Proper Imputation Techniques for Missing Values in Data sets

Tahani Aljuaid and Sreela Sasi
Department of Computer and Information Science
Gannon University
Erie, PA 16541, USA
tahani.aljuaid.n@gmail.com, sasi001@gannon.edu

Abstract— **Data mining requires a pre-processing task in which the data are prepared and cleaned for ensuring the quality. Missing value occurs when no data value is stored for a variable in an observation. This has a significant effect on the results especially when it leads to biased parameter estimates. It will not only diminish the quality of the result, but also disqualify for analysis purposes. Hence there are risks associated with missing values in a dataset. Imputation is a technique of replacing missing data with substituted values. This research presents a comparison of imputation techniques such as Mean\Mode, K-Nearest Neighbor, Hot-Deck, Expectation Maximization and C5.0 for missing data. The choice of proper imputation method is based on datatypes, missing data mechanisms, patterns and methods. Datatype can be numerical, categorical or mixed. Missing data mechanism can be missing completely at random, missing at random, or not missing at random. Patterns of missing data can be with respect to cases or attributes. Methods can be a pre-replace or an embedded method. These five imputation techniques are used to impute artificially created missing data from different data sets of varying sizes. The performance of these techniques are compared based on the classification accuracy and the results are presented.**

Keywords—Data Pre-processing; Expectation Maximization; Hot-Deck; K-Nearest Neighbor; Decision Tree classification; C5.0;

I. INTRODUCTION

The preprocessing phase in the data mining process helps the user to understand the data and to make appropriate decisions to create the mining models [1]. In this phase, it is important to identify the data inconsistencies such as missing data or wrong data. It also tries to fix them using appropriate techniques because these problems can influence the results of the model. Missing data is a common problem that has become a rapidly growing area and is the current focus of this research. Many techniques have been developed in order to solve this problem. Missing data might not cause any issue particularly when the data contains a small number of missingness. In this case, the simplest method is the "Deletion techniques" that are used to eliminate the attributes or cases. This method tends to be the default method for handling missing data. However, in many cases a large amount of missing data could drastically influence the result of the model. It is more constructive and practically viable to consider imputation for replacing the missing values. Imputation is a technique for replacing missing data with substituted values. If an important feature is missing for a particular instance, it can be estimated from the data that are present by using these imputations.

Selection of an imputation method always depends on the given data set, missing data mechanism [2], patterns [2], [3], and methods of handling missing values [4]. The problem with these techniques is that some imputation techniques may perform very well in some datatypes while the others may not.

This research presents a comparison of imputation techniques such as Mean\Mode, K-Nearest Neighbor, Hot-Deck, Expectation Maximization and C5.0 for Missing Values (MVs). Section II describes the missing data mechanism, pattern of MVs, and methods of dealing with MVs. Section III defines different imputation techniques. Section IV describes the difference between these imputation techniques based on the literature review. Section V focuses on the results of simulations done on the artificially created missing data from different data sets of varying sizes. Section VI presents the conclusion and future work which is followed by the references.

II. BACKGROUND RESEARCH

Various imputation techniques are available in literature. The selection of imputation technique may be based on datasets or may be related to the mechanisms. Yet some other research further dig into the patterns of MVs [2] [3] and on the methods of handling the missingness [4]. Some imputation techniques work very well with Integer while some others work only with categorical, yet some others can work with mixed datasets. Missing data mechanism is a key factor to decide if missing values can be imputed using some methods or discard the missingness. It is critical to know the characteristics of missingness because it contributes to the success or failure of the analytical process. However, the missing data mechanism is classified as Missing Completely at Random (MCAR), Missing At Random (MAR) or Not Missing At Random (NMAR).

