Analysis and Comparison of Different Missing Value Imputation (MVI) Techniques

Ehsan Ul Haque

CSE 5717: Big Data Analytics Project Report

Abstract

In this project, we reviewed the nuances of missing value imputation (MVI) as part of the data cleanup process. Cleaning up data is a significant phase of data mining, as the quality of the data and the accuracy of the mining results depends on how well issues like inconsistencies, impurities, and incompleteness are handled. We compared popular MVI techniques across different datasets for two different missing rate scenarios and in a multivariate missing data setup. Our finding indicates mixed results in terms of performance of the MVI techniques. We also noticed a distinction between the best measures between the two missing rate scenarios.

1. Introduction

One of the key parts of any data mining or big data analytics tasks is the refinement of raw data to construct the usable, sort of "smart" data that contains the necessary characteristics required for the mining purposes. This process is referred to as data cleanup and is often done as part of data pre-processing phase of data mining. Cleanup is always necessary because the raw data often can be noisy, or, can have outliers, or inconsistencies, which contribute to inaccuracy in the data. Among all these, availability of missing values in the raw dataset is one of the most prevalent sources of data inconsistencies, and it is detrimental to the quality of the data to be mined. In fact, data scientists are often significantly concerned about how to deal with missing data (Brown & Kros, 2003). There can be several sources that contribute to the missingness of the data. For example, data can get omitted at the data entry process due to a human or mechanical error, can correspond to transmission failure due to network issue, database failures, inconsistency across different data sources, or can be due to not being collected or simply refused to be provided (e.g., sensitive information like gross income etc.). No matter what the reason is, data scientists pay significant time to plan for and resolve the incompleteness to ensure the quality of the data. One trivial approach towards solving this can be just ignore the cases with missing data in it. While this approach is viable for some cases when missing rate is low, however, in most cases simply just ignoring or removing the data with missing information is not an option to consider. There are several reasons for this – First, If the missing rates are high in the raw data, pairwise removal of the incomplete cases may affect model strength and lead to misleading conclusion. Second, Missing data introduces ambiguity in the data analysis process, because traditional statistical and

machine learning algorithms are not often robust for missing values or other ambiguities in the data (Jadhav, Pramod, & Ramanathan, 2019). Third, often a small amount of missing data can contain important information that cannot be ignored. For example, e-commerce transaction records for an item of interest, even when purchaser's personal information, such as gender, or date of birth is missing. Other than removing the data with missing information, the most common way of handling missing data is filling up the missing values by some means, otherwise known as missing value imputations (MVI). As the name suggests, missing value imputation is the strategy where researchers come up with the best possible value to fill up the missing data. The approaches of missing value imputation can vary widely, from simple null or central value imputation to complex prediction models to predict the missing values based on observed data. As part of this report, we look at different MVI techniques from the literature and understand how these techniques work from the theoretical perspective. We also investigate some of the most common MVI techniques and see how they compare with each other.

2. Overview of Data Cleanup as part of Data Preprocessing

The fundamental challenge for any big data analytics or data mining task is to ensure data quality. In this world of information, data is everywhere. Everyday millions of bytes of data are flowing over the internet. However, the availability of data does not directly correlate the information it contains. Before any data mining task, raw data must be processed and shaped into quality data. Quality of the data is crucial to produce accurate outcome after the data mining process. The purpose of data preprocessing is to analysis and control the quality of the raw data. Accuracy, consistency, and completeness are the characteristics of quality data, whereas raw data tends to be inconsistent, inaccurate, and incomplete (Han, Pei, & Kamber, 2011). To convert raw data into quality data, different data cleanup tasks are performed. Several tasks are done as part of data cleanup process, such as cleaning up noisy data, resolving outliers, handling missing values, and removing other inconsistencies. Here are the components that lead to data inconsistency and incompleteness and needs to be resolved as part of data cleanup.

2.1. Noise: Noise referred to some mix-up of arbitrary values with the original value. For attributes noisy value indicate deviations from original values. There can be several factors that contribute to noisy data. For example, noise can be associated with attribute values due to faulty data collection systems, data transmission errors, instrument errors, data entry issues, or for technology limitations. Approaches like, binning or statistical procedures, such as regression, clustering etc. can be used to smooth noisy data.

