A Novel Framework Using Two Layers of Missing Value Imputation

Md. Geaur Rahman
Md Zahidul Islam

Center for Research in Complex Systems (CRIcS), School of Computing and Mathematics
Charles Sturt University, Bathurst, NSW 2795, Australia.
Emails: {graham, zislam}@csu.edu.au

Abstract

In this study we present a novel framework that uses two layers/steps of imputation namely the Early-Imputation step and the Advanced-Imputation step. In the early imputation step we first impute the missing values (both numerical and categorical) using existing techniques. The main goal of this step is to carry out an initial imputation and thereby refine the records having missing values so that they can be used in the second layer of imputation through an existing technique called DMI. The original DMI ignores the records having missing values. Therefore, we argue that if a data set has a huge number of missing values then the imputation accuracy of DMI may suffer significantly since it ignores a huge number of records. In this study we present four versions of the framework and compare them with three existing techniques on two natural data sets that are publicly available. We use four evaluation criteria and two statistical significance analyses. Our experimental results indicate a clear superiority of the proposed framework over the existing techniques.

Keywords: Data pre-processing; data cleansing; missing value imputation; EM algorithm; Decision Trees

1 Introduction

The existence of missing values in data sets is a common problem. Due to various reasons including human errors and misunderstanding, equipment malfunctioning, faulty data transmission, propagation and measurements, collected data often have missing or incorrect values (Rahman & Islam 2011, Farhangfar et al. 2008). If the data are collected through a survey then often we can have missing values just because of the existence of some survey questions that a user may not feel comfortable to answer. For example, even if the identity of a participant is protected still s/he may not feel comfortable to answer the questions that are related to the sensitive disease (such as HIV positive) or financial condition (Young et al. 2011). Various studies show that the amount of missing values can be approximately 5% or more unless an organization takes extreme care during data collection (Zhu et al. 2004, Maletic & Marcus 2000).

We consider a data set \( D = \{R_1, R_2, \ldots, R_N\} \) and columns represent attributes \( A = \{A_1, A_2, \ldots, A_m\} \). The attributes can be either numerical (like 4 and 5.54) or categorical (like Canberra and Bathurst). Categorical values do not have any natural ordering in them. A numerical attribute has its domain \( A_j = [\text{low}, \text{up}] \) where \( \text{low} \) is the lower limit and \( \text{up} \) is the upper limit. A categorical attribute \( A_j = \{a_1, a_2, \ldots, a_k\} \) has a domain with \( k \) (i.e. \( |A_j| = k \)) different values. The size of a data set \( |D| \) or \( |R| \) is \( N \), which is the number of records. We consider that \( R_{ij} \) is the \( j \)th attribute value of the \( i \)th record. By “missing values” we mean that some of the \( R_{ij} \) values are missing/absent for various reasons. A missing value is denoted as \( R_{ij} = ? \). If a record \( R_i \in R \) contains one or more missing values then we consider that \( r_m \subset R_i (m < M) \) is a \( 1 \times m \) matrix having \( m \) number of attributes with missing values and \( r_a \subset R_i (a = M - m) \) is a \( 1 \times a \) vector having a number of attributes with available values.

The data sets, collected by the organizations, are typically used for various data mining processes. However, the performance of a data mining technique can significantly be disturbed due to the existence of missing or incorrect values in the data sets (Khoshgoftaar & Van Hulse 2005). Moreover, the presence of missing values in data sets can cause an inaccurate and non-sensible decision which may make the whole process of data collection and analysis useless for the users (Han & Kamber 2000).

Therefore, it is crucial to have an effective data pre-processing framework for dealing with missing values. One important data preprocessing task is the imputation of missing values. A number of techniques have been proposed for imputing missing values (Aydilek & Arslan 2013, Rahman & Islam 2011, Cheng et al. 2012, Schneider 2001, Zhu et al. 2011).

For dealing with missing values an early method just deletes the records having missing value/s (Derjani Bayeh & Smith 1999). However, the usability of the data sets for various statistical analyses can generally be reduced if the records, having missing values, are deleted from a small sized data set. Moreover, the results of the analysis can be misleading due to the use of a data set having insufficient number of records (Osborne & Overbay 2008).

Another early method uses the mean of all available values of an attribute for imputing missing values (Schafar & Graham 2002). However, it is shown that the mean imputation approach can often produce more misleading results (from data mining and statistical analysis) than the simple record deletion approach (Osborne & Overbay 2008).

For imputing missing value/s of a record an advanced technique called k-Nearest Neighbour Imputation (kNNI) (Batista & Monard 2003) first finds the (user-defined) \( k \)-most similar records (of the record having missing value/s). If the missing value belongs to a categorical attribute, the technique imputes the missing value by using the most frequent value, of the same attribute, within the \( k \)-Nearest Neighbour (\( k \)-NN) records. Otherwise for imputing a numerical value the technique makes use of the attribute mean value for the \( k \)-NN records. kNNI is a simple technique, and it performs better than...
the normal mean/mode imputation technique which calculates mean/mode from a whole data set, instead of the horizontal segment having k-NN records (Liu et al. 2010). However, for a large data set the technique can be found expensive since it finds the nearest neighbours of each record having missing value(s) (Batista & Monard 2003, Wu et al. 2008).

Instead of using k-NN records, a more advanced approach called EMI (Schneider 2001, Junninen et al. 2004) considers a whole data set for imputing numerical missing values. The technique utilizes the mean values of all numerical attributes, and the correlations between attributes having missing values and attributes having available values of a record in order to impute the numerical missing value(s) of the record. However, EMI does not work for a numerical attribute having the same value in all records. Moreover, it is not useful for imputing a record where all numerical attribute values are missing.

