REVIEW: EVOLUTION OF FRACTIONAL HOT DECK IMPUTATION FOR CURING INCOMPLETE DATA - FROM SMALL TO ULTRA LARGE SIZES

In Ho Cho[1], Jae-Kwang Kim[2], Yicheng Yang[3], Yonghyun Kwon[4], and Ashish Chapagain[3]

[1]Department of Civil, Construction, and Environmental Engineering (CCEE), Iowa State University (ISU), Ames, USA
[2]Department of Statistics (STAT), ISU, Ames, USA
[3]CCEE, ISU, Ames, USA
[4]STAT, ISU, Ames, USA

ABSTRACT

Machine learning (ML) advancements hinge upon data - the vital ingredient for training. Statistically-curing the missing data is called imputation, and there are many imputation theories and tools. Butthey often require difficult statistical and/or discipline-specific assumptions, lacking general tools capable of curing large data. Fractional hot deck imputation (FHDI) can cure data by filling nonresponses with observed values (thus, "hot-deck") without resorting to assumptions. The review paper summarizes how FHDI evolves to ultra data-oriented parallel version (UP-FHDI).Here, "ultra" data have concurrently large instances (big-n) and high dimensionality (big-p). The evolution is made possible with specialized parallelism and fast variance estimation technique. Validations with scientific and engineering data confirm that UP-FHDI can cure ultra data(p >10,000& n > 1M), and the cured data sets can improve the prediction accuracy of subsequent ML. The evolved FHDI will help promote reliable ML with "cured" big data.

KEYWORDS

Big Incomplete Data, Fractional Hot-Deck Imputation,Machine Learning, High-Dimensional Missing Data

1. INTRODUCTION

The data- and machine learning (ML)-driven research paradigm gradually became mainstream, offering ground-breaking solutions to daunting questions in broad science and engineering domains. The primary driving force is large data from various sensors, computational simulations, high-precision experiments, multifaceted surveys, and even social networks. However, large data suffer from missing values due to hardware breakdowns, software malfunctions, and human inconsistencies, which can result in severe accuracy deterioration in subsequent ML predictions and statistical inference.

Still, to fill in the missing values, naive methods are widely used - e.g., simple deletion of instances involving missing values or a replacement with the means of the observed values. It is well known, however, that such naive methods can result in considerable bias [1,3] and may mislead to incorrect statistical inferences and ML predictions [1].

David C. Wyld et al. (Eds): ICDIPV, CBIoT, ICAIT, WIMO, NC, CRYPIS, ITCSE, NLCA, CAIML -2023
pp. 191-204, 2023. CS & IT - CSCP 2023 DOI: 10.5121/csit.2023.131315

A robust statistical approach exists to handling incomplete data, the so-called imputation method, which replaces a missing value with statistically plausible values to create complete data. One of the most popular imputation methods is multiple imputation (MI) [2,6] which fills in missing data by creating separate data sets, accounting for variances within and between imputations. Many serial programs of variants of MI methods are already available in the global statistical platform*R* (e.g.,*mice* [7], *mi* [8], *AmelianII* [9], and *VIM* [10]). However, for broader engineering researchers, there exists a difficult hurdle for the routine use of the MI. MI cannot be easily applied to data sets obtained from a complex sampling design [11], and MI also requires the so-called "congeniality" and "self-efficiency" conditions [12,13]. Without satisfying these conditions, MI may cause substantial bias and incorrect inference.

High-performance computing technology has been harnessed for large-scale imputation. Researchers in various disciplines developed parallel imputation methods and software - e.g., [14, 15] for bioinformatics data, [16] for big enterprise data, [17,18] for epidemiology data, and so on. However, these HPC-based imputation methods and software depend heavily on domain-specific knowledge. Their capability to handle general and/or ultra-large incomplete data (concurrently big-n and big-p) is not confirmed.

Therefore, various existing approaches to handling missing data often require statistical and/or discipline-specific distributional assumptions of data, which are difficult for general users in broad science and engineering. Furthermore, existing theories and tools are not suitable for curing incomplete "ultra" data, i.e., large data with currently large instances (big-n) and high dimensionality (big-p).This review paper summarizes how the fractional hot-deck imputation (FHDI) has been evolving from a serial version (Section 2.1) to a parallel version for big-n or big-p data (Section 2.2) and even to the most advanced parallel version for curing ultra-large data (concurrently big-n and big-p, in Section 2.3).

2. FHDI FOR SMALL TO ULTRA-LARGE DATA

FHDI is a non-parametric imputation method and creates a complete data set with fractional weights after imputation while preserving the joint probability of the observed data. Some of the authors of this paper developed an R package, FHDI, available on CRAN [4,19] and its initial parallel version [5]. Yet, these tools have several limits to curing ultra-large incomplete data. This section presents the consistent evolution of serial version FHDI to HPC-based FHDI (called P-FHDI) and even to ultra-large data-oriented parallel FHDI (named UP-FHDI).

2.1. Serial Fractional Hot Deck Imputation (FHDI)

FHDI takes several advantages: First, imputed values are built upon observed responses, not artificial values, thereby preserving the distribution features of original data; Second, a strong model assumption is not necessary for imputation [3]; Third, it works well for general-purpose of estimations without self-efficient or congeniality conditions; Thus, it is free from the improper imputation issue following the frequentist's EM framework [3]. The detailed formulations and example codes are available in [4], and this section summarizes the key equations and procedures of the serial FHDI.