MCAR is the highest level of randomness. It occurs when the probability of a record having a missing value for an attribute does not depend on either the observed data or the missing data. This can be solved by discarding all the cases of

978-1-5090-1281-7/16/$31.00 ©2016 IEEE

the missing attribute values. However, it may reduce the number of observations in the data set. MAR is when the probability of a record having a missing value for an attribute could depend on the observed data, but not on the value of the missing data itself. The way to handle this characteristic is by imputing the MVs using the existing data. NMAR occurs when the probability of a record having a missing value for an attribute could depend on the value of the attribute. Missing data mechanism that is considered as NMAR is non-ignorable. This can be solved by accepting the bias or impute the MVs using imputation.

There are different patterns of missing data. Some patterns associate with the cases while others associate with the attribute. Both these patterns help to understand the real data sets and the missingness. The case patterns include simple, medium, complex and blended. Simple case is when a record has at most one missing value. Medium case is when a record has missing values for a number of attributes ranging from 2% to 50% of the total number of attributes. Complex case is when a record has a minimum of 50% and a maximum of 80% attributes with missing values. Blended case is when a combination of records from all these three cases are missing. The attribute patterns consist of Univariate patterns, Monotone patterns and Arbitrary patterns. The Univariate patterns have all the missing values in one feature while the Monotone patterns have all the missing values at the end of the last three features. The Arbitrary patterns have missing values in random features. These are shown in Table 1.

TABLE 1. Pattern of Missing Values

Cases perspectives [3]	Attribute perspectives [2]
Simple	Univariate
Medium	Monotone
Complex	Arbitrary
Blended	

The Methods of handling the Missingness have two stages to solve the problem of MVs in the data sets. This can be done before the analysis or during the analysis according to [4] as shown in Fig. 1. In [4] methods are classified into

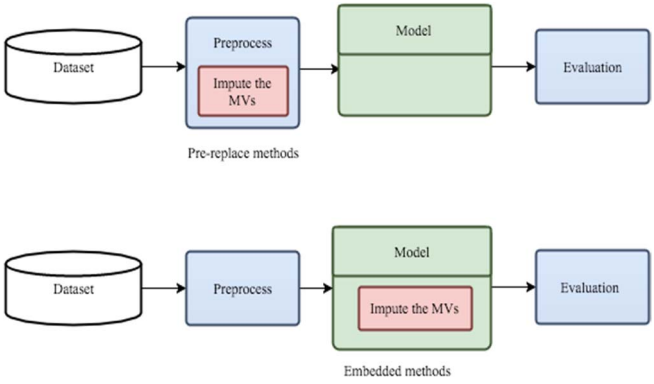

Fig. 1. Pre-replace and Embedded Methods

Pre-replacing method and Embedded method. The Pre-replacing method replaces the missing values before the data mining process. It works in the pre-process phase. The Embedded method is able to handle missing values during the data mining process. The missing values is imputed in the same time while creating the model. There are many imputations with different features to impute the missing data for these methods. The Pre-replacing method includes Mean-and Mode, Linear Regression, K-Nearest Neighbor (KNN), Expectation Maximization (EM), Hot-Deck (HD), and Autoassociative Neural Network techniques. The Embedded method includes the Casewise deletion, Lazy decision tree, Dynamic path generation, C5.0, and Surrogate split techniques. These are shown in Table 2.

TABLE 2. Methods of handling Missing Values

Pre-replace methods	**Embedded methods**
Mean /Mode	Casewise deletion
Linear Regression	Lazy decision tree
KNN	Dynamic path generation
EM	C5.0
HD	Surrogate split
Autoassociative Neural Network	

III. IMPUTATION TECHNIQUES

a. Mean/mode

The easiest way to impute the MVs is to replace each missing value with the mean of the observed values for that variable according to [5]. The mean of the attribute is computed using the non-missing values and is used it to impute the missing values of that attribute.

b. K-Nearest Neighbor

K-Nearest Neighbor is a pre-replace method that replaces the missingness before the data mining process as presented in [6]. It classifies the data into groups and then it replaces the missing values with the corresponding value from the nearest-neighbor. The nearest-neighbor is the closest value based on the Euclidean distance [7]. The missing values are imputed considering a given number of instances that are mostly similar to the instance of interest. The similarity of two instances is determined using Euclidean distance.