Another recent technique called DMI (Rahman & Islam 2011) identifies that the correlations of attributes within a leaf of a decision tree (i.e. a horizontal segment of a data set) are higher than the correlations of attributes within a whole data set. Therefore, the technique first finds a set of horizontal segments, where the records in each segment are considered to be similar to each other, from a data set using an existing decision tree algorithm such as C4.5 (Quinlan, 1996), and then applies an EMI algorithm within each segment for imputing numerical missing value(s). It is shown that the imputation accuracy and the improvement of data quality through DMI are higher than through EMI (Rahman & Islam 2013a, 2011).

However, the imputation accuracy of DMI declines for a data set having higher missing ratios (see Figure 2). This is perhaps due to the approach of considering only the complete records (i.e. the records having no missing values at all) while building a decision tree in order to find horizontal segments for the application of EMI. It does not take the records having any missing values into consideration for building the tree and therefore for the application of EMI. If a data set has a huge number of missing values then DMI ignores many records resulting in having only a small number of records for building the tree and applying the EMI.

We argue that the imputation accuracy of DMI, on a data set having a huge number of records with missing values, can be improved if it considers the records having missing values during the creation of decision trees. Therefore, we propose a framework which imputes missing values of a data set by the improved use of the records having missing values. The framework consists of two imputation steps namely 1. “Early-Imputation” and 2. “Advanced-Imputation”. In the Early-Imputation step we use an existing algorithm in order to initially impute the missing values, and in the Advanced-Imputation step we use DMI for final imputation in order to get better imputation accuracy.

In this study we use four difference versions of the proposed framework, namely “EMI”, “ESI”, “LDI” and “LSI” for imputing both numerical and categorical missing values. They use two layers of imputation: an early-imputation and an advanced-imputation. We also evaluate the performances of our proposed techniques by comparing them with three high quality existing techniques namely DMI, EMI and IBLLS in terms of four evaluation criteria namely co-efficient of determination (R²), Index of agreement (d₂), root mean squared error (RMSE) and mean absolute error (MAE) on two real data sets namely Autoimmune and Yeast that are publicly available such in the UCI machine learning repository (Frank & Asuncion 2010). For simulating missing values we use four missing patterns namely Simple, Medium, Complex and Blended, four missing ratios (1%, 3%, 5% and 10%), and two missing models namely Overall and Uniformly Distributed (UD). The initial experimental results indicate that our proposed technique EDI performs significantly better (based on 95% confidence interval analysis and statistical t-test analysis) than the existing techniques.

The organization of the paper is as follows. Section 2 presents a literature review. Our framework is presented in Section 3. Section 4 presents experimental results and Section 5 gives concluding remarks.

2 Background Study

For imputing missing values a number of techniques have been proposed recently (Aydilek & Arslan 2013, Olivetti de Franca et al. 2013, Dorri et al. 2012, Zzhang et al. 2011, Zhu et al. 2011, Liew et al. 2011, Liu et al. 2010, Twala & Phorah 2010, Farhangfar et al. 2007, Cai et al. 2006, Kim et al. 2005, Li et al. 2004, Rahman & Islam 2013b). Three existing techniques namely “Expectation-Maximisation based Imputation (EMI)” (Junninen et al. 2004), “Iterative Bi-Cluster based Local Least Square based Imputation (IBLLS)” (Cheng et al. 2012), and “Decision Tree and EMI based Imputation (DMI)” (Rahman & Islam 2011) are used in the experimentation of this study to compare them with our proposed techniques and thereby evaluate the performance of the proposed techniques. We also briefly discuss the EMI, IBLLS and DMI techniques here.

2.1 EMI

EMI (Schneider 2001, Junninen et al. 2004) uses correlations of the attributes having missing values and the attributes having available values of a whole data set for imputing numerical missing values of Rᵢ. Let Q = Σ⁻¹ m a be a matrix in which Σ a is the covariance matrix of available attribute values and Σ m is the covariance matrix of available and missing values. Also let µ a and µ m be the mean vectors of missing and available values, respectively. Based on the mean vectors and the correlations, the technique then imputes the missing value (r m) by using the following linear regression model (Schneider 2001).

\[ r_m = \mu_m + (r_a - \mu_a)Q + e \]

where \( e = [\mu_0 + HZT] \) is a residual error in which \( \mu_0 \) is a mean vector having zero value/s, H is a cholesky decomposition of the correlation matrix Q and Z is a vector having Gaussian random values (Muralidhar et al. 1999).

Similarly, EMI imputes all other records, of D,F, having missing values. Once the records, having missing values, are imputed the technique re-calculates the mean vectors (µ m and µ a) and the correlation matrix (Q). Using the re-calculated µ m, µ a and Q, EMI re-imputes the missing values of D,F. EMI repeats this process of imputation until the change of µ m, µ a and Q of two consecutive iterations is under a user-defined threshold.

2.2 IBLLS

Unlike EMI, IBLLS (Cheng et al. 2012) first finds k-nearest neighbour (k-NN) records of Rᵢ from D,F for imputing numerical missing values of Rᵢ. The value of k is determined automatically through applying a heuristic approach (Kim et al. 2005). Now let A̅ᵫₖₓₓ be the matrix that contains values from the k-NN records for the attributes having available values (rₜ) and B̅ₖₓₘ is the matrix that contains values from k-NN records for the attributes having missing values (rₘ). Using A and B, the
method calculates the correlation matrix $R_m \times a$ for the attributes having missing values and the attributes with available values.

