

RESEARCH

Open Access



Multiview clustering of multi-omics data integration by using a penalty model

Hamas A. AL-kuhali¹, Ma Shan^{1*}, Mohammed Abduljabbar Hael³, Eman A. Al-Hada¹, Shamsan A. Al-Murisi¹, Ahmed A. Al-kuhali⁴, Ammar A. Q. Aldaifl⁵ and Mohammed Elmustafa Amin²

*Correspondence:
mashan@lzu.edu.cn

¹ School of Mathematics and Statistics, Lanzhou University, Lanzhou, China

² Department of Mathematics, Faculty of Science and Technology, Omdurman Islamic University, Khartoum, Sudan

³ Department of Data Science and Information Technology, Taiz University, Taiz, Yemen

⁴ School of Management, Lanzhou University, Lanzhou, China

⁵ School of Information Engineering, Wuhan University of Technology, Wuhan, China

Abstract

Background: Methods for the multiview clustering and integration of multi-omics data have been developed recently to solve problems caused by data noise or limited sample size and to integrate multi-omics data with consistent (common) and differential cluster patterns. However, the integration of such data still suffers from limited performance and low accuracy.

Results: In this study, a computational framework for the multiview clustering method based on the penalty model is presented to overcome the challenges of low accuracy and limited performance in the case of integrating multi-omics data with consistent (common) and differential cluster patterns. The performance of the proposed method was evaluated on synthetic data and four real multi-omics data and then compared with approaches presented in the literature under different scenarios. Result implies that our method exhibits competitive performance compared with recently developed techniques when the underlying clusters are consistent with synthetic data. In the case of the differential clusters, the proposed method also presents an enhanced performance. In addition, with regards to real omics data, the developed method exhibits better performance, demonstrating its ability to provide more detailed information within each data type and working better to integrate multi-omics data with consistent (common) and differential cluster patterns. This study shows that the proposed method offers more significant differences in survival times across all types of cancer.

Conclusions: A new multiview clustering method is proposed in this study based on synthetic and real data. This method performs better than other techniques previously presented in the literature in terms of integrating multi-omics data with consistent and differential cluster patterns and determining the significance of difference in survival times.

Keywords: Multi-omics data, Multiview clustering, Data integration, Penalty model

Background

Cancer is a complicated disease with phenotypic manifestations connected to various molecular signatures, such as gene expression and DNA methylation. That is, genetic mutations and epigenetic landscapes can be used to determine cancer types and subtypes. Consequently, any causal analysis based exclusively on one component or single



© The Author(s) 2022. **Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>. The Creative Commons Public Domain Dedication waiver (<http://creativecommons.org/publicdomain/zero/1.0/>) applies to the data made available in this article, unless otherwise stated in a credit line to the data.

omics will exhibit a causal reductionism resulting in unsatisfactory results [1]. The rapid advancement of high-production technology has produced a massive amount of data generated from patients with different types of cancer, facilitating the collection of different genome-scale datasets to address clinical and scientific challenges. The Cancer Genome Atlas (TCGA) is one of the most prominent projects. It provides a considerable amount of omics data obtained from different platforms (e.g. DNA methylation and gene expression). Therefore, developing methods for integrating different types of omics data and creating a comprehensive picture of a given disease or biological process is necessary to improve disease detection, treatment and prevention [2].

Clustering is a popular technique for exploratory data analysis [3], and it has been used as a fundamental step in the comprehensive analysis of omics data to identify cancer subtypes [4–6] and detect correlated gene expression patterns [7]. Traditional clustering methods for omics data analysis involve one data type, such as DNA methylation [8]. However, information from one data type may be inconsistent due to a small number of samples compared with a large number of measurements, scale differences, collection bias, noise in each dataset and the complementary nature of information provided by different types of omics data. Several methods have focused on multi-omics data clustering to integrate information from different types of omics data [9–12] or used concordant data structure to perform clustering [10, 12–16].

Although multiview clustering methods have significantly improved clustering performance, some methods assume that underlying clusters for different data types are consistent clusters [11, 12, 16], whilst others assume that such clusters are differential clusters [17–19]. In such case, multi-omics data should have simultaneous consistent and differential cluster patterns in accordance with our intuition. A few integrative clustering methods have been proposed recently to integrate multiview data with consistent and differential cluster patterns [20–23]. The authors of [24] proposed a multiview clustering method to solve the multiview spectral clustering optimisation problem [17]; this method used the linear search technique. However, all previous studies reported low performance in integrating different data types. The optimisation problem exhibits orthogonality constraints. Several methods have been proposed to solve the optimisation problem with orthogonality, such as gradient-based methods [24–26], conjugate gradient methods [27, 28], projection-based methods [29], a constraint-preserving updating scheme [30, 31], a multiplier correction framework [32] and penalty function methods [33].

In the present study, we proposed a new approach for multiview clustering that aimed to integrate different types of data with consistent and differential cluster patterns. The current work developed the computational framework for multiview clustering presented in [23] to improve performance in integrating different types of omics data with consistent and differential cluster patterns. The procedures of the proposed method were based on the penalty model, and the first-order algorithm in [33] was used to solve the multiview spectral clustering optimisation problem with orthogonality constraints [21]. Moreover, we evaluated the performance of the proposed method on synthetic data. On the basis of the obtained high accuracy, we used the method on four different types of real multi-omics data. The experiment results demonstrated that the proposed method outperformed other methods presented in the literature.

The major contributions of this work are as follows. Firstly, a new multiview clustering method is provided for integrating multi-omics data with consistent and differential cluster patterns. Secondly, the proposed method is used to determine the significance of difference in survival times. Finally, comprehensive experiments on several state-of-the-art multiview clustering methods are performed to validate the proposed method. The remaining sections of this paper are organised as follows. "Related work" section provides a review of related work. "Proposed method" section presents the proposed method. "Results and discussion" section describes the experiments performed on datasets by using state-of-the-art algorithms for comparison. "Conclusion" section concludes the study.

Related work

In this section, we review related work on multiview clustering methods for integrating different types of data.

Currently available approaches can be classified into three types. The first type is based on a concordant data structure when performing clustering [10, 12–16]. iCluster is an integrative clustering method based on a Gaussian latent variable model with lasso-type penalty terms to induce sparsity in coefficient matrices for feature selection. This approach's significant computational complexity necessitates gene preselection. Therefore, clustering outcomes are dependent on this step [12, 15]. To address the issue of gene preselection, a fused network method is proposed. This method uses sample networks as a foundation for integration and merges similarity networks built for each data type into a single combined similarity network via an iterative approach based on message passing. The final clusters for a fused network are obtained using spectral clustering [13]. A multiple kernel learning method that aims to reduce the dimensions of data from different sources conserves the distance of neighbours for all data types by extending spectral clustering to accept several affinity matrices as input and then fuses matrices by using a linear combination with weight optimisation [12].

The second type is based on spectral clustering. The affinity aggregation spectral clustering method provides a framework for learning the spectral clustering similarity matrix to improve the robustness of spectral clustering by reducing the effect of unreliable and irrelevant features [34]. The authors of [21] and [22] aimed to maximise consistency between clusters from different perspectives by using various cluster consistency measures. The formulation of an optimisation problem in [22] included an alternative computation of the eigenvectors of the regularised Laplacian matrix; thus, the method is less stable and more likely to yield the local optimum.

