A Joint Optimization Framework Integrated with Biological Knowledge for Clustering Incomplete Gene Expression Data

Dan Li (ldan@dlut.edu.cn)
Dalian University of Technology Faculty of Electronic Information and Electrical Engineering
https://orcid.org/0000-0002-6363-820X
Hong Gu
Dalian University of Technology Faculty of Electronic Information and Electrical Engineering
Qiaozhen Chang
Dalian University of Technology Faculty of Electronic Information and Electrical Engineering
Jia Wang
The Second Hospital of Dalian Medical University
Pan Qin
Dalian University of Technology Faculty of Electronic Information and Electrical Engineering

Research Article

Keywords: Gene clustering, Joint optimization, Multi-objective clustering, Imputation, Gene ontology

Posted Date: November 30th, 2021

DOI: https://doi.org/10.21203/rs.3.rs-1087790/v1

A joint optimization framework integrated with biological knowledge for clustering incomplete gene expression data

Dan Li[1] · Hong Gu[1] · Qiaozhen Chang[1] · Jia Wang[2] · Pan Qin[1*]

[1] Faculty of Electronic Information and Electrical Engineering, Dalian University of Technology, Dalian 116024, China
[2] Department of Breast Surgery, Second Hospital of Dalian Medical University, Dalian 116023, China

*Corresponding author: qp112cn@dlut.edu.cn

Abstract

Clustering algorithms have been successfully applied to identify co-expressed gene groups from gene expression data. Missing values often occur in gene expression data, which presents a challenge for gene clustering. When partitioning incomplete gene expression data into co-expressed gene groups, missing value imputation and clustering are generally performed as two separate processes. These two-stage methods are likely to result in unsuitable imputation values for clustering task and unsatisfying clustering performance. This paper proposes a multi-objective joint optimization framework for clustering incomplete gene expression data that addresses this problem. The proposed framework can impute the missing expression values under the guidance of clustering, and therefore realize the synergistic improvement of imputation and clustering. In addition, gene expression similarity and gene semantic similarity extracted from the Gene Ontology are combined, as the form of functional neighbor interval for each missing expression value, to provide reasonable constraints for the joint optimization framework. Experiments on several benchmark data sets confirm the effectiveness of the proposed framework.

Keywords Gene clustering · Joint optimization · Multi-objective clustering · Imputation · Gene ontology

1 Introduction

The recent development of biological experiments has generated vast amounts of gene expression data. Thus, extracting the intrinsic patterns from the enormous number of genes has become a significant challenge. Clustering, as an essential unsupervised data mining method, is often applied to analyze gene expression data. One of the major tasks in gene expression data clustering is to identify co-expression gene groups, which can provide a useful basis for the further investigation of gene function and gene regulation in the field of functional genomics (Chen et al. 2019; Giri and Sara 2020; Maulik et al. 2009; Sara et al. 2013; Stegmayer et al. 2012).

Biological experiments inevitably generate data with missing values in acquiring gene expression data, which adversely affects the clustering analysis (Moorthy et al. 2014; Yu et al. 2017). A straightforward way is to discard the genes with missing components and perform clustering on the remained complete matrix. The discarded genes cannot be partitioned for further analysis and would cause information loss. For this reason, various imputation strategies have been proposed for incomplete microarray data. The typical approaches are to use the global or local information from within the expression data to fill up the missing values (Acurna and Rodriguez 2004; Buuren and Oudshoorn 2011; Kim et al. 2005; de Souto et al. 2015; Oba et al. 2003; Troyanskaya et al. 2001; Yu et al. 2017). Besides, the prior biological knowledge, like Gene Ontology (GO) (Ashburner et al. 2000) has been applied to the imputation of missing expression values (Tuikkala et al. 2006), and the use of domain knowledge is beneficial to improve the imputation accuracy beyond the purely data-driven approaches (Moorthy et al. 2014). When identifying co-expressed

genes from incomplete gene expression data, imputation approaches often act as an important preprocessing step. Then clustering techniques can be applied to the recovered gene expression data (de Souto et al. 2015; Kim et al. 2005; Moorthy et al. 2014; Sara et al. 2013; Yu et al. 2017). These two-stage methods are quite popular and simple to implement. However, missing value imputation and gene clustering are inherently related, as both tasks use the correlation information within the data. These two-stage methods prevent the collaborative optimization of the two learning processes, thus may lead to imputation values that are unsuitable for clustering and unsatisfying clustering performance.

In this paper, we focus on the problem of clustering incomplete gene expression data. To address the drawbacks of the two-stage methods, we propose a novel clustering method called MOC-FNI (multi-objective clustering algorithm based on functional neighbor interval) to integrate the imputation and gene clustering tasks into one joint optimization framework. In MOC-FNI, functional neighbor intervals for missing expression values are constructed based on the combination of gene expression similarity and GO semantic similarity, which act as interval constraints in the joint optimization. The contributions of the proposed method can be summarized as follows:

(1) The proposed functional neighbor intervals can introduce multi-source information, including gene expression information and GO semantic information, which are beneficial to provide reasonable constraints to guide the optimization process.

(2) MOC-FNI can realize synergistic improvement of imputation and clustering in the framework of nondominated sorting GA-II (NSGA-II) (Deb et al. 2002). Therefore, with the constraints of functional neighbor intervals, MOC-FNI can obtain imputations guided by both multi-source information and cluster validity, as well as gene clusters based on meaningful and reasonable imputations.

The rest of this paper is organized as follows. Section 2 gives a brief review of related works on the imputation and multi-objective clustering of gene expression data. Section 3 presents the proposed MOC-FNI method. Experiments on several bench-mark data sets are shown in Section 4. Conclusions are summarized in the last section.

2 Related works

Microarray data often contain missing values. The leading causes include hybridization failures, artifacts on the microarray, image noise and corruption, etc (Song et al. 2014; Tuikkala et al. 2006; Yu et al. 2017). Owing to the economic and experimental limitations, it is not always practical to repeat the experiments. Thus, many efforts have been devoted to exploring the missing value imputation, which makes it possible to realize a reliable analysis of incomplete gene expression data.

In most cases, the missing values in incomplete gene expression data were imputed based on the global or local statistical information of gene expression data. A simple and commonly used method was the mean imputation (MEANimpute) (Acurna and Rodriguez 2004), where each missing value was filled in with the average of the corresponding attribute values of all genes with complete expression values. Considering the similarity between neighboring genes, Troyanskaya et al. (2001) applied the k-nearest neighbors rule to the imputation problem. In the KNNimpute method (Troyanskaya et al. 2001), for each target gene with missing components, k complete neighboring genes were first found and the missing value was filled in with the weighted average of the corresponding attribute values of these genes. In the singular value decomposition (SVD) imputation (SVDimpute) (Troyanskaya et al. 2001), the SVD was employed to obtain a set of mutually orthogonal expression patterns, called eigengenes. Then, several most significant eigengenes were selected and their linear combination were used to reconstruct the missing values. It was claimed that KNNimpute was superior to SVDimpute in accuracy and robustness (Troyanskaya et al. 2001). Bayesian principal component analysis (BPCA) (de Souto et al. 2015; Oba et al. 2003) was another widely

used global imputation method, which involved principal component regression to estimate the missing part in the expression vector, Bayesian estimation and expectation-maximization like repetitive algorithm to estimate the posterior distributions for the model parameter. For the data with a global covariance structure, BPCA always outperformed KNNimpute and SVDimpute methods (Oba et al. 2003; de Souto et al. 2015; Yu et al. 2017). Besides, local least squares imputation (LLSimpute) (Kim et al. 2005) introduced a multiple linear regression model to impute each missing value based on neighboring genes. Based on LLSimpute, Yu et al. (2017) considered the different importance of the target gene's neighbors, and proposed an iterative locally auto-weighted least squares method. Another notable approach was the multivariate imputation by chained equations (MICE) (Buuren and Oudshoorn 2011), whose aim was to produce multiple imputations and integrate these results to fill in the missing values.