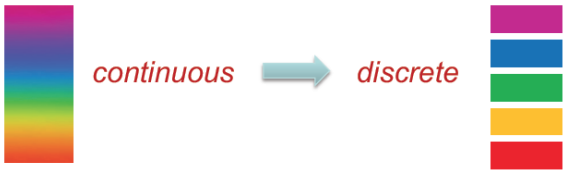


Figure 1. The key procedures of FHDI and an illustration of discretization of continuous variables (y) to discrete variables (z).

Fig. 1 presents the key procedures of FHDI and an intuitive illustration of the variable discretization. The basic setup of FHDI is as follows. Suppose that we have a finite population of size N, indexed by $U = \{1, 2, \ldots, N\}$, with two continuous variables y_1 and y_2. Let z_1 and z_2 be discretized values of y_1 and y_2, respectively. $z_1$is assumed to take discrete values $\{1, \ldots, G\}$ and z_2 takes $\{1, \ldots, H\}$. Let δ_p $(p = 1, 2)$, a response indicator function of y_p, i.e., 1 if y_p is observed and 0 otherwise. The finite population U can be subdivided into $G \times H$ cells based on z_1 and z_2, and we assume a cell mean model on the cells such that

$$y \mid (z_1 = g, z_2 = h) \sim (\mu_{gh}, \Sigma_{gh}), g = 1, \ldots, G, h = 1, \ldots, H,$$

where $y = (y_1, y_2), \mu_{gh} = (\mu_{1,gh}, \mu_{2,gh})$is a vector of cell means and $\Sigma_{gh}$is the variance-covariance matrix of y in cell (gh).Let y_{obs} and y_{mis} be the observed and missing part of y, respectively. We assume that the data are missing at random (MAR) in the sense that$P(\delta \mid y) = P(\delta \mid y_{obs})$where $\delta = (\delta_1, \delta_2)$. Let A be the index set of the sample elements selected from the finite population U. Let A_R be the index set of the respondents who answered both items y_1 and y_2. Similarly, define $A_M$as the set of nonrespondents who have at least one missing value, i.e.,$A_M = \{j \in A;\ \delta_{1j}\delta_{2j} = 0\}$. Denote $n_R = n(A_R)$ and $n_M = n(A_M)$, respectively.

The key procedures of FHDI initially proposed by [19] consist of the following steps.The first cell construction step constructs imputation cells. The imputation cell variable z can be given in advance or can be obtained using the estimated sample quantiles.From the realized values of $z_{1i}$and z_{2i} (i.e., the i-th entity of z_1 and z_2, respectively), we can construct two sets of observed patternsof (z_1, z_2) for A_R and A_M. Let V_R be the set of all observed combinations of z_1 and z_2 in A_R. $n(V_R)_{max}$ is $G \times H$ at maximum, but it can be smaller in the realized samples. Similarly, we obtain V_M basedon the observed parts of nonrespondents. For example, we may have V_M = {(NA,NA), (NA, 1), (NA, 2), (1,NA), (2,NA)}in the case of two binary outcomes.

Once the imputation cells are finalized from the above discretization, the next step needs to estimate the cell probabilities π_{gh}defined by

$$\pi_{gh} = P(z_1 = g, z_2 = h), g = 1, \ldots, G;\ h = 1, \ldots, H.$$

The initial cell probabilities are obtained using only the respondents in A_R. These initial cell probabilities are updated using the expectation maximization(EM) method, modified from the EM by weighting [20]. Details of the modified EM algorithm are given in [4]. In essence, the EM algorithm seeks to adjust each donor's weight so that the joint probability distribution can be as smooth as possible. The central importance lies in how to prepare the fractional weight for each donor robustly, i.e. w_{ij} where i corresponds to the i-thdonor while j corresponds to the j-th recipient. Complete mathematical formulae for the EM algorithm and w_{ij} are presented in [4].

The unbiased fully efficient fractional imputation (FEFI) employs all respondents as donors to each recipient in the same cell and then assigns the FEFI fractional weights to each donor. However, this FEFI may not be attractive in practice due to its huge size. Instead of using all the respondents, we can select just *M* donors among the FEFI donors with the selection probability proportional to FEFI fractional weights and then assign equal fractional weights. As for donor selection, we used a tailored systematic sampling method given in [4]. Variance estimation after imputation is vital to offer an uncertainty measure to researchers. FHDI uses the Jackknife variance estimation scheme.

As a simple validation, Table 1 presents the standard errors of the three mean estimators. The sample data is generated as $n = 100$ for the multivariate data vector $y_i = (y_{1i}, y_{2i}, y_{3i}, y_{4i}), i = 1, \ldots, n$. The standard normal distributions are used for random value generation, and the Bernoulli distribution is used for random missingness for each variable as $\delta_k \sim B(p_k)$,where $(p_1, p_2, p_3, p_4) = (0.6, 0.7, 0.8, 0.9)$. Although the response indicators are generated based on the missing completely at random (MCAR) assumption for simplicity, the FHDI method also holds for other response models based on MAR. The Naive estimator is justa simple mean-based estimator computed using only observed values. Since the partially observed valuesare used in the mean estimation, the two estimators (i.e., FEFI and FHDI) obtained using fractional hot deck imputationproduce smaller standard errors than the Naive estimator.

Table 1. Standard errors of three mean estimators confirming the superiority of FHDI.

Estimator	$E(y_1)$	$E(y_2)$	$E(y_3)$	$E(y_4)$
Naïve (mean-based)	0.135	0.135	0.150	0.138
FHDI	0.129	0.121	0.137	0.131
FEFI	0.128	0.121	0.137	0.130

Table 2. Regression coefficient estimates with standard errors (SE).