For the third type, multiview clustering methods have demonstrated remarkably improved clustering performance, although some of these methods assume that the underlying clusters across different data types are consistent clusters. For example, methods based on the application of cancer patients assume that the underlying subtypes are the same across different data types [11, 12, 16]. The flaw of these methods is that they do not check whether genes, microRNA (miRNA) and other small molecules exhibit the same cluster patterns across different subtypes. Several methods based on the assumption that underlying clusters across various data types are different have also been proposed by comparing clusters identified in different data or by merging

information that demonstrates differences in single objects [17–19]. Meanwhile, a few methods have focused on the problem of integrating different datasets with consistent and differential cluster patterns. The method in [22] tends to obtain the local optimum, whereas the methods in [21, 35] overly relax the original multiview ratio cut model, likely producing the information in each data type. The multiview clustering method based on manifold was proposed to solve the multiview spectral clustering optimisation problem [23] by using a linear search technique. All these methods exhibit lower performance in integrating different data types. Therefore, we proposed a new multiview clustering method to obtain better performance.

Proposed method

In this section, we introduce the penalty model, the proposed method steps, the algorithm, and the example for our method as follows.

Penalty model

By considering the following matrix optimization problem with orthogonality constraints:

$$\min_{X \in \mathbb{R}^{n \times p}} f(X), \text{ s.t. } X^T X = I_p, \quad (1)$$

where I_p is any $p \times p$ identity matrix, and $f : \mathbb{R}^{n \times p} \mapsto \mathbb{R}$ satisfies the following assumption throughout this section. We start with the merit function that can be written as follows:

$$h(X) = f(X) - \frac{1}{2} \langle \psi(\nabla f(X)^T X, X^T X - I_p) \rangle + \frac{\beta}{4} \|X^T X - I_p\|_F^2 \quad (2)$$

This merit function was firstly defined in Gao et al. [36]. Equation (4.2) evaluates the function value reduction of the proximal linearised augmented Lagrangian algorithm (PLAM), where $\psi : \mathbb{R}^{p \times p} \rightarrow \mathbb{R}^{p \times p}$, and $\psi(X) = \frac{X^T + X}{2}$ denotes the linear operator for symmetrisation. The penalty model that minimises $h(X)$ under a compact convex constraint (PenC) is as follows:

$$\min_{X \in M} h(X), \quad (3)$$

where M is a compact convex set that contains $St_{n,p}$.

Xiao et al. [33] demonstrated that the original problem 1 and the penalty model 3 are equivalent. They proposed the first-order method (PenCF) for solving problem 3. In our work, we used PenCF to solve our problem because our objective function was formulated as

$$f(X) = Tr(X^T L X). \quad (4)$$

Steps of the proposed method

The proposed method has five steps. Firstly, to normalise each data type to have a standard normal distribution [10], we convert each data type into a patient–patient k-nearest

neighbour (NN) similarity network on the basis of a spectral clustering method [37, 38]. Secondly, we obtain a multiview spectral clustering optimisation problem from the multiview network clustering model to integrate multiple similarity networks [21, 35]. Thirdly, we use the penalty model, i.e. the first-order algorithm from [33], to solve the multiview spectral clustering problem with orthogonality constraints. Fourthly, we repeat the processing of the penalty model until convergence is achieved, and we obtain the values of X that are represented by the matrix that contains the label for each patient. Finally, we use k -mean to cluster the matrix X into $C_1; C_2; C_3; \dots; C_K$. We present the details of these steps as follows.

Normalization

We start to normalise each feature across all data types to have a standard normal distribution. We use the following form:

$$\check{g} = \frac{g - E(g)}{\sqrt{Var(g)}}, \tag{5}$$

where \check{g} denotes the feature after normalization, g denotes a feature of any data type, $E(g)$ represents the mean of features, and $Var(g)$ represents the variance of features.

Construction of the k -NN similarity network

Consider M types of omics data measurements $\{X_m\}_{m=1}^M$ (each with a dimension of \check{g}_m) collected from N patient samples, such that X_m is a $\check{g}_m \times N$ matrix, where \check{g}_m is the number of features of m . For each data type X_m , we construct a patient-to patient similarity network G_m to model the local neighbourhood relationships between the samples. Let $G_m = (V_m; E_m; A_m)$ denote a patient similarity network for data type m , where $V_m = \{v_1; v_2; v_3 \dots; v_N\}$ denotes the set of the nodes, E_m denotes the edge set, and A_m denotes the adjacency matrix. The nodes represent N patients, and the edges represent the connection between patients.

The adjacency matrix A_m of the network G_M is a symmetric matrix whose entry $a_m(i, j)$ represents the edge weight if there is an edge between node v_i and v_j , otherwise, $A_m(i, j) = 0$. To construct this similarity network, we firstly compute a similarity matrix for measuring pairwise similarity between each sample pair. Here, we use a common similarity measure, called Gaussian similarity, as the similarity metric [37, 38]:

$$S_m(i, j) = \exp\left(-\frac{d^2}{2\alpha^2}\right) \quad i = 1, 2, \dots, N, j = 1, 2, \dots, N; \tag{6}$$

where $d = \|x_m(i) - x_m(j)\|$ is the Euclidean distance between $x_m(i)$ and $x_m(j)$ patients, and parameter α controls the width of the neighbourhoods. Several options are available for setting parameter α , here we use the most common choice by setting parameter α as the standard deviation of patients $\|x_m(i) - x_m(j)\|$. Then, we construct a k -NN network from the similarity matrix S_m . We symbolise N_i as a set of node v_i neighbours with node v_i , and the size of N_i is equal to k . We then connect v_i and v_j with an undirected edge with the edge weight as $S(i, j)$ if $v_i \in N_i$, as shown in Eq. (7).

$$A_m(i, j) = \begin{cases} \mathbf{S}_m(i, j), & \text{if } v_i \in \mathbf{N}_i \\ 0, & \text{otherwise.} \end{cases} \tag{7}$$

Thus far, the k -NN similarity network G_m for each data type is constructed.

Multiview network clustering model for integrating the multiple similarity networks

To integrate all k -NN similarity networks, we first compute the Laplacian matrix L_m for each k -NN similarity network G_m for all data types.

$$L_m = D_m - A_m, \tag{8}$$

where A_m is the adjacency matrix for the k -NN similarity network G_m ; and D_m is the diagonal matrix for the k -NN similarity network G_m , whose entries are the degree of all the nodes(patients) in the m -th G_m .

Then, we present the multiview network clustering model [21, 35]. We start by denoting the number of clusters as K across the M networks and S^m as the assignment of the N nodes into K clusters for the network G_M .

We set

$$S_{i,k}^m = \begin{cases} 1, & \text{if } i \in G^m \text{ belongs to the } k - \text{th cluster} \\ 0, & \text{otherwise.} \end{cases} \tag{9}$$

$i = 1; 2; \dots; N, m = 1; 2; \dots; M, k = 1; 2; \dots; K.$

We firstly align the clusters obtained for different networks to show consistent and differential cluster patterns across multiple networks. The computation is complex if we directly align the identified cluster patterns in each network. Thus, Zhang et al. proposed identifying cluster patterns for each network and aligning clusters for multiple networks. The cluster patterns in each network are identified using spectral clustering. For cluster alignment in different networks, the similarity between the m th cluster in network G_m and the h th cluster in network G_h is defined as $\frac{S_{.,k}^m S_{.,k}^h}{\|S_{.,k}^m\|_2 \|S_{.,k}^h\|_2}$. Zhang et al. [21] and

Chen et al. [35] aimed to maximize the similarities of the corresponding clusters in all networks, and the objective function of a multiview network clustering model is

$$\varphi(S^1, S^2, \dots, S^M) = \sum_{m=1}^M \sum_{k=1}^K \frac{(S_{.,k}^m)^T (D_m - A_m) S_{.,k}^m}{(S_{.,k}^m)^T S_{.,k}^m} - \beta \sum_{m,h=1}^M \sum_{k=1}^K \frac{(S_{.,k}^m)^T S_{.,k}^h}{\|S_{.,k}^m\|_2 \|S_{.,k}^h\|_2}, \tag{10}$$

where β is the parameter for controlling the contributions from intra- and inter-network connections. The optimization problem is formulated as

$$\begin{aligned} & \min \varphi(S^1, S^2, \dots, S^M) \\ & \text{s.t. } S_{i,k}^m \in \{0, 1\}, i = 1, 2, 3, \dots, N; m = 1, 2, 3, \dots, M; k = 1, 2, 3, \dots, K \\ & \sum_{k=1}^K S_{i,k}^m = 1 \quad \text{for } m = 1, 2, 3, \dots, M, \end{aligned} \tag{11}$$

where $S^m_{,k}$ represents the k -th column of S^m . We can cluster the nodes in each network from the first term in the objective function and obtain the alignment of clusters in different networks from the second term. The aligned clusters in all the views are obtained by solving optimization problem (11).

