5

Recursive Integration of Synergised Graph Representations for Cancer Subtypes Identification

5.1 INTRODUCTION

Cancer is a heterogeneous disease with diverse pathogeneses, and clinical features that can develop in different tissues and cell types (Stingl and Caldas, 2007). A cancer subtype can be defined as a subcategory of specific cancer; for example, Cervical cancer can be further grouped into Adenocarcinomas and Squamous cell carcinomas. Multiple subtypes are distinguishable based on molecular profiles, histology, or sometimes specific mutation. Patient-specific medicines are provided in personalized medicine practices rather than generic medicine. Therefore, for effective treatment of any cancer, it is crucial to identify the proper cancer subtype to provide an effective prognosis (Liang et al., 2015).

With the advancement of technologies, it has become very easy to generate high-dimensional multi-omics data for an individual. Multi-omics data include miRNA and mRNA expressions, DNA methylation, reverse protein phase assays, and others. These datasets are publicly available at various databases like TCGA (Tomczak et al., 2015). Accumulating various omics and transcriptome data opens the opportunity to develop novel computational methods that can integrate the tremendous amount of multi-view information available for cancer subtype identification. The usual practice to identify cancer subtypes is clustering cancer patient data. By grouping the cancer patients based on their genetic profiles, one can better understand the pathogenic mechanisms behind the disease. This will later help in the development of subtype-specific anticancer treatments. However, several challenges exist in grouping the cancer patients and integrating multi-omics data.

The multi-view omics data integration and clustering of cancer patients are considerably new research areas. Few algorithms are developed to address the challenges of this research area. A decade ago, researchers used single omics data to cluster cancer subtypes. Several studies are done using only gene expression data (Therese et al., 2001; Bhattacharjee et al., 2001; Monti et al., 2003b) or DNA methylation data (Teschendorff et al., 2007) or Copy number data (Zhang et al., 2017) to identify cancer subtypes. These algorithms perform clustering across the samples to capture the homogeneity present within the patients based on expression levels of a specific biomarker. Since acquiring cancer hallmarks requires molecular alterations at multiple levels, these algorithms fail to establish the causal relationship between molecular signatures. This biological phenomenon indicates the need for algorithms that integrates multi-omics data to identify cancer subtype. In this regard, integrative clustering-based approaches help capture underlying molecular mechanisms behind deadly cancer. Further, these algorithms can be categorized into two groups. The first group of algorithms separately identifies clusters from each omic data. Later, it combines these clustering results to obtain a global cluster that represents cancer subtypes (Network, 2012c,b; Hoadley et al., 2014; Gabasova et al., 2017). These algorithms mostly perform final clustering on individual clusters obtained from different omic datasets using a voting mechanism. Different voting mechanisms generate different clustering solutions. The second group of integrative clustering-based approaches integrates the multi-view omics data and then applies clustering to obtain cancer subtypes (Bo et al., 2014; Shen et al., 2009; Ronglai et al., 2012; Zhang et al., 2013b). Sometimes the multi-view data are concatenated or stacked together, and clustering is performed to identify cancer subtypes. Data concatenation may lead to information loss and amplifies the curse of dimensionality (Zhang et al., 2013b). On the other hand, to overcome the above-mentioned limitation, some algorithms are developed to extract informative subspace

from each of the omics datasets and then perform clustering on the integrated dataset (Shen et al., 2009; Ronglai et al., 2012; Zhang et al., 2013b; Tibshirani, 2015; Khan and Maji, 2020, 2021a). There are some methods available that model the variation of each view first with the help of similarity graphs and integrate them to identify clusters (Bo et al., 2014; Xu et al., 2016b; Khan and Maji, 2021a; Jiang et al., 2019). The challenge here is to find the best possible way of integration to capture the essence of all the views coming from different types of genomic information available for the same set of samples. The research area devoted to this type of problem is multi-view learning (Long et al., 2008; Xia et al., 2010; Zhou and Burges, 2007).

In this study, a novel algorithm named RISynG is presented. The proposed approach treats multi-omics data clustering as multi-view clustering, where information from multiple omics platforms is integrated to identify clinically important sub-groups within cancer. The proposed approach works in three steps to capture the variation across the multi-omics dataset. At first, for each view, two sample similarity matrices are computed using graph representation matrices, namely, the Gramian matrix and the Laplacian matrix. This step acknowledges the statistical diversity in the multi-view omics data, directly influencing the quantification of similarity between the samples. Later, it involves the integration of representation matrices for the respective omic-view using a parameterized combination function to generate synergy matrices. In the second step, the variation captured through synergy matrices for each omic-view is fused. The proposed approach first arranges all the synergy matrices based on their relevance. Then, a recursive function is designed to merge each synergy matrix so that the less relevant matrix only slightly influences the final cluster structures. At the end of this process, the final accretive basis of the accretive subspace is obtained, whose first k eigenvectors hold the cluster structure. At last, k-means clustering is applied on the rows of the accretive basis matrix to generate cluster labels. The efficacy of the proposed algorithm is extensively studied on five multi-omics cancer datasets and compared with existing multi-view clustering approaches used for cancer subtypes identification.

5.2 PROPOSED APPROACH FOR CANCER-SUBTYPES IDENTIFICATION

This section describes the novel algorithm designed in this study to integrate multi-omics data for cancer subtypes identification. The proposed method integrates multi-view data using a recursive multi-kernel integration function. To harness the best picture of sample similarities from each omic view, the proposed approach uses graphical representation and explores each view's statistical property. The schematic workflow of RISynG is presented in Figure 5.1. Before moving to the proposed algorithm's steps, the required analytical formulations are first discussed.

5.2.1 Gramian Matrix and Kernel Trick

Gramian matrix, $G = [g_{ij}]_{n\times n}$ is a Hermitian matrix, in which each element is a pairwise Hermitian inner product of the vectors in a Hausdorff pre-Hilbert space, V=$\{v_1, v_2, v_3, ..., v_n\}$.

$$G(v_1,\ldots,v_n) = \begin{bmatrix} < v_1, v_1 > & \ldots & < v_1, v_n > \\ < v_2, v_1 > & \ldots & < v_2, v_n > \\ \vdots & \ddots & \vdots \\ < v_n, v_1 > & .... & < v_n, v_n > \end{bmatrix} \quad , v_i \in \mathbb{R}^d$$

The Hermitian inner product space is accompanied by the geometric notions associated with the vectors, such as the length and the angle between two vectors. Since G is a Hermitian matrix, it inherits all the properties portrayed by a Hermitian matrix. A few of the relevant properties are enlisted below Jha (2016).

Property 1. *All the eigenvalues of G are real.*
Proof: Eigenvalues of a matrix are the roots of its characteristic equation. The characteristic equation for matrix G is written as:

$$\det(\lambda I - G) = 0 \tag{5.22}$$

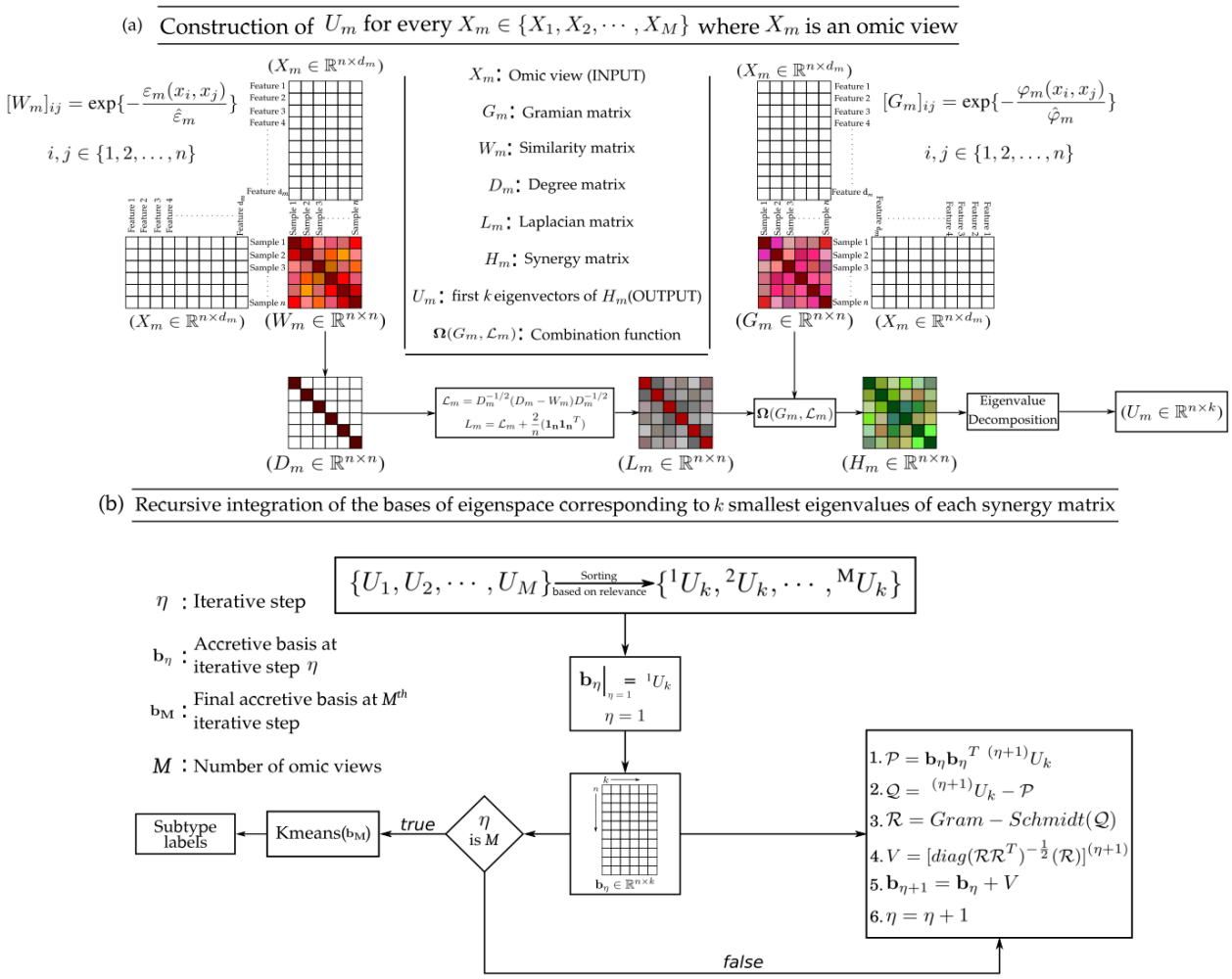


Figure 5.1. : Schematic flow diagram of the proposed approach for cancer subtypes identification

Let the root be some complex number $\lambda = a + ib, a, b \in \mathbb{R}, b \neq 0$ and I be the identity matrix of same order. Since, at this value of λ, the characteristic equation has a non-empty kernel, there must exist a vector $u = x + iy, x, y \in \mathbb{R}$ such that:

$$Gu = \lambda u \tag{5.23}$$

or,

$$G(x + iy) = (a + ib)(x + iy) \tag{5.24}$$

Taking adjoint of this equation we get,

$$G(x - iy) = (a - ib)(x - iy) \tag{5.25}$$

If $x + iy$ and $x - iy$ were two different eigenvectors of matrix G, then their inner product $x^2 + y^2$ would have been 0 because of the mutual orthogonality among the eigenvectors. That is not possible until x and y are 0, in which case, (5.24) and (5.25) would be indifferent. That is possible only if the initial assumption is contradicted and b is allowed to be 0 for all eigenvectors x. Hence, all the eigenvalues of G are real.

Property 2. *G is symmetric and positive semi-definite matrix.*
Proof: Pertaining to the fact that $v_i \in \mathbb{R}^d$, the following should hold for some set of vectors x.

$$x^{\mathsf{T}} G x = \sum_{i,j} x_i x_j \langle v_i, v_j \rangle = \sum_{i,j} \langle x_i v_i, x_j v_j \rangle \tag{5.26}$$

According to the elementary property of inner products,
$\langle x+y, x+y \rangle = \langle x,x \rangle + \langle x,y \rangle + \langle y,x \rangle + \langle y,y \rangle$. It implies that the sum of inner products in (5.26) can be taken forward as,

$$\left\langle \sum_i x_i v_i, \sum_j x_j v_j \right\rangle = \left\| \sum_i x_i v_i \right\|^2 \geq 0. \tag{5.27}$$

Therefore, G is positive semi-definite matrix.

Property 3. *All the eigenvalues of G are non-negative.*
Proof: Property 2 implies $x^{\mathsf{T}} G x \geq 0$. Substituting the value of Gx from (5.23) into it,

$$x^{\mathsf{T}} G x = \lambda x^{\mathsf{T}} x \geq 0 \tag{5.28}$$

Since $x^{\mathsf{T}} x$ is positive for all eigenvectors, therefore, $\lambda \geq 0$.

The previously described premise is often used in various methods of dimensionality reduction. Algorithms like Principal Component Analysis and its variants utilize kernel trick to map the observations into a higher dimension to make the data linearly separable. It is equivalent to projecting the mean-centered data onto a subspace on which its variance is maximum (Collins et al., 2001). It is shown by Bernhard Scholkopf (Schölkopf et al., 1998) that algorithms like KPCA use a kernel function κ to essentially learn a mapping function ϕ for the input space $\mathbb{R}^n$ into a high-dimensional Hilbert space F, which can be called as feature space. The process is demonstrated in (5.29) and (5.30).

$$\phi : \mathbb{R}^n \to \mathbf{F} \tag{5.29}$$

Therefore, for a data point $v = (x_1, \ldots, x_n), x_i \in \mathbb{R}^d$, mapping into a feature space $\mathbb{R}^{n+k}$ will be given by

$$\phi(v) = (x_1, \ldots, x_n, p_1, \ldots, p_k) \in \mathbb{R}^{n+k} \tag{5.30}$$

where, the value of p_i depends upon the kernel that has been used for the mapping; however, kernels do not explicitly project the data into that high dimensional feature space; rather, it generates a Gramian matrix G of the mapped data in the aforementioned feature space $\mathbf{F}$. Generated Gramian matrix enables the input data to be operated in that high-dimensional feature space (Raykar, 2004). If $X = (x_1 \ldots x_n), x_i \in \mathbb{R}^d$ represent the input data. The corresponding Gramian matrix is given by

$$[G]_{ij} = \kappa(x_i, x_j) = \phi(x_i)\phi(x_j)^T, x_i, x_j \in X \tag{5.31}$$

Let $G = U \Sigma U^T$ represent the eigen decomposition of G, where U is a matrix containing the eigenvectors of matrix G, arranged column-wise in descending order of their corresponding eigenvalues, which are present in the same fashion in the diagonal matrix Σ as shown in (5.32) and (5.33).

$$U = [u_1, \ldots, u_n] \tag{5.32}$$

$$\Sigma = diag(\lambda_1, \lambda_2, \ldots, \lambda_n) \tag{5.33}$$

Here, $\lambda_1 \geq \cdots \geq \lambda_n \geq 0$ (see *Property 3* of Gramian matrix), $u_i^T u_i = 1$ for $i \in \{1, 2, \ldots, n\}$ and $Gu_i = \lambda_i u_i$. Also note that in context of PCA *Principal Components* refers to the projection of the input data points

onto the principal direction where the variance of the data is maximum. For PCA, the projection is given by $y_i = U_k^T x_i$ for all $i \in \{1,2,\ldots,n\}$, where U_k is a matrix of first k eigenvectors of G. However, in case of KPCA, the spectrum of G itself gives the projection of X Sch¨olkopf et al. (1998). Note that when $\phi(v) = v$, Gramian matrix transforms into covariance matrix. Generalising both, if U_k represent k principal axes, the algorithm finds a basis of an optimal low-dimensional subspace where the L_2-norm of reconstruction error is minimum Welling (2005). That is, for a test sample x

$$\arg\min_{U_k} \|\phi(x) - U_k U_k^T \phi(x)\|^2 \tag{5.34}$$

In addition to dimensionality reduction, PCA can also be used for k-clustering using a heuristic based k-means algorithm. This is done by performing k-means clustering in the projected space, as shown in heuristic k-means algorithm described in (Xu and Franti, 2004).