2.2. Outliers: Outliers often refers to as extreme deviations from the data distributions. The factors that contribute to noisy data, can also contribute to the presence of outliers in the raw data. Several statistical techniques are available to detect outliers that is usually incorporated as part of the data cleanup process to detect and resolve the outliers present in the raw data.

2.3. Data Inconsistency: Aside from noise or outliers, there can be discrepancy between records that cause data inconsistency. For example, for a record if the values for the attributes age and date of birth can show discrepancy if the deducted age from the birthdate does not match the actual age. For some attribute that indicate user rating, data discrepancy may occur if the rating systems differ across records. For example, in some records the rating was given in a numeric scale whereas the other records show ratings in alphabetic letter scale. There can also be duplicate records in the data (possibly caused due to merging raw data from multiple sources) where some attributes indicate different values. For example, same person having different email addresses in two different records can indicate discrepancy. As part of the data cleanup process, these discrepancies need to be addressed to ensure consistency across the data, which further contribute to data quality.

2.4. Missing Values: Missing values contribute to incomplete data and reduces data quality. There is a difference between empty and missing value. Empty value indicates that there is no value that can be assigned to the corresponding attribute whereas missing value indicate that there exists some value for the variable but not available or not captured in the dataset. Thus, the association of missingness with a dataset is often attributed to some sort of human or mechanical error in the phase of data collection or data entry. Transmission of data over networks and limitations of the measurement instrument can also introduce missing values. When data is collected in an interview or survey setup, participants' refusal to answer can also introduce missingness. This happens particularly when the collected data can be considered as sensitive or meant not to be disclosed, for example, participant's social security number, personal contact information, gross income, etc.

3. Concepts and Terminologies regarding Missing Values

3.1. Missingness Mechanism

Proper planning to resolve missing value problem is often critical and needs consideration for why data is missing. Understanding the true reason for missing value both helps identifying the proper methods to be applied to resolve the problem as well as reduces the risk of adding bias to the analysis. Three mechanisms were proposed to explain the missingness in the data: Missing Completely at Random (MCAR), Missing at Random (MAR), and Missing Not at Random (MNAR) (Little & Rubin, 2019).

3.1.1. Missing Completely at Random (MCAR): MCAR assumes that the pattern of missing value is completely random and does not depend on any observed or unobserved portion of data. This is favorable considering that the missing information is unrelated to the observed values in any variable or unobserved portion of the dataset and the probability of missingness is same for all cases across the dataset. For example, if a weighting scale run out of battery, weight of some of

the elements will be missing. However, if there was no weight group ordering of the objects that were measured, the missing weights for can be considered MCAR. In such case, the probability of missing weight is same for each object that were measured. Though favorable, often this is unrealistic. Resolution measures considering MCAR in cases where there is some unnoticed cause for missingness can create bias.

3.1.2. Missing at Random (MAR): When the missing data depends only on the observed portion of the data, and not on the unobserved portion of the data, the missingness is considered as MAR. In MAR, the probability of missing is same across groups defined by the data already observed and not by the unobserved data. MAR is broader class than MCAR and much more realistic in real life scenarios because MAR assumes some correlations between the attribute containing missing value and other attributes on the dataset. For example, if a weighting machine does not work properly within objects beyond certain dimensions. If we have data on the dimension ranges and we can assume MCAR of the weights of the objects in the dimension ranges, the missingness can be considered MAR. Here, the probability of missingness is not random across all cases but depends on the dimension of the objects. Controlling for the dimension of the objects, the missingness is completely random.

3.1.3. Missing Not at Random (MNAR): When both MCAR and MAR cannot be assumed, it indicates the reason for missingness is not known to the researchers. In other words, when missingness depends on the observed data instead of observed data, it is said to be MNAR. In the case of MNAR, assumption of missing neither occurs on random nor associated with observed variables in the dataset. Thus, MNAR is least favorable while making decisions about the resolution approach of the missing values. Often when the variable is too sensitive and missingness depend on the item itself, it falls in MNAR category where the missingness did not happen at random. For example, if the weight machine functions less reliably overtime and produce missing value more due to being used for a long time, it may be hard to notice and thus falls into MNAR. If the weight machine does not produce wight beyond a certain weight limit of the objects, missing weights for the objects beyond the weight limit are MNAR as the missing values neither happened randomly nor can be assumed from the observed data on different variables. In most cases, MNAR is the least favorable mechanism and sometimes the only option to resolve missingness to completely change the measurement unit or collect more data that may explain the cause.