In recent years, knowledge-assisted methods had gradually become a research hotspot in gene analysis. Gene Ontology (GO) (Ashburner et al. 2000) is one of the most widely used publicly available knowledge bases, which describes biological annotations in the form of directed acyclic graphs. GO contains three dynamic sub-ontologies, and consists of a set of terms related to biological process (BP), cellular component (CC), and molecular function (MF) along with relations between terms. For the imputation problem of gene expression data, a popular way to apply GO annotation information was to calculate the semantic similarity of GO terms and that of genes. Tuikkala et al. (2006) proposed to integrate GO annotations into the KNN imputation algorithm (GOimpute). The experimental results verify the positive effect of GO in improving imputation accuracy.

Once the missing values are imputed, machine learning techniques, including clustering, can be applied to analyze the complete data sets. In recent works, multi-objective optimization (Deb et al. 2002) had been employed to deal with the gene clustering problem and exhibited better performance than single-objective clustering methods (Bandyopadhyay et al. 2007; Faceli et al. 2009; Giri and Sara 2020; Maulik et al. 2009; Mukhopadhyay et al. 2013; Sara et al. 2013; Sara et al. 2018). A multi-objective optimization problem can be formulated as (Deb et al. 2002):

$$\min F(\boldsymbol{x}) = \left(f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \cdots, f_K(\boldsymbol{x})\right) \tag{1}$$

where K is the number of objective functions, vector $\boldsymbol{x}$ is an element in the decision space, and $f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \cdots, f_K(\boldsymbol{x})$ are K objectives to be minimized. Unlike simple-objective optimization, multi-objective optimization generally yields a set of optimal solutions, called Pareto optimal set (PS). Bandyopadhyay et al. (2007) developed a multi-objective clustering algorithm by adopting a variable string length genetic scheme to obtain the number of clusters automatically. Faceli et al. (2009) integrated cluster ensemble and multi-objective clustering for gene expression data with the help of specially designed initial population and crossover schemes. Maulik et al. (2009) proposed a supervised learning method that combined Pareto optimal clusters to identify co-expressed genes. The approach selected the genes that were always clustered together by most of the Pareto optimal solutions to form the training set and classified the remaining genes using a support vector machine. For the case that clusters satisfied the point symmetry property, Sara et al. (2013) proposed a multi-objective clustering method, which optimized fuzzy symmetry-based and separation-based cluster validity indices simultaneously. Considering the adaptive selection of cluster validity indices, Mukhopadhyay et al. (2013) proposed an interactive multi-objective clustering approach that simultaneously found the patterns of gene expression data and the best set of validity measures. By using a link-based clustering ensemble technique, Sara et al. (2018) proposed two multi-objective symmetry-based clustering methods. To involve GO-based biological knowledge during the clustering process, Giri et al. (2020) proposed a multi-view multi-objective clustering approach, which treated the GO-based and expression-based similarities of genes as complementary views. All these methods focused on the clustering analysis of complete gene expression data, in which non-dominated

sorting GA-II (NSGA-II) (Deb et al. 2002) was one of the most prominent multi-objective optimization frameworks applied to gene clustering.

When identifying co-expressed genes from incomplete gene expression data, the aforementioned methods were often designed for a two-stage scheme. In this scheme, the imputation was first implemented as a preprocessing step and clustering techniques were consequently applied to the recovered gene expression data (Kim et al. 2005; Sara et al. 2013; de Souto et al. 2015; Yu et al. 2017). These two-stage methods were quite popular for their simplicity and easy implementation. However, the literature has pointed out that these two-stage methods hinder the collaborative improvement of the two learning processes and thus affect the clustering performance (Liu et al. 2020). Besides, existing imputation methods generally filled in each missing value separately and ignored the overall impact of imputation values on data analysis, such as clustering. Therefore, our goal in this paper is to develop a multi-objective joint optimization framework for clustering incomplete gene expression data, with a view to improving imputation and clustering accuracy synergistically. In addition, biological knowledge is integrated into our framework.

3 Proposed MOC-FNI for incomplete gene expression data

To identify the co-expressed gene groups from incomplete gene expression data, we propose a novel clustering method, called multi-objective clustering algorithm based on the functional neighbor interval (MOC-FNI). In MOC-FNI, the imputation of missing expression values and clustering are optimized synergistically in the framework of NSGA-II. To provide reasonable constraints for the optimization, functional neighbor intervals are constructed for the missing expression values based on the combination of gene expression similarity and GO semantic similarity. Our method makes full use of multi-source information, including gene expression information and GO semantic information. The proposed multi-objective joint optimization can promote the synergistic improvement of imputation and clustering. Thus, MOC-FNI can obtain imputation values guided by both multi-source information and cluster validity, as well as clusters based on meaningful and reasonable imputation results.

3.1 Determination of functional neighbor intervals

Let $\widetilde{\boldsymbol{G}} = [\tilde{g}_{ij}]_{N\times M}$ denote an incomplete data matrix with N genes and M samples, in which the vector $\widetilde{\boldsymbol{g}}_i (i = 1,2,\dots,N)$ denotes expression level of the ith gene in M experiments. Let $\boldsymbol{G} = [g_{ij}]_{N\times M}$ denote the corresponding recovered complete data matrix. For the target gene $\widetilde{\boldsymbol{g}}_b (1 \le b \le N)$ which retains at least one, but not all, missing component, the functional neighbor interval $\left[g_{bj}^{-}, g_{bj}^{+}\right]$ of its missing $\tilde{g}_{bj} (1 \le j \le M)$ can be determined by the combination of GO semantic similarity and expression-based similarity.

For a set of genes to be analyzed, each gene can be annotated with several GO terms. Thus, the functional similarity between genes can be deduced based on the term similarity. In MOC-FNI, an aggregate information content (AIC) (Song et al. 2014) based approach is adopted to measure the semantic similarity of GO terms.

For the given GO terms t_1 and t_2, the AIC semantic similarity is defined as follows (Song et al. 2014):

$$sim_{GO}(t_1, t_2) = \frac{\sum_{t\in T_{t1}\cap T_{t2}} 2\times SW(t)}{SV(t_1)+SV(t_2)} \tag{2}$$

where

$$SW(t) = \frac{1}{1+e^{-\frac{1}{IC(t)}}} \tag{3}$$

$$SV(t) = \sum_{t' \in T_t} SW(t') \tag{4}$$

with T_t being the set of ancestors of term t including t itself in the GO graph. $p(t)$ is the probability of t occurring in the GO database and $IC(t) = -\log p(t)$ reflects the information content of t. Therefore, based on the knowledge represented by $1/IC(t)$, $SW(t)$ measures the semantic weight and $SV(t)$ is the semantic value of GO term t by adding the semantic weights of all its ancestors.