5.2.2 Graph Laplacian

Any set of observations appear to have an emergent behaviour to evince the properties of a graph when operated in a clustering pipeline. Therefore, given a set of data points $X = (x_1, x_2, \ldots, x_n) \in \mathbb{R}^{d \times n}$ and a notion of similarity between any two points x_i,$x_j \in X$, an undirected similarity graph $S = (V,E)$ can be constructed out of them such that each vertex $v_i \in V$ represent a data point x_i, and $(v_i, v_j) \in E$ represent the edge between vertices v_i and v_j. With each edge, there is an associated edge weight e_{ij} that represent the similarity between the corresponding data points. Let the similarity matrix be $W(i,j) = [e_{ij}]_{n \times n}$. The degree $d(v_i)$ associated with each node v_i is given by

$$d(v_i) = |\{v_j \in V | \{v_j, v_i\} \in E \text{ or } \{v_i, v_j\} \in E\}| = \sum_{j=1}^{n} e_{ij} \tag{5.35}$$

The degrees of all the nodes/vertices can be wrapped in matrix form as shown in (5.36)

$$D = diag(d_1, d_2, \ldots, d_n) \tag{5.36}$$

These matrices act as a precursor for constructing a matrix of algebraic importance, called *Laplacian matrix*. The data can be composed as a discrete graph form by making graph Laplacian of its continuous representations like vector space or Riemannian manifolds. Laplacian matrix has many variants, so much so, that depending on the problem and available data, authors device their own version of graph Laplacian matrix (Luxburg, 2007). The simplest graph Laplacian, is given by $(D - W)$. It is called unnormalise graph Laplacian matrix. However, in the proposed algorithm, the normalised graph Laplacian matrix has been used. That is,

$$\mathscr{L} = D^{-1/2}(D - W)D^{-1/2} = I - D^{-1/2} W D^{-1/2} \tag{5.37}$$

where $D^{-1/2} = diag(d_1^{-1/2}, d_2^{-1/2}, \ldots, d_n^{-1/2})$ and I is the identity matrix of appropriate order. Considering the fact that similarity matrix is a Gramian matrix, it is apparent that Gramian and Laplacian are not much different. Laplacian can be characterised as the Gramian normalised over the degree matrix. The distinction between unnormalised and normalised graph Laplacian is better apparent in light of spectral clustering. Consider a strongly connected graph $S = (V,E)$. The purpose of clustering is to come up with the subsets of points according to their similarity, such that the similar points lie in the same subset. It is equivalent to finding the *partitions* of a graph such that the edge between different partitions has minimum weights. For two disjoint subsets $A, B \subset V$ corresponding to two different partitions, the cut size is given by

$$cut(A,B) = \sum_{i \in A, j \in B} e_{ij} \tag{5.38}$$

Let there be k clusters in the data. The aim of clustering is to find k such partitions $\mathbf{A} = (A_1, A_2, \ldots, A_k)$, such that the size of the cuts, as shown in (5.38), over all the partitions is minimum. That is

$$\min_{A_1,\ldots,A_k} \text{cut}(A_i : 0 \geq i \geq k) := \sum_{i=1}^{k} cut(A_i, \bar{A_i}) \tag{5.39}$$

Where $\bar{A}_i$ is the complement of A_i. This is called the *mincut* problem. However, solving (5.39) alone does not achieve reliable clustering results. For example, for $k = 2$, partitioning one vertex from the rest of the graph can also be a valid solution as per mincut. In clustering, each cluster needs to accommodate a reasonably large partition to be considered credible. Therefore, the objective function is redefined in following two ways

$$\min_{A_1,\ldots,A_k} \text{RatioCut}(A_i : 1 \geq i \geq k) := \sum_{i=1}^{k} \frac{cut(A_i,\bar{A}_i)}{|A_i|} \tag{5.40}$$

$$\min_{A_1,\ldots,A_k} \text{NCut}(A_i : 1 \geq i \geq k) := \sum_{i=1}^{k} \frac{cut(A_i,\bar{A}_i)}{vol(A_i)} \tag{5.41}$$

Where $|A_i|$ represent the number of vertices in partition A_i and $vol(A_i) = \sum_{v_j \in A_i} d_j$.

However, solving these minimisation problems is NP hard. Laplacian matrix is an utility that can be used to approximate these minimisation problem. Consequently, unnormalised Laplacian serves in the approximation of the minimization of RatioCut, while normalised Laplacian serves in the approximation of the minimization of NCut. Therefore, the approximated objective function using normalised Laplacian is given by (5.42).

$$\min_{U_k} \mathbf{tr}(U_k^T \mathscr{L} U_k), \text{ subjected to } U_k^T U_k = I \tag{5.42}$$

The above expression is minimum when $U_k \in \mathbb{R}^{n \times k}$ is a matrix containing eigenvectors corresponding to k smallest non-zero eigenvalues of matrix $\mathscr{L}$. This matrix is used to embed the data into a k dimensional euclidean space spanned by the vectors in matrix U, in which grouping of the data points is arguably easy even with simpler techniques like k-means. The described practice is known as *Laplacian embedding*. The embedded data is then subjected to k-means clustering algorithm for cluster discovery, as shown in Normalised Spectral Clustering presented in (Ng et al., 2002). For a strongly connected graph with single component, the eigenvector corresponding to the trivial solution (i.e. $\lambda = 0$) of the eigenvalue problem of matrix $\mathscr{L}$ is a column vector of n ones. Therefore, $\mathscr{L}\mathbf{1}_n = 0$ where $\mathbf{1}_n = (1,\ldots,1)^T$. If the graph happens to have more than one components, then the multiplicity k of eigenvalue 0 if equal to the number of connected components in the graph. Nonetheless, with respect to clustering, the eigenvector(s) corresponding to eigenvalue 0 should be omitted while performing Laplacian embedding. It can be done by introducing a minor change in the matrix.

$$L = \mathscr{L} + \frac{2}{n}(1_n \mathbf{1}_n{}^T) \tag{5.43}$$

If the eigenpairs of $\mathscr{L}$ are given by

$$\mathbf{\Gamma}(\mathscr{L}) = \{(\lambda_1, f_1), (\lambda_2, f_2), \ldots, (\lambda_n, f_n)\}$$

then, the eigenpairs of (5.43) are given by

$$\mathbf{\Gamma}(L) = \{(\lambda_2, f_2), (\lambda_3, f_3), \ldots, (\lambda_1 + 2, f_1)\}$$

$$\text{where, } 0 = \lambda_1 < \lambda_2 \cdots \leq \lambda_n \leq 2 \text{ and } f_1 = \mathbf{1}_n$$

Hence, the new eigenvalue problem becomes

$$Lv = \mathscr{L}v + \frac{2}{n}(1_n 1_n{}^T)v = \lambda v \tag{5.44}$$

By modifying the matrix to L, the initial k eigenvectors can be taken right away. This trick works because of the fact that for all the pairs in $\mathbf{\Gamma}(\mathscr{L})$ except (λ_1, f_1), the matrix L gets reduced to $\mathscr{L}$. Hence, set $\mathbf{\Gamma}(L)$ is going to have all the eigenpairs that are in $\mathbf{\Gamma}(\mathscr{L})$, except (λ_1, f_1). While at $v = f_1 = \mathbf{1}_n$,

$$L\mathbf{1}_n = \mathscr{L}\mathbf{1}_n + \frac{2}{n}(1_n 1_n{}^T)\mathbf{1}_n = \lambda_1\mathbf{1}_n + 2\mathbf{1}_n = (\lambda_1 + 2)\mathbf{1}_n \tag{5.45}$$

Therefore, in the new set $\mathbf{\Gamma}(L)$, the rank of all the eigenvalues greater than λ_1 gets reduced by one and $\mathbf{1}_n$ becomes the eigenvector corresponding to the largest eigenvalue. Laplacian matrix has certain properties which are exploited by many clustering techniques like the one shown above. Some of the relevant properties are as following.

Property 1. *For every vector $f \in \mathbb{R}^n$, $\mathscr{L}$ satisfies the following condition*

$$f'\mathscr{L}f = \frac{1}{2}(\sum_{i,j=1}^{n} e_{ij}(\frac{f_i}{\sqrt{d_i}} - \frac{f_j}{\sqrt{d_j}})^2) \tag{5.46}$$

Proof: By the definition of degree, $d_i = \sum_{j=1}^{n} e_{ij}$. Therefore,

$$\begin{aligned}
f'\mathscr{L}f &= f'(I - D^{-1/2}WD^{-1/2})f \\
&= \sum_{i=1}^{n} f_i^2 - \sum_{i,j=1}^{n} \frac{f_i}{\sqrt{d_i}}\frac{f_j}{\sqrt{d_j}}e_{ij} \\
&= \frac{1}{2}(\sum_{i=1}^{n}\frac{f_i^2}{d_i}d_i + \sum_{j=1}^{n}\frac{f_j^2}{d_j}d_j - 2\sum_{i,j=1}^{n}\frac{f_i}{\sqrt{d_i}}\frac{f_j}{\sqrt{d_j}}e_{ij}) \\
&= \frac{1}{2}(\sum_{i,j=1}^{n}\frac{f_i^2}{d_i}e_{ij} + \frac{f_j^2}{d_j}e_{ij} - 2\frac{f_i}{\sqrt{d_i}}\frac{f_j}{\sqrt{d_j}}e_{ij}) \\
&= \frac{1}{2}(\sum_{i,j=1}^{n} e_{ij}(\frac{f_i}{\sqrt{d_i}} - \frac{f_j}{\sqrt{d_j}})^2)
\end{aligned}$$

Hence proved.

Property 2. *$\mathscr{L}$ is symmetric and positive semi-definite matrix.*
Proof: From (5.37), the symmetry of the matrix is fairly evident. Also, from the property 1, $f'\mathscr{L}f \geq 0$ for all $f \in \mathbb{R}^n$. Hence, $\mathscr{L}$ is symmetric and positive semi-definite matrix.

Property 3. *All eigenvalues of $\mathscr{L}$ are non-negative.*
Proof: Property 1 implies $f'\mathscr{L}f \geq 0$. Substituting $\mathscr{L}f = \lambda f$, we get $f'\mathscr{L}f = \lambda x^{\mathsf{T}}x \geq 0$. Since $f'f$ is positive for all eigenvectors, therefore, $\lambda \geq 0$.

5.2.3 RISynG Algorithm

For grouping the cancer patients into clusters, each omic view is represented as graph using two representation matrices, that is, Gramian matrix and Laplacian matrix. Each of the representation matrices attribute the similarity network of the samples with a notion of similarity between the samples. Consider a view $X_m = (x_1, x_2, \ldots, x_n)$, $x_i \in \mathbb{R}^{d_m}$ corresponding to mth omic-source. If $\rho(x_i, x_j)$ denotes the distance between x_i and $x_j \in X_m$, then the similarity $w(x_i, x_j)$ between them is given by

$$w(x_i, x_j) = \exp\{-\frac{\rho(x_i, x_j)}{\sigma}\} \tag{5.47}$$

where σ is a free parameter adjusted as per the intrinsic properties of the data when subjected to clustering model. For the cancer data used in this study, the σ is given by $\sigma = \frac{\max(\rho(x_i, x_j))}{2}$ for all $x_i, x_j \in X_m$. It has been

assumed in the proposed method that multi-views may constitute different cluster manifolds when learnt on a particular similarity measure. Therefore, predicted clusters would be apparent, and in strong concordance with the clinical clusters if pairwise sample similarity is computed in data-dependent multi-kernel approach. It was found that in some views correlation distance was prominently reflecting cluster manifold that concurred with the natural clusters, while some of them showed proclivity towards Euclidean distance, and the rest seemed to accommodate parts of both. All things considered, two different graph representation matrices have been formulated, Gramian matrix and Laplacian matrix, both with different measures of similarity. Let for X_m, the correlation distance between x_i and x_j be given by $\varphi_m(x_i,x_j)$ and the squared Euclidean distance be given by $\varepsilon_m(x_i,x_j)$. If $\hat{\varphi}_m$ and $\hat{\varepsilon}_m$ denotes the maximum pairwise correlation distance and squared Euclidean distance respectively, then Gramian matrix G_m and similarity matrix W_m are given by

$$[G_m]_{ij} = w_G(x_i,x_j) = \exp\{-\frac{\varphi_m(x_i,x_j)}{\hat{\varphi}_m}\}, \text{ where } i,j \in \{1,2,\ldots,n\} \tag{5.48}$$

$$[W_m]_{ij} = w_L(x_i,x_j) = \exp\{-\frac{\varepsilon_m(x_i,x_j)}{\hat{\varepsilon}_m}\}, \text{ where } i,j \in \{1,2,\ldots,n\} \tag{5.49}$$

The matrix articulated in (5.49) is a crucial precursor for the construction of Laplacian matrix. Laplacian matrix is constructed by normalising W_m by the degree matrix D_m of its associated graph as in Equation (5.36) and (5.37). Hence, required representation matrices for each view X_m, $m \in \{1,2,\ldots,M\}$ are given by (5.48) and (5.50).

$$\mathscr{L}_m = D_m^{-1/2}(D_m - W_m)D_m^{-1/2} = I - D_m^{-1/2}W_mD_m^{-1/2} \tag{5.50}$$

So obtained laplacian matrix is then modified as described in Equation (5.43)

$$L_m = \mathscr{L}_m + \frac{2}{n}(1_n \mathbf{1}_n{}^T) \tag{5.51}$$

It is apparent from the discussion presented under the heading Gramian Matrix and Kernel Trick and Graph laplacian that the matrix U_k from Gramian matrix has the same role as that from Laplacian matrix. Therefore, for combining the information encoded in these matrices, a parameterised combination function $\mathbf{\Omega}(\cdot,\cdot)$ can be used, hence obtaining a synergy matrix of representation matrices. If G_m is the Gramian matrix and L_m is the Laplacian matrix of omic-view X_m, then the synergy matrix is given by:

$$\mathbf{\Omega}(G_m,\mathscr{L}_m) = H_m = \beta G + (1-\beta)L, \text{ where } 0 \leq \beta \leq 1 \tag{5.52}$$

Consequently, the corresponding objective functions, (5.34) and (5.42) also combines to optimise over $U_k \in \mathbb{R}^{n\times k}$.

$$\min_{U_k} \beta \|X - U_kU_k^TX\|_F + (1-\beta)\boldsymbol{tr}(U_k^T\mathscr{L}U_k), \text{ subjected to} U_k^TU_k = I \tag{5.53}$$

Some of the relevant properties of synergy matrix H_m are:

Property 1. *H_m is symmetric and positive semi-definite matrix.*
Proof: H_m can be called a positive semi-definite matrix if and only if $v^TH_mv \geq 0$ for all $v \in \mathbb{R}^n$. Also, from the properties of the Graph Laplacian and the Gramian, it is evident that both L and G satisfies this condition. Therefore,

$$v^TH_mv = \beta v^TGv + (1-\beta)v^TLv \geq 0 \tag{5.54}$$

In addition to that, since H_m is a summation of symmetric matrices, it is also symmetric. Hence, H_m is a symmetric and positive semi-definite matrix.

Given *Property 1*, rest of the properties are its direct consequence.
Property 2. *All the eigenvalues of H_m are real.*
Property 3. *All the eigenvalues of H_m are non-negative.*

Recursive Multi-kernel Integration

After generating synergy matrices for all the views of the data set, next step is to integrate the information obtained from each of them. However, before moving on to the integration step, the proposed approach needs these matrices to be arranged based on their relative relevance for cluster discovery. It is apparent that the better views would encode the cluster structure better. As a consequence of that, they would depict better cluster validity indices as well. Therefore, the sorting of synergy matrices have been done on the basis of cluster validity indices such as silhouette index. Suppose $\boldsymbol{H} = \{H_1, \ldots, H_M\}$ be the set of synergy matrices of a data set with M views. Let the sorted set be $\boldsymbol{H'} = \{{}^1H, \ldots, {}^MH\}$, where the superscript i denotes the relevance of the corresponding synergy matrix ${}^iH$, ${}^1H$ being the most relevant. Additionally, let every ${}^iU_k$ from the set $\boldsymbol{U} = \{{}^1U_k, \ldots, {}^MU_k\}$ represent the basis of eigenspace corresponding to k smallest eigenvalues of matrix ${}^iH$.