Estimator	Intercept	SE of Intercept	Slope	SE of Slope
True	0		0.5	
Naïve (mean-based)	-0.074	0.305	0.588	0.142
FHDI	0.023	0.111	0.472	0.052
FEFI	0.035	0.103	0.466	0.048

Table 2 presents the positive impact of FHDI on the subsequent regression model. Table 2 shows the regression coefficient estimates with standard error (SE) for the three estimators. Point estimates of the FEFI and FHDI estimators are much closer to the true values than the Naive estimators. Also, two fractional imputation estimators have smaller standard errors than the naive estimator. All R codes to obtain these results are given in [4].

As shown in Fig. 2, incomplete data may lead to biased decisions by a few % errors or more than 10%, depending on the data type. Such a small error may have significant scientific, economic, and social impacts. [1] showed that FHDI can improve the accuracy of subsequent ML and statistical inference, and its positive impact on the root-mean-square-error (RMSE) of ML and statistical model can be a few percent to more than 20% depending upon data and ML and statistical models (Fig. 2).

The adopted ML methods include artificial neural networks (ANN), support vector machine (SVM), and extremely randomized trees (ERT). The adopted advanced statistical model is the non-parametric generalized additive model (GAM). These ML and statistical methods are popular in broad science and engineering fields. Many ML and statistical packages tend to have naïve imputation methods as default, i.e., simply deleting incomplete rows or instances of the

input data sets before training ML models. Therefore, the improvement of accuracy of the subsequent ML and statistical models, as shown in Fig. 2 holds overarching implications for comprehensive data- and ML-driven research. If the data-driven decision allows a slight compromise of prediction errors, such simple imputation methods before ML and statistical inferences may be acceptable. However, if the decision involves critical influence on society, human health, politics, scientific results, and so on, a few percent of loss of accuracy should be handled by proper data-curing methods.

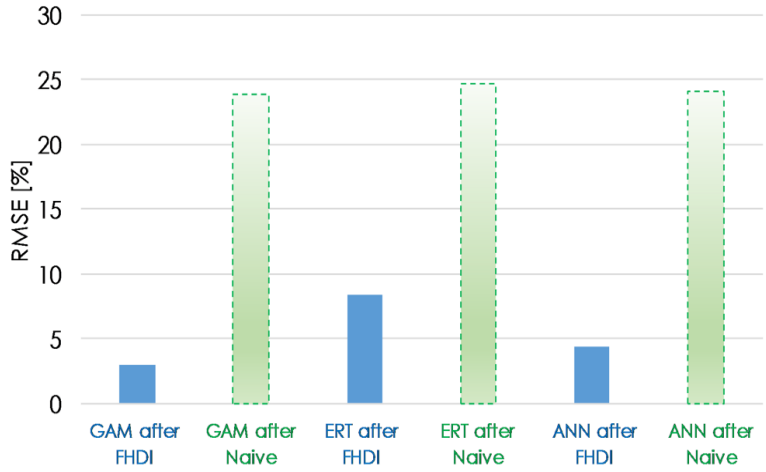


Figure 2. The positive impact of fractional hot-deck imputation (FHDI) on the subsequent ML and statistical predictions (adapted from [1]). GAM (generalized additive model), ERT (extremely randomized trees), and ANN (artificial neural networks)

2.2. Parallel FHDI (P-FHDI) for Large Data

The serial version FHDI is a general-purpose, assumption-free imputation method for handling multivariate missing data by filling each missing item with multiple observed values without resorting to artificially created values.The corresponding *R* package *FHDI*[4] holds generality and efficiency. Still, it is not adequate for tackling large-sized incomplete data due to the requirement of excessive memory and long running time. Some of the authorsof this paper [5] developed the first version of a parallel FHDI (P-FHDI) program suitable for curing large-sized incomplete datasets. Results show a favorable speed-up when the P-FHDI is applied to large datasets of millions of instances or 10,000 variables. It should be noted the target data sets are either big-n or big-p, not concurrently big-n and big-p. This capability is illustrated in Fig. 3.The developed P-FHDI program inherits all the advantages of the serial FHDI and enables a parallel variance estimation (i.e., parallelized Jackknife).

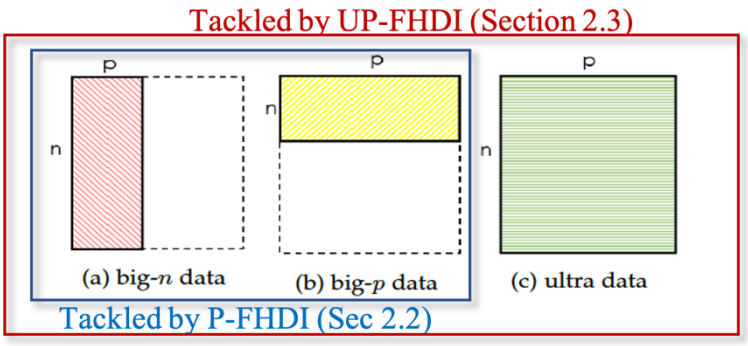


Figure 3. Types of incomplete data sets: (a) big-n data with large instances; (b) big-p data with high dimensionality. (a-b) are tackled by P-FHDI; (c) ultra data, concurrently big-n and big-p. UP-FHDI can tackle all types of large to ultra data sets (adapted from [5]).

Recall that Fig. 1 briefly summarizes the key procedures of FHDI. Figs. 4(a-b) shows the two parallel schemes adopted for developing P-FHDI. The two distinct schemes are needed since the primary global loops for many tasks are "implicit". Thus, a direct divide and conquer scheme is not applicable, as parallelization focuses on the separately parallelizable internal tasks without breaking the implicit loop. In contrast, some embarrassingly parallelizable tasks, such as Jackknife variance estimation, are tackled by the typical divide-and-conquer scheme. To achieve load balance during the P-FHDI, the cyclic distribution (Fig. 4c) is selectively chosen to balance the work domain among slave processors effectively.