Note that IBLLS imputes the missing values of $R_i$ by one by one. For imputing the $j$th missing value $r_{m,j}$ of $r_m$, the technique finds a set of $k$-NN records of $R_i$ from $D_F$ by considering the correlation matrix $Q$ with a weighted Euclidean distance measure (Cheng et al. 2012). It then partitions the $k$-NN records vertically by considering only the attributes having high correlation with $r_{m,j}$. IBLLS then finds the $r_{m,j}^{B_i}$ of $A^i$ and $B^j$ for $r_{m,j}$ from the data segment (Bi-Cluster) which is partitioned both horizontally and vertically. Finally, IBLLS imputes $r_{m,j}$ by using the following regression model (Cheng et al. 2012).

\[ r_{m,j} = r_{m}^{B_i} C_j \]  

where $C_j$ is the matrix that contains the regression coefficients that are obtained by minimising the following Least Square equation (Kim et al. 2005).

\[ \arg \min_{C_j} \| A^j C - B^j \|_2 \]  

The solution of Equation (3) can be obtained as follows.

\[ C_j = (A^j)^T B^j \]  

where $(A^j)^T$ is a pseudo inverse of $A^j$. Thus, the missing value $r_{m,j}$ can be imputed as

\[ r_{m,j}^{B_i} = r_{m}^{B_i} (A^j)^T B^j \]  

Similarly, IBLLS imputes all other missing values (if any) of $R_i$, and all other records having missing values. For each iteration (from iteration 2), it calculates the Normalised Root Mean Squared Error ($NRMSE$) (Cheng et al. 2012) by comparing the imputed values of the current iteration with imputed values of the previous iteration. Once the $NRMSE$ value is under a user-defined threshold the technique stops the process.

2.3 DMI

For imputing both numerical and categorical missing values DMI (Rahman & Islam 2011) uses a decision tree algorithm and an Expectation-Maximization algorithm (EMI) (Schneider 2001). Since EMI uses the correlations of attribute values of a data set, and generally the correlations of attribute values within a leaf are higher than the correlations of attribute values within the whole data set (Rahman & Islam 2011), the imputation accuracy of DMI is expected to be better through applying EMI for the records within a leaf rather than for the records within the whole data set.

DMI first divides $D_F$ into two sub data sets namely $D_C$ and $D_I$. The data set $D_C$ contains records without any missing values whereas $D_I$ contains records with missing values. If an attribute $A_j \in A$ has missing values, the technique then builds a decision tree $T_I$ from $D_C$ through applying a decision tree algorithm such as C4.5 algorithm (Quinlan 1996) by considering the attribute $A_j$ as the class attribute. For a numerical attribute $A_j$ DMI first generalizes the values of the attribute into $N_C$ categories, where $N_C$ is the squared root of the domain size of $A_j$. Note that the output of each tree is a set of logic rules where each logic rule represents a leaf. For each logic rule $L_{kj}$, DMI generates a sub data set $D_{kj}$ by assigning the records of $D_C$ that satisfy the logic rule $L_{kj}$, where $L_{kj}$ is the logic rule representing the $k$th leaf of the $j$th tree. DMI assigns each record $R_i \in D_I$ into a sub data set $D_{kj}$ corresponding to the logic rule $L_{kj}$ where the record $R_i$ falls in $L_{kj}$.

Once the records are assigned to the sub data sets, DMI imputes the missing values in the sub data sets one by one. If a missing value $R_{ij}$ is numerical then for imputing $R_{ij}$, DMI uses the EMI algorithm (Junninen et al. 2004, Schneider 2001) within the sub data sets where $R_{ij}$ belongs to. If $R_{ij}$ is categorical then it considers the majority class value of the sub data set as the imputed value. A majority class value is the class having the highest frequency, and a class value is the value of the attribute that is considered as the class attribute for building the tree (Quinlan 1993).

3 A Novel Imputation Framework

We present a novel imputation framework in four different versions namely “EDI”, “ESI”, “LDI” and “LSI”. The framework uses two main steps/layers: an early-imputation step and an advanced-imputation step. In the early-imputation step we impute the missing values by using an imputation technique such as EMI (Junninen et al. 2004, Schneider 2001) or IBLLS (Cheng et al. 2012). In advanced-imputation step, we then apply DMI (Rahman & Islam 2011) on the early-imputed data set. Before we discuss the framework in details we first introduce the basic concepts.

3.1 Basic Concept

DMI divides a given data set $D_F$, having missing values, into two sub data sets namely $D_C$ (having only records without missing values) and $D_I$ (having only records with missing values). It uses the C4.5 algorithm on $D_C$ in order to build the decision trees (DTs). DMI then applies the EMI technique on the records of each leaf of a tree. If the number of records within a leaf is big then the imputation accuracy is typically higher than when the number of records for the leaf is small. If a data set has high number of missing values then DMI creates a data set $D_{C}$ with a small number of records resulting in a small number of records in the leaves of the tree built from $D_C$. When EMI is applied on these small number of records, it typically produces a poor quality imputation. It was reported that DMI performs better in a data set having low number of missing values than a data set having high number of missing values (Rahman & Islam 2011). We understand that one reason of the low accuracy of DMI is the existence of large number of records with missing values in $D_I$.