Next, a method for combination has been proposed which distills the cluster information from each of the synergy matrix one by one, in an iterative fashion. While doing that, it subtly takes care of enriching the information coming from the relevant matrices. The way that the synergy matrices has been made, it is apparent that it is their basis of the eigenspace that brings out the latent cluster structure in the corresponding view. Therefore, the proposed method uses a recursive function to exploit this fact for integration as well as enrichment of the relevant views of the data set. The recursive formula can be written as:

$$\mathbf{k}_{\eta+1} := \mathbf{k}_\eta \otimes \mathscr{N}(\mathbf{k}_\eta, {}^{(\eta+1)}U), \text{ where } \mathbf{k}_1 = {}^1H \text{ and } \eta = 1, \ldots, M \tag{5.55}$$

Here $\boldsymbol{k}_\eta$ is called accretive matrix of ηth recursive step. Non-cumulative operator $\otimes$ signifies the integration operation. That is, for $A \in \mathbb{R}^{n\times n}$ and $U \in \mathbb{R}^{n\times k}$, where A has its k smallest eigenvectors in $V \in \mathbb{R}^{n\times k}$, and U is a basis matrix, the expression $A \otimes U$ evaluates to an accretive matrix $A' \in \mathbb{R}^{n\times n}$ with k smallest eigenvectors given by $V + U$. Other eigenvectors of A are irrelevant for this discussion. Let the basis of eigenspace of A' be known as accretive basis and associated subspace as accretive subspace. Also, let the accretive basis corresponding to k smallest eigenvectors of $\boldsymbol{k}_\eta$ be given by $\boldsymbol{b}_\eta$.

In extension to that, for enriching relatively relevant views, the proposed method uses an orthogonalising-normalising function $\mathscr{N}(\cdot,\cdot)$. To ensure the accumulation of only the essential cluster information, the proposed approach acquires the basis of that projection of synergy matrix eigenspace which is orthogonal to the accretive subspace at that recursive step. The idea is similar to eigenspace updation for integrative clustering as performed in (Khan and Maji, 2020). This function does not normalise the synergy matrix per se, rather, it normalises the basis of the described projection subspace. The computation starts by instantiating $\boldsymbol{k}_1 = {}^1H$ so that $\boldsymbol{b}_\eta$ becomes ${}^1U_k$. Therefore, at $(\eta+1)$th recursive step ($\eta \in \{0, 1, \ldots, M\}$), one should have accretive matrix $\boldsymbol{k}_\eta$ and eigenspace basis ${}^{(\eta+1)}U_k$ of synergy matrix ${}^{(\eta+1)}H$. Subsequently, processing within orthogonalising-normalising function $\mathscr{N}(\mathbf{k}_\eta, {}^{(\eta+1)}U_k)$ renders the final basis matrix in four steps:

First, computing the basis $\mathscr{P}$ of the projection subspace, which is given by:

$$\mathscr{P} = \boldsymbol{b}_\eta \boldsymbol{b}_\eta{}^T \, {}^{(\eta+1)}U_k \tag{5.56}$$

Second, computing the residual component of the synergy matrix eigenspace $\mathscr{Q}$ which is given by subtracting the above-mentioned projected component from ${}^{(\eta+1)}U_k$ as:

$$\mathscr{Q} = {}^{(\eta+1)}U_k - \mathscr{P} \tag{5.57}$$

In the third step, $\mathscr{Q}$ is subjected to Gram-Schmidt orthogonalisation to yield the final basis $\mathscr{R}$. This basis cannot be integrated with the eigenspace of accretive matrix, therefore it needs to be normalised on the basis of its relevance. So, the fourth step of normalization is performed as:

$$\mathscr{N}(\mathbf{k}_\eta, {}^{(\eta+1)}U_k) = V, \text{ where } V = [diag(\mathscr{R}\mathscr{R}^T)^{-\frac{1}{2}}(\mathscr{R})]^{(\eta+1)} \tag{5.58}$$

Here the notation $[\ \cdot\ ]$ denotes that the subsequent operations are done in element-wise fashion. The resultant V matrix is called as orthogonalised-normalised basis matrix. After the end of the process, the final accretive matrix $\mathbf{k}_M$ is obtained whose first k eigenvectors in the matrix $\mathbf{b}_M \in \mathbb{R}^{n\times k}$ holds the cluster structure. Hence, performing k-means on the rows of matrix $\mathbf{b_M}$ returns the cluster labels for each sample. The proposed algorithm is described in Algorithm 2.

Algorithm 2 RISynG: Proposed Algorithm

Input: Similarity matrices $W_1,\ldots,W_M$, Gramian matrices $G_1,\ldots,G_M$ and number of clusters k.
Output: Clusters $A_1, A_2, \cdots, A_k$

function INTEGRATE($\boldsymbol{b}, \eta$)
  if η is not 1 **then**
    $\boldsymbol{b} \leftarrow INTEGRATE(\boldsymbol{b}, \eta - 1)$
  end if
  Construct orthogonalised-normalised basis V as per (5.56),(5.57) and (5.58)
  $\boldsymbol{b} \leftarrow \boldsymbol{b} + V$
  return **b**
end function
for $m \leftarrow 1$ `to` M **do**
  Construct a degree matrix D_m as in (5.36)
  Construct Laplacian L_m as in (5.37)
  for $\beta \leftarrow 0$ `to` 1 `STEP` $+\alpha$ **do**
    Construct synergy matrix H_m as in (5.52)
    Perform k-means on the rows of first k eigenvectors of matrix H_m
    Record the silhouette index of predicted labels
  end for
  Reconstruct H_m matrix at that value of β where the silhouette value is maximum
  Record the first k eigenvectors of matrix H_m in $U_m \in \mathbb{R}^{n\times k}$
  Perform k-means on the rows of U_m and record the silhouette value of so obtained sample labels as $\boldsymbol{f}_m$
end for
Sort U_i in the ascending order of the corresponding silhouette value $\boldsymbol{f}_i$, for all $i \in \{1,\ldots,M\}$. Let the sorted array be $U^{`} = ({}^1U, {}^2U, \cdots, {}^{\mathrm{M}}U)$
$\boldsymbol{b} \leftarrow {}^1U_k$ and $\eta \leftarrow M$
$\boldsymbol{b} \leftarrow$INTEGRATE($\boldsymbol{b}$, η)
Perform k-means clustering on the rows of matrix **b**
Return: Clusters $A_1, A_2, \cdots, A_k$

5.2.4 Significance of Proposed Algorithm

There are some aspects of the proposed algorithm that enhances it's performance and makes it unique from the other algorithms that are designed to identify cancer subtypes. Although, each omic-view in the cancer data set has its own distinct cluster structure, however, the knowledge of cancer biology suggests that none omics-source, to which each view belong can dictate the final cancer subtype alone. Rather, all the omics-sources collectively manifest the cancer subtype in a sample. Therefore, multi-view integration is critical to a sensible and clinically relevant clustering. The proposed approach can be broken down into three operative steps: (1) construction of representation matrices for each view, (2) construction of synergy matrix for each view, and (3) construction of accretive basis through recursive multi-kernel integration of synergy matrices. These steps make the proposed algorithm more effective in the following manner:

1. *Construction of representation matrices:* To group the cancer patients into clusters, each of the omic-view first has to be represented as similarity graphs. These similarity graphs can be interpreted through various representation matrices like the Gramian matrix, Laplacian matrix, and Adjacency

matrix. Each of the representation matrices attributes the samples' similarity network with a notion of similarity between the samples. The proposed method assumes that multiple information sources may constitute different cluster manifolds when learned on a particular similarity measure. Therefore, predicted clusters would be apparent and in strong concordance with the clinical clusters if pairwise sample similarity is computed in data-dependent multi-kernel approach (Gönen and Alpaydın, 2011). In some views, Correlation distance was prominently reflecting cluster manifold that concurred with the natural clusters. Whereas some of them showed proclivity towards Euclidean distance, the rest seemed to accommodate both. All things considered, two different graph representation matrices have been formulated, the Gramian matrix and Laplacian matrix, both with different measures of similarity.

2. *Construction of synergy matrices:* Representation matrices so constructed has two noteworthy aspects to it: (1) G_m represents a similarity graph formed using correlation-based distance. In the correlation-based distance, two objects are considered similar if the trends among its elements are highly correlated. That means the correlation distance between two perfectly correlated samples will be 0, even though they are far apart in the euclidean space of their dimension. It is instinctive to assume the omics data to behave like that. (2) Laplacian, on the other hand, preserves the intrinsic manifold structure in the data casted on a low embedding space. To integrate these representation matrices, a combination function has been devised that takes a convex combination of both the matrices. This method of combining matrices rectifies any bias that may create because of the dissimilarity in distance measurement used while constructing the similarity graphs. The combination function defined in (5.52) utilises the parameter $\beta \in [0,1]$ to capture graphs constituted by the Gramian and Laplacian. Parameter β can only take a positive value, making the combination a convex combination of representation matrices. This parameter's optimal value is learnt by iterating it from 0 to 1 at some incremental step size $\alpha \in (0,1)$. The data-sets used in this study tend to pick up the optimal value of β at a step size of $\alpha = 0.1$. It is crucial to choose the incremental step size wisely as the number of iterations t_β is directly proportional to the algorithm's time complexity. Because the synergy matrix is ultimately going to affect the cluster assignment, the best way to evaluate the appropriate value of β is to perform a provisional cluster validity test on the synergy matrix constructed with that β using a cluster validity index like silhouette index. Algorithm-2, step 15 to 19 formulates the described provisional cluster validity test using silhouette as a criterion.

3. *Construction of accretive basis:* After the similarity between the cancer patients are captured in a refined form with the help of synergy matrices, the next step is to integrate them. Property 1 of the synergy matrix proves that H_m is a positive semi-definite matrix. That makes the integration of synergy matrices a multi-kernel integration. The proposed algorithm does that by recursive multi-kernel integration by iteratively integrating each of the synergy matrices' relevant subspace. Here, relevant subspace refers to that subspace of the matrix that purely encodes the cluster information,which in the case of synergy matrix is its eigenspace corresponding to k eigenvalues. Finally, an accretive basis matrix is generated. This accretive matrix is required to have more cluster information coming from relevant views. Therefore, the orthogonalizing-normalizing function is made such that the accretive basis at each recursive step gets less influenced by irrelevant matrix.

5.3 DESCRIPTION OF DATA-SETS

For analyzing the efficiency of the proposed algorithm in identifying cancer subtypes, it is applied to five cancer datasets taken from TCGA (`https://cancergenome.nih.gov/`). The datasets used are Cervical cancer (CESC), Breast cancer (BRCA), Ovarian cancer (OV), Lower-grade glioma (LGG) and Stomach cancer (STAD). Different studies have identified 4 clinically important subtypes for BRCA (Network, 2012c) and STAD (Network, 2017a), 3 for CESC (Network, 2017b) and LGG (Network, 2015) and 2 for OV (Matsuno et al., 2013). The cancer genome is neither simple nor independent but is complicated and dysregulated by multiple levels in the biological system through genomic, epigenomic,

transcriptomic, and proteomic levels (Network, 2012a). miRNA, one of the important regulators of gene expression, can be integrated with gene expression to identify the selective inhibition of translation or selective degradation(Borel et al., 2011; Lu and Clark, 2012; Liu et al., 2020). Furthermore, in terms of epigenetic regulation, histone modification or DNA methylation can serve to regulate gene expression in cancer (Dudziec et al., 2012; KW et al., 2017). Also, protein expression data can be utilized for the diagnostic prognosis of cancer patients (Kim et al., 2019). Therefore, four genomic views, namely, gene expression (mRNA), microRNA expression (miRNA), DNA methylation (metDNA), and reverse-phase protein assays (RPPA) are utilized for CESC, BRCA, and LGG datasets. For STAD and OV datasets, mRNA and miRNA expression are only considered because most samples' metDNA and RPPA information are unavailable. To avoid involving features with too many missing values features having more than 5% of missing values in all of the omic views are removed, and the rest of the missing values are replaced with 0. Sequence-based expression data are log-transformed to make the data more or less normally distributed (Zwiener et al., 2014). Therefore the 0 entries from miRNA and mRNA expression data are replaced with one and then log-transformed with base 10. For metDNA datasets, beta values are considered. At last, variance filtering is applied to mRNA and metDNA omic views for all cancer datasets, and 2000 most variable genes and CpG locations were only considered. Table 5.1 describes the final processed data used for this study. The proposed algorithm can be applied to other large-scale multi-omics datasets if available; the run time will increase with the increase in sample size or the number of omic views, as shown in Figure 5.2. With the increase in sample size from 124 to 474, the runtime increases from 0.22 seconds to 0.47 seconds. Even though the BRCA dataset has lesser samples (398) than the OV dataset (474), the runtime for BRCA (0.56 sec) is more than OV (0.47 sec) because of the number of omic-views involved, which is 4 for BRCA and 2 for OV.

Table 5.1. : Data-sets description

		Number of features				
Data-sets	Number of samples	mRNA	miRNA	metDNA	RPPA	Number of clusters
CESC	124	2000	311	2000	219	3
BRCA	398	2000	278	2000	212	4
OV	474	2000	591	-	-	2
LGG	267	2000	333	2000	209	3
STAD	223	2000	524	-	-	4

5.4 EXPERIMENTAL RESULTS AND DISCUSSION

The performance of the proposed approach is compared with eleven other algorithms available for cancer subtype identification. Both two-stage clustering approaches and integrative clustering approaches are considered for method comparison. The methods used for comparison are SNF (Bo et al., 2014), Weighted Multi-View Low Rank Representation (WMLRR) (Sun et al., 2021), Consensus clustering (CC) (Monti et al., 2003b; Wilkerson and Hayes, 2010), Multi-view clustering approach with enhanced consensus (ECMC) (Cai and Li, 2017), SNF.CC (SNF merged with CC) (Xu et al., 2017), Cluster of cluster assignment (COCA) Network (2012c); Cabassi and Kirk (2019), Consensus non-negative matrix factorization (CNMF) (Brunet et al., 2004), Selective update of relevant eigenspaces (SURE) (Khan and Maji, 2020), Convex-combination of approximate laplacians (CoALa) (Khan and Maji, 2021a), iCluster (Therese et al., 2001), and Multi-manifold Integrative Clustering (MiMIC) (Khan and Maji, 2021b).

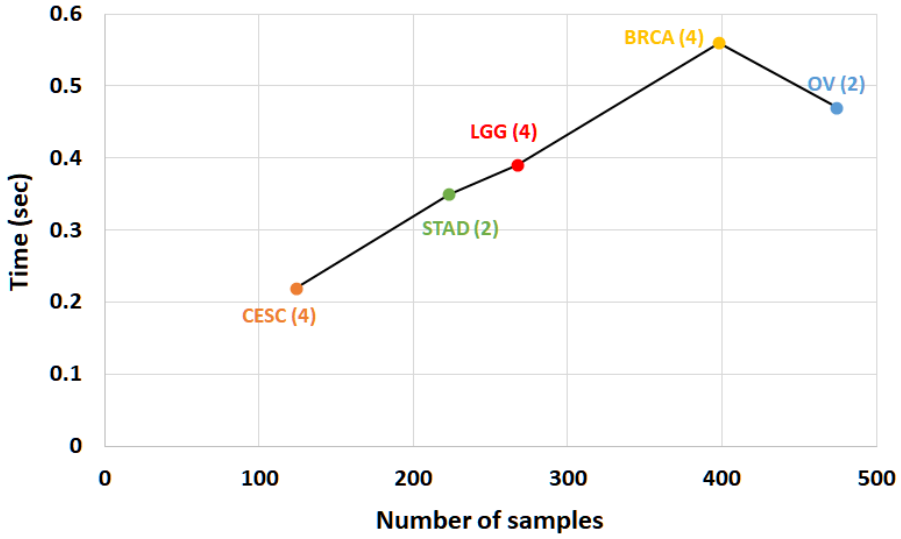


Figure 5.2. : Effect of sample size and number of omic-views on the runtime of the proposed algorithm. Values in the parentheses indicate the number of omic-views.

5.4.1 Performance Analysis on Multi-omics Cancer Data-sets

Proposed approach along with the above described methods are applied to five cancer data-sets, namely CESC, BRCA, OV, LGG and STAD taken from TCGA. The sample clusters identified by each of these methods are evaluated based on several internal and external cluster evaluation indices. The cancer subtypes identified by these methods are also evaluated for their biological relevance. Next, the detailed comparative analysis of the proposed algorithm is discussed.