3.1.4. Why Missingness Mechanism is Crucial

To determine the best approach to resolve incompleteness issue, proper understanding, and considerations of the missingness mechanism is needed. Missingness mechanism helps identify the statistical tools that may best work for a certain attribute with missing data in it. Moreover,

missingness mechanism are important in deciding whether discarding data with missing values is a possible option and will not affect performance on the data mining tasks. MNAR may cause bias in the analysis, whereas existence of MCAR and MAR may lead to loss of strength in the statical analysis if not considered and tackled properly (Schafer & Graham, 2002). Moreover, detecting the proper type of missingness is the data is often challenging for the researchers, as there are limited tools available to understand the type of missingness. Some statistical techniques may be used to identify whether the missingness mechanism is MCAR or not MCAR, however, there are no tool that identify whether the missingness is MAR or MNAR (Little & Rubin, 2019).

3.2. Missing Rate:

Percentage of missing values, otherwise known as missing rate, is another factor that affect selection of proper missingness resolution technique. Missing rates indicate the ratio of missing values compared to the total observed values in the dataset. Typically, dataset with low missing rates are easy to be handled and often does not require complicated resolution approaches. When missing rate exceeds 15%, careful considerations are required to observe and deal with the missing data (Acurna & Rodriguez, 2004). However, this is not any hard and fast rule and may vary based on the type or genre of that data, mechanisms of missingness, and/or other factors.

3.3. Approaches towards Dealing with Missing Values

3.3.1. Ignoring Missing Values

The easiest approach towards dealing with missing values is to simply ignore the missing values. This is often feasible when the missing rate is very low. It has been showed that when missing rate is less than 10%, ignoring missing values in the analysis does not significantly affect the mining results (Strike, El Emam, & Madhavji, 2001). When comes to the consideration for ignoring missing values, there are typically two strategies that can be used, a) listwise/case deletion, b) pairwise deletion. In the listwise deletion approach, a record is deleted if any of its attribute contain missing value. Thus, the final dataset contains only records that do not have missing value in any of the attributes. The main disadvantage of this approach is that the size of the dataset can significantly decrease after the deletion process as the whole record is omitted even if only a single attribute has missing value in it. Thus, this approach may contribute to bias and loss of precision in the final mining result (Schafer & Graham, 2002). On the other hand, in pairwise deletion approach, the deletion occurs based on the variable of interest. The idea is, for the attribute of interest, the analysis often only requires data for the attribute itself and will not affect the result if other variable has missing data, which is not the attribute of interest. Thus, unlike listwise deletion approach, a record is deleted only if there is missing value in the variable of interest. The advantage of this approach over the listwise approach is that not too many records are deleted and a record with missing value is still considered for analysis as log as the missing value is not in the variable that is being looked at. The disadvantage of this approach is that for different variables, the dataset size may vary significantly and the result across variables may not be compared or correlated

(Schafer & Graham, 2002). To summarize, ignoring missing cases can be beneficial if the rate of missing data is low, but may introduce bias if the data that is deleted are crucial for the mining tasks.

3.3.2. Missing Value Imputation (MVI):

Missing value imputation leverages statistical techniques to produce plausible values for the missing data. If properly applied, MVI techniques generate more "intelligent" substitute that better suits the data and thus improves data quality for mining or big data analytics. There are several approaches how plausible value is generated which range from easy and straightforward substitution to complex, sophisticated, and resource hungry calculation. The contribution of different MVI approaches in producing precise and effective mining results thus varies significantly. Several factors also influence the best possible MVI technique to use, e.g., the type of the data, missing rate, missingness mechanisms, attribute types etc. Thus, a proper understanding of different imputation techniques is required to make the most informed decision and choose the right one for the specific mining purposes. Typically, MVI techniques works better than omitting the missing data altogether but requires much careful considerations from researchers' end. As such, choosing a wrong MVI technique without closely examining the missingness mechanisms and other factors may introduce bias in the result. Nonetheless, MVI is a powerful tool for tackling the incompleteness issue and is widely used by data scientists. Figure 1 shows an experimental design procedure of MVI in dealing with missing data (Lin & Tsai, 2020).