In many cases, the size of $D_C$ can be very small compared to $D_F$. For example, Table 1 shows the number of records in $D_F$, $D_I$, and $D_C$ for the Autompg and Yeast data sets (Frank & Asuncion 2010) in terms of the 10% missing ratio and “Blended” missing pattern. Note that the data sets are publicly available in the UCI machine learning repository (Frank & Asuncion 2010). The details about the simulation of missing values (i.e. the 10% missing ratio and “Blended” missing pattern) are discussed in Section 4.1. For the Autompg data set, only 79 out of 392 records are used in $D_C$ (see Table 1).

Since the DTs are built on the small sized $D_C$, the knowledge extracted by the DTs may not be as useful as it could be if the DTs were built on the whole data set $D_F$. Therefore, DMI often fails to perform well for a data set having a large number of missing values. In order to explain this better, we consider here an example/toy data set
having 15 records and 4 attributes as shown in Table 2a. We then build a DT from the toy data set \( D_F \) by considering the attribute “Pos.” as a class attribute (see Figure 1(a)) where a leaf of the DT displays the information on the number of records belonging to a value of the class attribute. In the figure a leaf is represented by a rectangle and a node is represented by a circle. We then artificially create some missing values in \( D_F \) (see Table 2b). The data set \( D_F \) is then divided into \( D_C \) (Table 2c) and \( D_I \) (Table 2d).

Note that \( D_I \) only contains the records having missing values. Even if the records (such as “R4”, “R6”, “R10”, “R12” and “R14”) have only a single missing value, they are taken out of \( D_C \) and placed in \( D_I \). Thus, we often end up having a small number of records in \( D_C \). Therefore in this example, the leaves of the DT (see Figure 1(b)) built on \( D_C \) do not provide any information on the the value “L” of the class attribute “Pos.”. However, the DT (see Figure 1(a)) built on the full data set \( D_F \) provides information on the value “L”. Now if we classify the records “R4”, “R6”, “R10”, “R12” and “R14” of Table 2a by the DT (see Figure 1(a)) obtained from \( D_F \) then we get the correct class value “L”, whereas the records are misclassified as “Ap” if we use the DT (see Figure 1(b)) that is obtained from \( D_C \). We get a higher classification accuracy by the first DT than the second DT and therefore we expect a better imputation accuracy when we use the first DT than the second DT. That is, the removal of many useful records from \( D_C \) just because they have one/two missing values may not be a good idea.

We also analyse the impact of the number of missing values in a data set, in terms of imputation accuracy (\( RMSE \)). Four missing ratios namely 1%, 3%, 5%, and 10% are used. Two publicly available data sets are used as shown in Figure 2. The missing values are imputed by

![Figure 1: Decision trees built by using \( D_F \) and \( D_C \).](image)

Table 1: The number of records in \( D_F \), \( D_I \), and \( D_C \) for the Autompg and Yeast data sets in terms of 10% missing ratio and “Blended” missing pattern.

<table>
<thead>
<tr>
<th>Data set</th>
<th>Total Records</th>
<th>Number of records in ( D_I ) (having missing values)</th>
<th>Number of records in ( D_C ) (without missing values)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Autompg</td>
<td>392</td>
<td>313</td>
<td>79</td>
</tr>
<tr>
<td>Yeast</td>
<td>1484</td>
<td>674</td>
<td>810</td>
</tr>
</tbody>
</table>

DMI (Rahman & Islam 2011) and EMI (Schneider 2001). Here, x% missing ratios means x% of the total attribute values (i.e. \( N \) records \( \times \) \( M \) attributes) of a data set are missing. Figure 2 shows that DMI outperforms EMI in terms of \( RMSE \) (the lower the better) on both data sets namely Autompg (see Figure 2(a)) and Housing (see Figure 2(b)). However, the imputation accuracies of both DMI and EMI drop (on both data sets) with the increase of the missing ratios. DMI performs significantly better than EMI for a data set having small missing ratio (see Figure 2(a) and Figure 2(b)).

![Figure 2: Performance comparison on the Autompg and Housing data sets in terms of \( RMSE \) for different missing ratios.](image)

We argue that if a data set has a large number of records with missing values, the imputation accuracy of DMI can be improved by taking the records of \( D_I \) (in a refined form) into consideration instead of totally ignoring them. Therefore, we propose a novel framework that imputes missing values of \( D_F \) by first refining the records of \( D_I \) and then using them in \( D_C \). The framework therefore is a combination of two imputation steps/layers namely “Early-Imputation” and “Advanced-Imputation”. We now discuss the steps in details as follows.

### 3.2 Early-Imputation

The main goal of this step is to refine the records (that go to \( D_I \)) that have one or more missing values, so that the records can be considered in \( D_C \) in order to increase the
imputation accuracy. The records are refined by performing an early imputation for them.

In the early-imputation step, the mode value of the attribute \( A_j \) is imputed for each record \( R_i \). The framework then first initializes a missing matrix \( Y \) which is then used in the Advanced-Imputation step for indicating whether a value is originally missing or available. Each element \( y_{ij} \in Y \) (where \( 1 \leq i \leq N \) and \( 1 \leq j \leq |A| \)) contains either 0 or 1, which is calculated using Equation 6.

\[
y_{ij} = \begin{cases} 
1 & \text{if } R_{ij} \in D_F \text{ is missing} \\
0 & \text{if } R_{ij} \in D_F \text{ is available}
\end{cases}
\]  

(6)

The novel framework uses existing algorithms for imputing the missing values of \( D_F \) in this step. It first imputes the numerical missing values in \( D_F \) by using high quality techniques such as EMI (Junninen et al. 2004, Schneider 2001) and IBLLS (Cheng et al. 2012). If the exact leaf of \( Y_{ij} \) is missing. For SDMI, we identify the attribute \( A_j \) within the \( k \)-NN records is then considered as the imputed value. Based on the literature, the default value of \( k \) (for the \( k \)-NN) is set to 10 (Batista & Monard 2001, Bo et al. 2004, Troyanskaya et al. 2001).