When deducing the functional similarity between genes from the term similarities, the AVE method (Azuaje et al. 2005) is one of the most widely used schemes. The AVE method adopts the average inter-set similarity between terms that annotate the gens to measure the gene semantic similarity. Given a pair of genes $\boldsymbol{g}_a$ and $\boldsymbol{g}_b (a, b = 1,2, \dots, N)$, let $ann(\boldsymbol{g}_a)$, $ann(\boldsymbol{g}_b)$ be the sets of GO terms that annotate the two genes respectively. Then the semantic similarity of $\boldsymbol{g}_a$ and $\boldsymbol{g}_b$ can be determined by:

$$sim_{ave}(\boldsymbol{g}_a, \boldsymbol{g}_b) = \frac{1}{|ann(\boldsymbol{g}_a)||ann(\boldsymbol{g}_b)|} \sum_{\substack{t_1 \in ann(\boldsymbol{g}_a) \\ t_2 \in ann(\boldsymbol{g}_b)}} sim_{GO}(t_1, t_2) \tag{5}$$

where $|ann(\boldsymbol{g}_a)|$, $|ann(\boldsymbol{g}_b)|$ are the cardinalities of $ann(\boldsymbol{g}_a)$, $ann(\boldsymbol{g}_b)$, respectively. Note that Eq.(5) extracts the semantic similarity based on gene annotations available from GO. Therefore, there is no need to consider the incompleteness of gene expression data.

For the incomplete expression data set $\widetilde{\boldsymbol{G}} = [\tilde{g}_{ij}]_{N \times M}$, partial distance (Hathaway and Bezdek 2001; Li et al. 2013) can be used to extract the expression-based dissimilarity of $\widetilde{\boldsymbol{g}}_a$ and $\widetilde{\boldsymbol{g}}_b (a, b = 1,2, \dots, N)$:

$$dist_{expr}(\widetilde{\boldsymbol{g}}_a, \widetilde{\boldsymbol{g}}_b) = \sqrt{\frac{M}{\sum_{j=1}^{M} I_j} \sum_{j=1}^{M} (\tilde{g}_{aj} - \tilde{g}_{bj})^2 I_j} \tag{6}$$

where

$$I_j = \begin{cases} 1, & \text{if both } \tilde{g}_{aj} \text{ and } \tilde{g}_{bj} \text{ are nonmissing} \\ 0, & \text{otherwise} \end{cases} \tag{7}$$

In MOC-FNI, we take into account both the semantic-based and expression-based dissimilarity, the combined distance is defined as (Tuikkala et al. 2006):

$$dist_{comb}(\widetilde{\boldsymbol{g}}_a, \widetilde{\boldsymbol{g}}_b) = \left(1 - sim_{ave}(\boldsymbol{g}_a, \boldsymbol{g}_b)\right)^{\theta} dist_{expr}(\widetilde{\boldsymbol{g}}_a, \widetilde{\boldsymbol{g}}_b) \tag{8}$$

where the positive weight parameter $\theta > 0$ balances the contribution of two dissimilarities.

In this paper, we consider the case that gene expression values are missing completely at random (MCAR). For each target gene $\widetilde{\boldsymbol{g}}_b$ with missing values, the combined distance (8) is applied to guide the selection of functional neighbors. Then, the functional neighbor interval of its missing value $\tilde{g}_{bj}$ can be determined. Specifically, we search for q functional neighbors of $\widetilde{\boldsymbol{g}}_b$ with non-missing feature j, where g_{bj}^{-} and g_{bj}^{+} are the minimum and maximum of the neighbors' jth expression values, respectively. Therefore, $\tilde{g}_{bj}$ can get its functional neighbor interval as $\left[g_{bj}^{-}, g_{bj}^{+}\right]$.

3.2 Objective functions

MOC-FNI is a multi-objective joint optimization method, where the imputation and clustering results are optimized simultaneously in the framework of NSGA-II. In multi-objective clustering problems, the objective functions should conflict with each other and represent different aspects of clustering performance.

The cluster validity indices J_m (Bezdek 1981) and XB (Xie and Beni 1991) measure the intra-cluster compactness and inter-cluster separation respectively. These two indices are commonly used as objective functions and have achieved satisfying clustering performance in various multi-objective clustering algorithms (Bandyopadhyay et al. 2007; Maulik et al. 2009; Mukhopadhyay et al. 2013). MOC-FNI simultaneously optimize these objectives, which are formulated as follows (Bezdek 1981; Xie and Beni 1991):

$$J_m = \sum_{k=1}^{N}\sum_{i=1}^{C} u_{ik}^{m}\|\boldsymbol{g}_k - \boldsymbol{v}_i\|_2^2 \tag{9}$$

$$XB = \frac{\sum_{k=1}^{N}\sum_{i=1}^{C} u_{ik}^{2}\|\boldsymbol{g}_k - \boldsymbol{v}_i\|_2^2}{N \times \min\limits_{i \neq j}\|\boldsymbol{v}_i - \boldsymbol{v}_j\|_2^2} \tag{10}$$

where

$$u_{ik} = \left[\sum_{t=1}^{C}\left(\frac{\|\boldsymbol{g}_k - \boldsymbol{v}_i\|_2^2}{\|\boldsymbol{g}_k - \boldsymbol{v}_t\|_2^2}\right)^{\frac{1}{m-1}}\right]^{-1} \tag{11}$$

represents the degree of $\boldsymbol{g}_k$ in the ith cluster with $u_{ik} \in [0,1](\forall i,k)$ and $\sum_{i=1}^{C} u_{ik} = 1(\forall k)$. $\boldsymbol{g}_k = [g_{1k}, g_{2k}, \dots, g_{Mk}]^{\mathrm{T}}$ is a gene expression vector in the recovered complete data matrix $\boldsymbol{G}$. C is the total number of clusters. $\boldsymbol{v}_i \in \mathbb{R}^M$ is the ith cluster prototype. The parameter $m \in (1,\infty)$ influences the fuzziness of the partition. $\|.\|_2$ stands for the Euclidean norm. Lower values of J_m and XB imply better compactness and separation of the yielded clusters.

3.3 Chromosome encoding and population initialization

To optimize the imputation and clustering results simultaneously, a mixed chromosome encoding strategy is adopted in MOC-FNI. Each chromosome includes the cluster prototypes and the imputation values of missing expression values.

Let $\boldsymbol{E}$ be the population and F be the population size. For an incomplete gene expression data set $\widetilde{\boldsymbol{G}} = [\tilde{g}_{ij}]_{N\times M}$ with h missing values, we first sort and renumber the h functional neighbor intervals of missing expression values by their appearance order in $\widetilde{\boldsymbol{G}}$. Specifically, for each missing value $\tilde{g}_{bj}$, its functional neighbor interval $[g_{bj}^{-}, g_{bj}^{+}]$ is renumbered and represented as $[e_o^{-}, e_o^{+}](1 \le o \le h)$ with o being the order of $\tilde{g}_{bj}$ in all of the missing expression values in $\widetilde{\boldsymbol{G}}$. Taking the C cluster prototypes into account, each chromosome has $C \times M + h$ components. Figure 1 shows the mixed chromosome encoding strategy. For each chromosome $\boldsymbol{E}_f(l)$ with $1 \le f \le F$ at generation l, let the cluster prototype part (with $C \times M$ components) be $\boldsymbol{E}_f^{(clu)}(l) = [v_{11,f}, \dots, v_{1M,f}, \dots, v_{C1,f}, \dots, v_{CM,f}]$, and the imputation part (with h components) be $\boldsymbol{E}_f^{(imp)}(l) = [e_{1,f}, e_{2,f}, \dots, e_{h,f}]$, where $[v_{i1,f}, \dots, v_{iM,f}](1 \le i \le C)$ denotes the ith cluster prototype, and $e_{o,f}(1 \le o \le h)$ denotes the oth imputation value that satisfies the interval constraint $[e_o^{-}, e_o^{+}]$.

