Pattern Recognition 45 (2012) 919–933

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

# A comparison of imputation methods for handling missing scores in biometric fusion

Yaohui Ding *, Arun Ross

*Lane Department of Computer Science and Electrical Engineering, West Virginia University, Morgantown, WV, USA*

ARTICLE INFO

*Article history:*
Received 20 January 2011
Received in revised form
6 July 2011
Accepted 2 August 2011
Available online 11 August 2011

*Keywords:*
Missing data
Imputation
Multibiometric fusion

ABSTRACT

Multibiometric systems, which consolidate or fuse multiple sources of biometric information, typically provide better recognition performance than unimodal systems. While fusion can be accomplished at various levels in a multibiometric system, score-level fusion is commonly used as it offers a good trade-off between data availability and ease of fusion. Most score-level fusion rules assume that the scores pertaining to *all* the matchers are available prior to fusion. Thus, they are not well equipped to deal with the problem of missing match scores. While there are several techniques for handling missing data in general, the imputation scheme, which replaces missing values with predicted values, is preferred since this scheme can be followed by a standard fusion scheme designed for complete data. In this work, the performance of the following imputation methods are compared in the context of multibiometric fusion: K-nearest neighbor (KNN) schemes, likelihood-based schemes, Bayesian-based schemes and multiple imputation (MI) schemes. Experiments on the MSU database assess the robustness of the schemes in handling missing scores at different missing rates. It is observed that the Gaussian mixture model (GMM)-based KNN imputation scheme results in the best recognition accuracy.



## 1. Introduction

Biometrics is the science of establishing human identity based on the physical or behavioral attributes of an individual [1]. These attributes include fingerprints, face texture, iris, hand geometry, voice, gait and signature. A biometric system is essentially a pattern recognition system that operates by acquiring biometric data from an individual, extracting a feature set from the acquired data, and comparing this feature set against the template stored in the database [2]. Multibiometric systems overcome many practical problems that occur in single modality biometric systems, such as noisy sensor data, non-universality and/or lack of distinctiveness of a biometric trait, unacceptable error rates and spoof attacks, by consolidating multiple biometric information pertaining to the same identity [3]. Biometric fusion can be implemented at various levels, such as raw data level, image level, feature level, rank level, score level and decision level. Fusion at the score level is the most popular approach discussed in the literature [2,3].

In score-level fusion, there are multiple biometric matchers. Each biometric matcher generates a match score indicating the proximity of the input biometric data (known as the probe) with the template data stored in the database (known as gallery). Thus, the set of scores pertaining to these matchers may be viewed as a score vector. Most techniques for score-level fusion are designed for a complete *score vector*[1] where the scores to be fused are assumed to be available. These techniques cannot be invoked when score vectors are incomplete.

Various factors can result in incomplete score vectors in multibiometrics: (a) failure of a matcher to generate a score (e.g., a fingerprint matcher may be unable to generate a score when the input image is of inferior quality); (b) absence of a trait during image acquisition (e.g., a surveillance multibiometric system may be unable to obtain the iris of an individual); (c) sensor malfunction, where the sensor pertaining to a modality may not be operational (e.g., failure of a fingerprint sensor due to wear and tear of the device); or (d) during enrollment, all the necessary biometric traits may not be available.

When encountering missing data, score-level fusion schemes may have to process the incomplete score vector prior to applying the fusion rule. Deletion methods, which omit all incomplete vectors, can result in biased results when complete cases are unrepresentative of the entire data [4–6], and are not suitable for use in biometric systems [7]. Certain "strong" decision tree methods, such as dynamic path generation [8] and the lazy decision tree approach [9,10], can utilize only the observed information without any deletion or replacement. However, the

* Corresponding author.
*E-mail address:* yding@mix.wvu.edu (Y. Ding).

[1] Here, the elements of the vector are the scores generated by the individual matchers.

0031-3203/$ - see front matter © 2011 Published by Elsevier Ltd.
doi:10.1016/j.patcog.2011.08.002

process of growing a decision tree is computationally expensive, and requires relatively large number of training examples.

Imputation methods, on the other hand, which substitute the missing scores with predicted values are better since (a) they do not delete any of the score vectors which may contain useful information for identification, and (b) their application can be followed by a standard score fusion scheme.

Certain simple imputation schemes that make some assumptions about the underlying distributions or models of the complete data, such as mean or median imputation, regression-based imputation and Hot-Deck imputation, can perform well when the fraction of the missing data is not large, but their shortcomings, such as the overestimation of association among variables[2] and the variance reduction within a variable, have to be considered [11,12].

Some complex imputation methods, such as Neighbor-based schemes (e.g. K-nearest neighbor (KNN)), likelihood-based schemes (e.g. multivariate normal models (MNs) and Gaussian mixture models (GMMs)), Bayesian-based schemes (e.g. Bayesian network (BN) [13] and Markov chain Monte Carlo (MCMC)) and multiple imputation (MI) schemes, have earned significant attention during the last decade. Some tools and packages based on these schemes have been built and implemented as standard methods in some research fields. However, they have received limited attention in the biometric literature.

The goal of the paper is to analyze whether these imputation methods are useful in the context of missing data in biometric fusion. While most imputation methods covered in this work are based on existing literature, they have not been suitably appropriated into the framework of multibiometric fusion. The discussion about these methods will be based on the matching performance after the application of a particular biometric fusion rule known as the simple sum rule.

The remainder of this paper is organized as follows. Section 2 introduces the approaches suggested in the literature for dealing with the missing data problem. The design of experiments which considers constraints and criteria in the context of biometrics is described in Section 3. The details of imputation schemes employed in this work are described in Section 4. The experiments and ensuing results of the different schemes are discussed in Section 5. Closing comments are provided in Section 6.

## 2. Related work

Various taxonomies have been developed to distinguish imputation methods. In most cases, an imputation method can only perform well under some specific assumptions either about the entire data or the missing variables. According to these assumptions, imputation methods can be grouped into three families: (a) the parametric family, such as the multivariate normal model, which is the most common assumption employed [12,14–16]; (b) the non-parametric family, such as the KNN scheme [17], the Hot-Deck scheme [18], and the kernel extension based schemes [19]; (c) the semi-parametric family, such as schemes based on GMMs [20–22], that allow for controlling the trade-off between parsimony of sample size and flexibility of model assumption. Particularly, the multivariate imputation by chained equations (MICE) schemes [23–25], which estimate the parameters under the multivariate normal assumption and then search the imputed value by nearest neighbor methods, combine aspects of the parametric family and the non-parametric family.

Additionally, according to the number of predictions generated for one missing value, the imputation methods can be divided into single imputation schemes and multiple imputation schemes. The single imputation schemes replace a missing value with a single predicted value that cannot reflect the uncertainty about that value. Multiple imputation is preferred when there are concerns about the accuracy and error bounds of the imputed values [11,25–27]. In multiple imputation schemes, appropriate models that account for the random variation in data are used and the imputation process is repeated several times. Then Rubin's Rules [27] are employed to combine these imputed values together to get a statistically valid estimation.

The specific process, which generates the imputed values for a particular incomplete vector, is the critical component of an imputation method. With this understanding, various imputation methods can be assorted into three categories:

- Regression-based schemes, where a linear or logistic regression is used to obtain the imputation of the missing variables (as responses) by the observed variables (as predictors) [15,24,28].
- Neighbor-based schemes, where a certain distance function is used to find the "closest" vector(s) imputation [17,18,23,29]. Here, the "closest" vector(s) are excepted to have similar characteristics as the incomplete vector.
- Sampling-based schemes, which are based on sampling algorithms such as Gibbs sampler and MCMC approaches, generate specific values based on the assumed model of the complete data [11,16,27]. Sampling-based schemes are frequently used in multiple imputation schemes, because the generated random samples always include intrinsic variation and uncertainty, as required by the MI schemes.

The problem of missing scores has recently received some attention in the biometric literature. Nandakumar et al. [30] designed a Bayesian approach utilizing both ranks and scores to perform fusion in an identification system. Instead of substituting the missing score(s) of the missing vector by the predicted score(s), the proposed method handles missing information by assigning a fixed rank value to the marginal likelihood ratio corresponding to the missing entity. As the result, the approach dealing with missing data does not need much change to their proposed rank-based fusion method. Fatukasi et al. [7] compared several simple imputation schemes, like zero imputation, mean imputation, KNN imputation and three different variants of the KNN schemes. An exhaustive fusion framework was designed for combining all possible combinations of available scores. The disadvantage of this framework is its exponential complexity as $2^k-1$ rules are required to cover a multibiometric system with $k$ modalities. Poh et al. [31] discussed an approach using support vector machines (SVM) with the neutral point substitution method. The experiments based on a multimodal data set demonstrated a better generalization performance than the sum rule fusion. However, the proposed method is strongly related to a particular training framework, viz., the SVM framework, and may not be applicable to other fusion schemes. Ding and Ross [32] used the Hot-Deck sampling method in conjunction with the GMM scheme to impute missing score values in a multimodal fusion framework employing the simple sum rule. Their experiments suggested the utility of the scheme under certain conditions.

## 3. Design of experiments

### 3.1. Database

The Michigan State University (MSU) database used in this study contains 500 genuine and 12,250 imposter score vectors. Take the $i$th score vector as an example; it is a 3-tuple: $(x_{i1},x_{i2},x_{i3})$, where $x_{i1}$, $x_{i2}$ and $x_{i3}$ correspond to the match scores obtained

[2] Each variable pertains to the scores corresponding to a single matcher. In some cases, "variables" are referred to as "attributes".

from face, fingerprint and hand-geometry matchers, respectively. The details of the database have been described by Ross and Jain [33]. The fingerprint and face data were obtained from user set I consisting of 50 users. Each user was asked to provide five face images and five fingerprint impressions (of the same finger). This data was used to generate 500 ($50 \times 10$) genuine scores and 12,250 ($50 \times 5 \times 49$) imposter scores for each modality. The hand geometry data was collected separately from user set II which also consists of 50 users. This also resulted in 500 genuine scores and 12,250 imposter scores for this modality. Each user in set I was randomly paired with a user in set II. Thus the corresponding genuine and imposter scores for all three modalities were available for testing.

It should be noted that the scores obtained from the face and hand-geometry matchers are distance scores, and those obtained from the fingerprint matcher are similarity scores. The fingerprint scores are converted into distance scores by subtracting from a large number (1500 in our experiments).

It should also be noted that the sample sizes of genuine scores and imposter scores are highly imbalanced in this database. Byon et al. [34] demonstrate that when the class sizes are highly imbalanced, classification methods tend to strongly favor the majority class, resulting in very low detection accuracy of the minority class. In order to simplify the problem and retain generality, the proportion of genuine scores and imposter scores is fixed at 1:4 in this paper. This means a total of 500 genuine scores and 2000 imposter scores are randomly selected from the original database. Fig. 1 shows the density plot of the dataset and the recognition performance of each modality.

### 3.2. Generation of missing data

In order to evaluate the performance of imputation methods, missing entries were synthetically introduced into a complete (that has no missing data) match score matrix. There are two different ways that are widely used to introduce missing data: the histogram-based scheme and the rate-based scheme [35].

In the histogram-based scheme, histograms are produced for each variable, and then entries are removed from the complete matrix based on these histograms. In this case, the histogram of the artificially missing entries will be similar to that of the original matrix. In the rate-based scheme, a specific proportion of the entries is randomly selected and then removed from the complete score matrix.

The former cannot be used in this work because the histograms or the estimates of densities are also used by some of the imputation methods, such as the GMM-based methods. If the histogram from the original score matrix fits the model assumed by the imputation method, the artificially missing data will also fit the assumed model well. Consequently, the imputation method will result in an optimistically biased performance. Therefore, the rate-based scheme was used to generate missing data in the following experiments.