### 3.3 Advanced-Imputation

The framework then applies DMI (Rahman & Islam 2011) on the early-imputed data set for the further improvement of the imputation quality for both the numerical and categorical missing values. It uses the matrix \( Y \) for identifying the missing values.

DMI builds a set of DTs \( T = \{ T_1, T_2, \ldots, T_M \} \) where each tree \( T_i \) considers an attribute \( A_i \) as the class attribute. In this study, we consider the following two options namely SDMI (Single DMI) and NDMI (Numerous DMI). In SDMI, we build a single DT (instead of a set of trees \( T \)) by considering the natural class attribute of \( D_F \) as the class attribute. Typically, every data set has a natural class attribute. For example, the natural class attribute of a patient data set can be “Diagnosis”.

In NDMI, we build a DT for each attribute of \( D_F \) as it is done in DMI. For numerical attribute we first generalizes the values of the attribute into \( N_C \) categories where \( N_C \) is the squared root of the domain size of the attribute. Therefore, for \( M \) attributes of \( D_F \) we have \( M \) decision trees in NDMI.

Following DMI, for both SDMI and NDMI we generate a sub data set for each logic rule of the DTs by assigning the records, of \( D_F \), which satisfy the conditions of the logic rule. The numerical missing values of each sub data set are then imputed by using the EMI algorithm. Let, a numerical attribute \( A_j \) has a missing value for the record \( R_i \), i.e. \( R_{ij} \) is missing. For SDMI, we identify the leaf where the record \( R_i \) falls in. For NDMI, we first find the leaf (of the tree \( T_j \)) where the record \( R_i \) falls in. EMI is then applied on all records belonging to the leaf for imputing \( R_{ij} \). If the exact leaf of a record \( R_i \) cannot be determined due to the existence of the missing values then we use the union of all possible leaves.

For the categorical imputation by SDMI, a missing value \( R_{ij} \) is imputed by the mode value of the attribute \( A_j \) within the records of the leaf where the record \( R_i \) falls in. On the other hand, in NDMI the missing value \( R_{ij} \) is imputed by the majority class value of the leaf (of the tree \( T_j \)) where the record \( R_i \) falls in. Note that if a record \( R_i \) has multiple categorical missing values then in NDMI multiple trees are used, one tree for one attribute.

### 3.4 Proposed Framework

The proposed framework uses two steps of imputation, the Early-Imputation step and the Advanced-Imputation step. In this study we use one of the two existing high quality imputation techniques namely EMI (Junninen et al. 2004, Schneider 2001) and IBLLS (Cheng et al. 2012) in the Early-Imputation step. For the Advanced-Imputation step we use either SDMI or NDMI.

Therefore, in this study we compare four versions of the framework namely EDI, ESI, LDI and LSI. EDI is the combination of EMI and NDMI, ESI is the combination of EMI and SDMI, LDI is the combination of IBLLS and NDMI, and LSI is the combination of IBLLS and SDMI. We now compare the performances of the techniques in the following section.

### 4 Experimental Results and Discussion

We implement our novel framework in four different versions namely EDI, ESI, LDI, and LSI, and three other high quality existing techniques namely DMI (Rahman & Islam 2011), EMI (Junninen et al. 2004, Schneider 2001) and IBLLS (Cheng et al. 2012). It was shown in the literature that the imputation accuracies of the existing techniques are better than many other techniques including Bayesian principal component analysis (BPCA) (Oba et al. 2003), LLSI (Kim et al. 2005), and ILSSI (Cai et al. 2006).

#### 4.1 Data Set

We apply the techniques on two real data sets, namely the Yeast data set and the Autompg data set that are available from UCI Machine Learning Repository (Frank & Asuncion 2010). A brief description of the data sets is presented in Table 3.

<table>
<thead>
<tr>
<th>Data set</th>
<th>Records</th>
<th>Num. attr.</th>
<th>Cat. attr</th>
<th>Missing</th>
<th>Pure Rec.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yeast</td>
<td>1484</td>
<td>8</td>
<td>1</td>
<td>No</td>
<td>1484</td>
</tr>
<tr>
<td>Autompg</td>
<td>398</td>
<td>5</td>
<td>3</td>
<td>Yes</td>
<td>392</td>
</tr>
</tbody>
</table>

The Yeast data set has 1484 records, 8 numerical and 1 categorical attributes. There are no records having natural missing values in the data set. So, we use all 1484 records as a pure data set in our experiment. On the other hand, the Autompg data set has 398 records, 5 numerical and 3 categorical attributes. There are a number of records having missing values. We first remove all records having missing values. Therefore, we get a pure data set having 392 records without any missing values. In our experiments we use the pure data sets. Note that for the experimentation purpose we artificially create missing values in the pure data sets, the actual value of which is known to us.

#### 4.2 Simulation of Missing Values

For experimentation, we artificially create missing values in the pure data set. We then impute the missing values by different techniques. Since we know the actual values of the artificially created missing value, we can evaluate the performances of the techniques by comparing the actual and the imputed values.