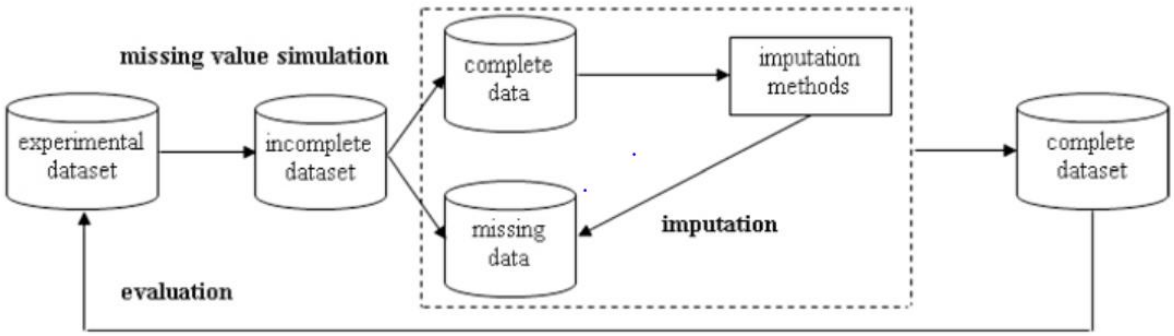

Figure 1: Experimental design procedure of MVI

4. Overview of Some of different MVI Methods

4.1. Constant Value Imputation: Probably the easiest MVI technique to replace the missing data with some constant, such as flagging as missing, assigning 0 etc. This method works better for

categorical variables with small range of possible values. However, for continuous variables this approach distorts the distribution of the variables and can lead to faulty results.

4.2. Mean Imputation: Mean substitution is one of the most used MVI techniques. It replaces the missing values with the sample mean of the observed values. Mean distribution works well for continuous variables and distribution is approximately normal. One of the disadvantages of mean distribution is that it does not consider the variability in the data and the correlative nature of the attributes. Let $x_1, x_2, \dots, x_n$ be the observed values of the dataset containing n variables. The estimation for the missing value, x_{miss} can be generated using the following formula:

$$\hat{x}_{miss} = \frac{\sum_{i=1}^{n} x_i}{n}$$

4.3. Median Imputation: Like the mean substitution approach, but the missing value is imputed with the estimation of median of the observed data. Median substitution works well when the distribution of the variable of interest in skewed in nature. Using the median imputation technique, estimation of missing value, x_{miss} is

$$\hat{x}_{miss} = \begin{cases} x_s & if \quad n\ is\ odd \\ \frac{x_s + x_{s+1}}{2} & if \quad n\ is\ even \end{cases}$$

Here, x_s is the middle value in the observed distribution.

4.4. Expectation Maximization (EM) Imputation: EM algorithm calculates the likelihood estimates for the incomplete data. This is an iterative approach that and cycles between two internal steps to come up with the best possible estimation. The first step attempts to estimate the missing data in the variables and known as E-step. In the next step, optimization of the parameters is done to best explain data, which is the reason the step is called as maximization step or M-step. EM algorithm follows the following iterative approach of handling missing data (Walczak & Massart, 2001).

1. Impute the missing values by their initial estimates
2. Estimate the parameters for the imputed dataset
3. Use the parameter estimations to optimize the initial estimates of the missing values
4. Re-estimate parameters after the imputation of missing values with the optimized estimates.

Until convergence of the parameter estimates, the algorithm alternates between step 3 and 4 in an iterative approach.

4.5. *k*-NN Imputation: The *k*-NN imputation leverages the *k*-nearest neighbor classification model to predict plausible values for the missing data. In general, the *k*-NN approach generates

the estimation of the substitute value by calculating similarity of the records in the dataset using some sort of distance function, such as Euclidean distance. Based on the similarity calculations, *k*-NN selects a set of *k* neighbors nearest to the target record and substitute the missing data of the given variable by averaging its neighbors' observed values. The algorithm for estimating a missing value x_{miss} using *k*-NN combines the following steps (Zhang, 2012):

1. Choose a value for *k* which is suitable. These number indicates the total neighbors generated for the calculations
2. For each variable *i* on the target observation (having the missing value), compute the distance between the missing observation and the observed values in the dataset using a distance function.