As shown below, the preceding optimisation problem is relaxed to a multiview spectral clustering optimization problem as follows:

$$\min_{X_m \in \mathbb{R}^{N \times K}} Tr(X^T LX) \text{ s.t. } X^T X = I, \tag{12}$$

where

$$L = \begin{pmatrix} L_1 & 0 & \cdots & 0 \\ 0 & L_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & L_M \end{pmatrix} - \beta \begin{pmatrix} 0 & I_n & \cdots & I_n \\ I_n & 0 & \cdots & I_n \\ \vdots & \vdots & \ddots & \vdots \\ I_n & I_n & \cdots & 0 \end{pmatrix} \quad X = \begin{pmatrix} X_1 \\ X_2 \\ \vdots \\ X_M \end{pmatrix}. \tag{13}$$

To relax both constraints in optimization problem (11), we transform the constraints for each network into a single equation with $X_m = \frac{S^m}{\|S^m\|_2}$. In the first constraint in optimization problem (11), the variables S^m are relaxed from binary values to real values. To combine the information of all networks, we relaxed the second constraint to $X^T X = I_K$. However, this relaxation will result in the loss of information in the network [23]. To preserve information, we solve the following optimization problem:

$$\min_{X_m \in \mathbb{R}^{N \times K}} Tr(X^T LX) \text{ s.t. } X_m^T X_m = I_K, \quad m = 1; 2; \dots; M. \tag{14}$$

Let $X_m = \frac{S^m}{\|S^m\|_2}$. Then, the constraint $X_m^T X_m = I_K$ is direct relaxation of $\sum_{k=1}^K S^m_{,k} = 1$, and it retains more information than in the optimization problem (11).

The problem (14) is an optimization problem with orthogonality constraints. The orthogonality constraints can be expressed as a Stiefel manifold $St_{n,p}$, which $X \in St_{n,p} = \{X \in \mathbb{R}^{n \times p} | X^T X = I_p\}$. Thus, problem (14) can be written as

$$\min_{\{X_m\}_{m=1}^M \in St_{K,N}} Tr(X^T LX), \tag{15}$$

which is an optimization problem defined in manifold.

Solving the optimization problem

In this step, we solve optimization problem (15) by using the penalty model, i.e. the first-order algorithm from [33]. This algorithm iteratively implements steps.

Step 1 As mentioned earlier, Xiao et al. [33] demonstrated that the original problem (1) and the penalty model 3 are equivalent. We begin by computing the gradient of the augmented Lagrangian [36] to approximate problem (15). The gradient of the augmented Lagrangian function with respect to X is formulated as

$$D_m = \nabla_X \varphi(X_m, \delta) = \nabla f(X_m) + X_m \delta(X_m) + \beta X_m (X_m^T X_m - I_p), \tag{16}$$

where $D_m = \nabla_X \varphi(X_m, \delta)$ denotes the gradient of the augmented Lagrangian, and $\nabla f(X_m)$ represents the gradient of the objective function 4. We computed the gradient of the objective function as

$$G = \nabla_X f(X) = \nabla_X \text{Tr}(X^T L X) = L X, \tag{17}$$

and $\delta(X)$ is the Lagrangian multiplier. The PLAM method in [36] is shown as the closed-form expression of the Lagrangian multipliers $\delta(X)$, which is defined as

$$\delta(X) = \psi(\nabla f(X)^T X), \tag{18}$$

where the operator ψ is defined from Eq. (2) as follows,

$$\psi(X) = \frac{X^T + X}{2}. \tag{19}$$

From Eqs. (18) and (19), we compute the Lagrangian multiplier to have the following form:

$$\delta(X) = \psi(\nabla \text{Tr}(X^T L X)^T X) = \psi((L X)^T X) = \frac{X^T (L^T + L) X}{2}. \tag{20}$$

We use Eq. (16) to calculate the gradient of the augmented Lagrangian.

Step 2 We perform the following iterations in this step. We compute $\tilde{X}_{m+1} = X_m - \mu_m D_m$, and then we check if $\|X_{m+1}\|_F$ is greater than the parameter r . If yes, then

$$X_{m+1} = \frac{r}{\|\tilde{X}_{m+1}\|_F} \tilde{X}_{m+1}. \tag{21}$$

If $\|X_{m+1}\|_F$ is not greater than r , then $X_{m+1} = \tilde{X}_{m+1}$, where μ_m is the stepsize, and r is the radius. We explain how we selected the two parameters in the “Results and discussion” section. We repeat the processing of the aforementioned steps until convergence. Finally, we obtain the values of X .

k-means

We begin by setting X as $N \times M$ points in R^K , and then use k -means clustering to obtain the clusters. The output is the class label of the patients (nodes) in each network. We call our method multiview clustering by using the penalty model (MVCPM).

Algorithm

In this subsection, we present the algorithm framework of the proposed method as shown in Algorithm 1. For convenience, we call it MVCPM.

Algorithm 1: Multiview clustering based on the penalty model (MVCPM)

Input: The original data.

1) Normalization of each feature across all data types to have a standard normal distribution by using $\tilde{g} = \frac{g - E(g)}{\sqrt{Var(g)}}$.

2) Construction of the k-NN similarity network.

Consider M types of omics data measurements $\{X_m\}_{m=1}^M$, such that X_m is a $\tilde{g}_m \times N$ matrix.

1- Compute the similarity matrix $S_m(i, j)$

$$S_m(i, j) = \exp\left(\frac{d^2}{2\alpha^2}\right), i = 1, 2, \dots, N, j = 1, 2, \dots, N;$$

2- Input the number of clusters K and the parameter β ;

3- Compute the adjacency matrix $A_m(i, j)$; $m = 1, 2, 3, \dots, M$

$$A_m(i, j) = \begin{cases} S_m(i, j), & \text{if } v_i \in \mathbf{N}_i \\ 0, & \text{otherwise.} \end{cases};$$

3) Compute the Laplacian matrices $L_m = D_m - A_m$ to get L ; Given an initial point of X^0 ;

4) Solve the optimization problem 15;

repeat until convergence

1- Compute the gradient of the augmented Lagrangian with respect to X :

$$D_m = \nabla_X \varphi(X_m, \delta) = \nabla f(X_m) + X_m \delta(X_m) + \beta X_m (X_m^T X_m - I_p)$$

2- Do the iterations:

$$\tilde{X}_{m+1} = X_m - \mu_m D_m$$

if $\|X_{m+1}\|_F > r$ then

$$X_{m+1} = \frac{r}{\|\tilde{X}_{m+1}\|_F} \tilde{X}_{m+1};$$

else

$$X_{m+1} = \tilde{X}_{m+1};$$

end

$m++$;

Return X_m .

5) Take X as $N \times M$ points in R^K , do k -means clustering to get the clusters C^1, C^2, \dots, C^K .