Generally the performances of an imputation technique depends on both the amount and the type/pattern of missing values (Junninen et al. 2004, Rubin 1976, Schneider 2001). Therefore, in this experiment we use various patterns of missing values such as simple, medium, complex and blended as discussed below.
A simple pattern permits a record to have at most one missing value, whereas a medium pattern permits a record to have minimum 2 attributes with missing values and maximum 50% of the attributes with missing values. Like wise, a complex pattern permits a record to have minimum 50% and maximum 80% attributes with missing values. A blended pattern allows a mixture of records from all three other patterns. We consider that a blended pattern simulates a natural scenario where we may expect a combination of all three missing patterns. In a blended pattern we have 25%, 50% and 25% records having missing values in the simple pattern, medium pattern and complex pattern, respectively (Junninen et al. 2004, Rahman & Islam 2011, Rahman et al. 2012).

For each missing pattern, we use four missing ratios: 1%, 3%, 5% and 10% where x% missing ratios means x% of the total attribute values (i.e. N records × M attributes) of a data set are missing. Note that for 10% missing ratios and simple pattern, the expected total number of records to have missing values may exceed the total records in some data sets. Therefore, in the simple missing pattern we consider 6% missing ratios (rather than 10% missing ratios) for all data sets.

In addition, two types of missing models namely Over all and Uniformly Distributed (UD) are considered. In the overall model, the attributes may not have equal number of missing values, and in the worst case scenario a single attribute can have all missing values. However, in the UD model the missing values are distributed equally in each attribute.

Based on the missing ratios, missing models, and missing patterns, we have a total of 32 missing combinations (id 1, 2, 3, . . . , 32). For each combination, we generate 10 data sets with missing values. For example, for the combination having “1%” missing values, “overall” missing model, and “simple” missing pattern (id 1, see Table 4) we generate 10 data sets with missing values. Therefore, we generate all together 320 data sets for each natural data set namely Yeast and Autompg.

4.3 Evaluation Criteria

We evaluate the imputation accuracies (performances) of the proposed and existing techniques in terms of four well known evaluation criteria namely co-efficient of determination (\(R^2\)), Index of agreement (\(d_2\)), root mean squared error (\(RMSE\)) and mean absolute error (\(MAE\)).

We now define the evaluation criteria briefly. Let \(L\) be the number of artificially created missing values, \(O_i\) (1 ≤ i ≤ L) be the actual value of the ith artificially created missing value, and \(P_i\) be the imputed value of the ith missing value. Also let \(\bar{O}\) and \(\bar{P}\) be the averages of the actual values \(O_i\); ∀i ∈ L and the imputed values \(P_i\); ∀i ∈ L, respectively. Let \(\sigma_O\) and \(\sigma_P\) be the standard deviation of the actual values and the imputed values, respectively.

The coefficient of determination (\(R^2\)) (Junninen et al. 2004) describes the imputation accuracy based on the degree of correlation between actual and imputed values. The output of \(R^2\) is a value between 0 and 1, where 1 indicates a perfect imputation.

\[
R^2 = \left[1 - \frac{\sum_{i=1}^{L} (P_i - O_i)^2}{\sum_{i=1}^{L} (|P_i - \bar{O}| + |O_i - \bar{O}|)^2}\right]^{2} 
\]

(7)

The index of agreement (\(d_2\)) (Willmott 1982) evaluates the degree of agreement between actual and imputed values. The output of \(d_2\) is also a value between 0 and 1. Similar to \(R^2\), a higher value of \(d_2\) indicates a better fit. It is calculated as follows:

\[
d = 1 - \frac{\sum_{i=1}^{L} (P_i - O_i)g}{\sum_{i=1}^{L} (|P_i - \bar{O}| + |O_i - \bar{O}|)^g} 
\]

(8)

where the value of \(g\) can be either 1 or 2. We use a value 2 for the index \(d_2\) throughout this experiment.

Finally, the mean absolute error (\(MAE\)) (Junninen et al. 2004) measures the average difference between the actual and the imputed values. Its value ranges from 0 to ∞, where a lower value indicates a better matching.

\[
MAE = \frac{1}{L} \sum_{i=1}^{L} |P_i - O_i| 
\]

(9)

4.4 Experimental Result Analysis on the Autompg and Yeast Data Sets

We present the performance of EDI, ESI, LDI, LSI, DMI, EMI, and IBLLS based on \(R^2\), \(d_2\), \(RMSE\), and \(MAE\) for 32 missing combinations on the Autompg data set in Table 4. The table shows the average values of performance indicators on 10 data sets with missing values for each missing combination. For example, we have 10 data sets with missing values for the combination (\(id = 1\)) of “1%” missing ratio, “Overall” missing model and “Simple” missing pattern. The average of \(R^2\) for the data sets having \(id = 1\) is 0.908 for EDI as reported in Table 4.

Bold values in the table indicate the best results among the seven techniques. Our proposed techniques (EDI, ESI, LDI and LSI) perform better than the existing techniques namely DMI, EMI and IBLLS in terms of all evaluation criteria. Moreover, the last row of the table, we present a score of each technique for each evaluation criteria, where a score “S” indicates that a technique performs the best among all the techniques in “S” (out of 32) number of missing combinations. The table shows that EDI outperforms all other techniques, the technique scores 31 (out of 32) for all evaluation criteria.