$$d(x_m, x_o) = \sqrt{\sum_{i=1}^{n} (x_{mi} - x_{oi})^2}$$

   Here, x_m and x_o indicates the missing record and observed records in the dataset and x_{mi} and x_{oi} indicate the value for the observations for variable *i*

3. Chose the *k* smallest distance records as the k nearest neighbors of the missing record x_m
4. Calculate the weights of the *k* nearest values and estimate the missing value as the weighted average of *k* nearest neighbors.

$$w_j = \frac{1}{d\left(x, x_j^n\right)^2}$$

$$W = \sum_{j=1}^{k} w_j$$

$$\hat{x}_{miss} = \frac{\sum_{j=1}^{k} w_j x_j^n}{W}$$

Here, $x_j^n, j = 1,2, \dots, k$ are the *k* nearest neighbors, w_j are the weight of the neighbors. k-NN works on both continuous and discrete variables. However, some disadvantage of the method is that the calculations can be time consuming when the number of records is very large and finding the optimal *k* value is difficult and is dependent on several factors (Zhang, 2012).

4.6. Multiple Imputation using Chained Equation (MICE) Imputation: In contrast to the single imputation approach, MICE uses many imputed values to substitute a missing value. Each set of generated imputed values are used to substitute the missing values, thus generating multiple imputed data sets. Further analysis and comparison are performed to combine the imputed datasets for the best results. Overall, there are three steps in the MICE mechanism.

- Generation: In an iterative approach, a total of *m* imputed datasets is generated
- Analyze: All *m* datasets are examined, and parameter of interest is estimated
- Combination: the best result is obtained by combining the *m* datasets

If $\hat{x}_j, j = 1,2, \dots, m$ are the estimated parameter of interest for each of the m possible imputed datasets, a combined estimation can be generated using following equations (Rubin, 1976).

$$\overline{x} = \frac{\sum_{j=1}^{m} \hat{x}_j}{m}$$

$$\overline{V} = \frac{\sum_{j=1}^{m} \hat{v}_j}{m}$$

$$B = \frac{\sum_{j=1}^{m} \left(\hat{x}_j - \overline{x}\right)^2}{m - 1}$$

Here, $\overline{x}$ is the average of the computed estimates for the m datasets, $\overline{V}$ is the variance across $\hat{x}$ (within imputed variance), B is the excess variance because of the missing values (between variance) (Murray, 2018). The benefit of MICE is that it can produces better unbiased results due to multiple iteration and combination, however, this also increases time complexity of the algorithms, specially when the data size is large.

5. Project Setup and Methodology

The aim of the project is to understand how different MVI techniques compare to each other when missing values are present on a multivariate setup. Prior work in the literature compared different MVI techniques in different setups (Mohammed, Zulkafli, Adam, Ali, & Baba, 2021) (Jadhav, Pramod, & Ramanathan, 2019), however, there are some limitations. First, some studies only compared the methods when missing values are observed in a single variable (univariate setup) (Mohammed, Zulkafli, Adam, Ali, & Baba, 2021). Second, even though EM computation was found one of the most popular MVI techniques in the literature (Lin & Tsai, 2020), however, the referenced works did not compare EM with the other techniques. Third, the missing rates were found to be limited up to 50% while comparing different methods, it would be interesting to see how the techniques perform in a more challenging setup where missing rate is pushed beyond 50%.

These considerations lead us to select the setup of the study. Particularly, we compared five different MVI techniques where missing values are present across all available variables (multivariate setup). We chose two level of missing rates to compare the methods: 5% and 55%. The MVI methods chosen for the study are: 1) Mean Imputation, 2) Median Imputation, 3) EM Imputation, 4) *k*-NN Imputation, and 5) MICE Imputations. For the *k*-NN algorithm, three different values of *k* are considered: 2, 5, and 10, the best performance among the three *k* values are reported. For the MICE technique, $m = 5$ was considered and best across the five imputed dataset performance was reported. Also, for MICE, the Predictive Mean Matching (PMM) algorithm was used. We used R and RStudio to perform the comparisons. Several R packages was as the implementation of the algorithms (Van Buuren & Groothuis-Oudshoorn, 2011) (Kowarik & Templ, 2016) (Afanador, Tran, Blanchet, & Baumgartner, 2016).