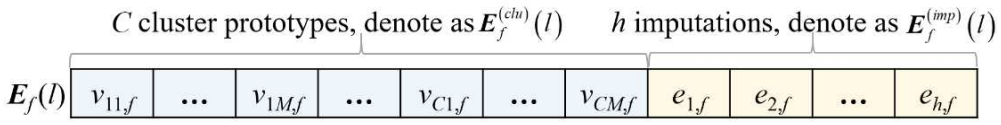


Fig. 1 The mixed chromosome encoding strategy used in MOC-FNI

In MOC-FNI, the two parts of each chromosome are initialized separately. For the fth chromosome $\boldsymbol{E}_f(1)$, each component in the initial imputation part $\boldsymbol{E}_f^{(imp)}(1)$ can be randomly generated in the corresponding functional neighbor interval. For the initial prototype part $\boldsymbol{E}_f^{(clu)}(1)$, we adopt the clustering by fast search and find of density peaks (DPC) (Rodriguez and Laio 2014) on the basis of recovered $\boldsymbol{G}$ imputed by $\boldsymbol{E}_f^{(imp)}(1)$. Then, select C genes with higher local density and having large distance from other density maxima points as the initial prototypes. The DPC algorithm, which has been proven to be effective in analyzing gene expression data (Mehmood et al. 2017), can identify reliable prototypes. The functional neighbor intervals can provide reasonable constraint on the imputation values. These advantages will contribute to improve the convergence speed and optimization ability of genetic search involved in the NSGA-II framework.

3.4 Genetic Operations

In the genetic process of MOC-FNI, we use the roulette wheel for implementing the selection scheme. As NSGA-II prefers solutions with lower non-domination rank (Deb et al. 2002), a lower-rank chromosome should be selected with a higher probability. Thus, for each chromosome $\boldsymbol{E}_f(l)$ with rank f_{rank}, we calculate its selection probability by the following rank-based evaluation function (Zhou and Zhu 2018):

$$P_{selection}\left(\boldsymbol{E}_f(l)\right) = \alpha(1-\alpha)^{f_{rank}-1} \tag{12}$$

where $\alpha \in [0,1]$ is a parameter that controls the selective pressure.

Because the imputation values of missing expression values should evolve within the corresponding functional neighbor intervals, a crossover operator based on competition and optimal selection (Ren and San 2007) is adopted. For the parent chromosomes $\boldsymbol{E}_{f_1}(l)$ and $\boldsymbol{E}_{f_2}(l)$ with $1 \le f_1, f_2 \le F$ at generation l, four offspring are first generated as follows (Ren and San 2007):

$$\boldsymbol{offsp}_1 = \frac{\boldsymbol{E}_{f_1}(l)+\boldsymbol{E}_{f_2}(l)}{2} \tag{13}$$

$$\boldsymbol{offsp}_2 = \frac{(\boldsymbol{e}_{max}+\boldsymbol{e}_{min})(1-\beta)+\left(\boldsymbol{E}_{f_1}(l)+\boldsymbol{E}_{f_2}(l)\right)\beta}{2} \tag{14}$$

$$\boldsymbol{offsp}_3 = \boldsymbol{e}_{max}(1-\beta) + max\left(\boldsymbol{E}_{f_1}(l), \boldsymbol{E}_{f_2}(l)\right)\beta \tag{15}$$

$$\boldsymbol{offsp}_4 = \boldsymbol{e}_{min}(1-\beta) + min\left(\boldsymbol{E}_{f_1}(l), \boldsymbol{E}_{f_2}(l)\right)\beta \tag{16}$$

where crossover factor $\beta \in [0,1]$ denotes the weight determined by users. Both $\boldsymbol{e}_{max}$ and $\boldsymbol{e}_{min}$ have $C \times M + h$ components, $\boldsymbol{e}_{max} = [1,1,\dots,1,e_1^+,e_2^+,\dots,e_h^+]$, $\boldsymbol{e}_{min} = [0,0,\dots,0,e_1^-,e_2^-,\dots,e_h^-]$, where 1 and 0 are the maximum and minimum values of the $C \times M$ prototype components (all expression values have been max-min normalized into $[0,1]$ beforehand). Vectors $max\left(\boldsymbol{E}_{f_1}(l), \boldsymbol{E}_{f_2}(l)\right)$ and $min\left(\boldsymbol{E}_{f_1}(l), \boldsymbol{E}_{f_2}(l)\right)$ are formed by the maximum and minimum of corresponding components in $\boldsymbol{E}_{f_1}(l)$ and $\boldsymbol{E}_{f_2}(l)$, respectively. Note that the imputation values and prototype components may spread all over the functional neighbor intervals and $[0,1]$, respectively. The above crossover operator can generate superior offspring than arithmetic crossover or heuristic crossover (Li et al. 2013; Ren and San 2007). Then, from the two parent chromosomes and four offspring, two chromosomes with lower rank values and lesser crowding distances can be chosen to substitute the parent chromosomes.

In the mutation process, we adopt the classic uniform mutation scheme. As all expression values have been normalized and the evolution of imputation values should satisfy the interval constraints, each component of the prototype part and the imputation part of a chromosome is replaced by a random value in [0,1] and the corresponding functional neighbor interval respectively, with a small probability P_m.

3.5 The procedure of the proposed method

For an incomplete gene expression data set $\widetilde{\boldsymbol{G}} = [\tilde{g}_{ij}]_{N\times M}$ with h missing values, the procedure of the proposed MOC-FNI can be described as follows:

Step 1: For each missing expression value $\tilde{g}_{bj}(1 \le b \le N, 1 \le j \le M)$, find q functional neighbors of the target gene $\widetilde{\boldsymbol{g}}_b$ with non-missing feature j using Equation (8), then determine the functional neighbor interval $\left[g_{bj}^{-}, g_{bj}^{+}\right]$. Renumber the functional neighbor intervals by their appearance order and get $[e_o^{-}, e_o^{+}]$ $(o = 1,2,\dots,h)$.

Step 2: Set the genetic population size F, maximal number of generations $L_{\max}$, selection factor $\alpha \in [0,1]$, crossover factor $\beta \in [0,1]$, mutation probability P_m, fuzzification parameter $m \in (1,\infty)$, and the number of clusters C. Initialize the genetic population $\boldsymbol{E}_f(1)$ $(f = 1,2,\dots,F)$ based on the functional neighbor intervals and DPC algorithm.

Step 3: When the genetic generation index is l $(l = 1,2,\dots,L_{\max})$, for each chromosome $\boldsymbol{E}_f(l)$ $(1 \le f \le F)$, recover $\widetilde{\boldsymbol{G}}$ by using $\boldsymbol{E}_f^{(imp)}(l)$ and yield complete data set $\boldsymbol{G}$. Decode cluster prototypes from $\boldsymbol{E}_f^{(clu)}(l)$ and calculate memberships and indices J_m, XB using Equations (9), (10), and (11) on $\boldsymbol{G}$. Calculate the non-domination rank and crowding distance of $\boldsymbol{E}_f(l)$.

Step 4: Perform roulette wheel selection, the crossover based on competition and optimal selection, and uniform mutation.

Step 5: Combine the parent and offspring population, select the best F solutions for the next iteration with respect to non-domination rank and crowding distance.

Step 6: If genetic generation index $l = L_{\max}$, stop and get the set of Pareto optimal solutions P_s; otherwise set $l = l + 1$ and return to Step 3.