Cluster Evaluation

The clusters (cancer subtypes) generated by all the methods are evaluated based on several internal and external cluster evaluation indices. These indices help get the idea of how well a method can group the samples into homogeneous clusters. Samples belonging to the same cluster should have higher similarity representing a cancer subtype, whereas samples belonging to different clusters should be highly dissimilar. How well an algorithm can capture the natural grouping present in the data can be quantified with internal validity indices. Four internal evaluation indices, namely Silhouette index (Rousseeuw, 1987), Dunn index (Bezdek and Pal, 1995), Davies–Bouldin index (Davies and Bouldin, 1979) and Xie-Beni (Xie and Beni, 1991) index are calculated in this study. In Table 5.3, the internal evaluation indices for every method in compared. CNMF outperformed in the case of CESC data-set and SNF.CC in case of BRCA and LGG data-sets. Whereas, for OV and STAD data-sets, different algorithms are performing better with respect to different internal indices. The proposed method has the best silhouette index for the OV data-set. If all the data-sets are considered together to rank the clustering efficiency of all the algorithms under study considering all the internal indices, SNF.CC stands first, followed by CNMF, SNF, CC, and proposed. SURE, CoALa, and iCluster are not in the picture.

The class distribution of the cancer data-sets used in this study is presented in Table 5.2. Except for the CESC data-set, all the other cancers have an imbalanced class. When clustering is applied to these data-sets, there are chances that most of the samples get clustered into one group leading to good values for internal indices. Still, in reality, the clustering is not efficient. The partitions created in these kinds of imbalanced data can be efficiently evaluated with external evaluation indices if the ground truth is available. In this study, five external evaluation, namely F-measure (FM) (de Souto et al., 2012), Adjusted Rand Index

(ARI) (Hubert and Arabie, 1985), NMI, Jaccard Index, and Purity (Qiang et al., 2016) are calculated for comparing the clustering efficiency of the different algorithms. Based on these five external evaluation indices, it is observed that the proposed algorithm outperforms in CESC, BRCA, and LGG data-sets. Also, in STAD data-set it outperforms with respect to F-measure, Jaccard index, and Purity. OV cancer is the only case where the proposed approach is not able to work that well. If all the data-sets are considered together to rank the clustering efficiency of all the algorithms under study considering all the external indices, the proposed method stands first by attaining a maximum value 18 times out of 25. The execution times reported in Table 5.3 show that RISynG is faster than other algorithms. All the cluster evaluation indices used in in this study are defined in Supplementary material A.

Table 5.2. : Cancer subtypes description: Actual class distribution

Data-sets	Subtypes	Number of samples
CESC	Keratin low squamous	37
	Keratin high squamous	58
	Adenocarcinoma	29
BRCA	Luminal A	80
	Luminal B	49
	Her-2 enriched	171
	Triple negative / basal like	98
OV	Neoplasm histological grade 3	417
	Neoplasm histological grade 2	57
LGG	IDH mutation without 1p/19q codeletion	134
	IDH mutation with 1p/19q codeletion	84
	Wild type IDH subtype	49
STAD	MSI (Microsatelite instability)	45
	EBV (Epstein–Barr virus)	17
	CIN (Chromosomal instability)	111
	GS (Genomically stable)	50

Importance of Multi-omics Data Integration

The proposed algorithm RISynG, iteratively integrates the relevant subspace of each of the synergy matrices. The relevant subspace corresponds to the k largest eigenvectors of the synergy matrices that holds the cluster structure. To exhibit the significance of this iterative integration and effectiveness of RISynG, it is compared with Spectral clustering performed on individual omics data-sets. The results presented in Figure 5.3 shows that the proposed algorithm outperforms the individual omic-views in CESC, BRCA, LGG, and STAD data-sets for all the external cluster validity indices. In the OV data-set, RISynG outperforms for F-measure, Jaccard, and Purity. However, miRNA view performs better for ARI and NMI indices. The performance of RISynG is significantly higher than the best individual view in case of CESC, BRCA, and LGG data-sets irrespective of any indices.

Table 5.3. : Comparative Cluster Analysis of Proposed and Existing Approaches

		Internal Evaluation Indices				External Evaluation Indices					
Datasets	Methods	Silhouette	Dunn	DB	Xie-Beni	F-measure	ARI	NMI	Jaccard	Purity	Time (sec)
	SNF	0.4009138	0.4454051	1.2271228	1.0368490	0.7258065	0.4066304	0.4872207	0.4482759	0.7258065	0.39
	CC	0.6558044	0.2434998	0.3457636	0.8466638	0.6774194	0.3682621	0.3919370	0.4441509	0.6774194	31.61
	CNMF	**0.8849808**	**0.5815151**	**0.1346099**	**0.1404269**	0.6693548	0.4280599	0.3887215	0.4761179	0.6693548	65.7
	ECMC	-	-	-	-	0.5943548	0.4296452	0.4684325	0.4978563	0.549342	89.35
	WMLRR	0.4174973	0.0279456	0.3756743	64.698374	0.5493742	0.4739754	0.4963865	0.3857491	0.5836492	72.46
	COCA	-	-	-	-	0.6370968	0.2907653	0.3949373	0.3713777	0.6370968	2.26
CESC	SNF.CC	0.6820409	0.4747143	0.2514171	0.2487638	0.7258065	0.4059437	0.4792426	0.4467519	0.7258065	6.46
	SURE	0.3533451	0.0643343	0.8705947	13.6558000	0.8387097	0.5969901	0.5731598	0.6027137	0.8387097	0.34
	CoALa	0.4750780	0.0459589	1.0234149	36.8191400	0.4677419	0.0712289	0.0944394	0.2599910	0.4677419	98.44
	iCluster	0.4133838	0.0248786	0.8147982	40.2522200	0.4435480	0.0270767	0.0364498	0.2436537	0.4435484	175.81
	MiMIC	0.4064141	0.4064141	1.1487670	5.7334720	0.5243129	0.4891097	0.4952194	0.4196284	0.7741935	100.204
	RISynG	0.4824000	0.0701205	0.6639514	14.5428100	**0.8951612**	**0.7191808**	**0.6639029**	**0.6975966**	**0.8951613**	**0.22**
	SNF	0.4936198	0.3458074	1.2421657	1.7509710	0.6934673	0.4010266	0.4625550	0.3945632	0.6934673	7.88
	CC	0.6653615	0.0702003	0.5397077	13.248510	0.4321608	0.2979244	0.3477434	0.3271579	0.5678392	239.52
	CNMF	0.6428795	0.0423558	0.4194482	33.133060	0.4899497	0.3161555	0.3537982	0.3503934	0.5954774	344.58
	ECMC	-	-	-	-	0.3857462	0.2846732	0.3285674	0.1846376	0.4695832	112.63
	WMLRR	0.4783742	**0.4593621**	**0.1496783**	13.97465	0.5478943	0.2385643	0.3486532	0.2957483	0.4768392	94.32
	COCA	-	-	-	-	0.4045226	0.2211292	0.3145460	0.2821191	0.4824121	12.58
BRCA	SNF.CC	**0.8923872**	0.2021579	0.1871910	**0.7428676**	0.4271357	0.4281475	0.4732544	0.4150974	0.7160804	140.11
	SURE	0.2966142	0.0417821	0.9317001	23.00041	0.7562814	0.4798912	0.5063654	0.4569119	0.7562814	2.2
	CoALa	0.3363002	0.0304929	0.8465351	28.189170	0.3919598	0.3907137	0.4685283	0.3923870	0.6758794	737.92
	iCluster	0.3673494	0.0202157	0.9099812	88.604970	0.4798990	0.2900248	0.3659365	0.3185041	0.5000000	145.61
	MiMIC	0.3283235	0.0327848	1.0975070	28.99286	0.3783270	0.4108080	0.4855910	0.3764962	0.6984925	1521.87
	RISynG	0.4296000	0.0459134	0.7135468	33.395980	**0.7613065**	**0.4987076**	**0.5260974**	**0.4716932**	**0.7613065**	**0.56**
	SNF	0.4378744	**0.4301527**	1.3849515	**1.7023463**	0.6877637	**0.02818935**	0.0038031	0.5263954	0.6877637	7.06
	CC	0.5903129	0.1032681	**0.4607749**	7.9070535	0.6265823	-0.0323654	**0.0081459**	0.4892693	0.6265823	664.01
	CNMF	0.3538763	0.1154637	0.8634937	435.7459476	0.7046413	0.0278468	0.0031328	0.5436891	0.7046414	263.46
	ECMC	-	-	-	-	0.5694732	0.00047853	0.0001874	0.3857693	0.5849563	172.51
	WMLRR	0.4867543	0.2275846	0.9859643	19.45342	0.5648932	0.0018593	0.0064382	0.4537682	0.6385932	132.12
	COCA	-	-	-	-	0.6370968	-0.0036014	0.0014148	**0.7846563**	**0.8776371**	4.03
OV	SNF.CC	0.3957387	0.1673645	0.7284081	2.6021595	0.6793249	0.0141705	0.0010843	0.5211995	0.6793249	62.57
	SURE	0.3222961	0.0111151	0.9517326	326.6007	0.5843258	-0.0033712	0.0033794	0.440303	0.5063291	1.74
	CoALa	0.3812581	0.0162638	0.9152212	144.9817	0.6540084	0.0078844	0.0004858	0.5005109	0.6540084	1189.76
	iCluster	0.5366303	0.0023473	0.6248983	4335.3610	0.5253164	0.0004292	0.0169881	0.4408962	0.5253165	599.562
	MiMIC	0.3542949	0.0119706	1.1338063	306.19587	**0.8272251**	0.0052532	0.0002227	0.4375362	0.6582278	1678.144
	RISynG	**0.6132400**	0.0043105	0.6058482	877.900712	0.6708865	-0.0386828	0.0079705	0.5168302	0.6624473	**0.47**
	SNF	0.5552045	0.331413	1.2073461	1.954655	0.6853933	0.3025052	0.3251302	0.4082171	0.6853933	3
	CC	0.6585338	0.2589659	0.3511019	0.7576893	0.8913858	0.6746313	0.6852894	0.6594185	0.8913858	101.73
	CNMF	0.8583117	0.0420026	0.1764646	17.21837	0.5131086	0.1517872	0.2211501	0.3099572	0.5131086	187.31
	ECMC	-	-	-	-	0.6467354	0.5785674	0.7637284	0.5563743	0.6845632	72.51
	WMLRR	0.5648732	0.3365783	0.3486573	72.85743	0.4485673	0.6759743	0.6493754	0.5873549	0.5704737	89.85
	COCA	-	-	-	-	0.6254682	0.2799432	0.3394982	0.4282438	0.6254682	6.31
LGG	SNF.CC	**0.8796842**	**0.5938181**	**0.1291197**	**0.0976532**	0.6853933	0.3025052	0.3251302	0.4082171	0.6853933	152.09
	SURE	0.3834002	0.0950595	0.8558895	6.137831	0.6329588	0.2814732	0.4197476	0.3817919	0.6329588	1.06
	CoALa	0.5106729	0.1025237	0.5474956	6.995131	0.6741573	0.3990866	0.5757631	0.5336079	0.6741573	330.52
	iCluster	0.5840541	0.0145988	0.5850834	158.9932	0.5767790	0.0965103	0.0782154	0.3346215	0.5767790	341.17
	MiMIC	0.5433994	0.0069822	0.7392175	119.9754	0.4472362	0.8460880	0.7071638	0.6392837	0.8812734	487.566
	RISynG	0.4542000	0.1240071	0.7823224	4.904607	**0.9513109**	**0.8752557**	**0.8179747**	**0.8562767**	**0.9513109**	**0.39**
	SNF	0.4390147	**0.4424347**	1.2665290	**1.2660670**	0.3049327	0.0126031	0.0339522	0.2039706	0.3139013	0.94
	CC	0.6517968	0.1587210	0.3735403	2.459431	0.3183857	0.0068337	0.0118838	0.1867525	0.3183857	49.97
	CNMF	**0.9261939**	0.0751574	**0.1297056**	3.592156	0.3049327	0.0055038	0.0085026	0.1800452	0.2959641	124.65
	ECMC	-	-	-	-	0.1847652	0.0018564	0.0035761	0.0174563	0.1785945	84.845
	WMLRR	0.5873752	0.2857689	0.3285963	16.74563	0.1847563	0.0075843	0.0084754	0.0985647	0.1568347	69.213
	COCA	-	-	-	-	0.2600897	0.0055526	0.0211873	0.1730754	0.3004484	3.48
STAD	SNF.CC	0.5330525	0.1745648	0.6411855	2.907296	0.3139013	-0.0139222	0.0342102	0.1871362	0.3363229	3.33
	SURE	0.3371392	0.0345696	0.7900013	53.22062	0.2825112	0.0250796	0.0243271	0.2243634	0.3991031	0.61
	CoALa	0.3646726	0.0354464	0.7797201	24.92721	0.3318386	0.0080341	0.0319674	0.1969155	0.309417	126.12
	iCluster	0.2700001	0.0551868	1.2110837	26.40223	0.2466360	0.0022785	0.0077143	0.1723892	0.3094170	111.22
	MiMIC	0.3104935	0.0591516	0.9772476	19.93442	0.3854797	0.0096791	0.0101976	0.1493762	0.3766816	214.767
	RISynG	0.3517051	0.0280232	0.7694432	54.50121	**0.3901345**	**0.0260441**	**0.1230677**	**0.2267073**	**0.3901345**	**0.35**

The proposed algorithm RISynG, iteratively integrates the relevant subspace of each of the synergy matrices. The relevant subspace corresponds to the k largest eigenvectors of the synergy matrices that holds the cluster structure. To exhibit the significance of this iterative integration and effectiveness of RISynG, it is compared with Spectral clustering performed on individual omics data-sets. The results presented in Figure 5.3 shows that the proposed algorithm outperforms the individual omic-views in CESC, BRCA, LGG, and STAD data-sets for all the external cluster validity indices. In the OV data-set, RISynG outperforms for FM, Jaccard, and Purity. However, miRNA view performs better for ARI and NMI indices. The performance of RISynG is significantly higher than the best individual view in case of CESC, BRCA, and LGG data-sets irrespective of any indices.

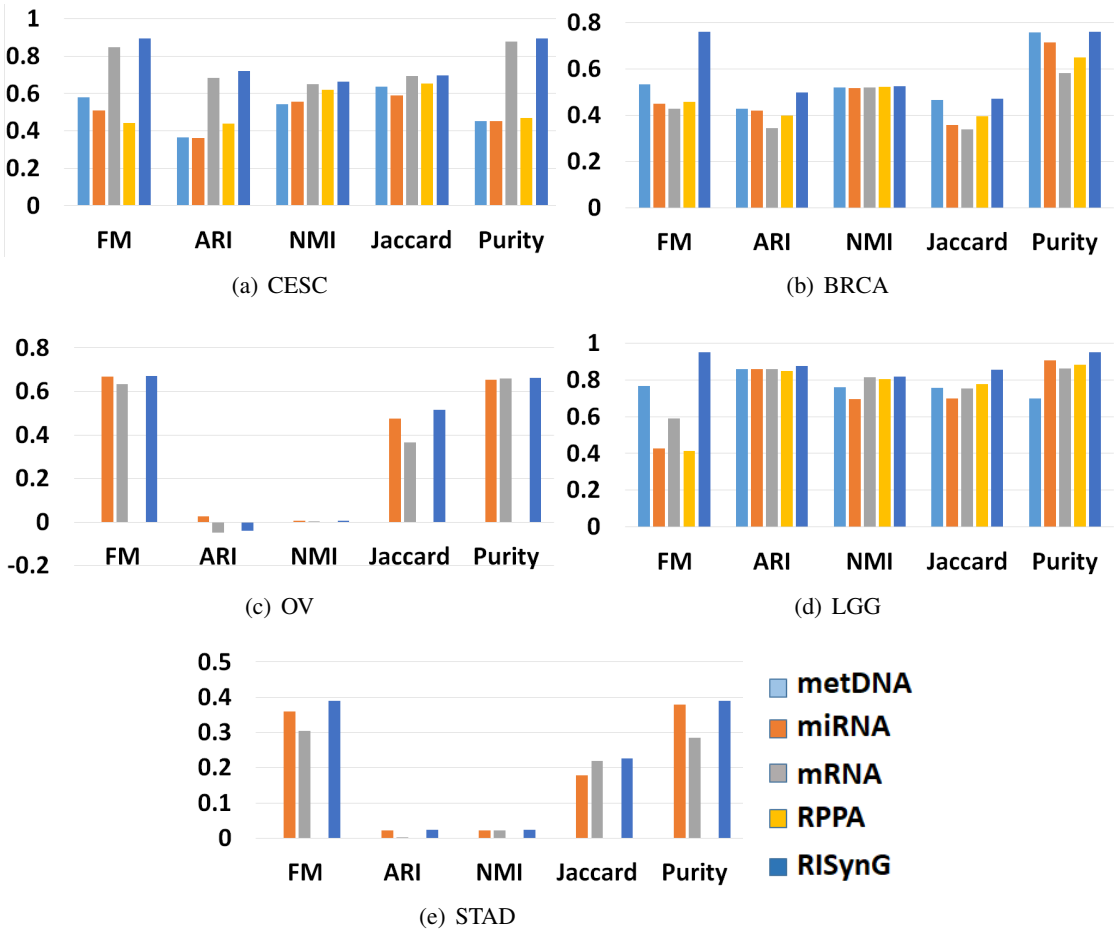
To express the cluster holding capacity of the integrated subspace obtained by the proposed approach scatter plots for the best k dimensions are plotted. The colours in the plots indicate the ground truth (cancer subtypes). Comparative plots are also presented in Figure 5.4, 5.5, 5.6, 5.7, and 5.8 to show that the integrated subspace obtained by RISynG are more informative than other subspace-based integrative-clustering approaches (SNF, SURE, CoALa, and iCluster), for most of the data-sets. Comparison with the best individual omic-view (CESC: mRNA, BRCA: metDNA, OV: miRNA, LGG: metDNA, and STAD:miRNA) is also presented to establish the significance of multi-omics data integration performed by the proposed approach. Considering the proposed approach, the scatter plots show that the clusters are well separated in case of CESC (Figure 5.4) and LGG (Figure 5.7) data-sets. Their is slight overlap between two groups in BRCA (Figure 5.5), but it is better than the other methods. Whereas, for OV (Figure 5.6) and STAD (Figure 5.8) data-sets overlap between subtypes is observed in subspace obtained by all the methods.