Similarly, Table 5 demonstrates the performance of the techniques in terms of all evaluation criteria for 32 missing combinations on the Yeast data set. The last row of the table indicates for all evaluation criteria EDI outperforms other techniques. EDI scores 32 (out of 32) for all evaluation criteria except \(MAE\) where EDI scores 29 and DMI scores 3.

4.5 Statistical Significance Analysis for All Data Sets

We present several statistical significance analysis on the Autompg and Yeast data sets. Since EDI (among the techniques we proposed in this paper) outperforms three other existing techniques, we present the statistical significance analysis of EDI, DMI, EMI and IBLLS as follows.

Figure 3 demonstrates 95% confidence interval analysis of EDI with DMI, EMI and IBLLS in terms of \(R^2\) (Figure 3(a)), \(d_2\) (Figure 3(b)), \(RMSE\) (Figure 3(c)), and \(MAE\) (Figure 3(d)) for all 32 missing combinations on the Autompg data set. It is clear from the figures that EDI performs better (i.e. better average value and no overlap
Table 4: Performance of EDI, ESI, LDI, DSL, DMI, EMI, and EBLLS based on $R^2$, $d_2$, $RMSE$, and $MAE$ for 32 missing combinations on AutoPromp data set

<table>
<thead>
<tr>
<th>Missing combination</th>
<th>$R^2$ (Higher value is better)</th>
<th>$d_2$ (Higher value is better)</th>
<th>$RMSE$ (Lower value is better)</th>
<th>$MAE$ (Lower value is better)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Overall</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Simple</td>
<td>0.790</td>
<td>0.789</td>
<td>0.756</td>
<td>0.762</td>
</tr>
<tr>
<td>Overall</td>
<td>0.790</td>
<td>0.789</td>
<td>0.756</td>
<td>0.762</td>
</tr>
<tr>
<td>Simple</td>
<td>0.790</td>
<td>0.789</td>
<td>0.756</td>
<td>0.762</td>
</tr>
<tr>
<td>Overall</td>
<td>0.790</td>
<td>0.789</td>
<td>0.756</td>
<td>0.762</td>
</tr>
<tr>
<td>Simple</td>
<td>0.790</td>
<td>0.789</td>
<td>0.756</td>
<td>0.762</td>
</tr>
<tr>
<td>Overall</td>
<td>0.790</td>
<td>0.789</td>
<td>0.756</td>
<td>0.762</td>
</tr>
<tr>
<td>Simple</td>
<td>0.790</td>
<td>0.789</td>
<td>0.756</td>
<td>0.762</td>
</tr>
<tr>
<td>Overall</td>
<td>0.790</td>
<td>0.789</td>
<td>0.756</td>
<td>0.762</td>
</tr>
</tbody>
</table>

Table 5: Performance of EDI, ESI, LDI, DSL, DMI, EMI, and EBLLS based on $R^2$, $d_2$, $RMSE$, and $MAE$ for 32 missing combinations on Yeast data set

<table>
<thead>
<tr>
<th>Missing combination</th>
<th>$R^2$ (Higher value is better)</th>
<th>$d_2$ (Higher value is better)</th>
<th>$RMSE$ (Lower value is better)</th>
<th>$MAE$ (Lower value is better)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Overall</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Simple</td>
<td>0.790</td>
<td>0.789</td>
<td>0.756</td>
<td>0.762</td>
</tr>
<tr>
<td>Overall</td>
<td>0.790</td>
<td>0.789</td>
<td>0.756</td>
<td>0.762</td>
</tr>
<tr>
<td>Simple</td>
<td>0.790</td>
<td>0.789</td>
<td>0.756</td>
<td>0.762</td>
</tr>
<tr>
<td>Overall</td>
<td>0.790</td>
<td>0.789</td>
<td>0.756</td>
<td>0.762</td>
</tr>
<tr>
<td>Simple</td>
<td>0.790</td>
<td>0.789</td>
<td>0.756</td>
<td>0.762</td>
</tr>
<tr>
<td>Overall</td>
<td>0.790</td>
<td>0.789</td>
<td>0.756</td>
<td>0.762</td>
</tr>
<tr>
<td>Simple</td>
<td>0.790</td>
<td>0.789</td>
<td>0.756</td>
<td>0.762</td>
</tr>
<tr>
<td>Overall</td>
<td>0.790</td>
<td>0.789</td>
<td>0.756</td>
<td>0.762</td>
</tr>
</tbody>
</table>

Table 6: Performance of EDI, ESI, LDI, DSL, DMI, EMI, and EBLLS based on $R^2$, $d_2$, $RMSE$, and $MAE$ for 32 missing combinations on CIFAR-10 data set

<table>
<thead>
<tr>
<th>Missing combination</th>
<th>$R^2$ (Higher value is better)</th>
<th>$d_2$ (Higher value is better)</th>
<th>$RMSE$ (Lower value is better)</th>
<th>$MAE$ (Lower value is better)</th>
</tr>
</thead>
</table>
except for those marked by the circles.

We can see from the figures that IBLLS in general performs worse for a high missing ratios, whereas EDI maintains almost the same performance even for a high missing ratios.

In Figure 5 we present a statistical significance analysis using t-test for all 32 missing combinations of all data sets. The figure demonstrates a considerably better performance of EDI over other techniques at $p = 0.05$ based on all evaluation criteria for the Autompg and Yeast data sets. The t-values are higher than the t(ref) values. We get the values of t (ref) using Student’s t distribution table (Distribution table: Students t [online available: http://www.statsoft.com/textbook/distribution-tables/]).