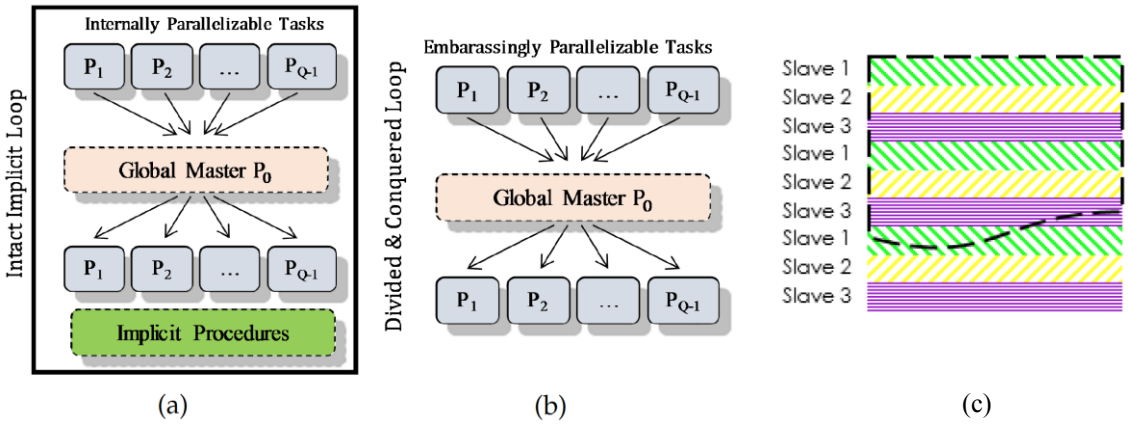


Figure 4. Adopted parallel computing schemes for P-FHDI: (a) Internal parallelization within the unbreakable implicit loop; (b) Typical divide and conquer for embarrassingly parallelizable explicit loop; (c) Cyclic job distribution over slave processors (dashed box means the computing jobs) (adapted from [5]).

[P-FHDI Procedure 1]Parallel Imputation Cell Construction: The determination of initial imputation cells may take considerably large iterations for the cell collapsing process to guarantee at least two donors for each recipient. The so-called cell collapsing algorithm (i.e., when donors are insufficient, merge adjacent imputation cells to make donors [4]) of serial FHDI is an implicit process that is non-parallelizable. Considering the inevitable obstacle, we employ internal parallelization within the unbreakable implicit iterations.

[P-FHDI Procedure 2] Parallel Joint Cell Probability Using EM Algorithm: The estimation of joint cell probability is an implicit and iterative process that does not support simple parallelism. The EM iterations run until the joint probability converges. In particular, the EM algorithm will terminate if changes in probabilities converge to a specific threshold (e.g., 10E-6).

[P-FHDI Procedure 3] Parallel Imputation: Imputation of the P-FHDI aims at selecting *M* donors for each recipient. The fractional weights for all possible donors assigned to each recipient are computed using the probability proportional sampling (PPS) method to select M donors randomly. In particular, it sorts all donors by the half-ascending and half-descending order to construct successive intervals.

[P-FHDI Procedure 4] Parallel Variance Estimation: The parallelized variance estimation is developed for the parallel Jackknife algorithm. A pre-processing function computes the cell probability for unique missing patterns recursively. Without the parallel Jackknife method, variance estimation of big-n or big-p data sets will be intractably expensive.

Systematic validations of P-FHDI were conducted with big-n or big-p data sets by [5]. To validate the P-FHDI, [5] adopted a variety of data sets (Table 3), including continuous, categorical, and hybrid data with instances up to millions and variables = 10,000. Both synthetic and practical data are used to confirm the general applicability of the P-FHDI.

Table 3. Some of the adopted datasets for validation of the P-FHDI. **U**(instances, variables, missing rate). Adapted from [5].

Data Set Type	**Variable Types**	**Dimensions and missing rate**
Synthetic	Continuous	U(1000000, 4, 0.25)
Practical (Air Quality)	Hybrid (Contin. and Categ.)	U(41757, 4, 0.1)
Practical (Nursery)	Categorical	U(12960, 5, 0.3)
Synthetic	Continuous	U(1000, 10000, 0.3)

The first validation focuses on the scalability of the P-FHDI with large instance data (big-n). Fig. 5(a) shows the desired speed-up with big-n data curing by P-FHDI. While fixing the large instance (n=1M), the impact of missing rates on the parallel performance of P-FHDI is investigated. Fig. 5(b) confirms the stable scalability of P-FHDI with varying missing rates of a fixed big-n data set. The following validation focuses on the parallel performance of the big-p data curing with P-FHDI. Fig. 6 shows a promising performance of the big-p data curing (p=10,000).