Fig. 2 illustrates the construction of training sets and test sets used in this study. Fifty percent of the score vectors were first randomly selected from the dataset as the training set. The proportion of genuine scores to imposter scores was set to 1:4. The remaining score vectors were used as the test set. Next, for each modality, 10% of the scores were randomly removed from

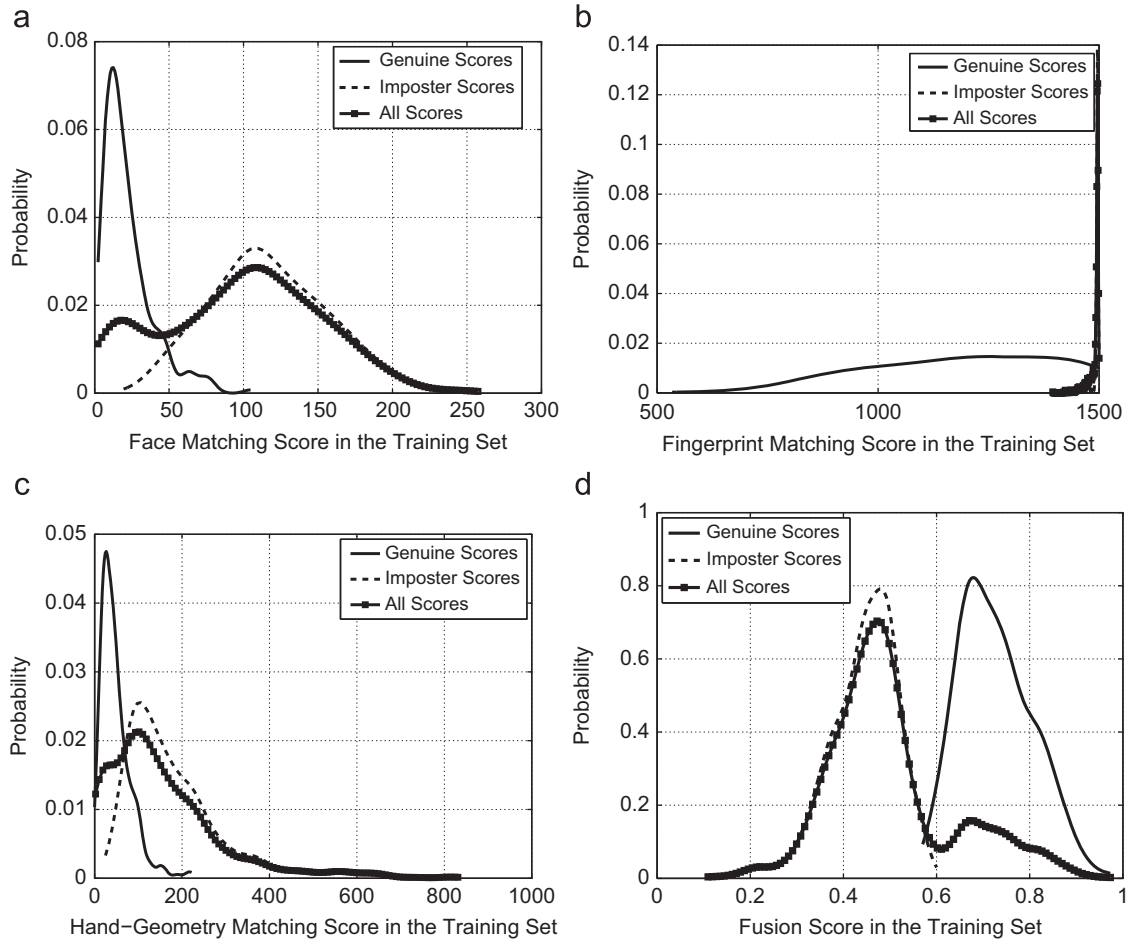


**Fig. 1.** Density plots of the genuine and imposter scores in the raw training set. Here, 50% of the dataset is used as training data. (a) Face, (b) fingerprint, (c) hand-geometry, (d) after fusion.

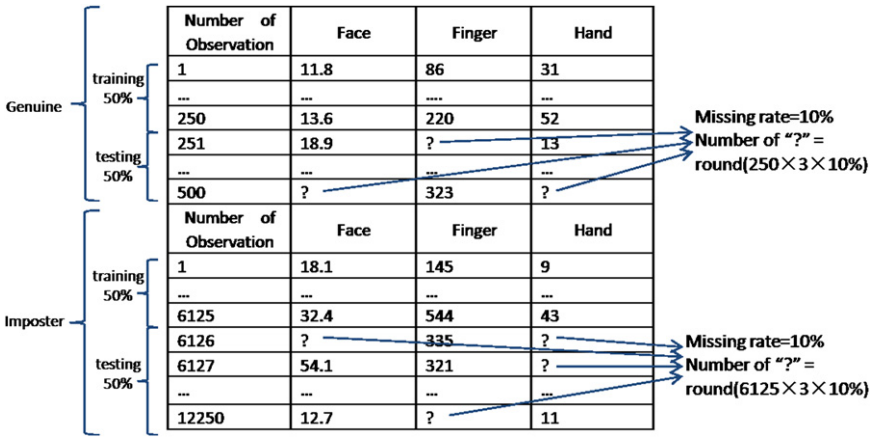


**Fig. 2.** Generation of the datasets used in the experiments. In this illustration, 50% of the dataset is used as training data, and a missing rate of 10% is specified for the test set.

the test set, in order to simulate missing data while making sure that each observation contained at least one observed score. As a result, a dataset with 50% of the observations as the training set and a 10% missing rate for the test set was generated.

Two different sizes for the training set, that were 10% and 50% of the entire dataset, were used for comparison. Similarly, two missing rates, 10% and 25%, were specified for the test sets. As a result, four different datasets encompassing different training rates and missing rates were generated.

In this work, K-fold cross-validation was used to account for the variability due to the random selection of the training sets. For example, when the training rate is 10%, the original data set is randomly divided into $K=10$ folds. Similarly, if the training rate is 50%, then $K=2$. The scores generated over multiple folds are used to compute the ROC curve.

### 3.3. The nature of missing data

Rubin [36] defines a taxonomy for different patterns of missing data:

- Missing completely at random (MCAR): The probability of an observation being missing does not depend on the value of the observed or unobserved data. In mathematical terms, this is written as:

$$Pr(X^m|X^{mis},X^{obs}) = Pr(X^m), \quad (1)$$

  where $X^m$ denotes the missingness mechanism,[3] and $X^{mis}$ and $X^{obs}$ denote the unobserved part and observed part, respectively. For example, if a participant decides whether or not to answer a question in a survey on the basis of coin flips, the missing data in this survey would conform to MCAR.
- Missing at random (MAR): the missingness mechanism does not depend on the unobserved data; rather, it depends only on the observed data. For example, in a medical test, if the blood work for a patient is conducted only if the temperature and blood pressure exceed certain thresholds, then the data related to the blood work will be missing for some patients. Mathematically,

$$Pr(X^m|X^{mis},X^{obs}) = Pr(X^m|X^{obs}). \quad (2)$$

- Missing not at random (MNAR): When neither MCAR nor MAR hold, we say that the data are missing not at random, abbreviated as MNAR. In other words, the mechanism of missing data depends on the unobserved data. For example, high income people are more likely to refuse to answer survey questions about income, thereby resulting in missing data pertaining to the income.

In this work, since missing data are synthetically generated by randomly removing entries from a complete dataset, the datasets appear to conform to the MCAR scenario.

### 3.4. Constraints in biometrics

In a standard classification problem, there are three possibilities involving missing data: (1) both training set and test set contain missing data; (2) only training set contains missing data; or (3) only test set contains missing data. In the context of biometrics, a fixed and complete *training set* is preferred because (a) imputation is based on this fixed set rather than a dynamically changing set, and (b) one can easily handle both complete and incomplete score vectors in the test set. Some recent research in biometrics discuss the challenges caused by a dynamically changing training set [37]. However, in most current biometric systems the training set is a fixed set as assumed in this work. Therefore, the computation of densities, nearest neighbours and regression coefficients is based on complete fixed training sets.

In the test set, the assumed independence among score vectors allows the biometric system to see only one score vector at a time. The constraint requires a vector-by-vector imputation process rather than a batch process where all missing scores corresponding to *all* incomplete vectors are imputed at the same time. With this understanding, only the observed part of this score vector and the training set can be used to perform the imputation. Any information from the other independent vectors cannot be incorporated.

Unlike the missing data problem in Gene-expression or other data mining applications where usually a large number of variables are used, most multibiometric systems discussed in the literature have fewer than five modalities. Therefore, exhaustive methods that consider all possible combinations of missing patterns, such as the exhaustive fusion framework [7], are likely to be useful in multibiometrics. On the other hand, some imputation methods like Bayesian network (BN) [13] which require more variables to compute probabilistic relationships between them, might be inefficient in a biometrics environment.

[3] In statistics, the cause for the missing data is referred to as missingness mechanism.

**Table 1**
Imputation methods and related tools used in this paper.

| Imputation methods | Related tools |
|---|---|
| K-nearest neighbor (KNN) | Implemented in R code |
| Maximum likelihood estimation in multivariate normal data (MLE-MN) | Package "mvnmle" in R with modification |
| Random draw imputation via gaussian mixture model (GMM-RD) | Package "mclust" in R with modification |
| KNN imputation via Gaussian mixture model (GMM-KNN) | Package "mclust" in R with modification |
| Predictive mean matching (PMM) | Package "mice" in R |
| Multiple imputation via Markov chain Monte Carlo (MI via MCMC) | The MI procedure in SAS |
| Multivariate imputation by chained equations (MICE) | Package "mice" in R |

**Table 2**
Properties of the imputation methods considered in this paper.

| | | KNN | MLE-MN | GMM-RD | GMM-KNN | PMM | MI via MCMC | MICE |
|---|---|---|---|---|---|---|---|---|
| Model assumption | Parametric | | √ | | | √ | √ | √ |
| | Non-parametric | √ | | | | √ | | |
| | Semi-parametric | | | √ | √ | | | |
| Imputation process | Regression-based | | √ | | | | | √ |
| | Neighbor-based | √ | | | √ | √ | | |
| | Sampling-based | | | √ | | | √ | |
| Number of imputation | Single imputation | √ | √ | √ | √ | √ | | |
| | Multiple imputation | | | | | | √ | √ |

### 3.5. Design of experiments

Table 1 shows seven different imputation schemes discussed in this work, and the related tools or packages used in the experiments. Table 2 shows the properties of each imputation scheme. Three factors, viz., model assumption, imputation process and the number of imputations, are considered. Based on this table, it is difficult to test the interaction between different factors, so this work focuses on testing the main effect of each factor. Experiments are implemented in four different groups based on the property of the schemes:

1. MLE-MN vs. PMM
2. GMM-RD vs. GMM-KNN
3. PMM vs. KNN vs. GMM-KNN
4. MI via MCMC vs. MICE

Receiver operating characteristic (ROC) curves are used to evaluate performance at multiple training set sizes and missing rates.

### 3.6. Criteria in biometrics

As stated by Marker et al. [38], two major criteria should be employed in assessing the performance of imputation methods: firstly, a good imputation method should preserve the natural relationship between variables in a multivariate dataset (in our case, the variables correspond to scores originating from multiple matchers); secondly, a good imputation method should embody the uncertainty caused by the imputed data by deriving variance estimates.

These two criteria are applicable for imputation in a biometric score dataset. Additionally, the matching accuracy which might be increased (or decreased) by the imputation is more critical than the similarity between the missing values and the imputed values in biometrics. So the use of imputed data should result in comparable matching performance to that of the original data containing no missing scores.