Output: The class label of the patients (nodes) in each network.

Example for the proposed method

Figure 1 presents an illustrative example of our method's steps.

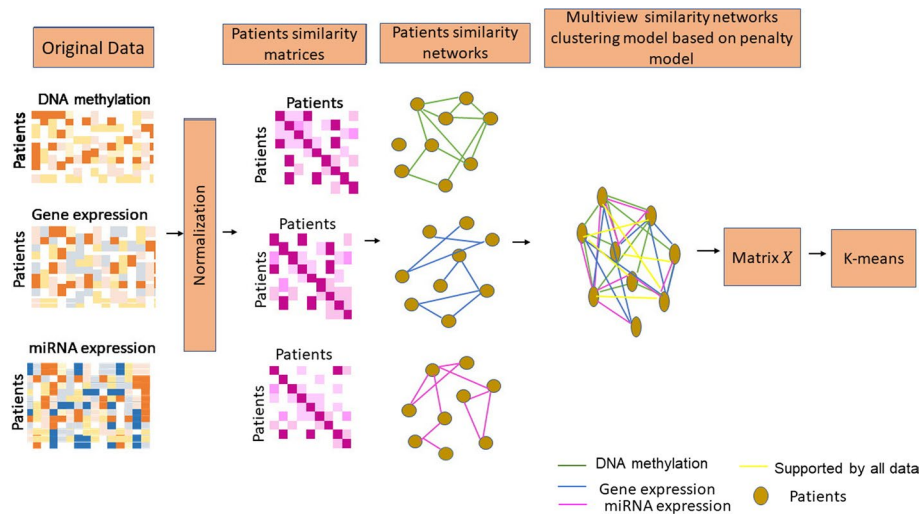


Fig. 1 Illustrative example of method steps

Results and discussion

We investigate the performance of our method (i.e. MVCPM) in this section. The performance of the proposed method is compared with several recently developed methods by using synthetic and real data. The comparison methods are the affinity aggregation for spectral clustering (AASC) algorithm [34], the multiview clustering based on manifold optimisation (MVCMO) method [23] and the multiview spectral clustering (MVSC) method [21].

Comparison using synthetic data

We begin by simulating network structures because all the methods mentioned earlier have been proposed using network tools. The networks are generated via the stochastic block model [39]. We begin by simulating network structures because all the methods mentioned earlier have been proposed using network tools. The networks are generated via the stochastic block model [39]. Assume that M views are present, and N is the number of nodes in each view. First, the number of N nodes is assigned to K . Then, the connections within and between different clusters are generated using a given probability matrix, in which diagonal entries specify connection probabilities within clusters. The other entries specify connection probabilities between corresponding clusters. To obtain the results when between cluster connections are changed, we use the following four connection probability matrices, with the within cluster connection probability being larger than that between clusters:

$$\begin{aligned}
 P_1 &= \frac{1}{n} \begin{pmatrix} 16 & 0 & 0 \\ 0 & 18 & 0 \\ 0 & 0 & 17 \end{pmatrix} & P_2 &= \frac{1}{n} \begin{pmatrix} 16 & 0.4 & 0.6 \\ 0.4 & 18 & 0.55 \\ 0.6 & 0.55 & 17 \end{pmatrix} \\
 P_3 &= \frac{1}{n} \begin{pmatrix} 16 & 0.8 & 1.2 \\ 0.8 & 18 & 1.1 \\ 1.2 & 1.1 & 17 \end{pmatrix} & P_4 &= \frac{1}{n} \begin{pmatrix} 16 & 1.2 & 1.8 \\ 1.2 & 18 & 1.65 \\ 1.8 & 1.65 & 17 \end{pmatrix}.
 \end{aligned} \tag{22}$$

The entries of main diagonal (l, l) for each probability matrix represent the connection probabilities within cluster l , whilst the entries (l, s); $l \neq s$ represent the connection probabilities between the cluster l and cluster s .

To evaluate its performance, the proposed method is compared with existing methods in the literature under different scenarios. Firstly, the performance of the numbers of M views and N nodes is analysed when setting the same cluster size. For example, we set the cluster size as $(50, 50, 50)N = 150$ and $M = 3$ to test the performance of the proposed method and as $(200, 200, 200)$, $N = 600$ and $M = 5$ to emphasise the superiority of our presented method. Meanwhile, performance is analysed under different cluster sizes, i.e. cluster size is set as $(50, 50, 50)$, $(30, 90, 30)$, $(40, 60, 50)$ with $N = 150$, $M = 3$, and $(200, 150, 250)$, $(230, 270, 100)$, $(180, 160, 260)$, $(150, 310, 140)$, $(130, 250, 220)$ when $N = 600$ and $M = 5$.

To see the performance of all the methods, we repeat the experiments 50 times for each setting. Then, we calculate the identification accuracy of the clusters, wherein identification accuracy is represented by the Rand index, which is defined as

Table 1 Comparison between the performance of different methods, when the clusters size the same across all M views, $N = 150$, $M = 3$, and cluster size: (50, 50, 50), (50, 50, 50), (50, 50, 50)

| Method | P_1 | P_2 | P_3 | P_4 |
|--------|------------|------------|------------|------------|
| MVCPM | 1.00(0.00) | 1.00(0.00) | 1.00(0.00) | 1.00(0.00) |
| MVSC | 1.00(0.00) | 1.00(0.00) | 1.00(0.00) | 0.99(0.01) |
| MVCMO | 1.00(0.00) | 1.00(0.00) | 1.00(0.00) | 0.99(0.01) |
| AASC | 1.00(0.00) | 1.00(0.00) | 1.00(0.00) | 1.00(0.00) |

Table 2 Comparison between the performance of different methods, when the clusters size is different across all M views, $M = 3$, $N = 150$, and cluster size: (50, 50, 50), (30, 90, 30), (40, 60, 50)

| Method | P_1 | P_2 | P_3 | P_4 |
|--------|------------|------------|------------|------------|
| MVCPM | 1.00(0.00) | 1.00(0.01) | 0.97(0.00) | 0.97(0.01) |
| MVSC | 0.99(0.01) | 0.98(0.01) | 0.94(0.12) | 0.89(0.16) |
| MVCMO | 0.99(0.01) | 0.99(0.01) | 0.97(0.07) | 0.97(0.02) |
| AASC | 0.73(0.00) | 0.73(0.00) | 0.73(0.01) | 0.73(0.01) |

Table 3 Comparison between the performance of different methods. When the size of the cluster is the same across all M views, $M = 5$, $N = 600$, and cluster size: (200, 200, 200)

| Method | P_1 | P_2 | P_3 | P_4 |
|--------|------------|------------|------------|------------|
| MVCPM | 1.00(0.00) | 1.00(0.00) | 1.00(0.00) | 1.00(0.00) |
| MVSC | 1.00(0.00) | 1.00(0.00) | 1.00(0.00) | 1.00(0.00) |
| MVCMO | 1.00(0.00) | 1.00(0.00) | 1.00(0.00) | 1.00(0.00) |
| AASC | 1.00(0.00) | 1.00(0.00) | 1.00(0.00) | 1.00(0.00) |

Table 4 Comparison between the performance of different methods. When the size of the cluster is different across all M views, $M = 5$, $N = 600$, and cluster size: (200, 150, 250), (230, 270, 100), (180, 160, 260), (150, 310, 140), (130, 250, 220)

| Method | P_1 | P_2 | P_3 | P_4 |
|--------|------------|------------|------------|------------|
| MVCPM | 0.92(0.01) | 0.93(0.00) | 0.93(0.01) | 0.89(0.02) |
| MVSC | 0.91(0.01) | 0.90(0.02) | 0.77(0.06) | 0.76(0.04) |
| MVCMO | 0.90(0.01) | 0.91(0.00) | 0.90(0.02) | 0.78(0.03) |
| AASC | 0.56(0.56) | 0.56(0.56) | 0.56(0.56) | 0.56(0.56) |

$$R = \frac{TP + TN}{TP + TN + FP + FN}, \tag{23}$$

where TP , TN , FP and FN represent the numbers of true positive, true negative, false positive and false negative, respectively. TP is the number of nodes in clusters i and j , and the others can be defined similarly.