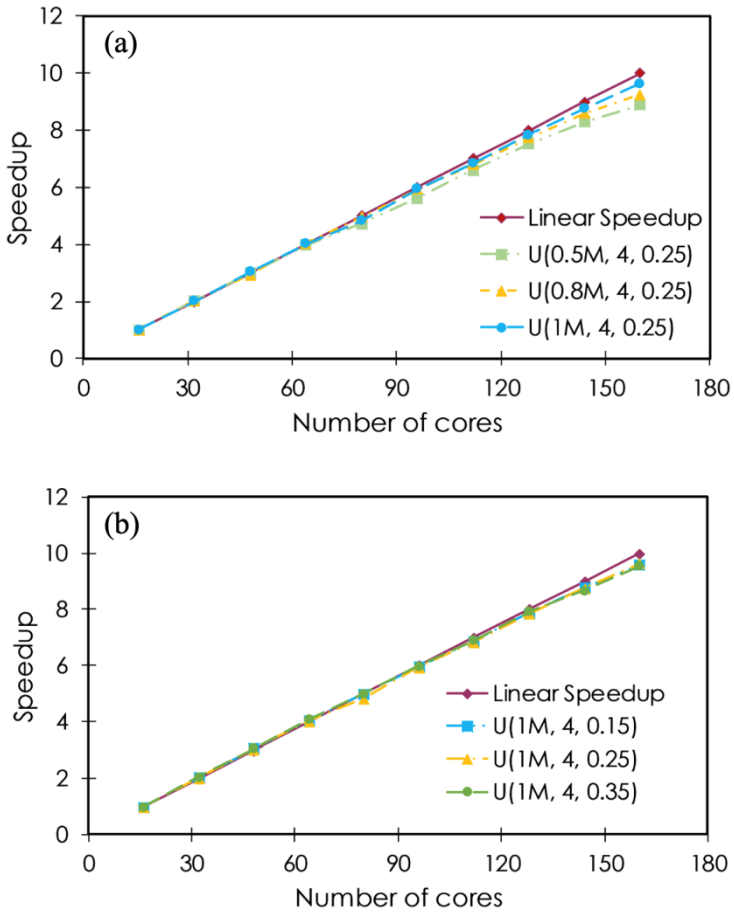


Figure 5. (a) Scalability of P-FHDI for Big-n Data Curing: Impact of the number of instances n on speed-ups of the entire P-FHDI (i.e., imputation and variance estimation) with datasets U(n; 4; 0.25) meaning four variables, 25% missing rate, and varying n; (b) Scalability of P-FHDI with Varying Missing Rate: Impact of the missing rate η on speed-ups of the entire P-FHDI (i.e., imputation and variance estimation) with datasets U(1M; 4; η): 1 million instances and four variables by varying η [adapted from [5]].

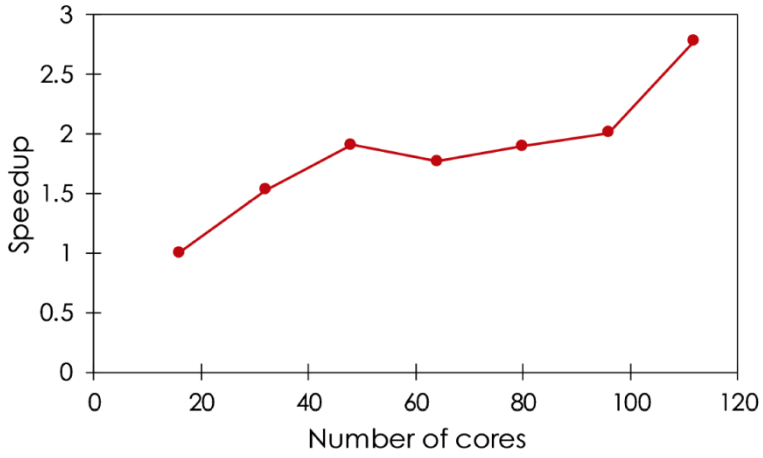


Figure 6. Initial Performance Test of P-FHDI for Big-p data curing: P-FHDI cured an extremely high-dimensional dataset U(1,000; 10,000; 0.3): 1,000 instances and 10,000 variables with 30% missingness. We adopt three selected variables with Fan and Lv (2018) [21] 's sure independence screening based on the big-p algorithm. (Adapted from [5]).

2.3. Ultra Large Data-Oriented Parallel FHDI (UP-FHDI)

P-FHDI is the first parallel version of FHDI that can cure big-n or big-p data sets separately. But, if the dataset is ultra-large, i.e., concurrently big-n and big-p, we need to have special parallel algorithms and ultra-data handling schemes. Like P-FHDI, UP-FHDI leverages parallelism for essential four steps of fractional hot-deck imputation theory: (1) parallel imputation cell construction, (2) parallel expectation maximization, (3) parallel imputation; (4) parallel variance estimation. While P-FHDI handles all the data on memory available, the sheer size and volume of ultra data require a new specialized data handling scheme and associated parallelism. As briefly described in Fig. 7, UP-FHDI adopts the OOOPS system [22] for optimal IO workload balance with local hard drives of the HPC environment. All the essential steps of UP-FHDI are parallelized so that it can easily handle ultra-data. Thus, as long as local storage is large enough, UP-FHDI has no limit on the number of instances and high dimensionality.

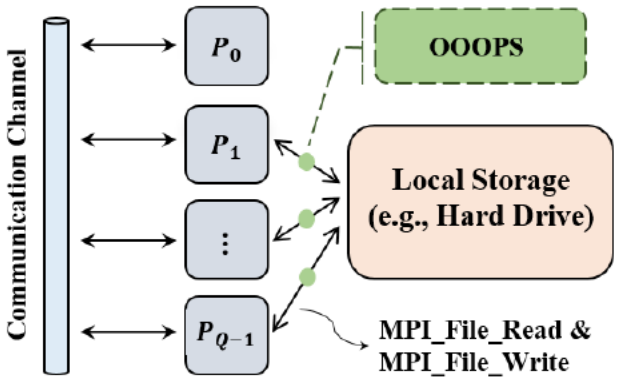


Figure 7. UP-FHDI's parallel file system on multiple writers and readers and the OOOPS optimally throttles the IO workload of ultra-large data (marked by solid green circles). Adapted from [23]