The min–max normalization scheme followed by the simple sum of scores has been observed to result in reasonable improvement in matching accuracy of a multimodal biometric system [2]. This scheme was used to generate ROC curves to summarize the fusion results of various imputation methods.

## 4. Description of imputation methods

### 4.1. Notation

In the context of multimodal biometric systems, a user $i$ offers $p$ biometric modalities. The system will generate a vector of match scores, $\boldsymbol{x}_i=(x_{i1},x_{i2},\ldots,x_{ip})$, where each match score corresponds to one modality. Suppose that there are $n$ users, then the score matrix with $n$ observations and $p$ variables can be written as:

$$\mathbf{D}=\begin{pmatrix}\boldsymbol{x}_1\\ \boldsymbol{x}_2\\ \vdots\\ \boldsymbol{x}_n\end{pmatrix}=\begin{pmatrix}x_{11} & x_{12} & \ldots & x_{1p}\\ x_{21} & x_{22} & \ldots & x_{2p}\\ \vdots & \vdots & x_{ij} & \vdots\\ x_{n1} & x_{n2} & \ldots & x_{np}\end{pmatrix},$$

where $x_{ij}$ denotes the match score from the $j$th modality of the $i$th user. Similarly, the training set can be expressed as $\mathbf{D}^{\mathrm{tr}}$.

If there is no missing data, the conventional fusion techniques can be implemented on each observation (row) separately. Any observation $\boldsymbol{x}_i$ containing missing scores can be written in the form $(\boldsymbol{x}_i^{\mathrm{obs}},\boldsymbol{x}_i^{\mathrm{mis}})$, where $\boldsymbol{x}_i^{\mathrm{obs}}$ and $\boldsymbol{x}_i^{\mathrm{mis}}$, respectively, denote the observed and missing variables (i.e., scores) for observation $i$. The missing values $\boldsymbol{x}_i^{\mathrm{mis}}$ can be replaced with the imputed value $\boldsymbol{x}_i^{imp}$ using the schemes considered below. For certain schemes, it is more clear to express the $p$ different modalities of the score matrix as $\mathbf{D}=(X_1,X_2,\ldots,X_p)$, and note here that the uppercase $X$ is used.

Different multivariate distributions will be assumed in the following schemes. Let $\Theta$ denote all the parameters to be estimated in a particular model. Take the MLE scheme as an example. The dataset $\mathbf{D}$ will be assumed to have a $p$-variate normal distribution with mean $\boldsymbol{\mu}=(\mu_1,\ldots,\mu_p)$ and covariance matrix $\Sigma$, so here $\Theta=(\boldsymbol{\mu},\Sigma)$ corresponds to the parameters of the multivariate normal distribution. Since several schemes use iterative algorithms for estimation, let $t$ denote an iteration counter, and then $\Theta^{(t)}$ denotes all the parameters to be estimated at the $t$th iteration.

### 4.2. K-nearest neighbor imputation

In a classical KNN imputation, the missing values of an observation are imputed based on a given number of instances (k) in $\mathbf{D}^{\mathrm{tr}}$ that are most similar to the instance of interest.

A measure of distance $d$ between two instances should be determined. In this work, a Euclidean distance function is considered. Let $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ be two observations; then $d$ is defined as:

$$d(\boldsymbol{x}_i,\boldsymbol{x}_j)=\sum_{h\in O_i\cap O_j}(x_{ih}-x_{jh})^2, \tag{3}$$

where $O_i=\{h|\text{the } h\text{th variable of the } i \text{ th observation is observed}\}$. In other words, only the mutually observed variables are used to calculate the distance between observations.

The KNN algorithm is described as follows:

(1) For each observation $\boldsymbol{x}_i$, apply the distance function $d$ to find the k nearest neighbor vectors in the training set $\mathbf{D}^{\mathbf{tr}}$;
(2) The missing variables $\boldsymbol{x}_i^{\text{mis}}$ are imputed by the average of the corresponding variables from those k nearest neighbors.

KNN imputation does not require the creation of a predictive model for each variable, and so it can easily treat instances with multiple missing values. However, there are some concerns with respect to KNN imputation. Firstly, which distance function should be used for a particular dataset? The choice could be Euclidean, Manhattan, Mahalanobis, Pearson, etc. In this work the Euclidean distance is employed. Secondly, the KNN algorithm searches through the entire dataset looking for the most similar instances, and can therefore be a very time consuming process. Thirdly, the choice of $k$ will impact the results. The choice of a small $k$ may produce a deterioration in the performance of the classifier after imputation due to overemphasis on a few dominant instances in the estimation process of the missing values. On the other hand, a neighborhood of large size would include instances that are significantly different from the instance containing missing values thereby impacting the estimation process, and the classifier's performance declines. According to our analysis (not shown here), we found $k=5$ to provide the best imputation accuracy on our relatively small dataset.

### 4.3. Imputation via the MLE in multivariate normal data

The imputation scheme using the MLE with a multivariate normal assumption (MLE-MN) was first described by Dempster et al. [39] in their influential paper on the EM algorithm. The key idea of EM is to solve a difficult incomplete-data estimation problem by iteratively solving an easier complete-data problem. Intuitively, "fill" in the missing data with the best guess under the current estimate of the unknown parameters (E-STEP), then re-estimate the parameters from the observed and filled-in data (M-STEP). An overview of EM has been given in [12,16,40].

In order to obtain the correct answer, Dempster et al. [39] showed that, rather than filling in the missing data values per se, the complete-data sufficient statistics should be computed in every iteration. The form of these statistics depends on the model under consideration. With the assumption of $K$-variate normal distribution, the hypothetical complete dataset $\mathbf{D}$ belongs to the regular exponential family. So $\sum_{i=1}^n x_{ik}$ and $\sum_{i=1}^n x_{ik}x_{ij}$ are sufficient statistics of samples from this distribution ($j,k=1,\ldots,K$). The modified $t$th iteration of E-STEP can then be written as:

$$E\left(\sum_{i=1}^n x_{ik}|\mathbf{D}^{\mathbf{tr}},\boldsymbol{x}_i^{\text{obs}},\Theta^{(t)}\right)=\sum_{i=1}^n x_{ik}^{(t)},\quad k=1,\ldots,K,$$

$$E\left(\sum_{i=1}^n x_{ik}x_{ij}|\mathbf{D}^{\mathbf{tr}},\boldsymbol{x}_i^{\text{obs}},\Theta^{(t)}\right)=\sum_{i=1}^n (x_{ik}^{(t)}x_{ij}^{(t)}+c_{ijk}^{(t)}),$$

where

$$x_{ik}^{(t)}=\begin{cases}x_{ik} & \text{if } x_{ik} \text{ is observed,}\\ E(x_{ik}|\mathbf{D}^{\mathbf{tr}},\boldsymbol{x}_i^{\text{obs}},\Theta^{(t)}) & \text{if } x_{ik} \text{ is missing,}\end{cases}$$

and

$$c_{ijk}^{(t)}=\begin{cases}0 & \text{if } x_{ik} \text{ or } x_{ij} \text{ is observed,}\\ \text{Cov}(x_{ik},x_{ij}|\mathbf{D}^{\mathbf{tr}},\boldsymbol{x}_i^{\text{obs}},\Theta^{(t)}) & \text{if } x_{ik} \text{ and } x_{ij} \text{ are missing.}\end{cases}$$

Missing values $x_{ik}$ are thus replaced by the conditional mean of $x_{ik}$ given the set of values $\boldsymbol{x}_i^{\text{obs}}$, available for that observation. These conditional means and the nonzero conditional covariances are easily found from the current parameter estimates by sweeping the augmented covariance matrix so that the variables $\boldsymbol{x}_i^{\text{obs}}$ are predictors in the regression equation and the remaining variables are outcome variables.

The M-STEP of the EM algorithm is straightforward and is a standard MLE process, i.e.,

$$\mu_k^{(t+1)}=\frac{1}{n}\sum_{i=1}^n x_{ik}^{(t)},\quad k=1,\ldots,K, \tag{4}$$

$$\sigma_{jk}^{(t+1)}=\frac{1}{n}E\left(\sum_{i=1}^n x_{ik}x_{ij}|\mathbf{D}^{\mathbf{tr}},\boldsymbol{x}_i^{\text{obs}}\right)-\mu_k^{(t+1)}\mu_j^{(t+1)}. \tag{5}$$

The algorithm will iterate repeatedly between the two steps until the difference between covariance matrices in subsequent M-STEPs falls below some specified convergence criterion. Although the classical EM algorithm will stop at this M-STEP, it is straightforward to get the imputed values by performing the E-STEP one more time, using the *sweep operator* [39] and the regression equations with $\boldsymbol{x}_i^{\text{obs}}$ as predictors.

### 4.4. Imputation via the GMM estimation

As mentioned earlier, the MLE method is based on the multivariate normal assumption to determine the form of the likelihood function and sufficient statistics. Although this assumption is mild, an obvious violation of normality often happens in biometrics because of the inherent discrimination between genuine and imposter scores.

Finite mixture models allow more flexibility, because they are not constrained to one specific functional form. As shown in Fraley and Raftery [20], many probability distributions can be well approximated by mixture models. At the same time, in contrast to non-parametric schemes, mixture models do not require a large number of observations to obtain a good estimate [41,21].

Let observations $\boldsymbol{x}_1,\ldots,\boldsymbol{x}_n$ be a random sample from a finite mixture model with $K$ underlying components in unknown proportions $\pi_1,\ldots,\pi_K$. Let the density of $\boldsymbol{x}_i$ in the $k$th component be $f_k(\boldsymbol{x}_i;\boldsymbol{\theta}_k)$, where $\boldsymbol{\theta}_k$ is the parameter vector for component $k$. In this case, $\Theta=(\pi_1,\ldots,\pi_K;\boldsymbol{\theta}_1,\ldots,\boldsymbol{\theta}_K)=(\boldsymbol{\pi},\boldsymbol{\theta})$, and the density of $\boldsymbol{x}_i$ can be written as:

$$f(\boldsymbol{x}_i;\Theta)=\sum_{k=1}^K \pi_k f_k(\boldsymbol{x}_i;\boldsymbol{\theta}_k),$$

where $\sum_{k=1}^K \pi_k=1,\pi_k\geq 0$, for $k=1,\ldots,K$.

Finite mixture models are frequently used when the component densities $f_k(\boldsymbol{x}_i;\boldsymbol{\theta}_k)$ are taken to be $p$-variate normal distributions $\boldsymbol{x}_i\sim N_p(\boldsymbol{\mu}_k,\Sigma_k)$, where the $i$th observation belongs to component $k$. This model has been studied by Titterington et al. [42], and by McLachlan & Basford [43]. Further details on the maximum likelihood estimates of the components of $\Theta$ can be found in McLachlan and Peel [44].

When Gaussian mixture models are used in imputation, two main steps will be essential: the density estimation using the

GMM assumption and the imputation itself based on this estimated density.

#### 4.4.1. Density estimation using GMM

The EM algorithm of Dempster et al. [39] is applied to the finite mixture model for density estimation. Let the vector of indicator variables, $\boldsymbol{z}_i=(z_{i1},\ldots,z_{iK})$, be defined by:

$$z_{ik}=\begin{cases}1 & \text{if observation } \ i\in \text{component } k,\\ 0 & \text{if observation } \ i\notin \text{component } k.\end{cases}$$

where $\boldsymbol{z}_i$, $i=1,\ldots,n$, are independently and identically distributed according to a multinomial distribution generated by a single trial of an experiment with $K$ mutually exclusive outcomes having probabilities $\pi_1,\ldots,\pi_K$.