To measure the performance of the techniques, five datasets are used and obtained from UCI Machine Learning Repository, which was found as one of the most used dataset sources for the MVI performance analysis (Lin & Tsai, 2020). The datasets are collected from (Jadhav, Pramod, & Ramanathan, 2019). Table 1 shows the description of the datasets.

Table 1: Dataset description

Dataset	No of Instances (n)	No of Attributes (m)	Mean of Attribute Means
Wine Dataset	178	12	12.6537
Glass Identification Dataset	214	9	11.26585
Concrete Comprehensive Strength Dataset	1030	7	299.7391
Indian Liver Patient Dataset	583	8	62.28544
Seeds Dataset	210	8	6.284152

5.1. Performance Measure and Procedure

We used Root Mean Square Error (RMSE) as our performance metric, as it is one of the most representative and commonly used measure in the MVI research (Schmitt, Mandel, & Guedj, 2015). The procedure of measuring algorithm performance is: First, missing data were introduced randomly across all the variables in the dataset. Next, imputation dataset is generated using the MVI algorithms. Then, RMSE values are calculated by comparing imputed and original dataset for each of the attributes, using the following equation.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(original_i - imputed_i)^2}{n}}$$

Here, *n* is the number of instances in the dataset. This gives *m* RMSE values, for each of the attributes. Finally, the mean of the RMSE values is calculated and reported as the measure of performance. Lower mean RMSE value indicate better performance of the algorithm.

$$Mean\ RMSE = \frac{\sum_{i=1}^{m} RMSE_i}{m}$$

6. Result and Discussion

Our findings regarding the performance of the techniques for each of the datasets are presented in Table 2, 3, 4, 5, and 6. We noticed a mixed result in terms of performance across the techniques. For 5% missing imputation, all the three sophisticated methods worked well compared to mean or

median imputation as the variability across the data is considered in those techniques. *K*-NN and MICE were the best approaches in imputing missing values when 5% missing rate was considered. However, we noticed a somewhat average performance from MICE for some of the datasets (i.e., concrete dataset, liver patient dataset) where the RMSE values were comparable or even worse than the mean imputation. There can be several possible reasons for this. First, we chose a very low *m* value for our analysis of MICE due to limitations of CPU resources, which can contribute to the mixed results. Second, the PMM algorithm of MICE may not be the suitable one for some of the datasets. Compared to MICE, *k*-NN was more consistent in imputing missing values.

For 55% missing rate case, from the more sophisticated algorithms, EM worked well, however, for 5% missing rate case EM were mostly comparable to mean and median imputation. This indicates the effectiveness of the algorithm in more challenging conditions. When 55% missing rate was considered, *k*-NN performed ordinarily. Same goes for MICE except for one dataset. Interestingly, for the 55% missing rate case, mean imputation worked better than the sophisticated approaches for multiple datasets. This can be attributed to the very high missingness of the datasets where the missing information may have significantly affected the variability across the datasets. Although, a higher *m* value for MICE or a different *k* value for *k*-NN might have worked better, which we did not test. Finally, as we expected, performance of the techniques on the 5% missing rate scenarios were better than performance on the 55% missing rate scenarios.

Table 2: Performance of the methods on wine dataset

MVI Method	Mean RMSE for 5% MR	Mean RMSE for 55% MR
Mean	0.570407	1.50274
Median	0.5500541	1.516291
EM	0.5537835	**1.457162**
k-NN	**0.4261469**	1.573839
MICE	0.4760224	1.936227

Table 3: Performance of the methods on glass dataset

MVI Method	Mean RMSE for 5% MR	Mean RMSE for 55% MR
Mean	0.133839	**0.4886721**
Median	0.1294382	0.5141306
EM	0.123807	0.5213542
k-NN	0.090786	0.5137899
MICE	**0.04209586**	0.5362384

Table 4: Performance of the methods on concrete dataset

MVI Method	Mean RMSE for 5% MR	Mean RMSE for 55% MR
Mean	11.36785	**37.89193**
Median	12.67285	40.20339
EM	10.75702	39.23908
k-NN	**6.890098**	39.85688
MICE	11.30725	47.22975