3.6 Selection of the final solution

After running the proposed MOC-FNI, a set of Pareto optimal solutions P_s will be achieved, from which the final imputation results and clustering partition can be extracted. We employ the projection similarity validity index ($PSVIndex$) (Xia et al. 2013; Zhou and Zhu 2018) to extract the best solution from P_s:

$$PSVIndex = \sum_{i=1}^{C}\sum_{a=1}^{n_i}\sum_{b=1,b\le a}^{n_i} SPDis(\boldsymbol{g}_a, \boldsymbol{g}_b) \tag{17}$$

where

$$SPDis(\boldsymbol{g}_a, \boldsymbol{g}_b) = \sum_{j=1}^{M} \log\left(\left|project_{aj} - project_{bj}\right| + 1.0\right) \tag{18}$$

C and M are the numbers of clusters and attributes, respectively. $\boldsymbol{G} = [g_{ij}]_{N\times M}$ is the complete matrix imputed by the imputation part of solutions in P_s. n_i $(1 \le i \le C)$ denotes the number of genes of the ith cluster. $project_{aj}$ is the projection coordinates of expression value g_{aj} that represents the projected interval of gene $\boldsymbol{g}_a$ on the jth dimension. If $\boldsymbol{g}_a$ and $\boldsymbol{g}_b$ belong to the same cluster, they should have the same or quite similar projection coordinates on the M dimensions. That is, $SPDis(\boldsymbol{g}_a, \boldsymbol{g}_b)$ should results in a small value. Therefore, we can select the solution with the smallest $PSVIndex$ value to get final imputation results and cluster partition.

4 Experimental results

4.1 Data sets

We apply MOC-FNI to the following four benchmark data sets to prove its performance.

Arabidopsis Thaliana: This data set consists of 138 Arabidopsis Thaliana genes. Each gene has 8 expression values that correspond to 8 time points. The number of clusters $C_{Arabidopsis} = 4$. To measure the gene semantic similarity, these genes are mapped to GO terms in the three sub-ontologies (BP, MF and CC) of GO. For the Arabidopsis Thaliana data set, the number of GO terms is 2189, with 1365 terms under biological process, 597 terms under molecular function, and 227 terms under the cellular component.

Yeast Cell cycle_384: This data set contains the expression levels of 384 genes involved in yeast cell cycle regulation at 17 time points. These data are related with five phases of cell cycle. Thus, the number of clusters $C_{Yeast\_384} = 5$. The genes in Yeast Cell cycle_384 also mapped to GO terms in BP, MF, and CC. Consequently, the number of GO terms is 4380, with 2872 terms under biological process, 962 terms under molecular function, and 546 terms under the cellular component.

Yeast Cell cycle_237: This data set consists of 237 genes, whose functions fall into four categories in the MIPS database, i.e. $C_{Yeast\_237} = 4$. For Yeast Cell cycle_237 data set, the number of GO terms is 3204, with 2139 terms under biological process, 637 terms under molecular function, and 428 terms under cellular component.

Human Fibroblasts Serum: This data set contains the expression levels of 517 human genes. The data set has 13 dimensions and $C_{Serum} = 6$. For this data set, the number of GO terms is 8469, with 5284 terms under biological process, 2026 terms under molecular function, and 1159 terms under cellular component.

To simulate MCAR, we randomly discard a specified percentage of components in the original data set $\boldsymbol{G}^{(ori)} = [g_{ij}^{(ori)}]_{N\times M}$ and generate the incomplete data set $\widetilde{\boldsymbol{G}} = [\tilde{g}_{ij}]_{N\times M}$.

4.2 Evaluation criteria

To evaluate the imputation performance of MOC-FNI, we use normalized root mean square error (NRMSE) (Cheng et al. 2012):

$$NRMSE = \sqrt{\frac{\frac{1}{h}\sum_{o=1}^{h}(e_o^{(ori)} - \hat{e}_o)^2}{\frac{1}{h-1}\sum_{o=1}^{h}(e_o^{(ori)} - \bar{e})^2}} \tag{19}$$

where h is the number of missing values in $\widetilde{\boldsymbol{G}}$, $\hat{e}_o (1 \le o \le h)$ is the imputation value of the oth missing value, $e_o^{(ori)}$ is its original (true) expression value in $\boldsymbol{G}^{(ori)}$, and $\bar{e}$ is the average of all missing values. NRMSE is the most commonly used evaluation criterion that measure the imputation accuracy. A smaller NRMSE indicates a better imputation result.

An internal cluster validity measure, called Silhouette index, is utilized to evaluate the clustering performance. Let $a(i)$ be the average distance between gene $\boldsymbol{g}_i$ and other genes in the same cluster, $b(i)$ be the minimum average distance between gene $\boldsymbol{g}_i$ and genes in other clusters. Then, the silhouette width of $\boldsymbol{g}_i$ is defined as (Rousseeuw 1987):

$$S(i) = \frac{b(i)-a(i)}{\max\{a(i),b(i)\}}, 1 \le i \le N \tag{20}$$

It can be seen that, for each gene $\boldsymbol{g}_i$, $S(i) \in [-1,1]$. The Silhouette index of the gene expression matrix is computed as the average value of the Silhouette width of all genes. A large Silhouette index indicates a good clustering result.

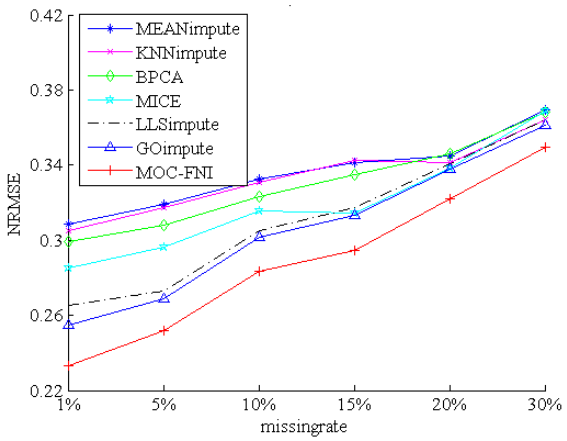
4.3 Experiments setting-up

We compare MOC-FNI with several popular and promising imputation methods, including MEANimpute (Acurna and Rodriguez 2004), KNNimpute (Troyanskaya et al 2001), BPCA (Oba et al. 2003; de Souto et al. 2015), MICE (Buuren and Oudshoorn 2011), LLSimpute (Kim et al. 2005), and GOimpute (Tuikkala et al. 2006). Then, the same NSGA-II framework as that in MOC-FNI is employed to perform clustering on these recovered gene expression matrices. These two-stage clustering methods are termed as MEANimpute+NSGAII, KNNimpute+NSGAII, BPCA+NSGAII, MICE+NSGAII, LLSimpute+NSGAII, and GOimpute+NSGAII, respectively. For each of the two-stage methods, as the imputation has been implemented in the preprocess, the chromosomes only encode cluster prototypes (with $C \times M$ components).

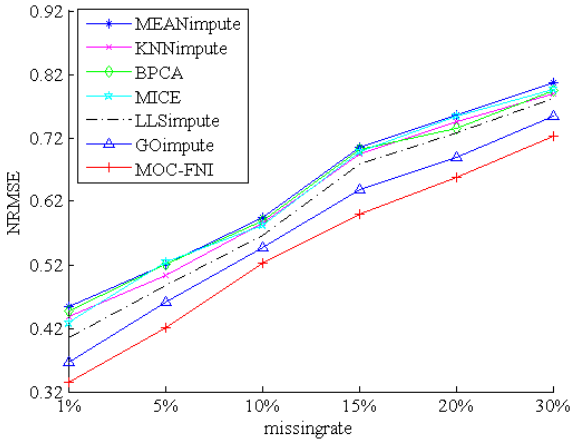
Set population size $F = 80$, maximal number of generations $L_{\max} = 60$, selection factor $\alpha = 0.3$, crossover factor $\beta = 0.3$, mutation probability $P_m = 0.1$, fuzzification parameter $m = 2$, the number of functional neighbors $q = 10$ for MOC-FNI. Motivated by GOimpute (Tuikkala et al. 2006), the parameter θ in Equation (8) is selected adaptively. First, we randomly designate 5% of non-missing expression values as missing artificially from the data set to be imputed. For the optimal θ, we search from 0.4 to 2.0 with grid 0.1 with respect to the minimum *NRMSE*. The optimal values for the four data sets are $\theta_{Arabidopsis} = 0.9$, $\theta_{Yeast\_384} = 1.4$, $\theta_{Yeast\_237} = 1.1$, $\theta_{Serum} = 1.2$, respectively. The parameters of the compared methods are set according to the optimal parameters suggested in the original papers. Various missing rates are randomly generated in the original matrices: 1%, 5%, 10%, 15%, 20% and 30%, respectively. We present the average results obtained over 10 trials with the same incomplete data set in each trial for each approach.