Another important advancement of UP-FHDI is the specialized variance estimation technique for ultra-large data curing. It is the linearized variation estimation technique. After performing imputation, estimating the imputed results' uncertainty is important. The well-known Jackknife variance estimation method is commonly used in the previous version of P-FHDI and serial version FHDI. However, the Jackknife method is unsuitable for ultra-data curing as the computation and memory cost increase exponentially with the number of instances and dimensions. Thus, UP-FHDI implements the efficient linearized variance estimation technique (detailed formulations are presented in [23]). When the number of instances is large, the linearized variance estimation technique reliably replaces the Jackknife estimation method (see Fig.8). Also, Fig. 8 confirms that as the number of instances increases, the difference between Jackknife and Linearized methods becomes small enough. Fig. 8 uses the absolute difference of standard errors (ADSE),which is defined as

$$ADSE \equiv \frac{1}{p}\sum_{l=1}^{p}\left|(\widehat{SE}_{linear,l} - \widehat{SE}_{Jack,l})/\widehat{SE}_{Jack,l}\right|$$

where $\widehat{SE}_{Jack,l}$ and$\widehat{SE}_{Jack,l}$ are the standard error of the mean estimator of the l-th variable using the Jackknife and linearized variance estimation methods, respectively.

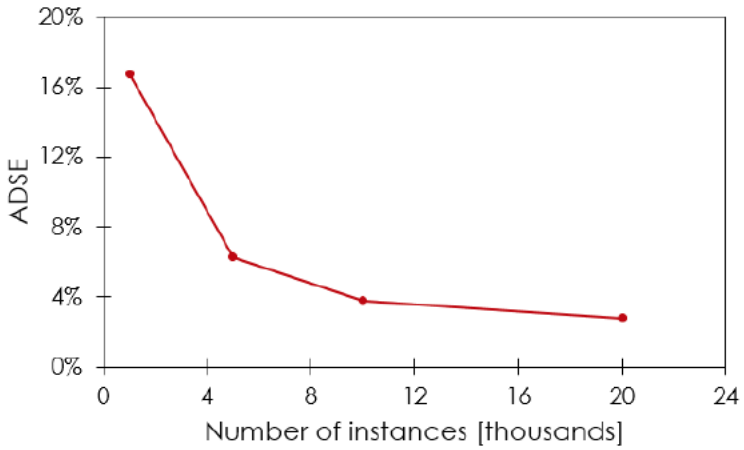


Figure 8. Impact of the increasing instances on the absolute difference of standard error (ADSE). (Adapted from [23]).

Table 4 summarizes the practical data sets used for testing the basic performance of the developed UP-FHDI, which emphasizes the generality of the data categories and disciplines of the data. Throughout the initial performance test, the computational gain of the linearized variance estimation is excellent. As shown in Fig. 9, the linearized variance estimation techniques cost only 2%-7% of the Jackknife estimation scheme. Thus, the linearized variance estimation is confirmed to be a successful substitute to the Jackknife method for ultra-large incomplete data imputation. To ensure the imputation accuracy of UP-FHDI, we compare the mean-based naïve imputation method against the UP-FHDI. Parts of four practical data sets of Table 4 are randomly removed and imputed by the mean-based naïve imputation and the UP-FHDI. As shown in Fig. 10, UP-FHDI outperforms the naïve imputation method by a factor of 2~5. Since such a mean-based naïve imputation is still prevalent in "big" data research communities and popular ML programs, these results underpin the significance of the UP-FHDI for ML and data science.

Table 4. Practical data sets used for the initial performance tests of UP-FHDI.

Dataset name	Number of Instances (*n*)	Number of Variables (*p*)	Discipline
CT [24]	53500	380	Medicine
p53 [25]	31159	5408	Genetics
Travel [26]	23772	50	Transportation
Swarm [27]	24016	2400	Biology

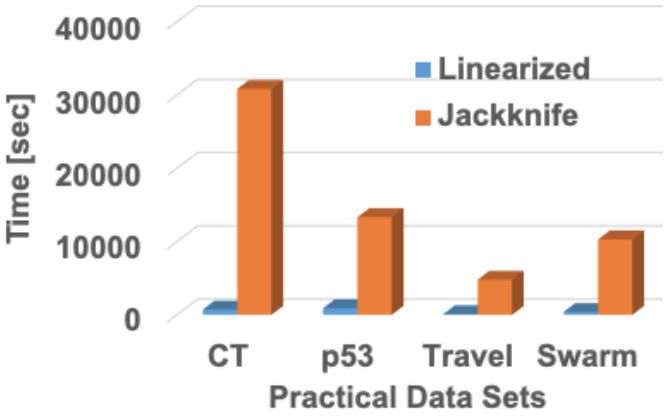


Figure9. Comparison of the total run time of the linearized and Jackknife variance estimation methods with four practical large data sets. The linearized variance estimation substantially outperforms the Jackknife method (Adapted from [23]).

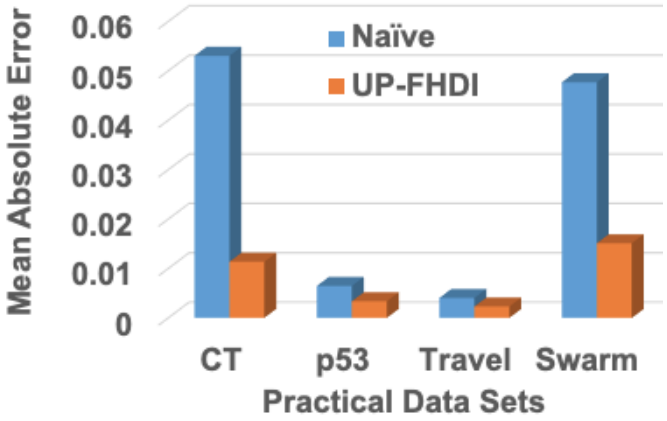


Figure 10. Comparison of UP-FHDI and Naïve (mean-based) method. The mean absolute error is calculated at the randomly deleted cells with original values(Adapted from [23]).