The results are presented in Tables 1, 2, 3 and 4. To see the results of the 50 times experiments, we list the mean Rand index and the standard deviation within brackets for each setting.

As shown in Tables 1 and 3, when cluster sizes are the same across all M views, all four methods can cluster the nodes with a mean Rand index close to 1 and a lower standard deviation. Thus, all the methods exhibit comparable performance and demonstrate the significance of data integration, with AASC and our method performing the best with a ratio of 0.01.

Tables 2 and 4 present the results of the four methods when cluster sizes are different. Our method outperforms all the other methods. The primary reason is solving optimization problem 15. Most of the other methods formulate the optimization problem under the assumption that cluster sizes are the same across different networks. Then, they focus on finding the common features of the cluster to perform clustering. In AASC, for example, the M vectors that minimize the ratio cut of the weighted combination of all the networks are computed. By clustering these M vectors, the common cluster for all the considered networks can be obtained. In Tables 2 and 4, the mean Rand index of the AASC method is lower than those of the other methods. Consequently, all the methods can detect cluster differences between different views, whereas the AASC method regards them as the same. Thus, MVCPM, MVSC and MVCMO exhibit better performance than the AASC method. The MVCPM method outperforms MVSC and MVCMO, which has the highest mean Rand index. The MVCPM method performs more consistently when networks are fewer, resulting in a considerably lower standard deviation of the Rand index.

Selection of the parameters of the proposed method

In this section, we describe the method for choosing the values of the penalty parameter β , the stepsize μ_k , and the radius r .

Penalty parameter β . In Theorem 4.1 [33], β should be sufficiently large to guarantee convergence. Although, we can estimate a suitable β that meets the requirements of previous theorems, such β is too large to be practically helpful. In our method, we set β to be less than $t = \|f(X_0)\|$. In the selected penalty model, the compact convex set as the ball B_r with radius $r > \sqrt{p}$.

Stepsize μ_k Practically, the upper bound of μ_k used in Theorem 4.1 is too restrictive. Therefore, we recommend using the alternating Barzilai–Borwein (ABB) step size [15] and the Barzilai–Borwein (BB) stepsize [40].

ABB step size

$$\mu_m^{ABB} = \begin{cases} \mu_m^{BB1}, & \text{for odd } m \\ \mu_m^{BB2}, & \text{for even } m. \end{cases} \tag{24}$$

BB step size

$$\mu_m^{BB1} = \frac{S_{m-1}^T S_{m-1}}{S_{m-1}^T Y_{m-1}} \quad \mu_m^{BB2} = \frac{S_{m-1}^T Y_{m-1}}{Y_{m-1}^T Y_{m-1}} \tag{25}$$

where $S_{m-1} = X_m - X_{m-1}$ and $Y_{m-1} = D_m - D_{m-1}$.

Moreover, we compare the penalty model for solving problem (15) with $n = 1000$, $p = 30$ and different pairs of β and r : $\beta = 10^{-6}t, 10^{-5}t, 10^{-4}t, 10^{-3}t, 10^{-2}t, 10^{-1}t, t, 10t$ and $r = \sqrt{p}, 1.01\sqrt{p}, 1.04\sqrt{p}, 1.06\sqrt{p}, 1.3\sqrt{p}, 1.8\sqrt{p}, 3\sqrt{p}, 7\sqrt{p}, 10\sqrt{p}, \sqrt{p}$.

The results are presented in Fig. 2.

From Fig. 2, we can conclude that our algorithm is insensitive to the choice of r when $r \leq 2$. Furthermore, a small β can lead to a fast convergence rate. Consequently, we suggest choosing $\beta = 0.01t$ whilst $r = 1.04\sqrt{p}$.

To guarantee fair comparison over the literature, we set parameter $\beta = 1$ to determine the performance of our method compared with those of other methods.

Real data experiments

In this section, we use five real different types of cancer data to evaluate the performance of the proposed method and compare it with those of other methods. We select two commonly used measures, namely, silhouette score to measure clustering performance [41] and the P-value in the Cox log-rank test to evaluate the significance of difference between survival times [42].

Dataset

This study uses real multi-omics data generated by TCGA. We use the five data types of cancer processed in [16]. The data types are glioblastoma multiforme (GBM), lung squamous cell carcinoma (LSCC), kidney renal clear cell carcinoma (KRCCC) and colon adenocarcinoma (COAD). Each data type of cancer contains miRNA expression, DNA methylation and gene expression data type.

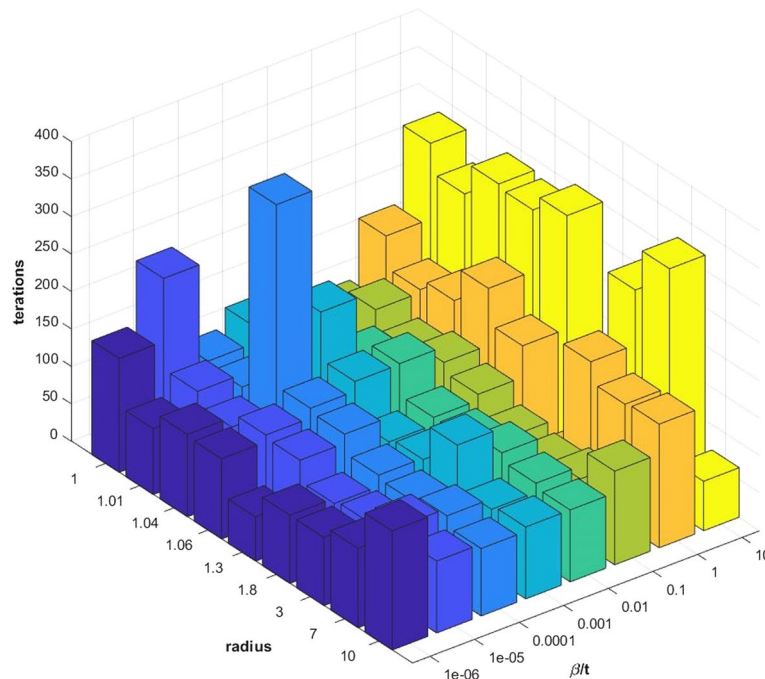


Fig. 2 Show the results of the different settings for the parameters of the penalty model for solving the problem (15)

Comparison using silhouette score

We begin by constructing the k -NN similarity networks for each data type, choosing the five nearest neighbours and setting the number of clusters K as that in [16]. Then, we apply the proposed method to the k -NN similarity networks to obtain the final result.

We use the silhouette score to measure the clustering performance of our method (MVCPM) compared with the performance of other methods [41]. For the AASC method, we use the code developed in [34]. For the MVCPM and MVSC methods, we use the code generated in [23]. We compute the silhouette score of the common clusters and the average silhouette score of all data types to obtain the cluster assignment in each dataset and the common clusters across the three datasets. We also calculate the silhouette score of integrative clustering by using k -NN to assign patients who are previously clustered in different clusters into one cluster because cluster assignment in the three data types is not the same. We denote the silhouette score of the common clusters, the average silhouette score and the silhouette score for integrative clustering as (MVCPM-com, MVCMOcom, MVSCcom), (MVCPMavg, MVCMOavg, MVSCavg) and (MVCPM-int, MVCMOint, MVSCint), respectively. The results are provided in Table 5.