Biological Analysis

Once the cancer-subtypes are obtained, the patient clusters' molecular characteristic feature is also evaluated to establish their biological relevance. To understand the varying expression of different biomarkers in different subtypes, differential expression analysis (DEA) of miRNAs and mRNAs are performed between the correctly identified groups of patients. A comparative analysis is performed between the true positives and true negatives obtained by all the algorithms. As there are three subtypes in the case of LGG and CESC data-sets; therefore, DEA is performed between three pairs (considering all possible pairs). Similarly, in case of STAD and BRCA data-sets since there are four subtypes, therefore, DEA is performed for six pairs, and for the OV data-set, there are two subtypes; therefore, DEA is performed for one pair. R package Limma (Smyth, 2005) is used to perform DEA. miRNAs, and mRNAs having Bejamini-Hochberg FDR adjusted p-value < 0.05 are considered as differentially expressed. Number of differentially expressed biomarkers obtained from different groups in CESC, BRCA, OV, LGG, and STAD data-sets are reported in Table 5.4, 5.5, 5.6, 5.7, and 5.8 respectively. To further explore and highlight the biological knowledge and process-specific functioning of the identified sets of differentially expressed biomarkers different types of enrichment analysis is also performed considering the hundred most differentially expressed biomarkers in each case.

Pathway Enrichment Analysis

Pathway enrichment analysis (PEA) helps to gain mechanistic insight into the set of differentially expressed biomarkers. It helps in the identification of those biological pathways that are enriched in a set of biomarkers more than would be expected by chance. In this study R package clusterProfiler (Yu et al., 2012) and DIANA Tools mirPath v.3 (Vlachosa et al., 2012) are used for performing PEA for genes and miRNAs respectively. The top 100 differentially expressed biomarkers are passed to these tools. In some cases, if the number of differentially expressed biomarkers are less than 100, then all of them are used. KEGG database is selected for PEA. All the pathway terms associated with the set of biomarkers having false discovery rate adjusted p-value < 0.05 (significant pathway terms) are only considered. If any of the differentially expressed biomarker sets are not associated with significant KEGG pathway terms, that set is said to be not biologically relevant with respect to KEGG pathway terms. For the quantification of PEA, two scores, namely, KPES (Paul and Madhumita, 2020) and AR (Consortium, 2000a) are also calculated. The higher the

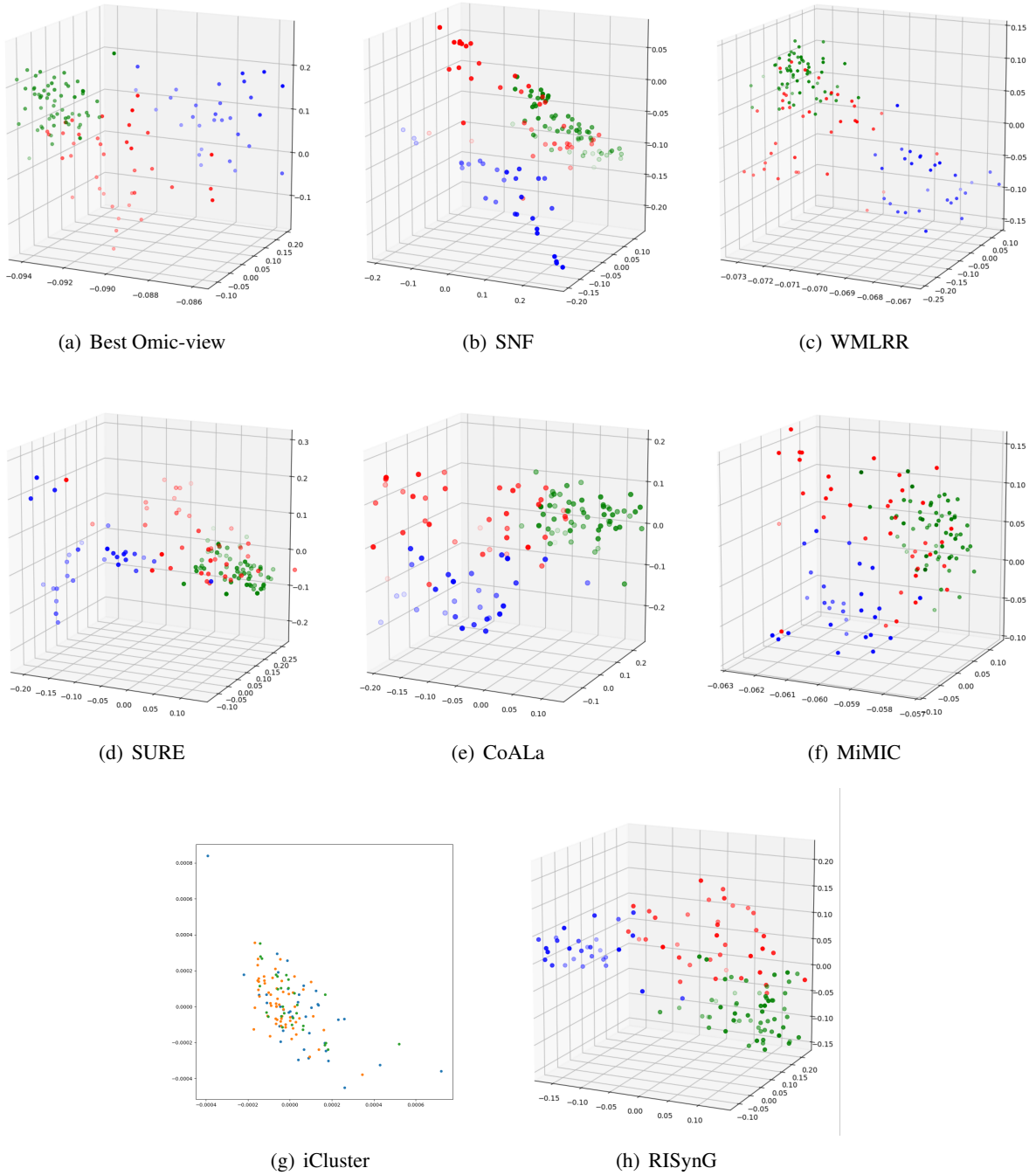


(a) CESC (b) BRCA

(c) OV (d) LGG

(e) STAD

Figure 5.3. : Comparative performance analysis of proposed approach and spectral clustering performed on individual omic-views

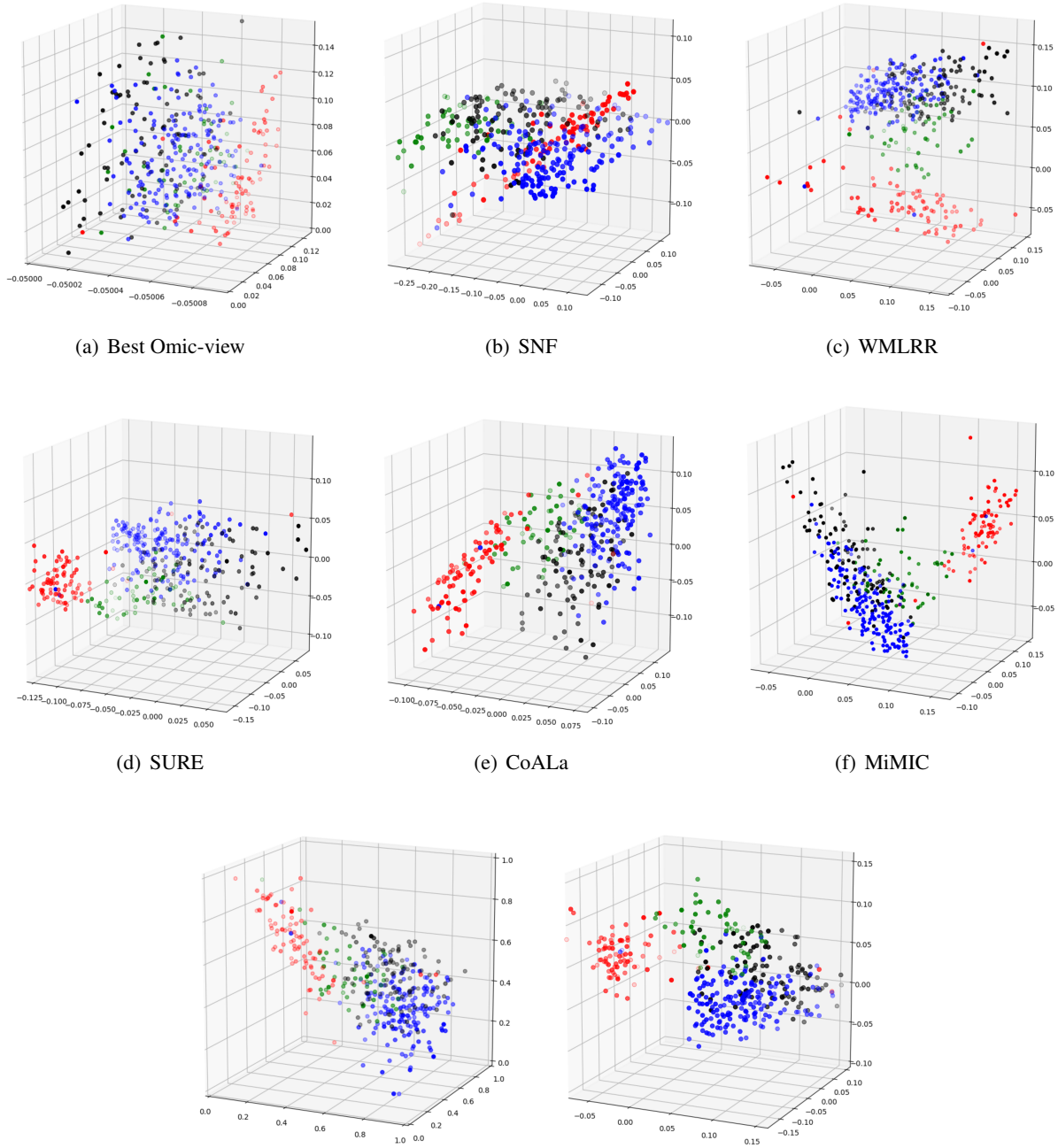


(a) Best Omic-view (b) SNF (c) WMLRR

(d) SURE (e) CoALa (f) MiMIC

(g) iCluster (h) RISynG

Figure 5.4. : Comparative analysis of different integrative sub-spaces on CESC data set

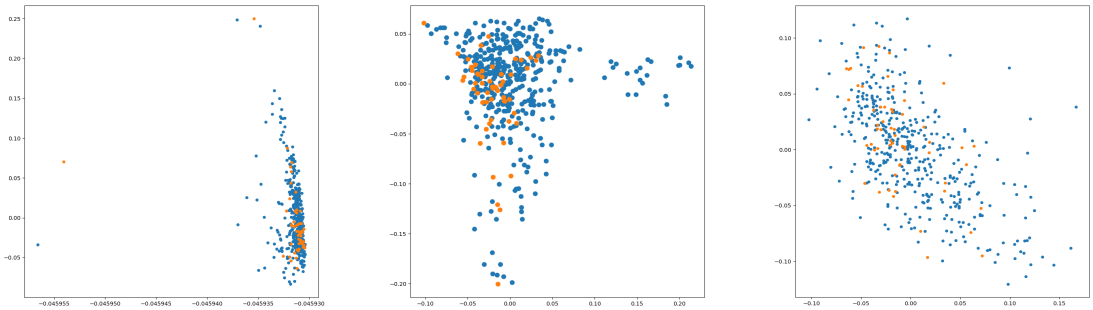


(a) Best Omic-view (b) SNF (c) WMLRR

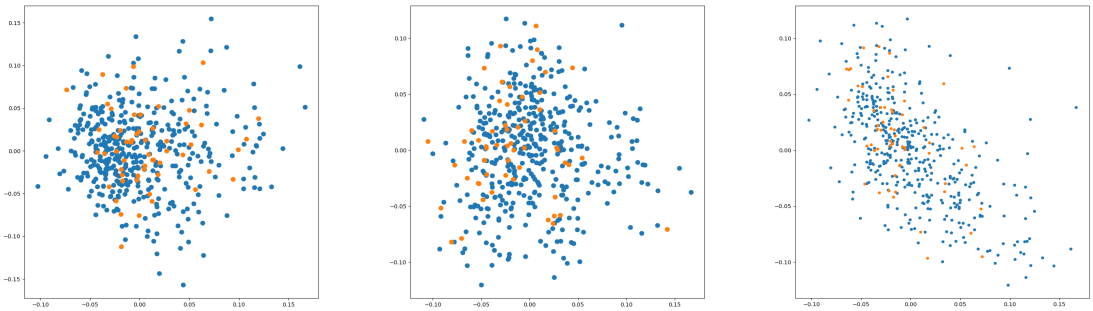
(d) SURE (e) CoALa (f) MiMIC

(g) iCluster (h) RISynG

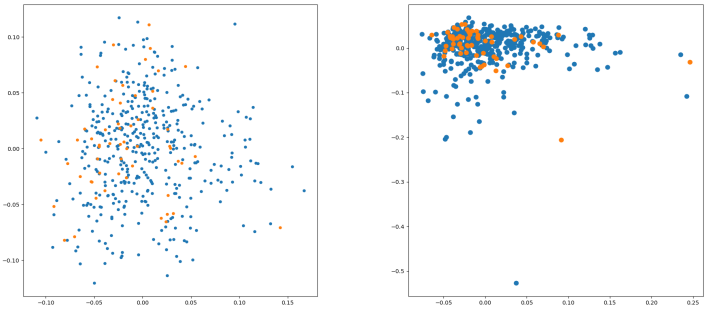
Figure 5.5. : Comparative analysis of different integrative sub-spaces on BRCA data set