c. *Expectation Maximization*

Expectation Maximization provides estimates of the means and covariance matrices [5] that can be used to get consistent estimates of the parameters of interest. It is based on an expectation step and a maximization step, which are repeated several times until maximum likelihood estimates are obtained. This method requires a large sample size.

d. Hot-Deck Imputation

A missing value is filled with an observed value that is closer in terms of distance as in [8]. In other words, the Hot-Deck (HD) randomly selects an observed value from a pool of observations that matches based on the selected covariates. HD is typically implemented into two stages. In the first stage, the data are partitioned into clusters. In the second stage, each instance with missing data is associated with only one cluster. The complete cases in a cluster are used to fill in the missing values. This can be done by a correlation matrix that is used to determine the most highly correlated variables.

e. C5.0

C5.0 is a decision tree algorithm that was developed as an improved version of the well-known and widely used C4.5 classifier technique. Both C4.5 and C5.0 can handle the missingness during the classification, but C4.5 method requires more memory as indicted in [9]. The C5.0 would classify the data in lesser time with minimum memory usage and would improve the accuracy compared to C4.5. There are two steps in which C5.0 and C4.5 deal with MVs as given in [10]; 'splitting criterion evaluation' and 'instances distribution'. The C5.0 implements the 'splitting criterion evaluation' step by ignoring all instances whose value of attribute is missing. Then imputation is done by using either the mean or the mode of all the instances in the decision node. The 'instances distribution' is done by weighting the splitting criterion value using the proportion of missing values. Then the MVs are imputed with either the mean or the mode of all the instances in the decision node whose class attribute is the same as the instance of the attribute value that is being imputed.

IV. COMPARISON OF IMPUTATION TECHNIQUES

Mean/ mode, KNN, EM, HD, and C5.0 are compared based on the literature and is shown in Table 3. Mean / Mode imputations are appropriate in MCAR mechanisms while HD and EM are good with MAR mechanism. KNN and C5.0 will work with different mechanisms and datatypes, but KNN utilizes more cost (memory) and time. C5.0 is the only embedded method that can impute the MVs using Mean / Mode algorithm during the analysis.

TABLE 3. Comparison of five imputation techniques

Imputation	Datatypes	Mechanism	Method	Pro	Cons
Mean / Mode	Numerical Categorical	MCAR	Pre-replace	- simple and easy - Faster	- Does not produce better classifiers. - Correlation is negatively biased [5]. - The distribution of new values is an incorrect representation of the population values because the shape of the distribution is distorted by adding values equal to the mean [11].
KNN	Numerical Categorical Mixed	MCAR MAR NMAR	Pre-replace	- Multiple MVs are easily handled - Improves the accuracy of classification [12].	-The process takes a lot of time because it searches all the instances having most of similar dataset [6]. - It is difficult to choose the distance function and the number of neighbors [7]. - Loses its performance with complex and Blended pattern
EM	Numerical	MAR	Pre-replace	- Increased accuracy if model is correct [11]	The algorithm takes time to converge and is too complex [11].
HD	Categorical Mixed	MCAR MAR	Pre-replace	- It is suitable for big data [8].	- Not efficient for small size of sample data. - Problematic if no other case is closely related in all aspects of the data set [11]. - requires at least one categorical attribute to be implemented.
C5.0	Numerical Categorical Mixed	MCAR MAR NMAR	Embedded	- It is faster because it can impute the MV during classification - It has lower error rates on unseen cases [9].	Does not use all the attributes for classification.

V. SIMULATION AND THE RESULTS

This study presents a comparison of the performance of Mean / Mode, KNN, EM, HD and C5.0 methods on different datasets. The architecture for the comparison is shown in Fig. 2.