From Table 5, the common clusters identified using our method provide the highest silhouette score. The proposed method can capture common cluster structures efficiently in multiple views. The average silhouette scores are also higher than those using other methods in three of the four datasets. This result shows that MVCPM maintains the clustering results in each dataset. The cluster assignment in the three data types are not the same; thus, we use k -NN to assign patients that are originally clustered into different clusters into one cluster and compute silhouette scores as MVCPM-int, MVCMOint, AASCint and MVSCint. In this case, the common silhouette score and average silhouette scores of all the methods are higher when compared with AASC, implying that all the methods, except for AASC, can identify the common clusters and capture the common cluster structures efficiently across multiview data. Moreover, all the methods have a high silhouette score for integrative clustering, except for AASC, indicating that all the methods can integrate clustering with good performance.

When we compare MVCPM, MVCMO and MVSC, MVCPM has the highest silhouette score for common clusters and the average silhouette score. MVCPM provides more detailed information within each data type, is better for integrating different types of

Table 5 Comparisons all methods with real multi-omics data

| Method | KRCC | COAD | GBM | LSCC |
|----------|--------|--------|--------|--------|
| MVCPMcom | 0.5021 | 0.4169 | 0.4206 | 0.4236 |
| MVCPMavg | 0.4948 | 0.4100 | 0.4200 | 0.4437 |
| MVCPMint | 0.4904 | 0.3631 | 0.3870 | 0.3431 |
| MVSCcom | 0.3869 | 0.3853 | 0.2740 | 0.2345 |
| MVSCavg | 0.3782 | 0.3619 | 0.2395 | 0.2127 |
| MVSCint | 0.3786 | 0.2952 | 0.2136 | 0.1732 |
| MVCMcom | 0.4177 | 0.3958 | 0.2797 | 0.2475 |
| MVCMOavg | 0.3969 | 0.3725 | 0.2548 | 0.2296 |
| MVCMOint | 0.3581 | 0.2836 | 0.2063 | 0.1812 |
| AASC | 0.2815 | 0.2926 | 0.1944 | 0.1803 |

omics data and simultaneously has consistent and differential cluster patterns. MVCP-Mcom and MVCPM avg are greater than MVCMO avg, MVCMOcom, MVSCavg and MVSCcom; therefore, our method has the highest silhouette score for integrative clustering. For example, MVCPMint in COAD is about 0.07 higher than MVCMOint, whilst MVCPMint is about 0.06 higher than MVSCint. Thus, MVCPM is the best method for integration the clustering.

Comparison by Cox survival P-values

In this subsection, we compare the performance of our proposed method with those of other methods by using the P-value in the Cox log-rank test to determine the significance of the difference in survival times [42]. The lowest P-values are used to determine the number of clusters for each cancer type in our proposed method. To ensure that the proposed method’s results are comparable with those of the other methods, we set the number of clusters for each cancer type for the other methods to be the same as the number of clusters for our proposed method. The results are provided in Table 6.

As indicated in Table 6, our method provides more significant differences in survival times across all types of cancer studied compared with the other methods. The P-value produced is comparable with the P-value produced by the other methods. Thus, the survival plots for all methods across COAD, GBM, KRCCC and LSCC cancers are presented in Figs. 3, 4, 5 and 6.

Conclusion

This paper proposed a new method for multiview clustering of This study proposed a new method for the multiview clustering of multi-omics data based on a penalty model. Synthetic data and four different types of real multi-omics data were used to test the performance of the proposed method. We adopted the block model to generate these data in synthetic data and applied the proposed method and other methods. To ensure the performance of the proposed method, we compared it with some recently developed multiview clustering methods by using synthetic data. The results showed that when the underlying clusters were consistent, the proposed method exhibited competitive performance with the recently developed methods. However, the performance of the proposed method was better when the underlying clusters were differential.

For real multi-omics data, we downloaded the gene expression, miRNA expression and DNA methylation datasets for five different types of real multi-omics data. These are GBM, COAD, LSCC and KRCCC from TCGA. To test the clustering performance of all the methods, we used two commonly used measures: silhouette score to measure clustering performance and P-value in Cox log-rank test to evaluate the significance

Table 6 Comparison of Cox survival p-values for all methods across all types of cancer

| Cancer type | Our method MVCPM | MVCMO | MVSC | AASC |
|-------------------|----------------------|----------------------|----------------------|----------------------|
| COAD (6clusters) | 3.7×10^{-2} | 6.5×10^{-2} | 3.4×10^{-2} | 13×10^{-2} |
| GBM (4clusters) | 1.2×10^{-4} | 1.8×10^{-2} | 1.9×10^{-3} | 4×10^{-3} |
| LSCC (3clusters) | 8.6×10^{-3} | 9×10^{-3} | 8.6×10^{-3} | 8.4×10^{-2} |
| KRCCC (3clusters) | 38×10^{-2} | 26×10^{-2} | 38×10^{-2} | 41×10^{-2} |

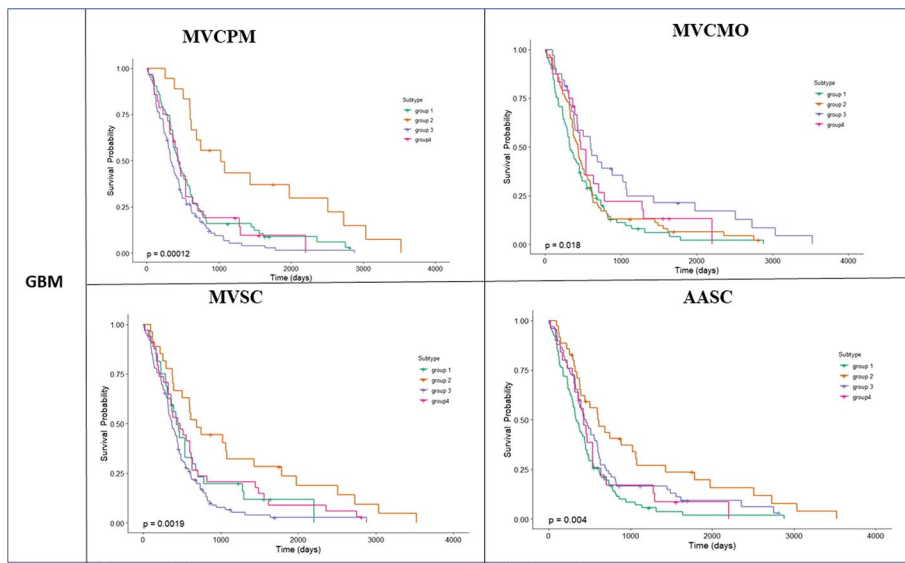


Fig. 3 Kaplan-Meier survival curves of glioblastoma multiforme (GBM) of all methods (p-values are recorded in Table 6)