(a) Best Omic-view (b) SNF (c) WMLRR

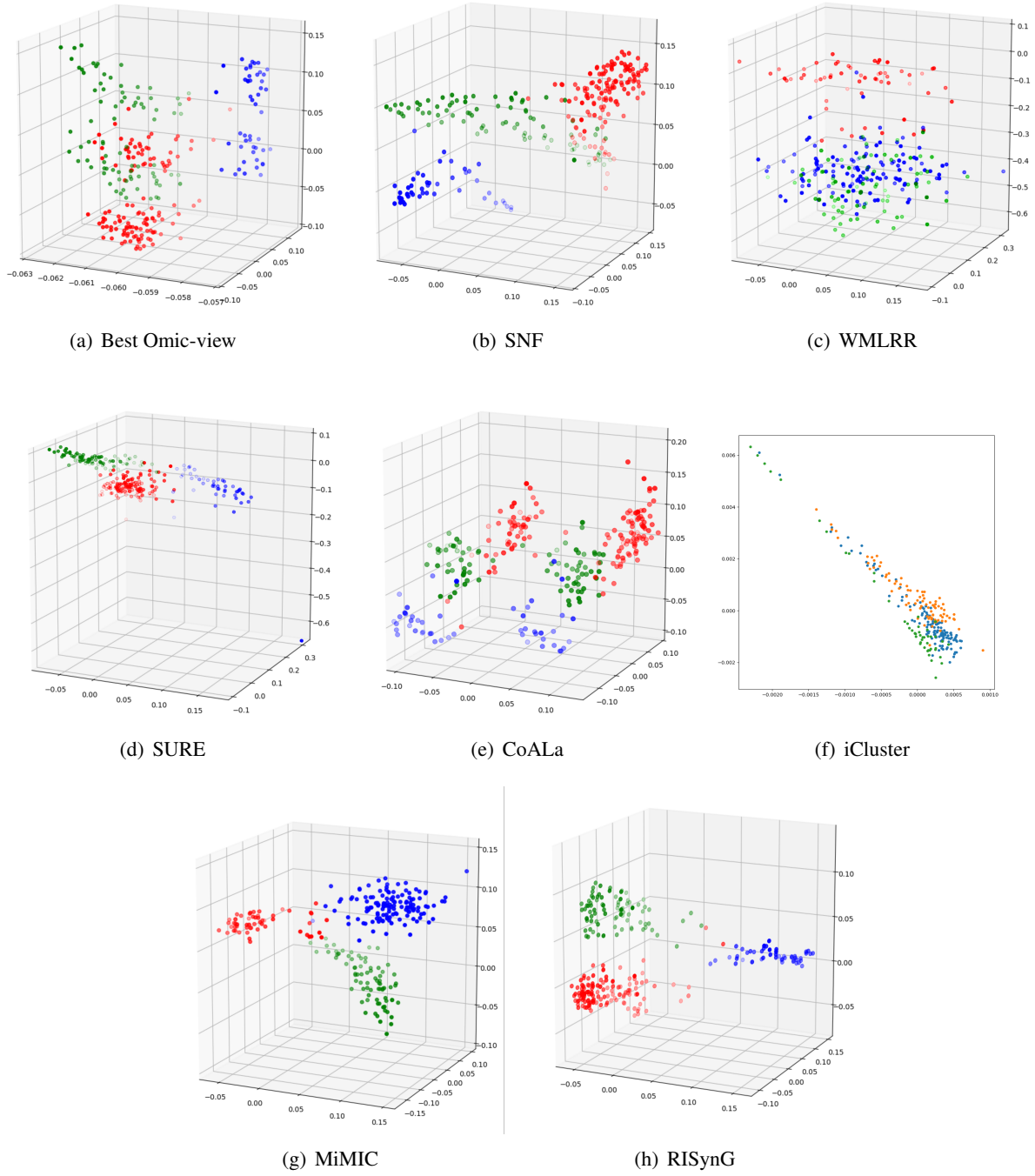


(d) SURE (e) CoALa (f) MiMIC



(g) iCluster (h) RISynG

Figure 5.6. : Comparative analysis of different integrative sub-spaces on OV data set

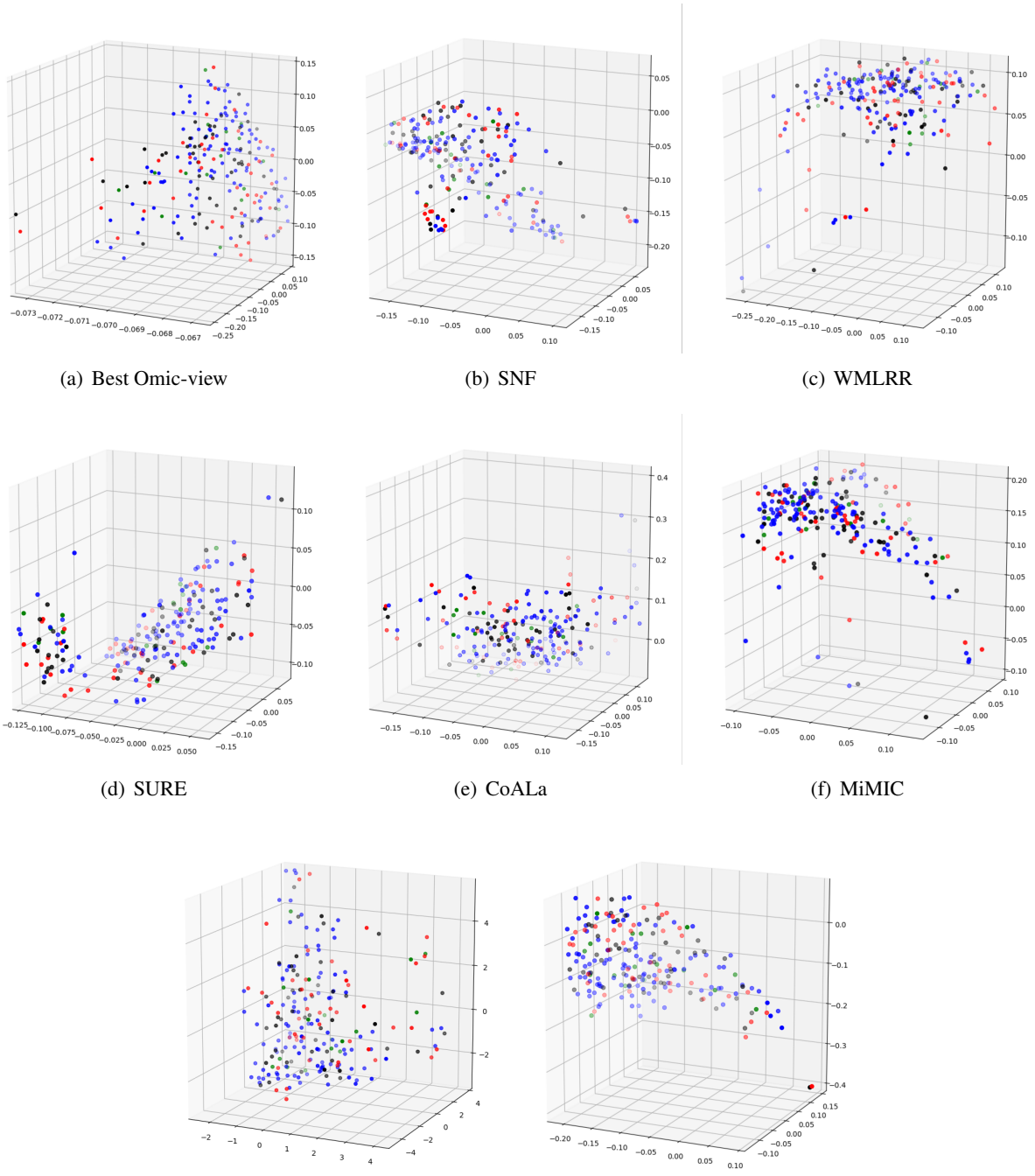


(a) Best Omic-view (b) SNF (c) WMLRR

(d) SURE (e) CoALa (f) iCluster

(g) MiMIC (h) RISynG

Figure 5.7. : Comparative analysis of different integrative sub-spaces on LGG data set



(a) Best Omic-view (b) SNF (c) WMLRR

(d) SURE (e) CoALa (f) MiMIC

(g) iCluster (h) RISynG

Figure 5.8. : Comparative analysis of different integrative sub-spaces on STAD data set

value of these scores better is the pathway enrichment; hence more biologically significant the differentially expressed biomarkers are that means better is the cancer sub-typing. Following are the equations for these scores:

$$KPES = \frac{1}{K}\sum_{k=1}^{K} -log_{10}(p-value_k) \tag{5.59}$$

$$AR = \frac{1}{K*G}\sum_{i=1}^{K} g_i \tag{5.60}$$

Here, K denotes the number of significant pathway terms, associated with a set of differentially expressed genes or miRNAs between two cancer subtypes identified by any of the clustering approaches. G denotes the total number of genes given to clusterProfiler for the enrichment analysis and g denotes gene count associated with an enrichment term. Comparative analysis of the cancer subtypes obtained by the proposed approach and other existing algorithms are performed and the KPES and AR for mRNA PEA and KPES for miRNAs PEA are reported in Table 5.4, 5.5, 5.6, 5.7, and 5.8 for different cancer data-sets. Some of the sets of differentially expressed miRNAs or mRNAs have no associated significant KEGG pathway terms; therefore, there is no scope for calculating KPES and AR. Also, in some cases, there is no differentially expressed biomarkers. All these cases are represented by $*$ in the tables.

Biological Process Enrichment Analysis

The hundred most differentially expressed biomarkers (miRNAs and mRNAs) obtained from cancer subtypes identified by different methods are also passed to Biological process enrichment analysis (BPEA). The gene ontology consortium contains the information of annotation of individual genes. The annotations are grouped into molecular function (MF), biological processes (BP), and cellular component (CC). BPEA helps in characterizing the relationship between genes or miRNAs by specifically annotating them to associated biological processes or a series of molecular functions of which they are part. It helps in the identification of the biological processes that are over-represented in our list, that can help in evaluating the the biological significance of the obtained cancer-subtypes. Like, PEA, R package clusterProfiler (Yu et al., 2012) and DIANA Tools mirPath v.3 (Vlachosa et al., 2012) are again used for performing BPEA for genes and miRNAs, respectively. All the biological process (BP) terms associated with the set of biomarkers having false discovery rate adjusted p-value < 0.05 (significant pathway terms) are only considered. If any of the differentially expressed biomarker sets are not associated with significant BP terms, that set is said to be not biologically relevant with respect to BP terms. For the quantification of BPEA, two scores, namely, BP enrichment score (BPES) (Paul and Madhumita, 2020), and Annotation ratio (AR) (Consortium, 2000b), are also calculated. The higher the value of these scores better is the BP enrichment; hence, the more biologically significant the differentially expressed biomarkers are, the better the cancer sub-typing. Following are the equations for these scores:

$$BPES = \frac{1}{B}\sum_{b=1}^{B} -log_{10}(p-value_b) \tag{5.61}$$

$$AR = \frac{1}{B*G}\sum_{i=1}^{B} g_i \tag{5.62}$$

Here, B is number of significant BP terms, associated with a set of differentially expressed genes or miRNAs between two cancer subtypes identified by any of the clustering approaches. G denotes the total number of genes gives to clusterProfiler for the enrichment analysis and g denotes gene count associated with a BP enrichment term. Comparative analysis of the cancer subtypes obtained by the proposed approach and other existing algorithms are performed and the BPES and AR for mRNA BPEA and BPES for miRNA BPEA are

reported in Table 5.4, 5.5, 5.6, 5.7, and 5.8 for different cancer data-sets. Some of the sets of differentially expressed miRNAs or mRNAs have no associated significant BP terms; therefore, there is no scope for calculating BPES and AR. Also, in some cases, there are no differentially expressed biomarkers. All these cases are represented by $*$ in the tables.

Disease Ontology Enrichment Analysis

To further map the clinical relevance of the cancer subtypes identified from high-throughput data Disease ontology enrichment analysis (DOEA) is also performed. R package DOSE (Yu et al., 2015) is used for this purpose. It calculates semantic similarities between Disease-ontology (DO) terms and genes that help explore the similarities of diseases and gene functions from a disease perspective. The output of DOES has associated disease terms. A gene set is said to be enriched with DO terms if the terms obtained by its DOEA have a FDR corrected p-value < 0.05. Hundred most differentially expressed genes identified between cancer subtypes-pairs of different cancer types identified by different methods are passed to DOEA. Like, PEA and BPEA, quantification of DOEA is also performed by calculating Disease ontology enrichment score (DOES) and AR as follows:

$$DOES = \frac{1}{D}\sum_{d=1}^{B} -log_{10}(p-value_d) \tag{5.63}$$

$$AR = \frac{1}{D*G}\sum_{i=1}^{D} g_i \tag{5.64}$$

Here, D denotes the number of significant DO terms, associated with a set of differentially expressed genes. G denotes the total number of genes gives to DOSE for the enrichment analysis and g denotes gene count associated with a DO term. Comparative analysis of the cancer subtypes obtained by the proposed approach and other existing algorithms are performed and the DOES and AR for mRNA DOEA are reported in Table 5.4, 5.5, 5.6, 5.7, and 5.8 for different cancer data-sets. Some of the sets of differentially expressed genes have no associated significant DO terms; therefore, there is no scope for calculating DOES and AR. Also, in some cases, there are no differentially expressed biomarkers. All these cases are represented by $*$ in the tables.

To compare the effectiveness of the proposed approach with the other algorithms in this study, the overall performance of all the methods is also evaluated. When all the five cancer datasets are considered together, the proposed approach outperforms concerning both cluster evaluation indices and biological enrichment analysis, as shown in Figure 5.9. The analysis is performed by considering the success frequency (number of times a method scored the highest value for respective indices when all the cases in all the cancer types are considered). The success frequency shows that the proposed approach outperforms when cluster validity indices are considered by scoring maximum values for 21 times, followed by SNF.CC (7), SNF (6), CNMF (5), CC (2), COCA (2), and WMLRR (1). Similarly, suppose the methods are ranked considering the success frequency for quantitative indices calculated for biological enrichment analysis. In that case, the proposed approach will again stand first by scoring the maximum value 67 times, followed by SNF (21), SNF.CC (20), CC (12), CoALa (10), CNMF (9), MiMIC (7), SURE (5), WMLFF (5), COCA (4), and iCluster (1). If the cluster validity indices are looked upon individually, the proposed approach also outperforms with respect to F-measure, ARI, NMI, Jaccard index, and Purity. Considering the indices for biological enrichment individually, the proposed algorithm again outperforms with respect to all the indices except for AR for BPES for mRNA enrichment, where it stands second.