![Figure 5: t-test analysis on Autompg, and Yeast data sets](image)

**4.6 Aggregated Performance Analysis for All Data Sets**

We now present aggregated performances of all techniques in terms of missing ratios, missing models, and missing patterns in Figure 6 for the Autompg data set. The figures demonstrate that EDI performs better (i.e. higher average imputation accuracy) than other techniques for all missing ratios, for all missing models, and for all missing patterns in terms of $R^2$ (Figure 6(a)), $d_2$ (Figure 6(b)), $RMSE$ (Figure 6(c)), and $MAE$ (Figure 6(d)).

Similarly for the Yeast data set we present the aggregated imputation accuracies in Figure 7. The figures demonstrate that EDI performs better (i.e. higher average imputation accuracy) than other techniques for most of the missing ratios, for all missing models, and for all missing patterns in terms of all evaluation criteria except $MAE$ (Figure 7(d)) where DMI also achieves the same accuracy for 5% and 10% missing ratios, and simple and medium missing patterns.

We also present overall performances (i.e. the average value of the accuracies for the 320 data sets) based on $R^2$, $d_2$, $RMSE$, and $MAE$ for the Autompg and the Yeast data set in Table 6. For the data sets the overall imputation accuracy of EDI is higher than the overall imputation accuracy of other techniques. For the Autompg data set the overall imputation accuracies of EDI, ESI, LDI and LSI, in terms of $R^2$, $d_2$, $RMSE$, and $MAE$, are higher than of DMLEMI and IBLLS. Similarly we get better imputation accuracy for EDI on Yeast data in terms of $R^2$, $d_2$, $RMSE$, and $MAE$.

Figure 8 presents the percentage of the combinations (out of the total 64 combinations for the two data sets) where the techniques perform the best. For example, EDI performs the best in 98.44% combinations in terms of $R^2$ (Figure 8(a)), $d_2$ (Figure 8(b)) and $RMSE$ (Figure 8(c)), and in 93.75% combinations in terms of $MAE$ (Figure 8(d)).

![Figure 8: Percentage of combinations for all data sets, where a technique achieves the best result.](image)

4.7 Experimentation on the Imputation of the Categorical Missing Values

Unlike EMI and IBLLS, the proposed techniques (i.e. EDI,ESI, LDI, LSI) can impute categorical missing values in addition to numerical missing values. Therefore, we now compare the performances of the techniques with only DMI, for the imputation of categorical values. Fig. 9 shows that EDI achieves lower $RMSE$ and $MAE$ values than DMI for all data sets. For each data set $RMSE$ and $MAE$ values are computed using all 32 combinations. Note that for $RMSE$ and $MAE$ a lower value indicates a better imputation.

4.8 Execution Time Complexity Analysis

We now present the average execution time (in milliseconds) for 320 data sets (32 combinations × 10 data sets per combination) with missing values for each real data set in Table 7. We carry out the experiments on a machine
Figure 3: 95% confidence interval analysis on Autompg data set in terms of 32 missing combinations.

Figure 4: 95% confidence interval analysis on Yeast data set in terms of 32 missing combinations.
Figure 6: Aggregated performance on Autompg data set in terms of Missing Ratios, Missing Models, and Missing Patterns.

Figure 7: Aggregated performance on Yeast data set in terms of Missing Ratios, Missing Models, and Missing Patterns.
having configuration $4 \times 8$ core Intel E7-8837 Xeon processors, 256 GB RAM. EDI takes less time than IBLLS, whereas it takes slightly more time than EMI to pay the cost of a significantly better quality imputation.

Table 7: Average execution time (in milliseconds) of different techniques on the two data sets.

<table>
<thead>
<tr>
<th>Data set</th>
<th>EDI</th>
<th>ESI</th>
<th>LDI</th>
<th>LSI</th>
<th>DMI</th>
<th>EMI</th>
<th>IBLLS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Autoppg</td>
<td>4.191</td>
<td>782</td>
<td>12.264</td>
<td>8.959</td>
<td>2.215</td>
<td>18</td>
<td>8.881</td>
</tr>
<tr>
<td>Yeast</td>
<td>6.913</td>
<td>4.164</td>
<td>182.670</td>
<td>193.147</td>
<td>3.024</td>
<td>92</td>
<td>173.209</td>
</tr>
</tbody>
</table>

5 Conclusion

In this paper we present a novel imputation framework that uses two layers of imputation, an Early-imputation and an Advanced-imputation step. We argue that an early imputation before the actual one should improve the imputation accuracy significantly. Especially for an existing technique called DMI the two layered approach of imputation should improve the accuracy significantly. We point out that if a big number of records have missing values then DMI may suffer from low accuracy. In this study we experimented four versions of the proposed framework on two data sets and four evaluation criteria. The experimental results show that the version called EDI (which is the combination of EMI and NDMI) gives the best results. The superiority of EDI over ESI, and LDI over LSI supports our belief on the supremacy of NDMI over SDMI. EDI performs better than ESI in 63 out of 64 combinations, and LDI outperforms LSI in 50 out of 64 combinations for the two data sets (see Table 4 and Table 5). Additionally, the superiority of EDI and the other three versions of the proposed framework over the three existing techniques justifies our argument in favour of the two layered approach. EDI outperforms all three existing techniques in 63 out of 64 total combinations for the two data sets. Our future research plans include the further development of the framework in order to reduce its time complexity.

References


URL: http://www.statsoft.com/textbook/distribution-tables/


