nat Med Vol. 8, Pp. 68-74.
- Singh, D., Febbo, P.G., Ross, K., Jackson, D.G., Manola, J., Ladd, C., Tamayo, P., Renshaw, A.A., D'Amico, A.V., Richie, J.P., Lander, E.S., Loda, M., Kantoff P.W., Golub, T.R. and Sellers, W.R. (2002) Gene Expression Correlates of Clinical Prostate Cancer Behavior, Cancer Cell, Vol. 1, No. 2, Pp. 203-209.
- Suyundikov, A., Stevens, J.R., Corcoran, C., Herrick, J., Wolff, R.K. and Slattery, M.L. (2015) Accounting for Dependence Induced by Weighted KNN Imputation in Paired Samples, Motivated by a Colorectal Cancer Study, PLOS ONE, Vol. 10, Issue 4, Article ID e0119876, Pp. 1-15.
- Umar, U. and Gray, A. (2023) Comparing Single and Multiple Imputation Approaches for Missing Values in Univariate and Multivariate Water Level Data, Water MDPI, Vol. 15, Article ID 1519, Pp. 1-21.
- West, M. and Blanchette, C., Dressman, H., Huang, E., Ishida, S., Spang, R., Zuzan, H., Olson, J.A., Marks, J.R. and Nevins, J.R. (2001) Predicting the clinical status of human breast cancer by using gene expression profiles, Proc Natl Acad Sci., Vol. 98, Pp. 11462–11467.
- WHO (2022) Cancer, https://www.who.int/news-room/fact-sheets/detail/cancer, Last Accessed During July, 2023.
- Zheng, H. and Huang, T. (2023) New incomplete data imputation based on k-nearest neighbor type framework, IEEE 3rd International Conference on Power, Electronics and Computer Applications, Pp. 769-774.