Overlap Analysis

The hundred most differentially expressed genes between all the subtypes-pairs in cervical cancer that were identified by RISynG and the other methods are explored further for experimental support. The genes are analyzed in terms of the degree of overlap with known cervical cancer genes that are

Table 5.4. : Comparative Biological Analysis of CESC dataset

		mRNA enrichment analysis						miRNA enrichment analysis		Number of differentially expressed	
CESC classes	Methods	KPES	AR	BPES	AR	DOES	AR	KPES	BPES	mRNAs	miRNAs
	SNF	2.253130	**0.013174**	2.124593	**0.014242**	1.492051	0.002597	*	*	811	9
	CC	3.688672	0.001579	2.227388	0.002396	1.533960	0.008514	*	*	829	5
	CNMF	1.533344	0.006379	2.021209	0.001684	1.383268	0.007361	*	*	719	3
	ECMC	1.658394	0.003756	0.002134	0.006945	0.385745	0.003684	*	*	529	6
	WMLRR	*	*	*	*	*	*	*	*	34	4
Keratin low squamous	SNF.CC	2.253130	**0.013174**	2.124593	**0.014242**	1.492051	0.002597	*	*	811	9
Vs	CoALa	1.650947	0.004808	2.381972	0.006842	1.397646	**0.010533**	*	*	632	5
Keratin high squamous	COCA	2.742077	0.010161	1.540280	0.010722	1.764804	0.000658	*	*	529	1
	SURE	10.322586	0.005417	4.595299	0.009796	1.656721	0.007143	*	*	915	7
	iCluster	6.456398	0.002674	3.564834	**0.014240**	1.465782	0.003657	3.459342	1.45698	939	92
	MiMIC	5.739475	0.003295	2.593754	0.004743	1.285647	0.004738	*	*	643	6
	RISynG	**19.023039**	0.003559	**4.483853**	0.007835	**1.993933**	0.003492	**4.730997**	**3.702912**	924	13
	SNF	2.228672	0.010167	2.060098	0.007604	1.586351	**0.030129**	**2.839092**	3.702912	748	23
	CC	1.524904	0.004545	2.005994	0.003854	1.396591	0.011918	**2.839092**	3.510903	545	19
	CNMF	1.837035	0.001731	*	*	**1.985854**	0.001143	2.838999	3.510903	398	13
	ECMC	0.568364	0.003956	2.967456	0.002756	1.486344	0.018934	1.459674	1.498264	544	54
	WMLRR	1.486745	0.002758	1.584754	0.001845	*	*	*	*	66	12
Keratin low squamous	SNF.CC	2.228672	0.003585	2.060098	0.007604	1.586351	**0.030129**	2.838999	3.510903	748	23
Vs	CoALa	*	*	1.819866	0.004375	1.530370	0.029351	2.838999	3.702912	489	17
Adenocarcinoma	COCA	2.095093	0.005333	2.266488	0.006105	1.812902	0.011081	2.838999	3.702912	600	14
	SURE	1.986232	0.002903	1.825145	**0.007835**	1.773099	0.021688	2.838999	4.853914	826	24
	iCluster	1.657453	0.003746	2.567439	0.006345	*	*	2.838998	3.675432	842	14
	MiMIC	1.698345	0.003521	2.674931	0.005392	1.436284	0.016385	*	*	819	2
	RISynG	**2.902596**	**0.010166**	**4.699835**	0.000833	*	*	2.186029	**7.052373**	966	34
	SNF	*	*	**3.405602**	0.004409	*	*	**2.838998**	7.034831	774	38
	CC	*	*	2.306625	0.004421	1.580965	**0.006619**	**2.838999**	7.034831	652	27
	CNMF	*	*	2.271344	0.003191	**1.822238**	0.004493	**2.838998**	7.034831	478	23
	ECMC	0.568364	0.003956	2.967456	0.002756	1.486344	0.018934	1.459674	1.498264	544	54
	WMLRR	1.385674	0.859432	1.674834	0.002856	1.495733	0.001745	*	*	633	17
Keratin high squamous	SNF.CC	*	*	1.701994	0.004409	*	*	**2.838998**	7.034831	774	39
Vs	CoALa	*	*	2.934221	0.005957	1.505341	0.006575	*	*	535	18
Adenocarcinoma	COCA	*	*	3.229027	0.004839	*	*	**2.838998**	7.105402	553	20
	SURE	*	*	2.931849	0.003871	1.432160	0.001884	**2.838998**	**7.819319**	790	52
	iCluster	*	*	2.627392	0.005734	*	*	1.645321	3.458233	637	96
	MiMIC	**2.564893**	**1.453983**	2.845623	0.001634	*	*	*	*	412	0
	RISynG	*	*	**3.405602**	**0.009148**	*	*	**2.838998**	**7.819319**	951	63

experimentally validated. The Cervical Cancer Gene Database (CCDB) (Agarwal et al., 2011) is used for finding the overlap. It is a manually curated catalog of experimentally validated genes involved in the different stages of cervical carcinogenesis. All the up-regulated and down-regulated genes in cervical cancer with evidence from the published literature available in CCDB are considered for this analysis. 367 genes are reported in CCDB that are differentially expressed in cervical cancer. This list contains 185 genes from a total number of 2000 genes that are used for cancer-subtype identification in this study. The statistical significance of the overlap analysis is reported in Table 5.9. In total, 30 genes out of 222 identified from the proposed approach are overlapping with cervical cancer-related genes. This is the maximum overlap when compared with the other methods. Fisher's exact test is used here to find the statistical significance of the contingency table created from the overlap analysis in Table 5.9 for different algorithms. At 95% confidence, it is observed that only the genes identified by the the proposed approach has significant overlap with experimentally validated genes curated from literature with a p-value of 0.026. Therefore, it indicates that the proposed approach has the potential to identify clinically important subtypes of cancer that have a characteristic molecular signature.

5.5 CONCLUSION

This chapter presents a multi-omics integration and clustering algorithm named RISynG that efficiently identifies cancer subtypes. RISynG acknowledges the statistical diversity in the multi-view omic data as it directly influences the quantification of sample similarity. The proposed method assumes multiple information sources may constitute different cluster manifolds when learned on a particular similarity measure. Therefore predicted clusters would be apparent and in strong concordance with the clinical clusters if pair-wise sample similarity is calculated in the data-dependent multi-kernel approach. Therefore, the method uses two different graph representations matrices with different similarity measures to quantify

Table 5.5. : Comparative Biological Analysis of BRCA dataset

		mRNA enrichment analysis						miRNA enrichment analysis		Number of differentially expressed	
BRCA classes	Methods	KPES	AR	BPES	AR	DOES	AR	KPES	BPES	mRNAs	miRNAs
	SNF	1.461915	0.002553	1.385690	0.000306	**2.477131**	0.001970	2.372875	5.449253	1198	106
	CC	1.478030	0.000755	1.365630	0.000306	*	*	2.632318	4.086584	864	49
	CNMF	**1.959958**	**0.0024**	*	*	2.461538	0.003279	*	**6.847070**	367	3
	ECMC	*	*	*	*	*	*	*	*	21	7
	WMLRR	*	*	*	*	*	*	*	*	0	5
Luminal A	SNF.CC	1.613135	0.001379	2.372456	0.006186	2.135156	0.082714	3.676487	6.071079	236	45
Vs	CoALa	1.418795	0.001311	**2.637358**	**0.006562**	2.162850	**0.089027**	**6.361579**	5.144748	225	36
Luminal B	COCA	*	*	*	*	*	*	*	*	0	0
	SURE	*	*	*	*	*	*	*	*	367	0
	iCluster	*	*	*	*	*	*	*	*	1186	89
	MiMIC	*	*	1.598348	0.0002643	*	*	*	*	586	14
	RISynG	*	*	*	*	*	*	*	*	0	0
	SNF	*	*	*	*	*	*	2.143165	5.339598	1522	181
	CC	*	*	*	*	1.679493	0.001791	1.975801	5.557729	1429	129
	CNMF	*	*	*	*	*	*	2.316795	*	1501	181
Luminal A	SNF.CC	1.528961	0.003288	2.484664	0.007766	**2.472259**	**0.081621**	4.772078	5.635141	204	34
	ECMC	1.645964	0.002194	0.763859	0.002856	*	*	*	*	516	19
	WMLRR	1.456983	0.0018465	1.678453	0.003956	1.649663	0.045843	3.856433	4.856377	1125	11
Vs	CoALa	2.053401	0.003889	1.352696	0.004352	2.418967	0.066857	**5.981546**	5.361591	175	39
Her-2 Enriched	COCA	*	*	*	*	*	*	2.182702	**6.704499**	1419	175
	SURE	*	*	*	*	*	*	2.143102	5.339598	1501	179
	iCluster	*	*	*	*	*	*	3.185943	4.675320	1486	177
	MiMIC	1.957394	0.002845	1.749245	0.002945	1.849382	0.025383	2.795843	3.738291	1891	181
	RISynG	**2.069400**	**0.004285**	**2.552696**	**0.008152**	*	*	2.143165	5.339598	1555	181
	SNF	*	*	1.968809	0.001237	1.671238	0.014714	*	6.208960	1514	134
	CC	*	*	*	*	2.650350	0.001944	2.099859	**7.595675**	1426	115
	CNMF	1.527132	0.001176	*	*	2.315198	0.002143	2.027684	4.551682	1420	111
	ECMC	*	*	*	*	*	*	*	*	112	0
	WMLRR	*	*	1.894784	0.001768	*	*	*	*	253	16
Luminal A	SNF.CC	2.301212	0.001296	**2.320499**	**0.011444**	1.586312	0.004521	*	*	155	5
Vs	CoALa	*	*	*	*	*	*	*	*	0	0
Basal Like	COCA	*	*	1.867577	0.001563	2.715869	0.001972	2.182702	6.704499	1255	115
	SURE	*	*	*	*	*	*	2.463865	4.086584	1420	66
	iCluster	1.567384	0.006354	*	*	*	*	1.945638	5.639721	1530	133
	MiMIC	1.563985	0.005932	1.486932	0.003285	1.748234	0.032863	0.638291	0.638294	512	110
	RISynG	**2.318606**	**0.02**	*	*	**3.529331**	**0.035571**	**2.518622**	4.086584	1153	61
	SNF	1.556018	0.006964	1.895925	0.001087	**1.711230**	0.004521	2.702356	**7.005326**	1074	111
	CC	1.848055	0.001961	*	*	1.611678	**0.043432**	**2.893976**	2.140340	734	27
	CNMF	*	*	*	*	*	*	*	*	32	0
Luminal B	SNF.CC	1.768648	0.001296	**2.439050**	**0.004365**	1.505375	0.037612	2.702356	**7.005326**	1173	119
	ECMC	1.245874	0.001749	0.846396	0.002469	1.285749	0.0175635	2.489275	4.867463	972	94
	WMLRR	**2.485737**	**0.185483**	1.946385	0.000174	0.946285	0.02857	1.674927	4.683956	1432	116
Vs	CoALa	1.516468	0.000806	2.296736	0.002857	1.422305	0.012857	2.702356	**7.005326**	1209	113
Her-2 Enriched	COCA	*	*	*	*	*	*	*	*	0	0
	SURE	*	*	*	*	*	*	*	*	32	0
	iCluster	*	*	*	*	*	*	2.567493	6.978346	135	90
	MiMIC	*	*	*	*	*	*	*	*	65	4
	RISynG	*	*	*	*	*	*	*	*	0	0
	SNF	**2.128049**	**0.0156**	*	*	**1.819763**	**0.060895**	**1.838692**	**3.481747**	1096	99
	CC	1.658311	0.006875	1.370137	0.000313	1.705471	0.044328	*	2.140340	897	66
	CNMF	*	*	*	*	*	*	*	*	15	0
Luminal B	SNF.CC	2.118175	0.008654	**2.476302**	**0.013333**	1.695800	0.009855	*	*	158	0
	ECMC	*	*	*	*	*	*	*	*	54	0
	WMLRR	1.756342	0.001856	1.674834	0.002849	*	*	*	*	856	0
Vs	CoALa	*	*	*	*	*	*	*	*	0	0
Basal Like	COCA	*	*	*	*	*	*	*	*	0	0
	SURE	*	*	*	*	*	*	*	*	15	0
	iCluster	*	*	*	*	*	*	*	*	1051	57
	MiMIC	*	*	*	*	*	*	*	*	84	0
	RISynG	*	*	*	*	*	*	*	*	0	0
	SNF	1.655414	0.003158	3.157786	0.002188	2.523476	0.110366	*	*	1120	152
	CC	1.920310	0.000714	2.702511	0.001505	1.800734	0.008451	*	*	450	11
	CNMF	2.143113	0.002097	3.196699	0.015612	2.245955	0.056154	2.749089	46.059107	1057	147
	ECMC	1.658453	0.006856	1.475986	0.001845	1.678432	0.027923	1.48234	4.674983	943	89
	WMLRR	1.287463	0.004738	2.476993	0.004845	1.283622	0.001745	1.945673	4.687384	1921	112
Her-2 Enriched	SNF.CC	2.558315	0.001212	3.033794	0.018085	1.774396	0.021528	*	*	112	0
Vs	CoALa	*	*	*	*	*	*	*	*	0	0
Basal Like	COCA	1.830650	0.002623	1.691438	0.003438	*	*	2.382039	4.804657	362	231
	SURE	*	*	*	*	*	*	2.615351	4.166632	1057	49
	iCluster	2.598346	0.002864	3.167489	0.017456	*	*	2.498563	3.758943	1175	128
	MiMIC	2.649362	0.003785	2.749734	0.001373	*	*	*	*	278	0
	RISynG	**2.844053**	**0.013333**	**3.384443**	**0.024848**	**2.307443**	**0.125301**	**2.938526**	**6.524382**	900	60

Table 5.6. : Comparative Biological Analysis of OV dataset

		mRNA enrichment analysis						miRNA enrichment analysis		Number of differentially expressed	
OV classes	Methods	KPES	AR	BPES	AR	DOES	AR	KPES	BPES	mRNA	miRNA
	SNF	1.898710	0.002381	**2.061189**	**0.001053**	*	*	*	7.871558	268	230
	CC	*	*	*	*	*	*	*	*	0	0
	CNMF	*	*	*	*	*	*	*	*	0	0
	ECMC	*	*	*	*	*	*	*	*	54	0
	WMLRR	*	*	2.148563	0.000584	*	*	*	*	123	11
Vs	CoALa	4.675430	0.003810	*	*	*	*	2.156299	10.129178	214	178
Neoplasm histological grade 3	SNF.CC	5.155090	0.003125	1.354431	0.000947	*	*	*	*	331	203
Neoplasm histological grade 2	COCA	*	*	*	*	*	*	*	*	0	0
	SURE	1.640585	0.010000	*	*	*	*	2.456219	5.679494	485	19
	iCluster	5.864734	0.016740	1.256483	0.001030	*	*	1.178456	7.457563	664	228
	MiMIC	5.327456	0.014639	1.645294	0.000143	**2.745934**	**0.763549**	2.458396	7.935281	981	172
	RISynG	**7.004104**	**0.020000**	*	*	*	*	**2.715358**	**13.354195**	256	209

Table 5.7. : Comparative Biological Analysis of LGG dataset

		mRNA enrichment analysis						miRNA enrichment analysis		Number of differentially expressed	
LGG classes	Methods	KPES	AR	BPES	AR	DOES	AR	KPES	BPES	mRNAs	miRNAs
	SNF	2.216865	0.004390	1.712976	0.010313	1.518906	**0.007586**	2.623947	7.443965	1530	180
	CC	2.060736	0.002167	1.828709	0.002371	*	*	2.381158	5.218678	1394	185
	CNMF	2.713893	0.004400	2.184500	0.023895	**1.737469**	0.003881	2.377525	6.805268	1475	197
	ECMC	2.569342	0.017453	2.983657	0.011547	1.385943	0.004856	3.684935	6.956385	1632	87
	WMLRR	2.957453	0.003758	2.567396	0.003854	0.946583	0.004867	2.956486	3.574869	943	114
IDH mutation without 1p/19q codeletion	SNF.CC	2.158081	0.001250	1.857196	**0.012580**	*	*	2.433907	6.981976	1346	186
Vs	CoALa	*	*	*	*	*	*	*	*	0	0
IDH mutation with 1p/19q codeletion	COCA	1.868207	0.005893	1.445307	0.002021	*	*	3.009416	**8.168954**	1091	209
	SURE	1.667323	0.000862	2.918052	0.002421	*	*	1.722047	2.988518	1022	46
	iCluster	2.6547342	0.00345628	2.674538	0.0045638	1.265483	0.0034562	3.256437	3.756417	1445	180
	MiMIC	2.983452	0.0352854	1.845372	0.011845	1.436956	0.004637	3.756281	5.342743	1223	124
	RISynG	**3.015521**	**0.047500**	**3.165604**	0.000729	*	*	**5.831749**	2.118806	1286	129
	SNF	5.334880	0.002250	2.177240	0.001848	*	*	2.191319	4.547026	1380	151
	CC	10.753373	0.002857	4.952588	0.002444	*	*	**3.276499**	3.693358	1334	255
	CNMF	3.828751	0.001905	1.827162	0.001739	*	*	2.640297	6.127825	1410	152
	ECMC	*	*	*	*	*	*	*	*	85	16
	WMLRR	17.47594	0.028475	3.674834	0.017385	*	*	*	*	323	35
IDH mutation without 1p/19q codeletion	SNF.CC	2.313690	**0.030833**	2.004395	**0.030232**	*	*	*	*	1151	0
Vs	CoALa	2.308687	0.005500	1.676198	0.012824	**1.835847**	**0.002712**	2.134683	6.374528	995	120
Wild type IDH subtype	COCA	*	*	*	*	*	*	2.191284	4.547026	1333	147
	SURE	8.396959	0.003750	4.759305	0.003146	*	*	2.598835	5.097597	1207	90
	iCluster	7.753974	0.002634	3.378298	0.004573	*	*	1.456382	4.634529	1476	203
	MiMIC	5.437984	0.016453	4.863478	0.024537	0.562849	0.000174	*	*	312	0
	RISynG	**16.665788**	0.002553	**6.554131**	0.001868	*	*	2.957544	**7.361838**	1299	162
	SNF	4.872952	0.001818	2.830515	0.003646	*	*	2.433906	6.981966	1433	186
	CC	1.770337	0.004118	1.845716	0.003118	2.150893	0.008475	**2.532364**	4.105869	1463	154
	CNMF	4.479526	0.004310	2.153374	0.008229	*	*	1.774068	5.301943	1299	177
	ECMC	*	*	2.584763	0.028496	*	*	*	*	122	5
	WMLRR	**5.968354**	0.005846	1.956384	0.017485	1.956486	1.956738	1.745867	5.956385	1765	165
IDH mutation with 1p/19q codeletion	SNF.CC	2.669711	**0.009000**	2.035048	0.026484	1.985843	0.009483	2.623947	**7.443965**	888	180
Vs	CoALa	*	*	*	*	*	*	*	*	0	0
Wild type IDH subtype	COCA	2.144910	0.007805	1.803831	0.020526	1.611270	0.006102	2.301836	7.175007	1320	195
	SURE	1.456283	0.002459	2.830515	0.004409	2.543007	0.009483	**2.532364**	4.105869	1321	118
	iCluster	1.456735	0.001645	1.456872	0.001674	*	*	1.456380	4.764537	1172	148
	MiMIC	3.956382	0.004281	2.459216	**0.038294**	2.453856	0.000352	1.463823	3.785645	1145	193
	RISynG	1.761283	0.002683	**2.835968**	0.004409	**2.866000**	**0.015172**	2.355656	5.218530	1321	144