Let $\hat{\Theta}$ denote the maximum likelihood estimate of $\Theta$. Then each observation, $\boldsymbol{x}_i$, can be allocated to component $k$ on the basis of the estimated posterior probabilities. The estimated posterior probability that observation $\boldsymbol{x}_i$, belongs to component $k$, is given by:

$$\hat{z}_{ik}=pr(\text{observation } i\in \text{component } k|\boldsymbol{x}_i;\hat{\Theta})=\frac{\hat{\pi}_k f_k(\boldsymbol{x}_i;\hat{\boldsymbol{\theta}}_k)}{\sum_{k=1}^{K}\hat{\pi}_k f_k(\boldsymbol{x}_i;\hat{\boldsymbol{\theta}}_k)}.$$

and $\boldsymbol{x}_i$ is assigned to component $k$ if:

$$\hat{z}_{ik}>\hat{z}_{ik'} \quad \text{for } k,k'=1,\ldots,K \quad k\neq k'.$$

The EM algorithm consists of defining an initial guess for the parameters to be estimated, and iteratively estimating the parameters until convergence of the expectation step (E-step) and the maximization step (M-step).

The E-step requires calculating the expectation of the log-likelihood of the complete data conditioned on the observed data and the current value of the parameters:

$$\hat{z}_{ik}=\hat{z}_{ik}^{(t)}=E(z_{ik}|\boldsymbol{x}_i^{\text{obs}};\Theta^{(t)})=\frac{\pi_k f_k(\boldsymbol{x}_i^{\text{obs}};\boldsymbol{\theta}_k^{(t)})}{\sum_{k=1}^{K}\pi_k f_k(\boldsymbol{x}_i^{\text{obs}};\boldsymbol{\theta}_k^{(t)})}.$$

That is, $z_{ik}$ is replaced by $\hat{z}_{ik}$, the estimate of the posterior probability that observation $i$ belongs to component $k$. With this estimate $\hat{z}_{ik}$, every component of our hypothetical complete data can be considered as a member of the regular exponential family with sufficient statistics:

$$\sum_{i=1}^{n} z_{ik}x_{ij} \quad \text{and} \quad \sum_{i=1}^{n} z_{ik}x_{ij}x_{ij'}, \quad j,j'=1,\ldots,K$$

So, the remaining calculations in the E-step are analogous to those required in the standard EM algorithm for incomplete normal data:

$$E(z_{ik}x_{ij}|\boldsymbol{x}_i^{\text{obs}};\boldsymbol{\theta}_k^{(t)})=\begin{cases}\hat{z}_{ik}x_{ij}, & x_{ij}\ \text{observed},\\ \hat{z}_{ik}E(x_{ij}|\boldsymbol{x}_i^{\text{obs}};\boldsymbol{\theta}_k^{(t)}), & x_{ij}\ \text{missing}.\end{cases}$$

$$E(z_{ik}x_{ij}^2|\boldsymbol{x}_i^{\text{obs}};\boldsymbol{\theta}_k^{(t)})=\begin{cases}\hat{z}_{ik}x_{ij}^2, & x_{ij}\ \text{observed},\\ \hat{z}_{ik}[(E(x_{ij}|\boldsymbol{x}_i^{\text{obs}};\boldsymbol{\theta}_k^{(t)}))^2+Var(x_{ij}|\boldsymbol{x}_i^{\text{obs}};\boldsymbol{\theta}_k^{(t)})], & x_{ij}\ \text{missing}.\end{cases}$$

For $j\neq j'$,

$$E(z_{ik}x_{ij}x_{ij'}|\boldsymbol{x}_i^{\text{obs}};\boldsymbol{\theta}_k^{(t)})=\begin{cases}\hat{z}_{ik}x_{ij}x_{ij'}, & x_{ij}\ \text{and}\ x_{ij'}\ \text{observed},\\ \hat{z}_{ik}x_{ij}E(x_{ij'}|\boldsymbol{x}_i^{\text{obs}};\boldsymbol{\theta}_k^{(t)}), & x_{ij}\ \text{observed},\ x_{ij'}\ \text{missing},\\ \hat{z}_{ik}E(x_{ij}|\boldsymbol{x}_i^{\text{obs}};\boldsymbol{\theta}_k^{(t)})x_{ij'}, & x_{ij'}\ \text{observed},x_{ij}\ \text{missing},\end{cases}$$

In summary, if any value of $\boldsymbol{x}_i$ is missing, it will be replaced by the conditional mean of the corresponding variable, given the set of values observed for that observation, $\boldsymbol{x}_i^{\text{obs}}$. When both $x_{ij}$ and $x_{ij'}$ are missing, the calculation will be:

$$E(z_{ik}x_{ij}x_{ij'}|\boldsymbol{x}_i^{\text{obs}};\boldsymbol{\theta}_k^{(t)})=\hat{z}_{ik}[E(x_{ij}|\boldsymbol{x}_i^{\text{obs}};\boldsymbol{\theta}_k^{(t)})E(x_{ij'}|\boldsymbol{x}_i^{\text{obs}};\boldsymbol{\theta}_k^{(t)})+Cor(x_{ij},x_{ij'}|\boldsymbol{x}_i^{\text{obs}};\boldsymbol{\theta}_k^{(t)})].$$

In the M-step of the algorithm, the new parameters $\boldsymbol{\theta}^{(t+1)}$ are estimated from the sufficient statistics of the complete data:

$$\hat{\pi}_k^{(t+1)}=\frac{1}{n}\sum_{i=1}^{n}\hat{z}_{ik}^{(t)} \quad \text{for } k=1,\ldots,K,$$

$$\hat{\mu}_{kj}^{(t+1)}=\frac{1}{n\hat{\pi}_k}E\left(\sum_{i=1}^{n}\hat{z}_{ik}^{(t)}x_{ij}|\boldsymbol{x}_i^{\text{obs}};\boldsymbol{\theta}_k^{(t)}\right),$$

$$\hat{\Sigma}_{kjj'}^{(t+1)}=\frac{1}{n\hat{\pi}_k}E\left(\sum_{i=1}^{n}\hat{z}_{ik}^{(t)}x_{ij}x_{ij'}|\boldsymbol{x}_i^{\text{obs}};\boldsymbol{\theta}_k^{(t)}\right)-\hat{\mu}_{kj}^{(t+1)}\hat{\mu}_{kj'}^{(t+1)}.$$

Although a mixture model has great flexibility in modeling, a restriction on the number of components $K$ is still required because, along with an increase in the number of parameters, the estimation of these parameters from the training data might imply a greater variance for each of the parameters. In this study, the Bayesian information criterion (BIC) [45] is employed. The BIC can be written as

$$BIC\equiv -2L(\hat{\Theta}|\boldsymbol{x}^{\text{obs}})+v_K\log(n_{tr}),$$

where $L(\hat{\Theta}|\boldsymbol{x}^{\text{obs}})$ is the maximized log-likelihood function given the observed data, $v_K$ is the number of parameters to be estimated in the assumed model, and $n_{tr}$ is the number of observations in training set. The target is to find that $v_K$ which minimizes BIC, and then a reasonable number of components $K$ is obtained.

#### 4.4.2. Two imputation methods via the GMM

With a reasonable density estimation method, various imputation schemes are possible. DiZio et al. [21] point out that for the preservation of the covariance structure, the Random Draw (RD) imputation process based on the GMM assumption (GMM-RD) is preferable over the conditional mean method (introduced by Nielsen [46]) based on the same model.

The estimates of the Gaussian mixture model parameters are obtained as:

$$f(\boldsymbol{x}_i;\Theta)=\sum_{k=1}^{K}\pi_{ik}N_p(\boldsymbol{x}_i;\boldsymbol{\mu}_k,\Sigma_k). \tag{6}$$

In practice, the random drawing of a value $\boldsymbol{x}_i^{\text{mis}}$ from the distribution of

$$f(\boldsymbol{x}_i^{\text{mis}}|\boldsymbol{x}_i^{\text{obs}};\Theta)=\sum_{k=1}^{K}\pi_{ik}N_p(\boldsymbol{x}_i^{\text{mis}}|\boldsymbol{x}_i^{\text{obs}};\Theta), \tag{7}$$

could be accomplished in two simple steps: First, draw a value $k$ from the multinomial distribution $Multi(1;\hat{\pi}_{i1},\ldots,\hat{\pi}_{iK})$; then, given $k$, generate a random value from the $p$-variate conditional Gaussian distribution $N_p(\boldsymbol{x}_i^{\text{mis}}|\boldsymbol{x}_i^{\text{obs}};\Theta)$ as the imputation of the missing value.

The KNN imputation process can also be used based on the GMM assumption (GMM-KNN). As a neighbor-based method, the main principle of the GMM-KNN method is to find the most similar vectors as "donors" in the training set. However, if the current training set is not large enough, only limited donors would be available, and this will reduce the accuracy of imputation. In the experiments, a larger simulated dataset ($n_{sim}=10n_{tr}$) based on the estimates of the mixture model parameters is used as the "imputation pool".

In this work, the Euclidean distance measurement $d$ is employed to find the "nearest" donors for incomplete score vectors. Recall that the distance measure $d$ between two observations $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ has been

defined in Eq. (3). The GMM-KNN scheme can be summarized in the following steps:

(1) Use the estimated parameters of GMM, $\Theta$, to simulate a dataset $\mathbf{D}^{\mathbf{sim}}$, having a larger size than $\mathbf{D}^{\mathbf{tr}}$.
(2) For each observation $\boldsymbol{x}_i$, apply the distance function $d$ to find $k=5$ nearest neighbors in the simulated set $\mathbf{D}^{\mathbf{sim}}$.
(3) The missing variables $\boldsymbol{x}_i^{\mathrm{mis}}$ are imputed by the average of corresponding variables from the nearest neighbors taken from $\mathbf{D}^{\mathbf{sim}}$.

### 4.5. Predictive mean matching (PMM)

Predictive mean matching (PMM) is an imputation scheme that combines some aspects of parametric and non-parametric imputation methods [47]. It imputes missing values by means of the neighbor-based schemes, where the distance is computed on the expected values of the missing variables conditioned on the observed variables, instead of directly on the values of the variables. In the PMM scheme, the expected values are computed through a linear regression model, such as in the MLE-MN scheme.

(1) The parameters of a multivariate normal distribution are estimated through the EM algorithm [27] using the training data which is complete.
(2) Based on the estimates from EM, for each incomplete score vector (recipient), predictions of the missing values are computed conditioned on the observed variables. These predictions are not directly used as imputation values; rather, they are used to compute the predictive means corresponding to the missing values.
(3) Each recipient is matched to the donor having the closest predictive mean with respect to the Mahalanobis distance, which is defined through the residual covariance matrix from the regression of the missing entries on the observed ones.
(4) Missing values are imputed to each recipient by transferring the corresponding values from its closest donor.

Although the imputation based on distance function is more robust than the imputation based on standard linear regression, the asymptotic properties of the neighbor-based methods are no longer guaranteed, because the measurement of distance used in PMM is not from a non-parametric model but from the results of a multivariate normal model. Thus, if the assumption is not appropriate, the performance is expected to be poor. Certain improved PMM methods that use more generalized model assumptions have been discussed in the literature [48].

### 4.6. Multiple imputation via MCMC

The primary shortcoming associated with the single imputation schemes discussed earlier – the inability to accommodate variability/uncertainty – can be attenuated by MI schemes. Proposed by Rubin [27], the MI scheme accounts for missing data by restoring not only the natural variability in the missing data, but by also incorporating the uncertainty caused by the estimation process. The general strategies of the MI scheme are as follows:

(1) Impute missing values using an appropriate model which can plausibly represent the data with random variation.
(2) Repeat this $m>1$ times to produce $m$ completed data sets.
(3) Perform the analysis of the complete data.
(4) Combine the results of these analysis to obtain overall estimates using Rubin's Rules [27].