4.4 Comparison of imputation performance

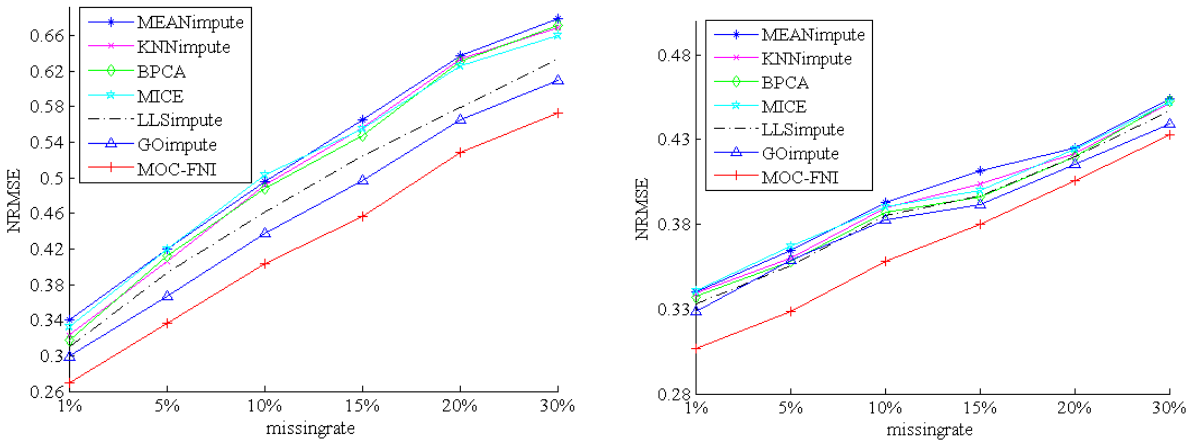
Figure 2 shows the average NRMSE values of MOC-FNI on four data sets at different missing rates in comparison to other six methods.



(a) Arabidopsis Thaliana



(b) Yeast Cell cycle_384



(c) Yeast Cell cycle_237 (d) Serum

Fig. 2 Average NRMSE values of seven methods at different missing rates for four data sets

From Figure 2, we observe an obvious trend that NRMSE values increase along with the missing rates for all imputation methods on all data sets. We also observe that MOC-FNI and GOimpute always achieved the first two smallest NRMSE values, suggesting that GO information improves the imputation accuracy. Taking the Yeast Cell cycle_384 data set as an example, compared to the MEANimpute, KNNimpute, BPCA, MICE and LLSimpute without using GO information, the MOC-FNI reduces the NRMSE values by 26.1%, 23.5%, 25.1%, 21.7%, and 17.0% at 1% missing rate, respectively. These percentages, along with the increase of missing rates, reduce to 7.5% - 10.5% at 30% missing rate. The results show that the proposed MOC-FNI dominates to other methods in imputation accuracy for various missing rates.

4.5 Comparison of clustering results

Tables 1, 2, 3 and 4 show the average Silhouette index values obtained by MOC-FNI and six two-stage methods on four data sets. The optimal solutions in each column are highlighted in bold and the suboptimal solutions are underlined. To give a visual comparison, we sort the average Silhouette index values in each column in descending order and obtain the sort index. Figure 3 gives the average sort index of the seven methods, and a smaller average sort index indicates a better clustering result.

Table 1. Average Silhouette index values of different algorithms on Arabidopsis Thaliana

Algorithm	Missing rate					
	1%	5%	10%	15%	20%	30%
MEANimpute+NSGAII	0.3880	0.3656	0.3597	0.3591	0.3272	0.2886
KNNimpute+NSGAII	0.3885	0.3785	0.3608	0.3468	0.3227	0.2928
BPCA+NSGAII	0.3974	0.3791	<u>0.3702</u>	0.3594	0.3252	**0.3094**
MICE+NSGAII	0.3855	0.3549	0.3691	0.3579	0.3290	0.2873
LLSimpute+NSGAII	0.3922	0.3901	0.3596	0.3515	0.3264	0.3016
GOimpute+NSGAII	<u>0.3989</u>	<u>0.3962</u>	0.3688	<u>0.3652</u>	<u>0.3303</u>	0.3063
MOC-FNI	**0.4039**	**0.4008**	**0.3954**	**0.3808**	**0.3416**	<u>0.3075</u>

Table 2. Average Silhouette index values of different algorithms on Yeast Cell cycle_384

Algorithm	Missing rate					
	1%	5%	10%	15%	20%	30%
MEANimpute+NSGAII	0.3868	0.3566	0.3613	0.3285	0.3105	0.3004
KNNimpute+NSGAII	0.3910	0.3736	0.3712	0.3576	0.3241	0.3165
BPCA+NSGAII	0.4026	0.3865	0.3688	0.3472	0.3153	0.3177
MICE+NSGAII	0.3850	0.3715	0.3553	0.3302	0.3138	0.3023
LLSimpute+NSGAII	0.4045	0.3879	0.3837	0.3581	0.3288	0.3148
GOimpute+NSGAII	0.4065	0.3956	0.3882	0.3567	0.3460	0.3287
MOC-FNI	**0.4235**	**0.4092**	**0.3953**	**0.3730**	**0.3709**	**0.3549**

Table 3. Average Silhouette index values of different algorithms on Yeast Cell cycle_237

Algorithm	Missing rate					
	1%	5%	10%	15%	20%	30%
MEANimpute+NSGAII	0.3930	0.3779	0.3562	0.3237	0.2935	0.2907
KNNimpute+NSGAII	0.3908	0.3690	0.3556	0.3266	0.3028	0.2945
BPCA+NSGAII	0.3849	0.3700	0.3657	0.3163	0.3266	0.3100
MICE+NSGAII	0.3874	0.3922	0.3624	0.3265	0.2976	0.2992
LLSimpute+NSGAII	0.3867	0.3815	0.3744	0.3321	0.3249	0.3115
GOimpute+NSGAII	0.3914	0.3857	0.3781	0.3388	0.3300	0.3115
MOC-FNI	**0.3971**	**0.3932**	**0.3846**	**0.3687**	**0.3479**	**0.3281**

Table 4. Average Silhouette index values of different algorithms on Serum

Algorithm	Missing rate					
	1%	5%	10%	15%	20%	30%
MEANimpute+NSGAII	0.3977	0.3866	0.3792	0.3396	0.3303	0.3221
KNNimpute+NSGAII	0.3875	0.3922	0.3632	0.3450	0.3254	0.3157
BPCA+NSGAII	0.4066	0.3953	0.3842	0.3658	0.3492	0.3232
MICE+NSGAII	0.4001	0.3901	0.3800	0.3497	0.3351	0.3197
LLSimpute+NSGAII	0.4003	0.3931	0.3890	0.3615	0.3389	0.3328
GOimpute+NSGAII	0.4053	0.4069	0.3990	0.3656	0.3536	0.3375
MOC-FNI	**0.4196**	**0.4106**	**0.4052**	**0.3980**	**0.3680**	**0.3575**