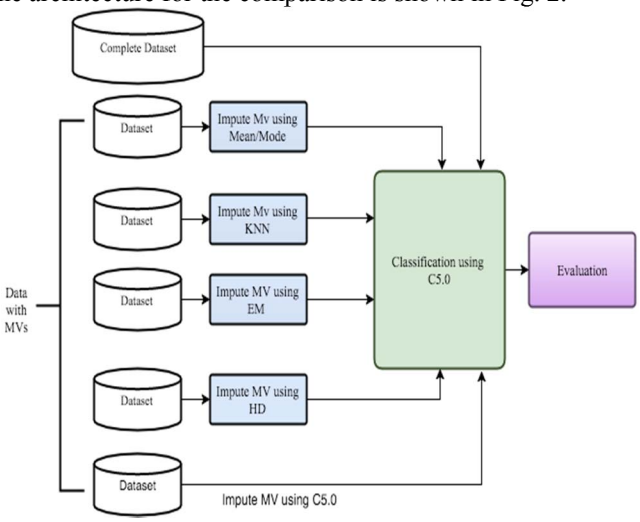

Fig. 2. Comparison Architecture

The data set is obtained from UCI Machine Learning Repository for simulation purposes [13]. The data sets used for the comparison are given in Table 4. Datatypes of Mixed, Categorical and Numerical are used. Categorical data consists of nominal and ordinal data. Numerical data consists of Real, Continuous and Discrete data. These data sets are used to create the missing data set by randomly removing some data from them. Blended patterns of missing values that consists of a combination of Medium and complex missing records, with different missing ratios up to 10%, have been used. Then imputation techniques are applied on these artificially created missing data sets and are used for classification. The efficiency of these techniques are compared either by comparing the classification rate error or Root Mean Squared Error. The results are summarized in Table 5 and 6. The Mean/Mode, KNN, and HD replaced the MVs during the pre-processing phase before the classification. C5.0 is the only embedded imputation method that has replaced the missingness during the classification. Simulation is done using R programming language.

TABLE 4. Data Sets

Datasets	Attributes	Instance	Datatypes	Missing ratio
Iris	4	150	Mixed	10%
Adult	13	30162	Categorical & continuous	20%
Glass	10	214	Numerical	15%
Wine	13	4898	Numerical	25%
Credit	16	690	Continuous & Nominal	15%

TABLE 5. Error rate in Classification for Categorical, Numerical and Mixed data sets

Classification	Complete Dataset	Imputed data by mean	Imputed data by HD	Imputed data by EM	Imputed data by KNN	Imputed data by C5.0
Iris	5/150= 0.0333	24/150= 0.16	**8/150= 0.0833**	Cannot be used because data is not numerical	14/150= 0.0933	6/150= 0.04 **Attribute usage** 91.76%
adult	5007/30162= 0.166	5258/30162= 0.1743	**5149/30162= 0.1707**	Cannot be used because data is not numerical	5215/30162= 0.1729	4541/30162= 0.15055 **attribute usage** 89%
Wine	1532 / 3428= 0.4469	1613/3428 = 0.4705	Cannot be used because data has not have categorical attributes	**1561/3428 = 0.4554**	1668/3428= 0.4866	1250/3428= 0.3646 **attribute usage** 84.58%
Credit approval	85/690= 0.12318	163/690= 0.2362	**119/690= 0.17246**	Cannot be used because data is not numerical	120/690= 0.17391	97/690= 0.14057 **attribute usage** 86%

TABLE 6. Root Mean Square Error (RMSE) for Numerical data sets

Data	Complete data	Imputed data by mean	Imputed data by EM	Imputed data by HD	Imputed data by KNN
Glass	0.1961161	1.322573	**0.359347**	0.4254544	0.4517767

HD offers good performance with minimum runtime based on the results. It performs better on larger data sets. HD has a slightly lower misclassification error rate compared to other imputations as shown in Table 5. So, HD is the better imputation technique for dealing with data that has Mixed and Categorical datatypes. HD and KNN show similar results on Adult data sets, but HD is more effective than KNN because the runtime overhead of KNN was five times longer than HD. C5.0 was not using all the attributes of the data set for classification; but it still provided a good classification accuracy on different datatypes. Mean imputation disturbs with the normality assumptions, as well as reduces association with other variables. EM performs better on the numerical data set and increases association with other variables. EM achieved more prospects than the other imputations on numerical attributes, but it is complex.