There are some options for choosing the appropriate model in step (1). The Markov Chain Monte Carlo (MCMC) methods are used in this work. In statistical applications, MCMC is used to generate pseudo-random samples from multidimensional and otherwise intractable probability distributions via Markov chains. Data augmentation (DA), originated by Tanner and Wong [49], is a very effective tool for constructing deterministic models using the MCMC technique, when a multivariate normal distribution is assumed.

The DA algorithm starts with the construction of the so-called augmented data, $\boldsymbol{x}_i^{\mathrm{aug}}$, which are linked to the observed data via a many-to-one mapping $M:\boldsymbol{x}_i^{\mathrm{aug}}\rightarrow\boldsymbol{x}_i^{\mathrm{obs}}$. A data augmentation scheme is a model for $\boldsymbol{x}_i^{\mathrm{aug}}$, $p(\boldsymbol{x}^{\mathrm{aug}}|\theta)$, which satisfies the following constraint:

$$\int_{M(\boldsymbol{x}_i^{\mathrm{aug}})=\boldsymbol{x}_i^{\mathrm{obs}}} p(\boldsymbol{x}_i^{\mathrm{aug}}|\Theta)\mu(d\boldsymbol{x}_i^{\mathrm{aug}}) = p(\boldsymbol{x}_i^{\mathrm{obs}}|\Theta). \tag{8}$$

With an appropriate choice of $p(\boldsymbol{x}_i^{\mathrm{aug}}|\Theta)$, sampling from both $p(\Theta|\boldsymbol{x}_i^{\mathrm{aug}})$ and $p(\boldsymbol{x}_i^{\mathrm{aug}}|\boldsymbol{x}_i^{\mathrm{obs}},\Theta)$ is much easier than sampling directly from $p(\Theta|\boldsymbol{x}_i^{\mathrm{obs}})$. Consequently, starting with an initial value, $\Theta^{(0)}$, a Markov chain can be formed as $(\Theta^{(t)},\boldsymbol{x}^{aug,(t)}), t>1$ by iteratively drawing $\boldsymbol{x}_i^{aug,(t+1)}$ and $\Theta^{(t+1)}$ from $p(\boldsymbol{x}_i^{\mathrm{aug}}|\Theta^{(t)},\boldsymbol{x}_i^{\mathrm{obs}})$ and $p(\Theta|\boldsymbol{x}_i^{aug,(t+1)})$, respectively.

The two steps are iterated long enough for the results to be reliable for a multiply imputed data set [16]. The goal is to have the iterations converge to their stationary distribution and then to simulate an approximately independent draw of the missing values.

After obtaining $m$ imputed data sets, the overall estimates are computed using Rubin's Rules [27]: the overall estimate is the simple average of the $m$ estimates, and the overall estimate of the standard error is a combination of the within-imputation variability, $W$, and the between-imputation variability, $B$:

$$T = W + \left[\left(1+\frac{1}{m}\right)*B\right]. \tag{9}$$

### 4.7. Multivariate imputation by Chained Equations

Multivariate imputation by chained equations (MICE) is an attempt to combine the attractive aspects of two schemes, Regression-based imputation and Multiple imputation. A conditional distribution for the missing data corresponding to each incomplete variable is specified by MICE. For example, the distribution can be in the form of a linear regression of the incomplete variables given a set of predictors, which can also be incomplete. It is assumed that the joint distribution can be factored into marginal distributions and conditional distributions, and then iterative Gibbs sampling from the conditional distributions can generate samples from the joint distribution.

Recall the score matrix can be written as $\mathbf{D}=(X_1,\ldots,X_p)$, where each variable $X_j$ may be partially observed, with $j=1,\ldots,k$. The imputation problem requires us to draw from the underlying joint distribution of $\mathbf{D}$. Under the assumption that the nature of missing data is MAR, one may repeat the following sequence of Gibbs sampler iterations:

For $X_1$ : draw $X_1^{t+1}$ from $P(X_1|X_2^t,X_3^t,\ldots,X_k^t)$
For $X_2$ : draw $X_2^{t+1}$ from $P(X_2|X_1^{t+1},X_3^t,\ldots,X_k^t)$
...
For $X_p$ : draw $X_p^{t+1}$ from $P(X_p|X_2^{t+1},X_3^{t+1},\ldots,X_{p-1}^{t+1})$

Rubin and Schafer [25] show that if $\mathbf{D}$ is multivariate normal, then iterating linear regression models like $X_1=X_2^t\beta_{12}+X_3^t\beta_{13}+\cdots+X_p^t\beta_{1p}+\varepsilon_1$ with $\varepsilon_1\sim N(0,\sigma_1^2)$ will produce a random draw

from the desired distribution. In this way, the multivariate problem is split into a series of univariate problems.

## 5. Results and conclusion

### 5.1. MLE-MN vs. PMM

Both the MLE-MN and PMM schemes employ the estimates obtained using the EM algorithm under the assumption of a multivariate normal distribution. Although both of them belong to the parametric class of methods, the imputation processes are different. The MLE-MN scheme uses the regression coefficients from the final M-Step of the EM algorithm to compute the imputed values, while the PMM scheme relies on a distance function to find nearest neighbors.

In Fig. 3, the density plots after using the MLE-MN and the PMM scheme demonstrate that PMM schemes are likely to generate longer "tails" for both genuine scores and imposter

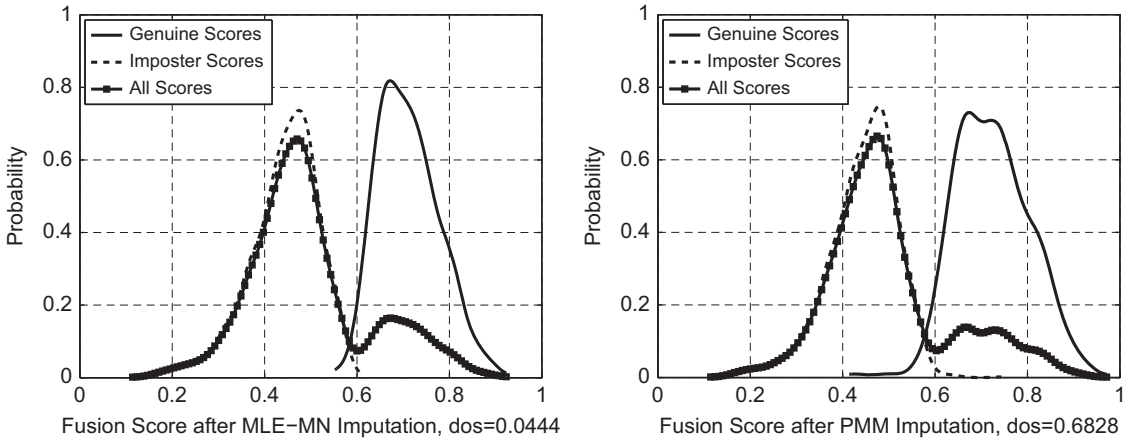


**Fig. 3.** Comparison of density plots after using the MLE-MN and the PMM scheme at training rate: 50%; missing rate: 10%.

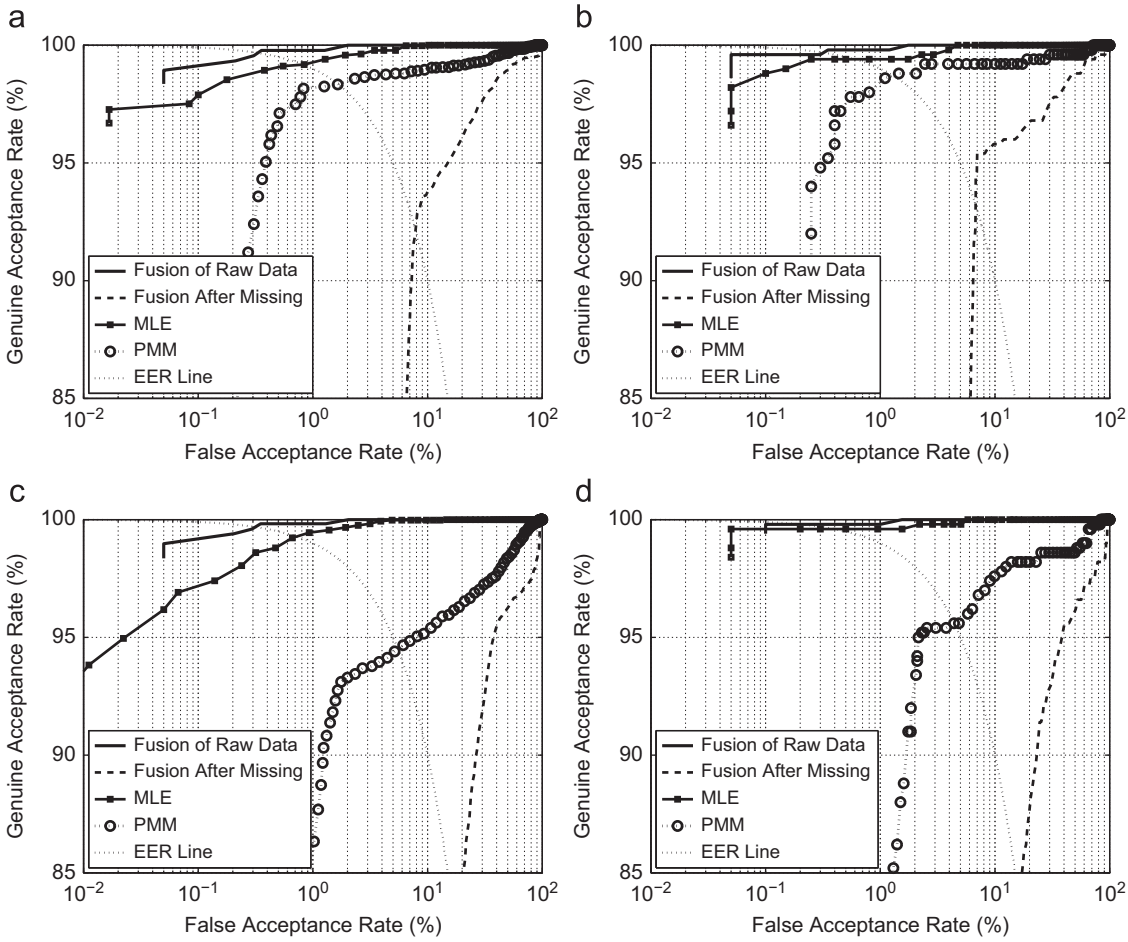


**Fig. 4.** Comparison between the MLE-MN and the PMM scheme at multiple training set sizes and missing rates. (a) Training set: 10%; missing rate: 10%, (b) training set: 50%; missing rate: 10%, (c) training set: 10%; missing rate: 25%, (d) training set: 50%; missing rate: 25%.

scores. So the overlap area between the two classes sharply increases leading to a much inferior recognition performance. Here, the degree of separation dos is defined as follows:

$$\mathtt{dos} = \frac{\mathtt{Number\ of\ vectors\ whose\ scores\ are\ within\ the\ overlap}}{\mathtt{Number\ of\ all\ score\ vectors}}.$$

According to the definition, the smaller the dos is, the better the performance.