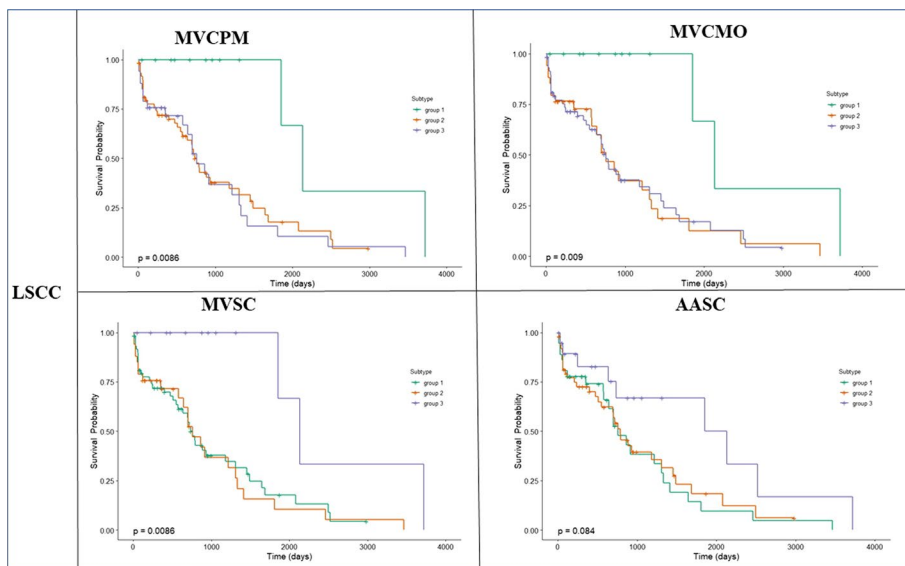


Fig. 4 Kaplan-Meier survival curves of lung squamous cell carcinoma (LSCC) of all methods (p-values are recorded in Table 6)

of difference between survival times. For the silhouette score, the proposed method obtained the highest silhouette score and provided more detailed information within each data type. In Cox survival P-values, the proposed method resulted in more significant differences in survival times across all studied data types.

Thus, the proposed method can be considered the best approach for integration and clustering. Although our method performed well in synthetic and real omics data analyses, some issues still need to be addressed. For example, we set parameter $\beta = 1$ and the number of clusters to be three throughout our study to guarantee fair comparison

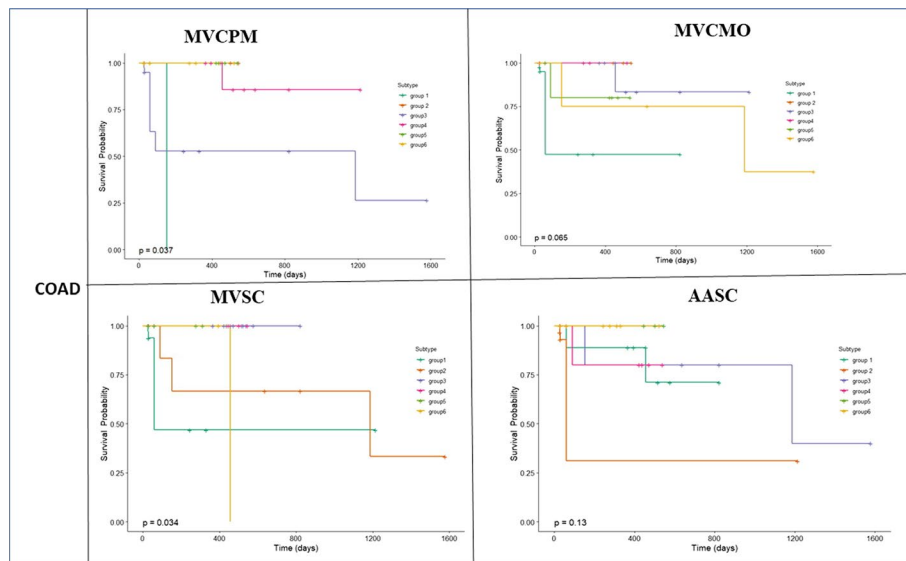


Fig. 5 Kaplan-Meier survival curves of colon adenocarcinoma (COAD) of all methods (p-values are recorded in Table 6)

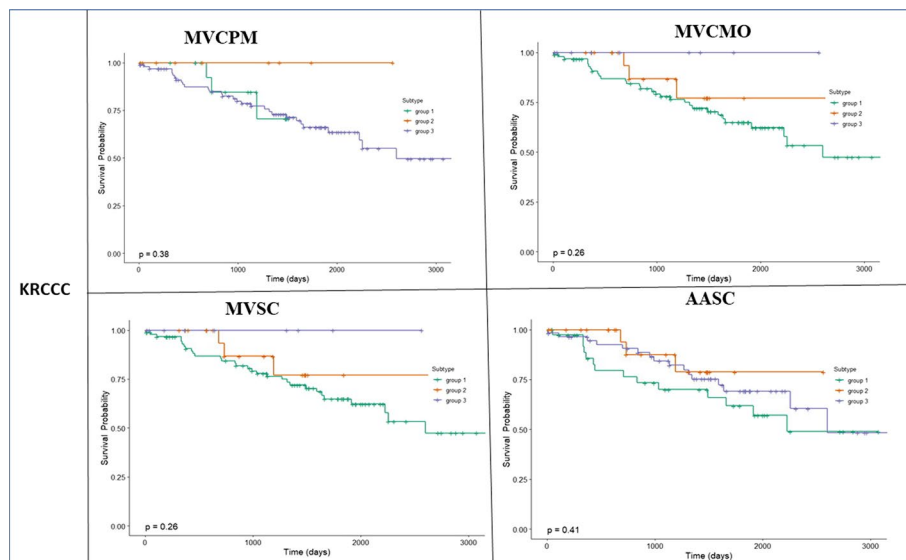


Fig. 6 Kaplan-Meier survival curves of kidney renal clear cell carcinoma (KRCCC) of all methods (p-values are recorded in Table 6)

with conventional methods. However, the number of clusters is not practically optional because it must be selected on the basis of the eigengap. Moreover, no guarantee theoretically exists to determine the optimal number of clusters. Consequently, developing an excellent statistical method for determining the optimal number of clusters in network-based clustering is still a worthwhile research.

Abbreviations

- MVCPM Our method
- AASC Affinity aggregation for spectral clustering algorithm

| | |
|-------|--|
| MVCMO | Multiview clustering based on manifold optimization method |
| MVSC | Multiview spectral clustering method |
| GBM | Glioblastoma multiforme |
| LSCC | Lung squamous cell carcinoma |
| KRCCC | Kidney renal clear cell carcinoma |
| COAD | Colon adenocarcinoma |

Acknowledgements

We would like to thank Prof. Ma for her insightful comments in reviewing the manuscript.

Author contributions

HAA-K: performed all analyses, generated the results, and wrote the manuscript. MEA wrote some functions code in Matlab. MS read and review the manuscript. EAA-H and SAA-M contributed to the interpretation and the discussion of results. AAA-K and AAQA checked for language and typos errors. All authors read and approved the final version of the manuscript.

Funding

This work was supported by grants from project of Interface system software design, development, and test of accelerator physics control system PACS, IMPJFCG20358.

Availability of data and materials

Data sets The datasets generated and/or analysed during the current study are available in the GitHub repository, <https://github.com/hamas200/MVCPM>.

Code availability

Matlab code for our method from (<https://github.com/hamas200/MVCPM>.) and Rstudio for analysis the p-value for all method described in this paper. The comparison methods code downloaded from <http://homepage.fudan.edu.cn/sqzhang/files/2018/12/MVCMOcode.zip>.

Declarations

Ethics approval and consent to participate

Only published data that are publicly available online were used in this paper.

Competing interests

The authors declare that they have no competing interests.