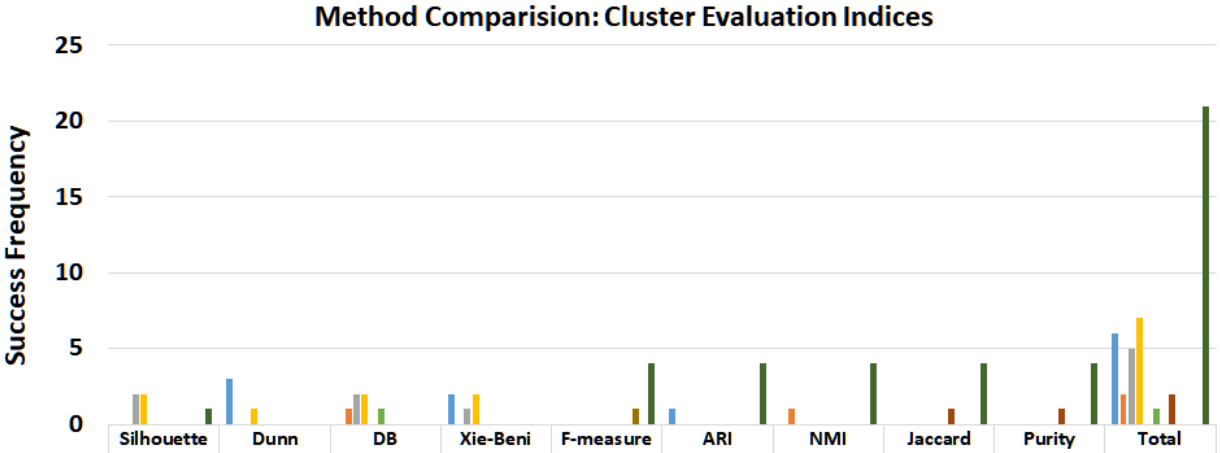
Table 5.8. : Comparative Biological Analysis of STAD dataset

		mRNA enrichment analysis						miRNA enrichment analysis		Number of differentially expressed	
STAD classes	Methods	KPES	AR	BPES	AR	DOES	AR	KPES	BPES	mRNAs	miRNAs
	SNF	*	*	**2.947976**	**0.007778**	*	*	3.886525	5.339598	51	139
	CC	*	*	*	*	*	*	3.475801	5.557729	0	117
	CNMF	*	*	*	*	*	*	2.286346	5.274368	0	173
	ECMC	*	*	1.486745	0.001754	*	*	*	*	76	19
	WMLRR	*	*	*	*	*	*	*	*	0	0
MSI	SNF.CC	**1.641207**	**0.009500**	1.808831	0.003590	*	*	4.663422	5.635141	51	308
Vs	CoALa	*	*	*	*	*	*	2.167354	5.631591	0	130
EBV	COCA	*	*	*	*	*	*	4.182702	5.649816	0	130
	SURE	*	*	*	*	*	*	2.143102	5.339598	1939	502
	iCluster	*	*	*	*	*	*	*	*	0	3
	MiMIC	*	*	*	*	*	*	*	*	0	0
	RISynG	*	*	*	*	*	*	**5.981546**	**6.704499**	0	138
	SNF	3.206072	0.006809	1.919825	0.012532	2.101852	0.005965	2.640297	4.547026	88	213
	CC	2.463472	0.004118	1.677523	0.014000	1.732361	0.004000	1.191319	3.593216	233	147
	CNMF	*	*	1.412950	0.005217	*	*	2.275633	3.547026	128	74
	ECMC	3.867453	0.002756	1.584732	0.001748	1.093748	0.004856	0.083956	3.956483	954	99
	WMLRR	1.856093	0.003756	2.256943	0.018463	2.056783	0.005734	1.756398	5.935744	432	118
MSI	SNF.CC	1.517881	0.001111	1.376410	0.001970	*	*	1.337563	5.097597	88	388
Vs	CoALa	3.152809	0.006471	2.094399	0.013291	2.033564	0.004918	2.718835	5.127825	288	219
CIN	COCA	3.152809	0.006471	2.094399	0.013291	2.033564	0.004918	0.387436	4.447824	288	219
	SURE	3.494658	0.005882	2.250916	0.013377	**2.194897**	0.002586	2.191284	5.376595	333	239
	iCluster	3.679345	0.003785	1.764538	0.003856	1.456396	0.006742	1.645297	4.762394	130	260
	MiMIC	4.678234	0.006453	1.845632	**0.0267453**	2.074592	0.005643	1.645372	4.756382	204	88
	RISynG	**5.672025**	**0.007447**	**2.361069**	0.011270	1.859974	**0.008033**	**2.957544**	**6.374528**	168	216
	SNF	1.625946	0.003333		0.013099	1.667743	0.013158	**3.534893**	2.140340	46	128
	CC	4.408086	0.006061	1.929352	0.014853	**2.178884**	0.008039	2.674398	2.238740	297	108
	CNMF	2.923027	0.001667	2.062190	0.016000	1.778082	**0.015600**	2.226550	2.347540	90	17
	ECMC	4.756823	0.002745	0.587384	0.027485	1.748396	0.003856	2.567498	2.986482	543	164
	WMLRR	3.056845	0.003856	0.574983	0.003758	2.489564	0.002745	1.859644	3.975844	643	219
MSI	SNF.CC	2.142436	0.007500	1.771466	0.000811	*	*	*	*	46	284
Vs	CoALa	3.008414	0.007143	1.836751	0.015467	1.701261	0.014107	2.615463	**4.376543**	220	129
GS	COCA	3.008414	0.007143	1.836751	0.015467	1.701261	0.014107	2.615463	**4.376543**	220	129
	SURE	3.245879	0.004889	1.889092	0.013288	2.090617	0.010727	3.286473	2.238740	124	42
	iCluster	3.651936	0.014563	0.674328	0.003549	1.845362	0.007453	3.254698	3.934657	27	112
	MiMIC	2.145936	0.004563	1.434328	**0.141283**	1.732362	0.005453	*	*	116	0
	RISynG	**5.266844**	**0.018500**	**1.960004**	0.001935	*	*	3.223198	3.674321	39	303
	SNF	*	*	*	*	*	*	1.481368	3.654953	0	237
	CC	*	*	*	*	*	*	2.228647	**4.578635**	0	214
	CNMF	*	*	*	*	*	*	2.193468	1.645390	0	35
	ECMC	*	*	*	*	*	*	*	*	0	15
	WMLRR	**2.856748**	**0.947567**	0.364867	0.001985	*	*	*	*	156	65
EBV	SNF.CC	*	*	*	*	*	*	*	*	0	0
Vs	CoALa	*	*	*	*	*	*	1.774584	2.984563	33	236
CIN	COCA	*	*	*	*	*	*	2.193468	2.537548	33	236
	SURE	*	*	1.375449	0.002295	*	*	2.306584	2.756483	1939	502
	iCluster	*	*	*	*	*	*	*	*	15	0
	MiMIC	*	*	*	*	*	*	*	*	7	0
	RISynG	*	*	**1.643941**	**0.002857**	*	*	**2.312785**	2.865474	18	139
	SNF	*	*	*	*	*	*	1.481368	3.654953	0	119
	CC	*	*	*	*	*	*	2.228647	**4.578635**	0	102
	CNMF	*	*	*	*	*	*	2.193468	1.645390	0	31
	ECMC	*	*	*	*	*	*	*	*	17	6
	WMLRR	*	*	0.678465	1.298456	*	*	*	*	119	12
EBV	SNF.CC	*	*	*	*	*	*	*	*	0	42
Vs	CoALa	*	*	*	*	*	*	1.774584	2.984563	0	134
GS	COCA	*	*	*	*	*	*	2.193468	2.537548	0	134
	SURE	*	*	1.375449	0.002295	*	*	2.306584	2.756483	1939	502
	iCluster	*	*	*	*	*	*	*	*	70	0
	MiMIC	1.274567	0.006543	1.274973	0.009451	1.374516	0.004373	2.264983	3.564832	110	67
	RISynG	**1.424199**	**0.010000**	**1.686765**	**0.008333**	*	*	**2.312785**	2.865474	6	60
	SNF	*	*	*	*	*	*	2.546392	3.058563	0	255
	CC	1.609349	**0.007222**	*	*	*	*	2.964875	3.337658	247	39
	CNMF	**2.689639**	0.004063	1.780668	0.007778	*	*	*	*	229	17
	ECMC	0.856377	0.003756	1.658499	0.004867	*	*	2.568493	2.583993	132	213
	WMLRR	*	*	*	*	*	*	*	*	0	32
CIN	SNF.CC	*	*	*	*	*	*	1.948756	2.354654	0	212
Vs	CoALa	*	*	1.637374	0.000625	*	*	0.864563	2.904373	64	222
GS	COCA	*	*	1.637374	0.000625	*	*	1.084653	3.569463	64	222
	SURE	1.967703	0.004737	1.455467	0.002667	*	*	3.564875	4.724974	98	325
	iCluster	2.645382	0.0037845	1.956734	0.000376	*	*	1.567354	2.645983	69	14
	MiMIC	2.534967	0.005378	1.432134	0.001453	*	*	2.134975	2.195342	112	54
	RISynG	*	*	**2.947976**	**0.010656**	*	*	**3.956474**	**4.765984**	92	283

the association between the samples. A function is designed to estimate synergism between the two representation matrices using a weighted convex combination. The weights determine the representation matrices' importance and are estimated by an exhaustive search. Once the synergy matrix for each omic view is generated, relevance-based integration is performed. RISynG does that by recursive multi-kernel integration, where each synergy matrices' relevant subspace is iteratively integrated. Here, relevant subspace refers to that subspace of the matrix that purely encodes the cluster information, which in the case of synergy matrix is its eigenspace corresponding to k eigenvalues. At last, k-means clustering is applied to the integrated subspace, and cancer subtypes are identified. To demonstrate the efficiency of the proposed algorithm, it is applied to five multi-omics cancer datasets and compared to several state-of-the-art methods in this domain. The results show that RISynG outperforms several cluster evaluation indices and biological relevance analyses.

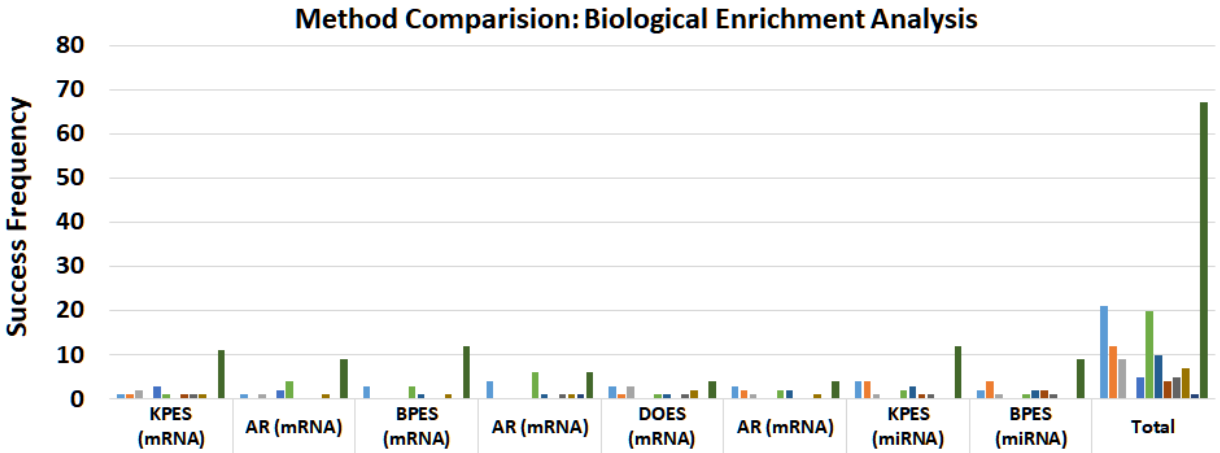
The severe data heterogeneity and high level of noise in the multi-omics data must be addressed simultaneously. In this regard, a novel deep neural network architecture-assisted pipeline is presented in the next chapter. The computational framework proposed in Chapter 6 learns the coherent manifold in the sample latent space for integrating multi-omics data and cancer subtyping. The proposed approach is designed using a sparse AE, which can perform early integration of nonlinear features of each omic view into a unified sample manifold. The information embedded in the latent space of the trained AE neural networks is used for cancer subtyping using Spectral clustering. The AE architecture in this study exhaustively searches for the best compression for multi-omics data by varying the number of neurons in the hidden layers and penalizing activations within the layers.

...



	Silhouette	Dunn	DB	Xie-Beni	F-measure	ARI	NMI	Jaccard	Purity	Total
SNF	0	3	0	2	0	1	0	0	0	6
CC	0	0	1	0	0	0	1	0	0	2
CNMF	2	0	2	1	0	0	0	0	0	5
SNF.CC	2	1	2	2	0	0	0	0	0	7
ECMC	0	0	0	0	0	0	0	0	0	0
WMLRR	0	0	1	0	0	0	0	0	0	1
CoALa	0	0	0	0	0	0	0	0	0	0
COCA	0	0	0	0	0	0	0	1	1	2
SURE	0	0	0	0	0	0	0	0	0	0
iCluster	0	0	0	0	1	0	0	0	0	0
MiMIC	0	0	0	0	0	0	0	0	0	0
RISynG	1	0	0	0	4	4	4	4	4	21

(a)



	KPES (mRNA)	AR (mRNA)	BPES (mRNA)	AR (mRNA)	DOES (mRNA)	AR (mRNA)	KPES (miRNA)	BPES (miRNA)	Total
SNF	1	1	3	4	3	3	4	2	21
CC	1	0	0	0	1	2	4	4	12
CNMF	2	1	0	0	3	1	1	1	9
ECMC	0	0	0	0	0	0	0	0	0
WMLRR	3	2	0	0	0	0	0	0	5
SNF.CC	1	4	3	6	1	2	2	1	20
CoALa	0	0	1	1	1	2	3	2	10
COCA	1	0	0	0	0	0	1	2	4
SURE	1	0	0	1	1	0	1	1	5
MiMIC	1	1	1	1	2	1	0	0	7
iCluster	0	0	0	1	0	0	0	0	1
RISynG	11	9	12	6	4	4	12	9	67

(b)

Figure 5.9. : Method Comparison

Table 5.9. : Overlap with experimentally validated gene-list

Methods		yes	no	Total	p-value
SNF	yes	22	202	224	0.715
	no	163	1613	1776	
CC	yes	20	198	218	1.000
	no	165	1617	1782	
CNMF	yes	26	197	223	0.219
	no	159	1618	1777	
ECMC	yes	9	272	281	0.053
	no	176	1543	1719	
WMLRR	yes	13	156	169	0.578
	no	172	1659	1831	
SNF.CC	yes	22	202	224	0.715
	no	163	1613	1776	
CoALa	yes	19	200	219	0.902
	no	166	1615	1781	
COCA	yes	23	194	217	0.457
	no	162	1621	1783	
SURE	yes	18	207	225	0.543
	no	167	1608	1775	
iCluster	yes	17	205	222	0.460
	no	168	1610	1778	
MiMIC	yes	19	216	235	0.631
	no	166	1599	1765	
RISynG	yes	30	192	222	0.026
	no	155	1623	1778	
Total		185	1815	2000	