A similar observation can be made from the ROC curves in Fig. 4. Here, the fusion performance with the original data (labeled as "Fusion of Raw Data") and after generating the missing data (labeled as "Fusion After Missing") are used as baselines in the comparison. It is evident that the regression-based method performs better than the neighbor-based method under the same density model (multivariate normal). However, the fusion performance after MLE imputation is lower compared to that of the raw complete dataset. So a multivariate normality test was conducted to determine whether the training set (at 50% training rate) conforms to a multivariate normal distribution. The E-statistic (energy) test was used for this purpose. The *p*-value

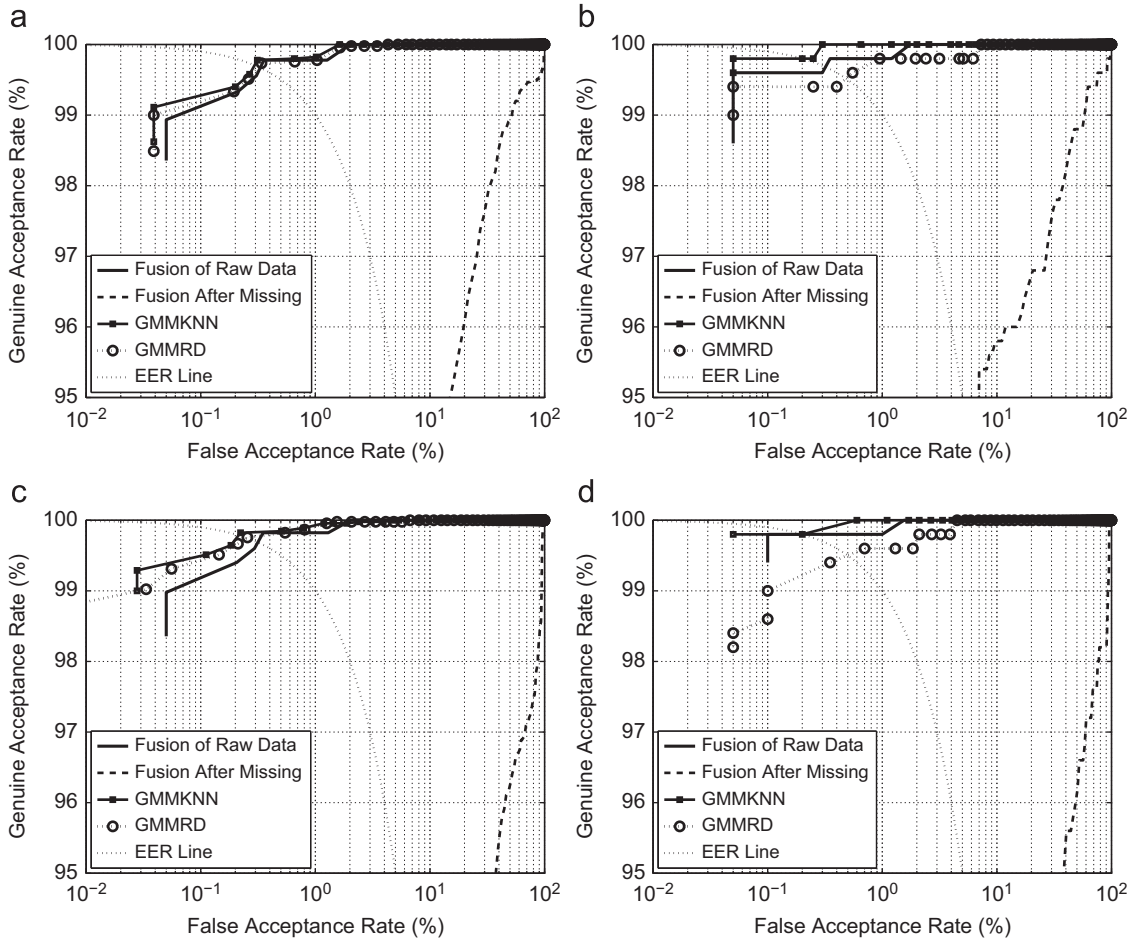


**Fig. 5.** Comparison between the GMM-RD and the GMM-KNN schemes at multiple training set sizes and missing rates. (a) Training set: 10%; missing rate: 10%, (b) training set: 50%; missing rate: 10%, (c) training set: 10%; missing rate: 25%, (d) training set: 50%; missing rate: 25%.

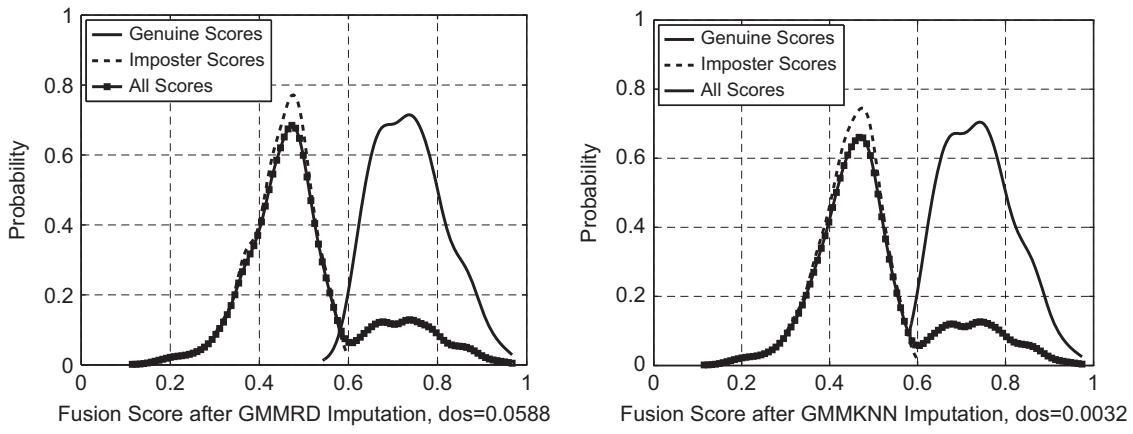


**Fig. 6.** Comparison of density plots after using the GMM-RD and the GMM-KNN schemes with training set: 50%; missing rate: 10%.

indicated that the data were not normally distributed. This violation of normality in the training set results in the obvious decrease in performance when using simple parametric density models.

Comparing (c) and (d) in Fig. 4, it can be observed that utilizing a larger size training set can reduce the impact on fusion performance brought about by an increasing missing rate.

### 5.2. GMM-RD vs. GMM-KNN

From Fig. 5, both the GMM-RD and GMM-KNN schemes show comparable performance with that of the raw complete dataset. This can also be observed in Fig. 6. Both schemes have similar looking density plots since they use the accurate parameters estimated under the GMM assumption.

There is still some subtle differences in the fusion performance due to the different imputation processes. In Fig. 6, it is observed that the GMM-KNN imputation scheme results in a decreased overlapping area between the two classes. One possible conclusion is that a good density model positively impacts the neighbor-based imputation process.

Additionally, GMM-KNN shows consistently better performance than the sampling-based process, and the difference becomes more noticeable when the training set is large (in (b) and (d) of Fig. 5). The possible reason has to do with the nuances of the imputation process of random sampling. Recall the value $k$ which was drawn from $Multi(1;\hat{\pi}_{i1},\ldots,\hat{\pi}_{iK})$, that played a critical role in the process, because the final imputed value depended upon the component that was chosen. A slightly biased value for $k$ will cause an enormous deviation from the true distribution corresponding to the missing score. In contrast, the GMM-KNN scheme does not rely on the value $k$, but uses the distance corresponding to the observed part to choose the "closest" neighbors in the simulated data.

### 5.3. PMM vs. KNN vs. GMM-KNN

The PMM, KNN and GMM-KNN schemes are all neighbor-based methods, but the model assumptions are different. The PMM scheme assumes a multivariate normal distribution of the complete data, while the GMM-KNN scheme assumes a Gaussian mixture model. The KNN scheme is a non-parametric method with no strict model assumptions.

From Fig. 7, it is observed that the KNN method based on the GMM assumption offers the best performance among the three schemes at multiple training set sizes and missing rates.

From Fig. 8, it is observed that the KNN scheme did not retain the shape of the genuine scores very well. The overlap area does not decrease because both the tails of the genuine scores and imposter scores are prolonged. Due to the presence of multiple peaks in the score distribution, the GMM is a better choice for modeling.

### 5.4. MI via MCMC vs. MICE

Fig. 9 demonstrates that the multiple imputation via MCMC scheme provides a much better performance than the MICE

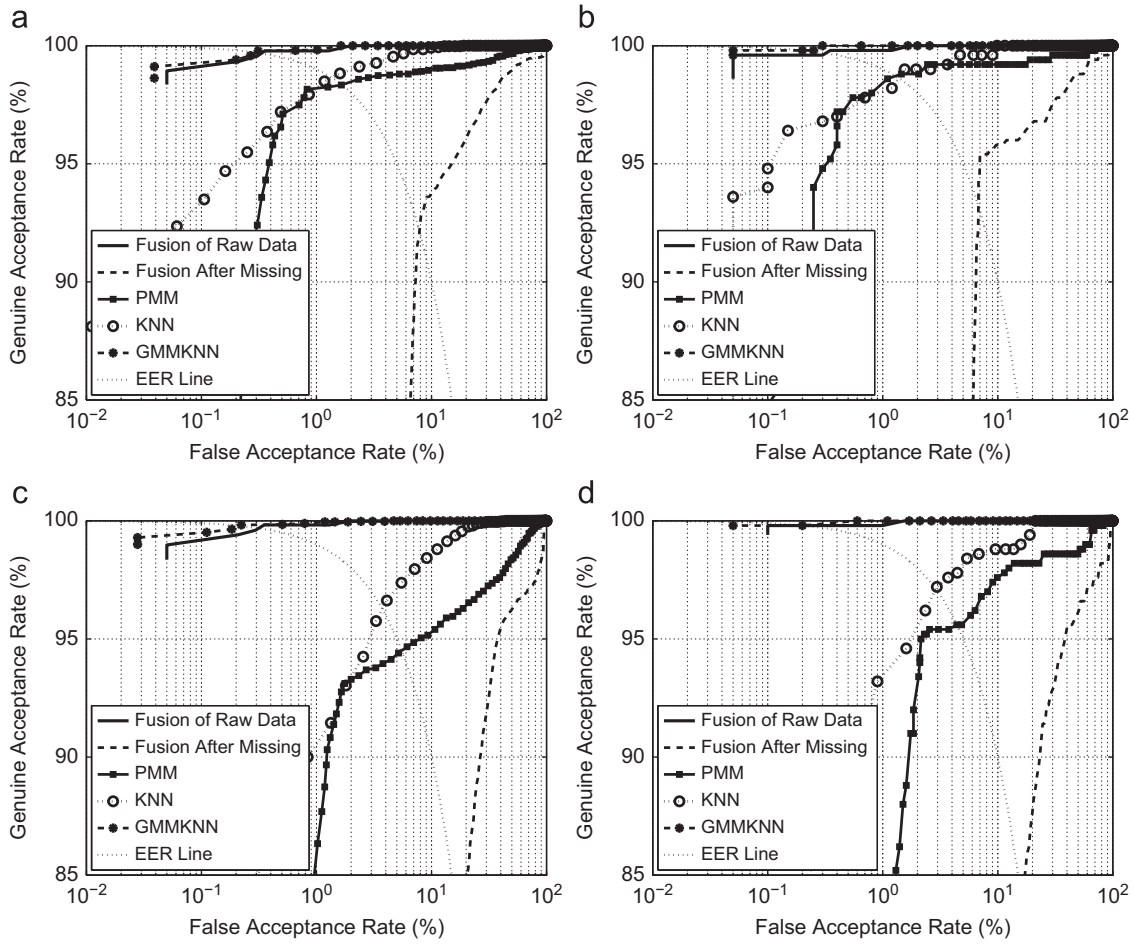


**Fig. 7.** Comparison between the PMM, KNN and GMM-KNN schemes at multiple training set sizes and missing rates. (a) Training set: 10%; missing rate: 10%, (b) training set: 50%; missing rate: 10%, (c) training set: 10%; missing rate: 25%, (d) training set: 50%; missing rate: 25%.