Received: 18 December 2021 Accepted: 20 June 2022

Published online: 21 July 2022

References

1. Nguyen ND, Wang D. Multiview learning for understanding functional multiomics. *PLoS Comput Biol*. 2020;16(4):1007677.
2. Shields CL, Mayo EL, Bas Z, Dockery PW, Yaghy A, Lally SE, Ganguly A, Shields JA. Ten-year outcomes of uveal melanoma based on the cancer genome atlas (TCGA) classification in 1001 cases. *Indian J Ophthalmol*. 2021;69(7):1839–45.
3. Von Luxburg U. A tutorial on spectral clustering. *Stat Comput*. 2007;17(4):395–416.
4. West M, Blanchette C, Dressman H, Huang E, Ishida S, Spang R, Zuzan H, Olson JA, Marks JR, Nevins JR. Predicting the clinical status of human breast cancer by using gene expression profiles. *Proc Natl Acad Sci*. 2001;98(20):11462–7. <https://doi.org/10.1073/pnas.201162998>.
5. Huang E, Cheng SH, Dressman H, Pittman J, Tsou MH, Horng CF, Bild A, Iversen ES, Liao M, Chen CM, et al. Gene expression predictors of breast cancer outcomes. *Lancet*. 2003;361(9369):1590–6.
6. Chang HY, Nuyten DS, Sneddon JB, Hastie T, Tibshirani R, Sørlie T, Dai H, He YD, van't Veer LJ, Bartelink H, et al. Robustness, scalability, and integration of a wound-response gene expression signature in predicting breast cancer survival. *Proc Natl Acad Sci*. 2005;102(10):3738–43.
7. Herrero J, Valencia A, Dopazo J. A hierarchical unsupervised growing neural network for clustering gene expression patterns. *Bioinformatics*. 2001;17(2):126–36.
8. Siegmund KD, Laird PW, Laird-Offringa IA. A comparison of cluster analysis methods using DNA methylation data. *Bioinformatics*. 2004;20(12):1896–904.
9. Li W, Liu C-C, Zhang T, Li H, Waterman MS, Zhou XJ. Integrative analysis of many weighted co-expression networks using tensor computation. *PLoS Comput Biol*. 2011;7(6):1001106.
10. Shen R, Olshen AB, Ladanyi M. Integrative clustering of multiple genomic data types using a joint latent variable model with application to breast and lung cancer subtype analysis. *Bioinformatics*. 2009;25(22):2906–12.
11. Shen R, Mo Q, Schultz N, Seshan VE, Olshen AB, Huse J, Ladanyi M, Sander C. Integrative subtype discovery in glioblastoma using iCluster. *PLoS One*. 2012;7(4):35236.
12. Speicher NK, Pfeifer N. Integrating different data types by regularized unsupervised multiple kernel learning with application to cancer subtype discovery. *Bioinformatics*. 2015;31(12):268–75.
13. Rappoport N, Shamir R. Nemo: cancer subtyping by integration of partial multi-omic data. *Bioinformatics*. 2019;35(18):3348–56.

14. Ding H, Sharpnack M, Wang C, Huang K, Machiraju R. Integrative cancer patient stratification via subspace merging. *Bioinformatics*. 2019;35(10):1653–9.
15. Dai Y-H, Fletcher R. Projected Barzilai-Borwein methods for large-scale box-constrained quadratic programming. *Numer Math*. 2005;100(1):21–47.
16. Wang B, Mezlini AM, Demir F, Fiume M, Tu Z, Brudno M, Haibe-Kains B, Goldenberg A. Similarity network fusion for aggregating data types on a genomic scale. *Nature Methods*. 2014;11(3):333–7.
17. Amar D, Safer H, Shamir R. Dissection of regulatory networks that are altered in disease via differential co-expression. *PLoS Comput Biol*. 2013;9(3):1002955.
18. Li D, Brown JB, Orsini L, Pan Z, Hu G, He S. Moda: Module differential analysis for weighted gene co-expression network; 2016. arXiv preprint [arXiv:1605.04739](https://arxiv.org/abs/1605.04739)
19. Ha MJ, Baladandayuthapani V, Do K-A. Dingo: differential network analysis in genomics. *Bioinformatics*. 2015;31(21):3413–20.
20. Maxwell R, Jackson CM, Lim M. Clinical trials investigating immune checkpoint blockade in glioblastoma. *Curr Treat Options Oncol*. 2017;18(8):1–22.
21. Zhang S, Zhao H, Ng MK. Functional module analysis for gene coexpression networks with network integration. *IEEE/ACM Trans Comput Biol Bioinform*. 2015;12(5):1146–60.
22. Kumar A, Rai P, Daume H. Co-regularized multi-view spectral clustering. *Adv Neural Inf Process Syst*. 2011;24:1413–21.
23. Yu Y, Zhang L-H, Zhang S. Simultaneous clustering of multiview biomedical data using manifold optimization. *Bioinformatics*. 2019;35(20):4029–37.
24. Abrudan TE, Eriksson J, Koivunen V. Steepest descent algorithms for optimization under unitary matrix constraint. *IEEE Trans Signal Process*. 2008;56(3):1134–47.
25. Manton JH. Optimization algorithms exploiting unitary constraints. *IEEE Trans Signal Process*. 2002;50(3):635–50.
26. Nishimori Y, Akaho S. Learning algorithms utilizing quasi-geodesic flows on the Stiefel manifold. *Neurocomputing*. 2005;67:106–35.
27. Abrudan T, Eriksson J, Koivunen V. Conjugate gradient algorithm for optimization under unitary matrix constraint. *Signal Process*. 2009;89(9):1704–14.
28. Dai X, Zhang L, Zhou A. Adaptive step size strategy for orthogonality constrained line search methods; 2019. arXiv preprint [arXiv:1906.02883](https://arxiv.org/abs/1906.02883)
29. Absil P-A, Mahony R, Sepulchre R. Optimization algorithms on matrix manifolds. Princeton: Princeton University Press; 2009.
30. Jiang B, Dai Y-H. A framework of constraint preserving update schemes for optimization on Stiefel manifold. *Math Program*. 2015;153(2):535–75.
31. Wen Z, Yin W. A feasible method for optimization with orthogonality constraints. *Math Program*. 2013;142(1):397–434.
32. Gao B, Liu X, Chen X, Yuan Y-X. A new first-order algorithmic framework for optimization problems with orthogonality constraints. *SIAM J Optim*. 2018;28(1):302–32.
33. Xiao N, Liu X, Yuan Y-X. A class of smooth exact penalty function methods for optimization problems with orthogonality constraints. *Optim Methods Softw*. 2020. <https://doi.org/10.1080/10556788.2020.1852236>.
34. Huang H-C, Chuang Y-Y, Chen C-S. Affinity aggregation for spectral clustering. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition, IEEE; 2012. p. 773–780
35. Chen C, Ng MK, Zhang S. Block spectral clustering methods for multiple graphs. *Numer Linear Algebra Appl*. 2017;24(1):2075.
36. Gao B, Liu X, Yuan Y-X. Parallelizable algorithms for optimization problems with orthogonality constraints. *SIAM J Sci Comput*. 2019;41(3):1949–83.
37. Ng AY, Jordan MI, Weiss Y. On spectral clustering: analysis and an algorithm. *Adv Neural Inf Process Syst*. 2002;14:849–56.
38. Veenstra P, Cooper C, Phelps S. Spectral clustering using the kNN-MST similarity graph. In: 2016 8th Computer Science and Electronic Engineering (CEECE), IEEE; 2016. pp. 222–227.
39. Bickel PJ, Chen A. A nonparametric view of network models and Newman–Girvan and other modularities. *Proc Natl Acad Sci*. 2009;106(50):21068–73.
40. Barzilai J, Borwein JM. Two-point step size gradient methods. *IMA J Numer Anal*. 1988;8(1):141–8.
41. Rousseeuw PJ. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *J Comput Appl Math*. 1987;20:53–65.
42. Lemeshow S, May S, Hosmer DW. Applied survival Analysis: regression modeling of time-to-event data. In: Hosmer DW, Lemeshow S, May S, editors. Wiley in probability and statistics. Wiley; 2011. <https://books.google.com/books?id=lvvOopIqzWsC>

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.