We compare the performance of UP-FHDI and baseline imputation methods, including naive imputation and the recently proposed Generative Adversarial Imputation Network (GAIN) (see Fig. 11). The naive imputation adopts a simple mean estimator computed using observed values. GAIN is a GAN-based framework that employs an imputer network to handle the missing data [28]. Experiments show that GAIN outperforms many state-of-the-art imputation techniques, and the summary of key theories of GAIN is presented in [23]. This study adopts the default settings to build the GAIN model. Considering the stochastic nature of GAIN, we conduct ten experiments for each dataset and average the performance measures. Using large real-world datasets (Earthquake, Bridge Strain, Travel Time, and CT Slices), UP-FHDI performs well

comparable to GAIN regarding RMSE results. Note that the default-setting GAIN aborted when applied to Swarm, p53, and Radar. It appears that GAIN may require specific extensions for large/ultra data curing.

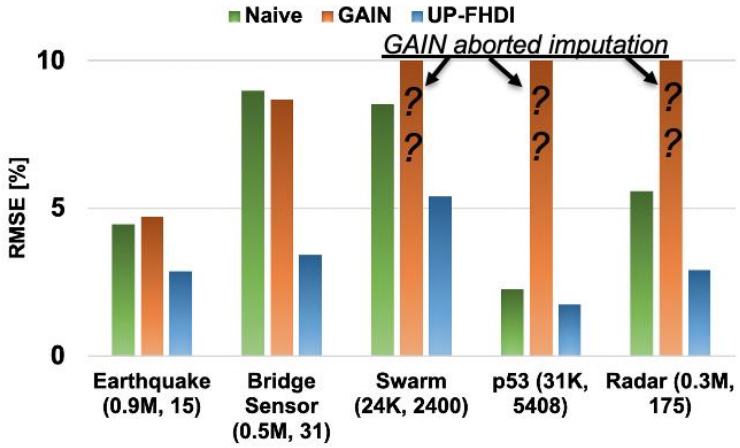


Figure 11. Positive impact and superior performance of UP-FHDI: Superior accuracy and stability of UP-FHDI compared to naive and GAIN (generative adversarial imputation nets), adapted from [23]. Diverse five large data sets of (instances n, the number of variables p) are used. With high-dimensional (big-p) data sets (Swarm, p53, Radar), default-setting GAIN aborted imputation during running, whereas UP-FHDI successfully imputed them with consistently high accuracy.

3. CONCLUSIONS

This review paper summarizes how the fractional hot-deck imputation (FHDI) method has evolved from a serial version to ultra-large data. FHDI has notable advantages compared to existing imputation methods since it does not require domain-specific and/or statistical assumptions. FHDI can thus become a general-purpose, assumption-free data-curing program for general users in science and engineering and beyond. By inheriting FHDI's generality and efficiency, several parallel computing algorithms enabled FHDI to become ultra data-oriented parallel FHDI (UP-FHDI). UP-FHDI can cure concurrently big-n and big-p (called "ultra") data with favorable scalability and accuracy. A specialized variance estimation technique also provides uncertainty measures of the UP-FHDI. Diverse validations with synthetic and practical data sets confirm that UP-FHDI outperforms naïve imputation methods as well as advanced imputation methods such as GAIN. Uncertainty estimation is also made possible with the developed special variance estimation scheme for UP-FHDI. FHDI and UP-FHDI also confirm that their cured data can improve the accuracy of the subsequent ML and statistical predictions. All data and programs are made publicly available via the relevant papers. The continued evolution of FHDI will help promote data- and ML-driven innovations and high-precision decision-making in broad science, engineering, and beyond.

ACKNOWLEDGMENTS

This research is supported by National Science Foundation (NSF) grant number OAC-1931380. The HPC@ISU equipment partially supports the high-performance computing facility used for this research at ISU, some of which have been purchased through funding provided by NSF CNS 1229081 and CRI 1205413. Ultra data applications of this paper used the Extreme Science and Engineering Discovery Environment (XSEDE), NSF ACI-1548562.

REFERENCES

[1] Song, I., Yang, Y.,Im, J., Tong, T.,Ceylan, H. &Cho, I. (2019)"Impacts of fractional hot-deck imputation on learning and prediction of engineering data",*IEEE Transactions on Knowledge and Data Engineering,* Vol. 32, No. 12,pp 2363-2373.

[2] Rubin, D. (1996) "Multiple imputation after 18+ years", *Journal of the American Statistical Association*, Vol. 91, pp 473-489.

[3] Yang, S & Kim, J.(2016) "A note on multiple imputation for method ofmoments estimation",*Biometrika,*Vol. 103, pp 244-251.

[4] Im, J., Cho, I.& Kim, J. (2018)"An R package for fractional hot deckimputation",*The R Journal,*Vol. 10, pp 140-154.

[5] Yang, Y., Kim, J.,& Cho, I. (2020)"Parallel fractional hot deck imputation andvariance estimation for big incomplete data curing"*IEEE Transactionson Knowledge and Data Engineering*, Vol. 34, No. 8, pp 3912-3926.

[6] Rubin, D. B. (1976)"Inference and missing data",*Biometrika*, Vol. 63, No. 3, pp 581-592.