Table 5: Performance of the methods on liver patient dataset

MVI Method	Mean RMSE for 5% MR	Mean RMSE for 55% MR
Mean	9.784007	74.09288
Median	9.829642	76.50923
EM	13.26972	**73.09774**
k-NN	**8.472461**	76.44595
MICE	10.13289	77.24802

Table 6: Performance of the methods on seeds dataset

MVI Method	Mean RMSE for 5% MR	Mean RMSE for 55% MR
Mean	0.2366261	0.7343151
Median	0.2433804	0.7332495
EM	0.06491941	0.4532404
k-NN	0.05616628	0.6174137
MICE	**0.05495519**	**0.39433083**

7. Limitations of the Study

There were several limitations that needs to be considered for the interpretation of the results. First, For MICE, we used a very low value of *m* for restrictions of CPU and RAM resources. A high value of *m* may have worked better. Similarly, for *k*-NN we did not look for the best possible *k* value for different datasets. There are several considerations to make for obtaining the optimal *k* values, which we did not do due to the time limitations. Second, we generated the missing data in a MCAR setup because of limited knowledge on the datasets. As such, we did not investigate how the algorithms would work if MAR or NMAR was present. Third, we did not code for the algorithm, instead we relied on several R packages for different techniques. Lastly, we did not perform any significance test on the difference across the techniques to identify whether the performance was significantly different from one another.

8. Conclusion

In this project, we analyzed different MVI techniques and compared them in an experiment setup. Findings show mixed results across the techniques which is expected. More sophisticated techniques performed well except for some of the cases when very high missing data was introduced. Overall, the project summarizes MVI as part of data cleaning and data preprocessing, which is an important part of the overall data mining and big data analytics process.

References

Acurna, E., & Rodriguez, C. (2004). The treatment of missing values and its effect in the classifier accuracy, classification, clustering, and data mining applications. *Proceedings of the meeting of the International Federation of Classification Societies (IFCS)*, (pp. 639–647).

Afanador, N. L., Tran, T., Blanchet, L., & Baumgartner, R. (2016). mvdalab-package 3. *Package 'mvdalab'*, 3.

Brown, M. L., & Kros, J. F. (2003). Data mining and the impact of missing data. *Industrial Management & Data Systems*.

Han, J., Pei, J., & Kamber, M. (2011). *Data mining: concepts and techniques.* Elsevier.

Jadhav, A., Pramod, D., & Ramanathan, K. (2019). Comparison of performance of data imputation methods for numeric dataset. *Applied Artificial Intelligence, 33*, 913–933.

Kowarik, A., & Templ, M. (2016). Imputation with the R Package VIM. *Journal of Statistical Software, 74*, 1–16.

Lin, W.-C., & Tsai, C.-F. (2020). Missing value imputation: a review and analysis of the literature (2006–2017). *Artificial Intelligence Review, 53*, 1487–1509.

Little, R. J., & Rubin, D. B. (2019). *Statistical analysis with missing data* (Vol. 793). John Wiley & Sons.

Mohammed, M. B., Zulkafli, H. S., Adam, M. B., Ali, N., & Baba, I. A. (2021). Comparison of five imputation methods in handling missing data in a continuous frequency table. *AIP Conference Proceedings*, *2355*, p. 040006.

Murray, J. S. (2018). Multiple imputation: a review of practical and theoretical findings. *Statistical Science, 33*, 142–159.

Rubin, D. B. (1976). Inference and missing data. *Biometrika, 63*, 581–592.

Schafer, J. L., & Graham, J. W. (2002). Missing data: our view of the state of the art. *Psychological methods, 7*, 147.

Schmitt, P., Mandel, J., & Guedj, M. (2015). A comparison of six methods for missing data imputation. *Journal of Biometrics & Biostatistics, 6*, 1.

Strike, K., El Emam, K., & Madhavji, N. (2001). Software cost estimation with incomplete data. *IEEE Transactions on Software Engineering, 27*, 890–908.

Van Buuren, S., & Groothuis-Oudshoorn, K. (2011). mice: Multivariate imputation by chained equations in R. *Journal of statistical software, 45*, 1–67.

Walczak, B., & Massart, D. L. (2001). Dealing with missing data: Part I. *Chemometrics and Intelligent Laboratory Systems, 58*, 15–27.

Zhang, S. (2012). Nearest neighbor selection for iteratively kNN imputation. *Journal of Systems and Software, 85*, 2541–2552.