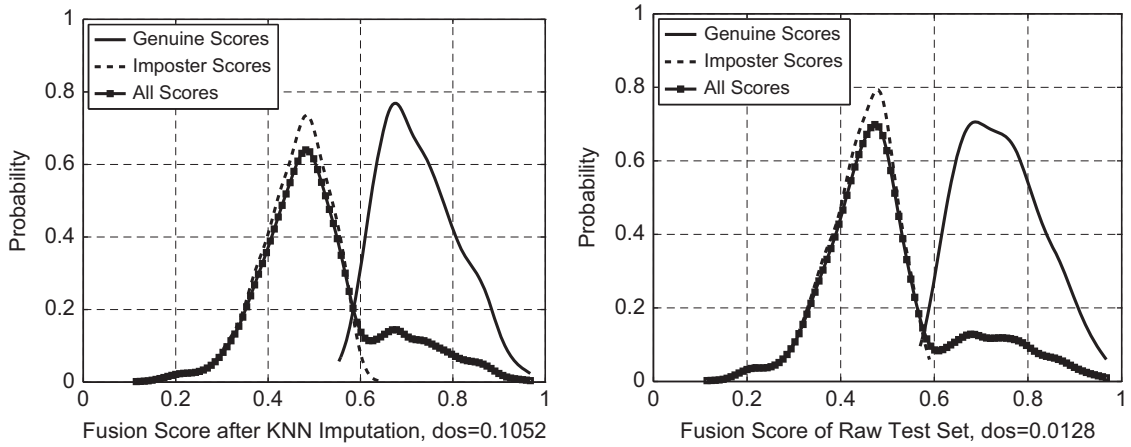


**Fig. 8.** Comparison of density plots between the raw test set and the imputation set after using the KNN scheme at training set: 50%; missing rate: 10%.

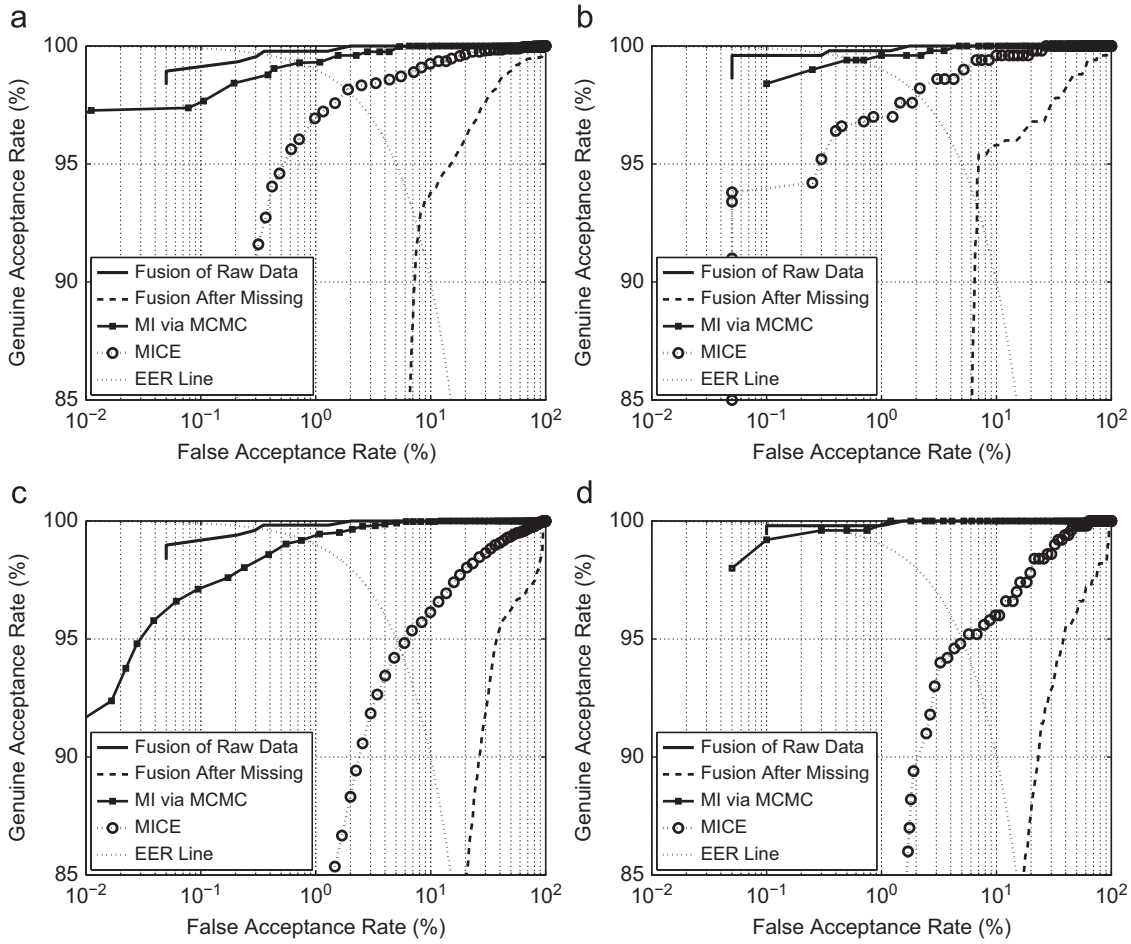


**Fig. 9.** Comparison between the MI via MCMC and the MICE schemes at multiple training set sizes and missing rates. (a) Training set: 10%; missing rate: 10%, (b) training set: 50%; missing rate: 10%, (c) training set: 10%; missing rate: 25%, (d) training set: 50%; missing rate: 25%.

scheme, even when the missing rate is high in (d). From Fig. 10, it is observed that the density plots after applying the MI via MCMC scheme has a very similar shape as that of the raw test set.

The squared residuals between the original fusion scores (the sample) and imputed scores (the estimated values) are computed for the MI via MCMC scheme and the GMM-KNN scheme. The sum of squared residuals of the MI via MCMC is less than that of the GMM-KNN. This indicates that the imputed scores from the MI via MCMC scheme are closer to the missing scores. However, the overall recognition accuracy is more critical than the similarity between the missing values and the imputed values. Both ROC curves and `dos` values indicate that a better recognition performance is offered by the GMM-KNN scheme.

Both MICE and MLE-MN schemes employ the parametric model combined with regression-based imputation, and cannot

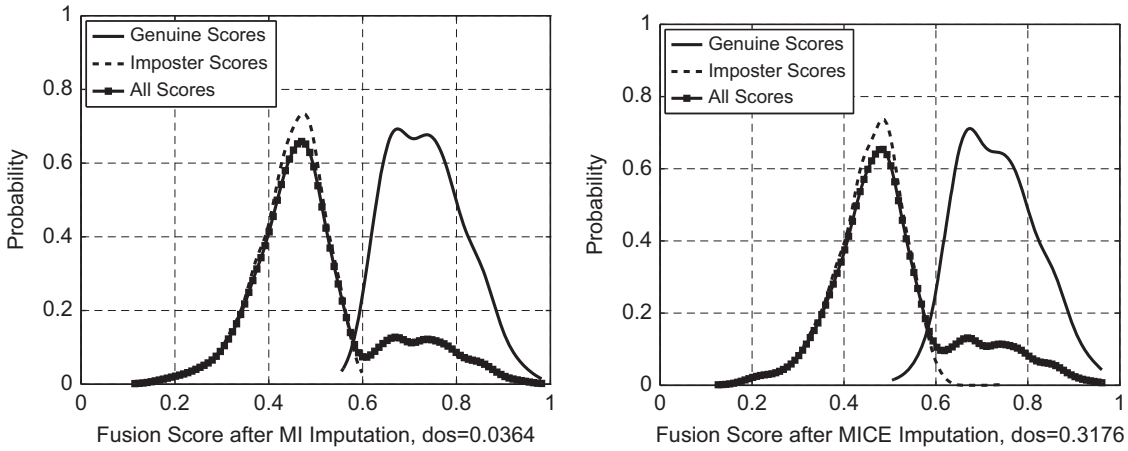


**Fig. 10.** Comparison of density plots after using the MI via MCMC and the MICE schemes with training set: 50%; missing rate: 10%.

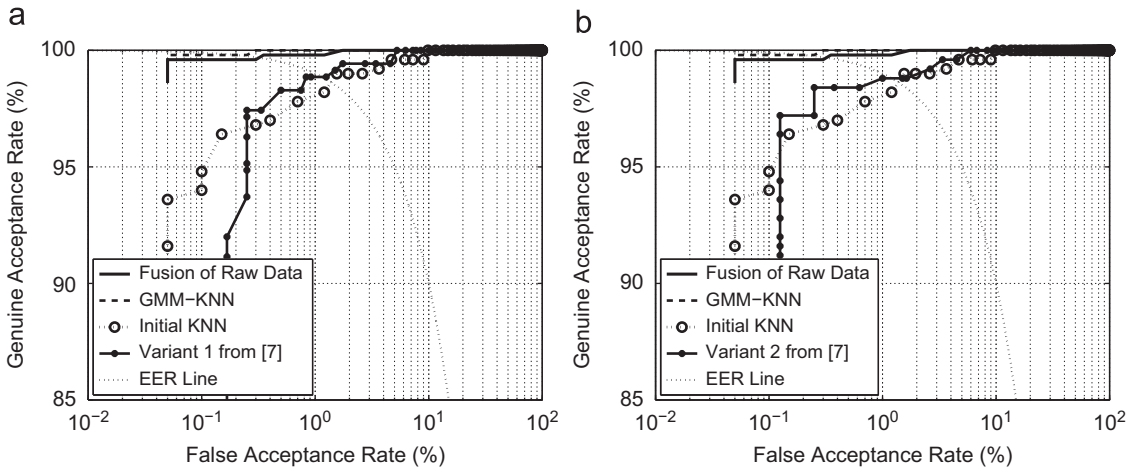


**Fig. 11.** Comparison between the GMM-KNN and two variants of KNN schemes proposed by Fatukasi et al. [7], when the training set size is 50% and missing rate is 10%. (a) Comparison with variant 1. (b) Comparison with variant 2.

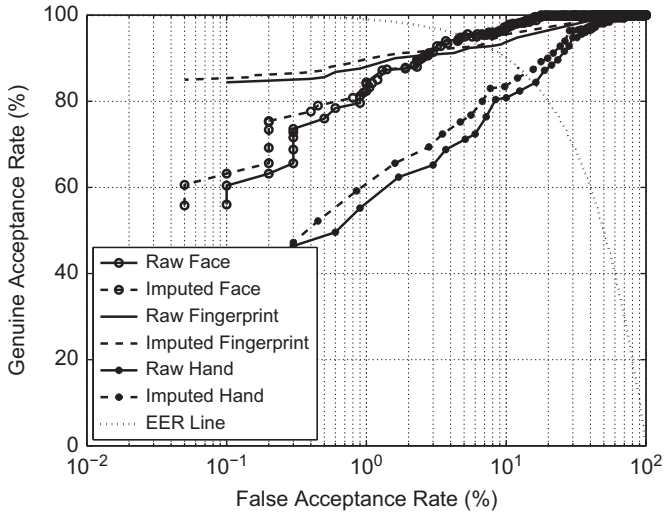


**Fig. 12.** Performances due to the original raw dataset and after applying the GMM-KNN scheme on the individual modalities when the training set size is 50% and missing rate is 10%.

yield a good imputed value while retaining the classification accuracy. The MICE, which averages the multiple imputed values together, is observed to result in the worst performance. This is because the averaging process can only reduce the variance of estimation, but cannot reduce the bias.