[7] van Buuren, S &Groothuis-Oudshoorn, K. (2011)"mice: Multivariate imputation by chained equations inR", *Journal of Statistical Software*, Vol. 45, pp 1-67.

[8] Su, Y. S., Gelman, A., Hill, H.&Yajima, M. (2011) "Multiple imputation with diagnostics (mice) in r: Opening windows into the black box",*Journal of Statistical Software*, Vol. 45, pp 1-31.

[9] Honaker, J., King, G.& Blackwell, M. (2011)"Amelia ii: A program for missing data",*Journal of Statistical Software*, Vol. 45, pp 1-47.

[10] Kowarik, A. &Templ, M. (2016) "Imputation with the R package VIM",*Journal of Statistical Software*, Vol. 74, pp 1-16.

[11] Kim, J. K.& Yang, S (2017)"A note on multiple imputation under complex sampling",*Biometrika*, Vol. 104, No. 1, pp 221-228.

[12] Meng, X. L. (1994)"Multiple-imputation inference with uncongenial sources of input",*Statistical Science*, Vol. 9, No. 4, pp 538-573.

[13] Nielsen, S. F. (2003) "Proper and improper multiple imputation",*International Statistical Review*, Vo. 71, No. 3, pp 593-607.

[14] Durham, T. J., Libbrecht, M. W.,Howbert, J.,Bilmes, J.& Noble, W. S. (2018)"Predicted parallel epigenomics data imputation with cloud-based tensor decomposition",*Nature communication*, Vo. 9, No. 1, pp 1402-1402.

[15] Hu, X. (2011)"Acceleration genotype imputation for large dataset on gpu",*Procedia Environmental Science*, Vol. 8, pp 457-463.

[16] Li, F.,Gui, Z., Wu, H., Gong, J., Wang, Y.& Tian, S. (2018)"Big enterprise registration data imputation: Supporting spatiotemporalanalysis of industries in China",*Computers, Environment and Urban Systems*, Vol. 70, pp 9-23.

[17] Stekhoven, D. J.&Buhlmann, P. (2012) "Missforest—non-parametric missing value imputation for mixed-type data",*Bioinformatics*, Vol. 28, No. 1, pp 112-118.

[18] Dominici, F., Caffo, B.,&Peng, R.(editor). (2011) "Parallel MCMC Imputation for Multiple Distributed Lag Models: A Case Study in Environmental Epidemiology",*The Handbook of Markov Chain Monte Carlo*.

[19] Im, J., Kim, J. K. & Fuller, W. A. (2015) "Two-phase sampling approach to fractional hot deck imputation",*In JSM Proceedings of Survey Research Methodology Section,* pp 1030-1043, Seattle,WA, USA.

[20] Ibrahim, J. G. (1990)"Incomplete data in generalized linear models", *JASA*, Vol. 85, pp 765-769.

[21] Fan, J. &Lv, J. (2008) "Sure independence screening for ultrahigh dimensional feature space"*Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, Vol. 70, No. 5, pp 849-911.

[22] Huang, L. & Liu, S. (2020) "Ooops: An innovative tool for io workload management on supercomputers"*IEEE 26th International Conference on Parallel and Distributed Systems (ICPADS)*, pp 486-493.

[23] Yang, Y., Kwon, Y., Kim, J. K. & Cho, I. (2022) "Ultra Data-Oriented Parallel Fractional Hot-Deck Imputation with Efficient Linearized Variance Estimation", *IEEE Transactions on Knowledge and Data Engineering* (under 2nd review).

[24] Graf, F., Kriegel, H.P., Schubert, M., Poelsterl, S. & Cavallaro, A. (2011) "Relative location of CT slices on axial axis data set", *UCI Machine learning Repository.*

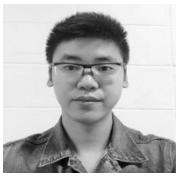
[25] Lathrop, R. H. (2010) "p53 mutants data set", *UCI Machine learning Repository.*

[26] Gao, C., Guo, H. & Sheng, W. (2021) "Travel time data of Chengdu road network", *IEEE Dataport.*
[27] Abpeikar, S., Kasmarik, K., Barlow, M. & Khan, M. (2020) "Swarm behavior data set", *UCI Machine learning Repository*.
[28] Yoon, Y., Jordon, Y.& van der Schaar, M. (2018)"Gain: Missing data imputation using generative adversarial nets"*35th International Conference on Machine Learning*, pp 5689-5698.

AUTHORS

In Ho Cho (corresponding author) received thePhD degree in civil engineering and minor in Computational Science and Engineering from California Institute of Technology, USA in 2012. He is currently an associate professor of CCEE department, ISU. His research interests include data-driven engineering and science, computational statistics, computational science and engineering, and parallel computing.

Jae-Kwang Kim received the PhD degree inStatistics from ISU in 2000. He is a fellow of American Statistical Association and Institute of Mathematical Statistics and currently a LAS Dean's professor in the department of statistics at ISU. His research interests include survey sampling, statistical analysis with missing data, measurement error models, multi-level models, causal inference, data integration, and ML.



Yicheng Yang received his PhD degree from the department of CCEE of ISU in 2021. He is currently a master's student in the department of computer science with an emphasis on data mining. His research interests include parallel imputation, ML, and data-driven engineering.

Yonghyun Kwon is a current PhD student at ISU and he is a graduate research assistant at Center for Survey Statistics & Methodology(CSSM). He received the BS degree in Statistics from Seoul National University in 2020. His research interests include survey sampling, missing data analysis, and ML.

Ashish Chapagain is a current PhD student at CCEE department of ISU. His research interests seek to combine data science, mechanics, and machine learning for machine learning- and data-driven science and engineering.