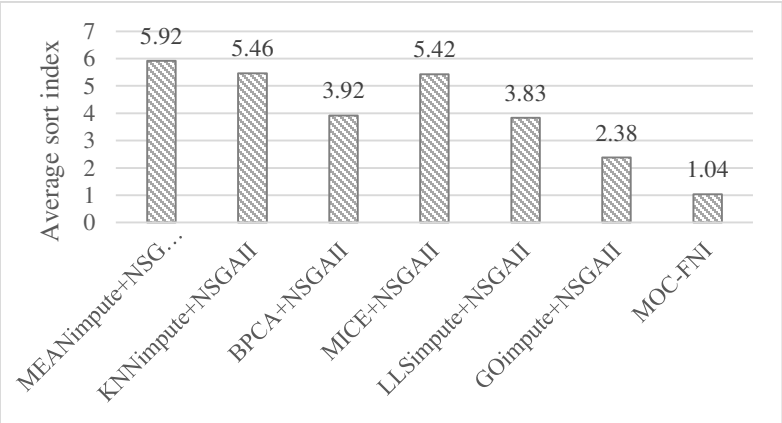
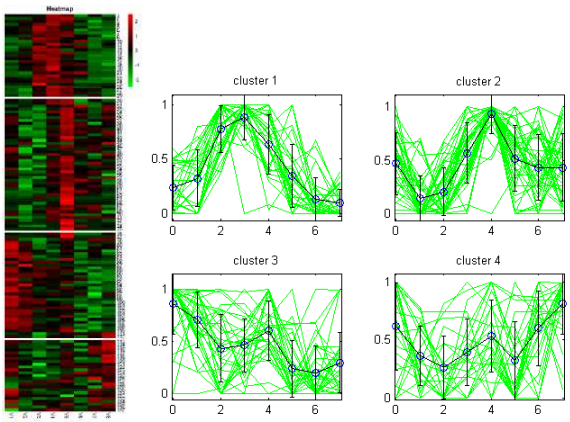


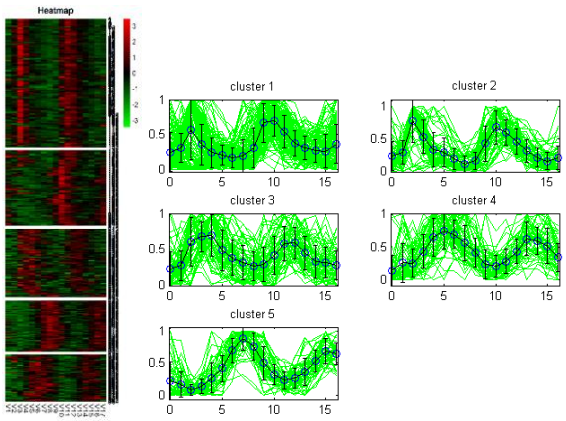
Fig. 3 Average sort index of seven methods at different missing rates for four data sets

From Tables 1, 2, 3 and 4, we observe that the Silhouette index values decline along with the missing rates on the overall trend. Compared to the two-stage methods, the proposed MOC-FNI is always the outperformer expect for the 30% case of incomplete Arabidopsis Thaliana data sets as a suboptimal solution. GOimpute+NSGAII can get the second largest Silhouettes index values in most cases. Meanwhile, the results shown in Figure 3 conform to the imputation accuracy shown in Figure 2, which illustrate the positive impact of high imputation accuracy on clustering. These experimental results validate the mutual promotion between imputation and clustering and the rationality of joint optimization.

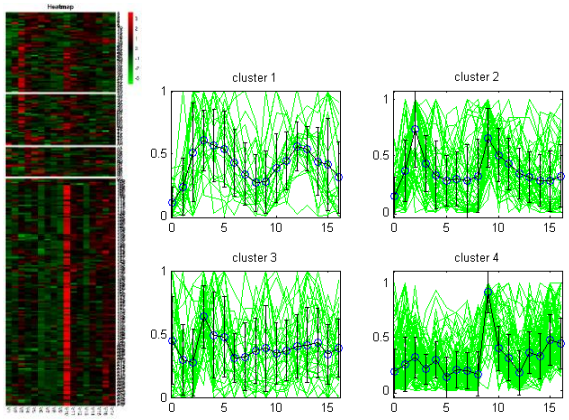
Figures 4 shows the Eisen plots and cluster profiles on the four data sets, respectively. Taking the Yeast Cell cycle_384 data set as an example, we find that the 5 clusters generated by MOC-FNI are very prominent as shown in the Eisen plot (Yeast Cell cycle_384 (a)), the cluster profiles (Yeast Cell cycle_384 (b)) indicate that the expression profiles of genes in each cluster are quite similar. Similar observations can also be obtained from the other data sets.



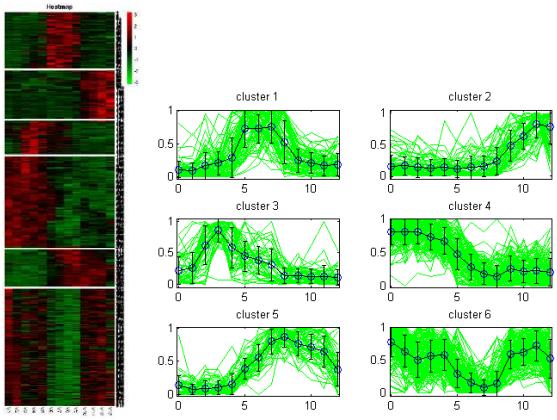
Arabidopsis Thaliana (a) Arabidopsis Thaliana (b)
(with 1% missing values)



Yeast Cell cycle_384 (a) Yeast Cell cycle_384 (b)
(with 5% missing values)



Yeast Cell cycle_237 (a) Yeast Cell cycle_237 (b)
(with 10% missing values)



Serum (a) Serum (b)
(with 20% missing values)

Fig. 4 Data sets clustered using the proposed MOC-FNI. (**a**) Eisen plot. (**b**) Cluster profiles

4.6 Biological significance

To test the functional enrichment of gene clusters obtained by MOC-FNI and the compared two-stage methods, we also perform biological relevance test with the help of GOTermFinder tool (https://www.yeastgenome.org/goTermFinder). The network analysis tool can find the significant shared GO terms that describe the genes in each cluster from a GO sub-ontologies (BP, CC, MF) and provide the corresponding *p*-values based on the hypergeometric distribution. The closer the obtained *p*-value is to 0, the more biologically significant the clustering result.

Taking the 5% case of incomplete Yeast Cell cycle_237 data set as an example, the biological significance test is conducted at the 1% significance level. We focus on the three most significant GO terms (with the first three smallest *p*-values) for each of the 4 clusters obtained by different methods. Figure 5 shows the plot of the average *p*-values. To illustrate the difference significantly, the *p*-values are negative log-transformed and the clusters are sorted in descending order according to the transformed values. Table 5 reports the three most significant GO terms and the corresponding *p*-values in each cluster obtained by MOC-FNI.