### 5.5. Comparing GMM-KNN with existing schemes

Fatukasi et al. [7] compared various imputation methods including zero imputation, mean imputation and three different variants of the KNN schemes. In Fig. 11, the GMM-KNN scheme proposed in our work is compared with the best two variants reported by Fatukasi et al.

It can be observed that, although both the variants of Fatukasi et al. provide an improved performance over the original KNN scheme, the proposed GMM-KNN scheme gives the highest performance on the MSU dataset. Since the GMM-KNN scheme uses a larger size dataset $\mathbf{D}^{\mathbf{sim}}$ that is synthetically generated for the donor pool, it has a better chance of selecting a "closer" nearest neighbor.

In Fig. 12, the performance of individual modalities are summarized using ROC curves. The GMM-KNN method delivers a low EER after fusion whilst simultaneously retaining the original shape of the ROC curves of individual modalities. It can also be noticed that among the three different modalities, the fingerprint is the most robust at various missing rates (not shown in this figure). Therefore, assigning a comparatively larger weight parameter to the fingerprint during fusion can increase the robustness of handling missing data for this dataset.

## 6. Summary

The results in the previous sections indicate that there are certain powerful imputation schemes, which can sustain the fusion performance at a high level when the missing rates are small. Specifically, the GMM-based scheme performs better than the other models, because it seems to capture the natural structure of the raw dataset. Further, under the GMM assumption, the non-parametric imputation process is preferred over sampling-based schemes. The experiments also indicate that utilizing a larger training set can mitigate the negative impact on performance at higher missing rates. On the other hand, there are some imputation schemes whose performance is not comparable to that of the raw (original) dataset; in fact a few of them such as PMM result in worse fusion performance than that of a single modality.

In the future, the robustness of the assumptions made for every scheme will be further analyzed. This is expected to offer additional guidance on how to choose appropriate imputation methods for a particular dataset. Also, we are looking at ways to combine the process of imputation with score normalization and fusion. Finally, these experiments will be repeated on large operational data sets using different fusion rules.

## Acknowledgments

This work was supported by the NSF Center for Identification Technology Research (CITeR). The authors would like to thank Dr. Anil Jain at Michigan State University for granting us access to their database. Thanks to the reviewers for their valuable comments which significantly improved the experimental analysis.

## References

[1] A.K. Jain, P. Flynn, A.A. Ross, Handbook of Biometrics, Springer, 2008.
[2] A. Jain, K. Nandakumar, A. Ross, Score normalization in multimodal biometric systems, Pattern Recognition 38 (12) (2005) 2270–2285.
[3] A.A. Ross, K. Nandakumar, A.K. Jain, Handbook of Multibiometrics, Springer, Secaucus, NJ, USA, 2006.
[4] G. King, J. Honaker, A. Joseph, K. Scheve, Analyzing incomplete political science data: an alternative algorithm for multiple imputation, American Political Science Review 95 (2001) 49–69.
[5] R. Brown, Efficacy of the indirect approach for estimating structural equation models with missing data: a comparison of five methods, Structural Equation Modeling 1 (1994) 287–316.
[6] J. Graham, S. Hofer, D. MacKinnon, Maximizing the usefulness of data obtained with planned missing value patterns: an application of maximum likelihood procedures, Multivariate Behavioral Research 31 (1996) 197–218.
[7] O. Fatukasi, J. Kittler, N. Poh, Estimation of missing values in multimodal biometric fusion, in: IEEE International Conference on Biometrics: Theory, Applications and Systems (BTAS), 2008.
[8] J. Quinlan, Induction of decision trees, Machine Learning 1 (1986) 81–106.
[9] O.O. Lobo, M. Noneao, Ordered estimation of missing values for propositional learning, Journal of the Japanese Society for Artificial Intelligence 1 (2000) 499–503.
[10] H.F. Friedman, R. Kohavi, Y. Yun, Lazy decision trees, in: The 13th National Conference on Artificial Intelligence, 1996, pp. 717–724.
[11] J. Schafer, J. Graham, Missing data: our view of the state of the art, Psychological Methods.
[12] R.J.A. Little, D.B. Rubin, Statistical Analysis with Missing Data, first ed., Wiley, New York, 1987.
[13] N. Friedman, D. Geiger, M. Goldszmidt, Bayesian network classifiers, Machine Learning 29 (2–3) (1997) 131–163.
[14] S.F. Buck, A method of estimation of missing values in multivariate data suitable for use with an electronic computer, Journal of the Royal Statistical Society 22 (2) (1960) 302–306.
[15] R.J.A. Little, Regression with missing x's: a review, Journal of the American Statistical Association 87 (420) (1992) 1227–1237.
[16] J. Schafer, Analysis of incomplete multivariate data, Chapman & Hall/CRC, London, 1997.
[17] J.K. Dixon, Pattern recognition with partly missing data, IEEE Transactions on Systems, Man and Cybernetics 9 (10) (1979) 617–621.
[18] G. Kalton, D. Kasprzyk, The treatment of missing survey data, Survey Methodology 1 (12) (1986) 1–16.
[19] M. Bramer, R. Ellis, M. Petridis, A kernel extension to handle missing data, Research and Development in Intelligent Systems 26 (2009) 165–178.
[20] C. Fraley, A. Raftery, Model-based clustering, discriminant analysis, and density estimation, Journal of the American Statistical Association 97 (2002) 611–631.
[21] M. Di Zio, U. Guarnera, O. Luzi, Imputation through finite gaussian mixture models, Computational Statistics and Data Analysis 51 (11) (2007) 5305–5316.
[22] J.Q. Li, A.R. Barron, Mixture density estimation, Advances in Neural Information Processing Systems, vol. 12, MIT Press, 1999, pp. 279–285.
[23] B. Stef Van, O. Karin (CGM), Multivariate Imputation by Chained Equations. MICE V1.0 User's manual, Vol. PG/VGZ/00.038., TNO Prevention and Health, Leiden, 2000.
[24] T.E. Raghunathan, J.M. Lepkowski, J.V. Hoewyk, P. Solenberger, A multivariate technique for multiply imputing missing values using a sequence of regression models, Survey Methodology 27 (2001) 85–95.
[25] D. Rubin, J. Schafer, Efficiently creating multiple imputations for incomplete multivariate normal data, Proceedings of the Statistical Computing Section (1990) 83–88.
[26] D.B. Rubin, N. Schenker, Multiple imputation for interval estimation from simple random samples with ignorable nonresponse, Journal of the American Statistical Association 81 (1986) 366–374.
[27] D.B. Rubin, Multiple imputation for nonresponse in surveys, Wiley, 1987.
[28] J. Schafer, J. Schenker, Inference with imputed conditional means, Journal of the American Statistical Association 95 (2000) 144–154.
[29] M. Ramoni, P. Sebastiani, Robust learning with missing data, Machine Learning 45 (2) (2001) 147–170.
[30] K. Nandakumar, A.K. Jain, A. Ross, Fusion in multibiometric identification systems: what about the missing data?, in: IEEE/IAPR International Conference on Biometrics, Springer, 2009, pp. 743–752.
[31] N. Poh, D. Windridge, V. Mottl, A. Tatarchuk, A. Eliseyev, Addressing missing values in kernel-based multimodal biometric fusion using neutral point substitution, IEEE Transactions on Information Forensics and Security 5 (3) (2010) 461–469.
[32] Y. Ding, A. Ross, When data goes missing: methods for missing score imputation in biometric fusion, in: Proceedings of SPIE Conference on Biometric Technology for Human Identification VII, vol. 7667, 2010.
[33] A. Ross, A. Jain, Information fusion in biometrics, Pattern Recognition 13 (2003) 2115–2125.
[34] E. Byon, A. Shrivastava, Y. Ding, A classification procedure for highly imbalanced class sizes, IIE Transactions 4 (42) (2010) 288–303.
[35] S. Oba, M. Sato, I. Takemasa, M. Monden, K. Matsubara, S. Ishii, A bayesian missing value estimation method for gene expression profile data, Bioinformatics 19 (16) (2003) 2088–2096.
[36] D. Rubin, Inference and missing data, Biometrika 63 (1976) 581–592.
[37] R. Singh, M. Vatsa, A. Ross, A. Noore, Online learning in biometrics: a case study in face classifier update, in: IEEE 3rd International Conference on Biometrics: Theory, Applications, and Systems, 2009. BTAS '09, 2009, pp. 1–6.
[38] D.A. Marker, D.R. Judkins, M. Winglee, Large-scale imputation for complex surveys, in: Survey Nonresponse, John Wiley and Sons, 1999.
[39] A. Dempster, N. Laird, D. Rubin, Maximum likelihood from incomplete data via the em algorithm, Journal of the Royal Statistical Society. Series B (Methodological) 1 (39) (1977) 1–38.
[40] G. McLachlan, T. Krishnan, The EM algorithm and extensions, Wiley New York, 1996.
[41] C. Priebe, Adaptive mixtures, Journal of the American Statistical Association 89 (1994) 796–806.
[42] D. Titterington, A. Smith, U. Makov, Statistical Analysis of Finite Mixture Distributions, Wiley, New York, 1985.
[43] G. McLachlan, K. Basford, Mixture models: inference and applications to clustering, Marcel-Dekker, New York, 1988.
[44] G. McLachlan, D. Peel, Finite Mixture Models, Wiley, New York, 2000.
[45] G. Schwarz, Estimating the dimension of a model, The Annals of Statistics 6 (2) (1978) 461–464.
[46] S. Nielsen, Nonparametric conditional mean imputation, Journal of Statistical Planning and Inference 99 (2001) 129–150.
[47] R. Little, Missing data adjustments in large surveys, Journal of Business and Economic Statistics 6 (3) (1988) 287–296.
[48] G. Durrant, C. Skinner, Missing data methods to correct for measurement error in a distribution function, Survey Methodology 32 (1) (2006) 25–36.
[49] M. Tanner, W. Wong, The calculation of posterior distributions by data augmentatio, Journal of the American Statistical Association 82 (398) (1987) 528–540.

**Yaohui Ding** received the B.S. degree in Electrical Engineering from Zhejiang University, China, in August 2003. He received the M.S. degree in Statistics from West Virginia University, USA, in August 2010. He is currently pursuing the Ph.D. degree in the Lane Department of Computer Science and Electrical Engineering at West Virginia University. His current research interest is biometrics fusion.

**Arun Ross** received the B.E. (Hons.) degree in Computer Science from the Birla Institute of Technology and Science, Pilani, India, in 1996, and the M.S. and Ph.D. degrees in Computer Science and Engineering from Michigan State University, East Lansing, in 1999 and 2003, respectively. Between 1996 and 1997, he was with the Design and Development Group of Tata Elxsi (India) Ltd., Bangalore, India. He also spent three summers (2000–2002) with the Imaging and Visualization Group of Siemens Corporate Research, Inc., Princeton, NJ, working on fingerprint recognition algorithms. He is currently a Robert C. Byrd Associate Professor in the Lane Department of Computer Science and Electrical Engineering, West Virginia University, Morgantown. His research interests include pattern recognition, classifier fusion, machine learning, computer vision, and biometrics. He is actively involved in the development of biometrics and pattern recognition curricula at West Virginia University. He is the coauthor of Handbook of Multibiometrics and co-editor of Handbook of Biometrics. Arun is a recipient of NSF's CAREER Award and was designated a Kavli Frontier Fellow by the National Academy of Sciences in 2006. He is an Associate Editor of the IEEE Transactions on Image Processing and the IEEE Transactions on Information Forensics and Security.