VI. CONCLUSION AND FUTURE WORK

Imputation techniques such as Mean\Mode, K-Nearest Neighbor, Hot-Deck, Expectation Maximization and C5.0 are used to impute artificially created missing data from different data sets of varying sizes. The performance of these techniques are compared based on the classification accuracy of original data and the imputed data. Identifying useful imputation technique will provide an accurate result in classification is presented in this research. 10% missingness for credit card data set and 25% missingness for Adult data set are used to demonstrate that these techniques will work better even though more missingness are present in the data.

The study found that: (1) HD imputation can improve the prediction accuracy to a statistically significant level on a large data sets; (2) C5.0 was not using all the attributes of the data set for classification; but it still provides a good classification accuracy on different datatypes; (3) both EM and KNN can be effective, but KNN consumed more time especially when dealing with large datasets. EM is too complex for implementation, but it performs better only on numerical attributes; (4) Mean imputation disturbs the normality assumptions, as well as reducing association with other variables. It can be used if less than 5% data are missing.

For future work, a combination of HD and EM with the classifier technique C5.0 might increase the accuracy of the classifications.

REFERNCES

[1] A. Saleem, K. H. Asif, A. Ali, S. M. Awan and M. A. Alghamdi, "Pre-processing Methods of Data Mining," *Utility and Cloud Computing (UCC), 2014 IEEE/ACM 7th International Conference on*, London, 2014

[2] B. Twala, M. Cartwright and M. Shepperd, "Comparison of various methods for handling incomplete data in software engineering databases", *International Symposium on Empirical Software Engineering, 2005.*

[3] M. Rahman and M. Islam, "A Decision Tree-based Missing Value Imputation Technique for Data Pre-processing", *Data Mining and Analytics 2011*.

[4] Y. Fujikawa and T. Ho, "Cluster-based Algorithms for Filling Missing Values", Lecture Notes in Computer Science, Vol. 2336, 2002, pp. 549-554.

[5] T. Schneider, "Analysis of incomplete climate data: Estimation of mean values and covariance matrices and imputation of missing values," Journal of Climate, vol. 14, pp. 853-871, 2001.

[6] O. Troyanskaya, M. Cantor, G. Sherlock, P. Brown, T. Hastie, R. Tibshirani, D. Botstein and R. Altman, "Missing value estimation methods for DNA microarrays", *Bioinformatics*, vol. 17, no. 6, pp. 520-525, 2001.

[7] V. Kumutha and S. Palaniammal, "An Enhanced Approach on Handling Missing Values Using Bagging k-NN Imputation", *International Conference on Computer Communication and Informatics*, Coimbatore, INDIA, 2013.

[8] R. Andridge and R. Little, "A Review of Hot Deck Imputation for Survey Non-response", *International Statistical Review*, vol. 78, no. 1, pp. 40-64, 2010.

[9] R. Pandya and J. Pandya, "C5. 0 Algorithm to Improved Decision Tree with Feature Selection and Reduced Error Pruning", *International Journal of Computer Applications*, vol. 117, no. 16, pp. 18-21, 2015.

[10] R. Barros, M. Basgalupp, A. de and A. Freitas, "A Hyper-Heuristic Evolutionary Algorithm for Automatically Designing Decision-Tree Algorithms", *Genetic and Evolutionary Computation Conference*, Philadelphia, 2012.

[11] S. Thirukumaran and A. Sumathi, "Missing value imputation techniques depth survey and an imputation Algorithm to improve the efficiency of imputation," *Advanced Computing (ICoAC), 2012 Fourth International Conference on*, Chennai, 2012

[12] V. Kumutha and S. Palaniammal, "An Enhanced Approach on Handling Missing Values Using

[13] "UCI Machine Learning Repository", *Archive.ics.uci.edu*, 2016. [Online]. Available: http://archive.ics.uci.edu/ml. [Accessed: 05- Dec- 2015].