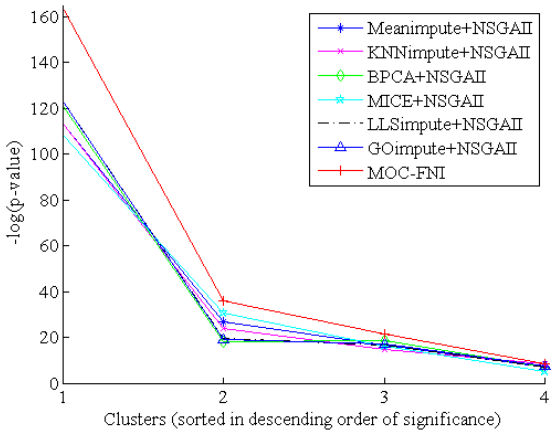


Fig. 5 Plot of the average *p*-values of the three most significant GO terms for each of the 4 clusters obtained by different algorithms

Table 5. The three most significant GO terms and the corresponding *p*-values for each of the 4 clusters obtained by MOC-FNI

Cluster	Significant GO term(GO ID)	*p*-value
Cluster1	structural constituent of ribosome(GO:0003735)	1.00e-171
	ribosomal subunit(GO:0044391)	2.70e-169
	ribosome(GO:0005840)	2.22e-163
Cluster2	DNA replication(GO:0006260)	7.80e-51
	DNA-dependent DNA replication(GO:0006261)	1.97e-48
	DNA metabolic process(GO:0006259)	3.56e-36
Cluster3	spindle pole body(GO:0005816)	9.50e-24
	microtubule organizing center(GO:0005815)	5.35e-23
	microtubule cytoskeleton(GO:0015630)	1.09e-21
Cluster4	sulfate assimilation(GO:0000103)	7.11e-11
	cellular amino acid metabolic process(GO:0006520)	9.92e-11
	toxin biosynthetic process(GO:0009403)	1.30e-08

From Figure 5, it is clear that MOC-FNI can always get the smallest average p-values in the 4 clusters. All the p-values of the significant GO terms listed in Table 5 are far less than 0.01. The above results indicate that the gene clusters obtained by MOC-FNI possess more biological significance.

5 Conclusion

This paper focused on the imputation and cluster tasks of incomplete gene expression data and presented a multi-objective clustering algorithm guided by biological information. In the proposed multi-objective joint optimization framework, we impute all the missing expression values as a whole rather than separately to serve the clustering task and realize the synergistic optimization of imputation and clustering. Besides, the GO annotation information integrated in our framework helps to provide reasonable constraints for the optimization process. Experimental results indicate that MOC-FNI outperforms the compared methods in terms of imputation, clustering accuracy, and biological significance. As an interesting future scope of work, we will introduce semantic similarity to the clustering process to further improve the clustering performance of incomplete gene expression data.

Acknowledgements This work is supported by the Fundamental Research Funds for the Central Universities (DUT21YG118).

Author contributions DL, HG and PQ proposed the theoretical method. QC conducted the experiments. JW contributed in the analysis of biological significance. DL and PQ performed the data analysis and wrote the manuscript.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

Informed consent Informed consent was obtained from all individual participants included in the study.

References

Acurna E, Rodriguez C (2004) The treatment of missing values and its effect in the classifier accuracy. In: Proceedings of the meeting of the international federation of classification societies, pp 639-648

Ashburner M, Ball CA, Blake JA et al (2000) Gene ontology: tool for the unification of biology. Nature Genetics 25(1):25-29

Azuaje F, Wang HY, Bodenreider O (2005) Ontology-driven similarity approaches to supporting gene functional assessment. In: Proceedings of the eight annual Bio-ontologies meeting, pp 9-10

Bandyopadhyay S, Mukhopadhyay A, Maulik U (2007) An improved algorithm for clustering gene expression data. Bioinf 23(21):2859-2865

Bezdek JC (1981) Pattern Recognition with Fuzzy Objective Function Algorithms. Kluwer Academic Publishers, Norwell

Buuren SV, Oudshoorn KG (2011) mice: multivariate imputation by chained equations in R. J Stat Softw 45(3):1-68

Chen XJ, Huang JZ, Wu QY et al (2019) Subspace weighting co-clustering of gene expression data. IEEE/ACM Trans Comput Biol Bioinform 16(2):352-364
Cheng KO, Law NF, Siu WC (2012) Iterative bicluster-based least square framework for estimation of missing values in microarray gene expression data. Pattern Recognit 45(4):1281-1289
de Souto MCP, Jaskowiak PA, Costa IG et al (2015) Impact of missing data imputation methods on gene expression clustering and classification. BMC Bioinf 16:1-9
Deb K, Pratap A, Agarwal S et al (2002) A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans Evol Comput 6(2):182-197
Faceli K, de Souto MCP, De Araujo DSA et al (2009) Multi-objective clustering ensemble for gene expression data analysis. Neurocomputing 72:2763-2774
Giri SJ, Saha S (2020) Multi-view gene clustering using Gene Ontology and expression-based similarities. In: Congress on Evolutionary Computation. IEEE, pp 1-8.
Hathaway RJ, Bezdek JC (2001) Fuzzy c-means clustering of incomplete data. IEEE Trans Syst Man Cybern B 31(5):735-744
Kim H, Golub GH, Park H (2005) Missing value estimation for DNA microarray gene expression data: local least squares imputation. Bioinf 21(2):187-198
Li D, Gu H, Zhang LY (2013) A hybrid genetic algorithm – fuzzy c-means approach for incomplete data clustering based on nearest-neighbor intervals. Soft Comput 17(10):1787-1796
Liu XW, Zhu XZ, Li MM et al (2020) Multiple kernel k-means with incomplete kernels. IEEE Trans Pattern Anal Mach Intell 42(5):1191-1204
Maulik U, Mukhopadhyay A, Bandyopadhyay S (2009) Combining pareto-optimal clusters using supervised learning for identifying co-expressed genes. BMC Bioinf 10:1-16
Mehmood R, Ashram SE, Bie RF et al (2017) Clustering by fast search and merge of local density peaks for gene expression microarray data. Sci Rep 7:45602
Moorthy K, Mohamad MS, Deris S (2014) A review on missing imputation algorithm for microarray gene expression data. Curr Bioinform 9(1):18-22
Mukhopadhyay A, Maulik U, Bandyopadhyay S (2013) An interactive approach to multiobjective clustering of gene expression patterns. IEEE Trans Biomed Eng 60(1):35-41
Oba S, Sato MA, Takemasa I et al (2003) A Bayesian missing value estimation method for gene expression profile data. Bioinf 19(16):2088-2096
Ren ZW, San Y (2007) Improvement of real-valued genetic algorithm and performance study. Acta Electronica Sinica 35(10):269-274
Rodriguez A, Laio A (2014) Clustering by fast search and find of density peaks. Science 344:1492-1496
Rousseeuw PJ (1987) Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. J Comput Appl Math 20:53-65
Sara S, Das R, Partha P (2018) Aggregation of multi-objective fuzzy symmetry-based clustering techniques for improving gene and cancer classification. Soft Comput 22(9):5935-5954
Sara S, Ekbal A, Gupta K et al (2013) Gene expression data clustering using a multiobjective symmetry based clustering technique. Comput Biol Med 43:1965-1977
Stegmayer G, Milone DH, Kamenetzky L et al (2012) A biologically inspired validity measure for comparison of clustering methods over metabolic data sets. IEEE/ACM Trans Comput Biol Bioinform 9(3):706-716
Song XB, Li L, Srimani PK et al (2014) Measure the semantic similarity of GO terms using aggregate information content. IEEE/ACM Trans Comput Biol Bioinform 11(3):468-476
Troyanskaya O, Cantor M, Sherlock G et al (2001) Missing value estimation methods for DNA microarrays. Bioinf 17(6):520-525
Tuikkala J, Elo L, Nevalainen OS et al (2006) Improving missing value estimation in microarray data with gene ontology. Bioinf 22(5):566-572
Xia H, Zhuang J, Yu DH (2013) Novel soft subspace clustering with multi-objective evolutionary approach for high-dimensional data. Pattern Recognit 46:2562-2575
Xie XL, Beni GA (1991) A validity measure for fuzzy clustering. IEEE Trans Pattern Anal Mach Intell 13(8):841-847

Yu Z, Li TR, Horng SJ et al (2017) An iterative locally auto-weighted least squares method for microarray missing value estimation. IEEE Trans NanoBioScience 16(1):21-33

Zhou ZP, Zhu SW (2018) Kernel-based multi-objective clustering algorithm with automatic attribute weighting. Soft Comput 22(6):3685-3709