Network Modeling Analysis in Health Informatics and Bioinformatics (2022) 11:24
https://doi.org/10.1007/s13721-022-00365-3

ORIGINAL ARTICLE

# SGAClust: Semi-supervised Graph Attraction Clustering of gene expression data

**Koyel Mandal**[1] · **Rosy Sarmah**[1]

Received: 24 September 2021 / Revised: 13 May 2022 / Accepted: 14 May 2022 / Published online: 21 June 2022


## Abstract

Gene expression data clustering groups genes with similar patterns into a group, while genes exhibit dissimilar patterns into different groups. Traditional partitional gene expression data clustering partitions the entire set of genes into a finite set of clusters which might not reflect co-expression or coherent patterns across all genes belonging to a cluster. In this paper, we propose a graph-theoretic clustering algorithm called GAClust which groups co-expressed genes into the same cluster while also detecting noise genes. Clustering of genes is based on the presumption that co-expressed genes are more likely to share common biological functions. However, it has been observed that the clusters produced by traditional methods often do not reflect true biological groups or functions. To address this issue, we propose a semi-supervised algorithm, SGAClust to produce more biologically relevant clusters. We consider both synthetic and cancer gene expression datasets to evaluate the performance of the proposed algorithms. It has been found that SGAClust outperforms the unsupervised algorithms. Additionally, we also identify potential gene biomarkers which will further help in cancer management.

**Keywords** Semi-supervised clustering · Gene expression data · Biomarkers · Cancer disease

## 1 Introduction

The central dogma is at the heart of molecular biology which represents the flow of information from DNA through RNA and finally into proteins. This is known as gene expression. Using modern high-throughput technologies, gene expression profiling quantifies the count of mRNA transcripts that in turn calculate the number of corresponding proteins at the transcription level. This profiling can reflect thousands of gene expressions simultaneously for the deep understanding of cellular function. To know the hidden information of gene expression data, one of the data mining methods, clustering takes active participation.

Clustering is proven to be a powerful exploratory technique to understand the functional relationship of genes in a biological process, sub-type of cells, and biological pathways. Cluster analysis helps researchers to formulate a new hypothesis to detect the relationship between genes and is effectively used to predict the function of unknown genes based on the genes of known functions with which it is co-expressed Mitra and Banka (2006). In other words, it is based on the assumption that similar expression patterns may exhibit a strong correlation with their functions in the biological activities Liu et al. (2004). It helps to identify the genes which shares the fundamental patterns like *co-expressed*, *co-regulated*, and *coherent patterns*Kerr et al. (2008); Jiang et al. (2004). Two genes are said to be co-expressed if they share a similar pattern. Co-expressed genes provide functionally enriched genes. This may also indicate co-regulation if it has a strong expression pattern Jiang et al. (2003). Co-regulated genes are those genes that are regulated (up or down) by some common transcription factor (a protein found in transcribing process). The characteristics of a coherent pattern are to share a common trend of a co-expressed cluster. The trend commonly means the mean (centroid) of the co-expressed group. Cluster analysis helps the researcher to understand the functions in a biological system, biological phenomena.

Clustering algorithms are unsupervised by nature, i.e., no prior knowledge is required for discovering interesting

Fully documented templates are available in the elsarticle package on CTAN.

✉ Koyel Mandal
koyel@tezu.ernet.in

[1] Department of Computer Science and Engineering, Tezpur University, Tezpur, India

patterns. There are various unsupervised clustering algorithms targeted to analyze gene expression data. In practice, K-means MacQueen (1967), SOM Tamayo et al. (1999), and Hierarchical Clustering (HC) Eisen et al. (1998) are widely applied in the context of gene expression data clustering. However, these approaches attempt to group all input genes into some sort of finite cluster. Thus, genes that are not co-expressed are also assigned to their "best-fitting" cluster, and as a result, co-expressed and non-co-expressed genes come under the same cluster Abu-Jamous and Kelly (2018). This outcome violates the basic property of biological clusters that no two clusters should have identical expression profiles; rather, it should form a cluster only with co-expressed genes.

Due to wet-lab experiments, often gene expression data are noisy; therefore, partial clustering is more suitable and appreciable in such cases. In partial clustering, full-space algorithm will not allow some of the genes (noise) to be present in a well-defined cluster that impacts the quality of the cluster. Additionally, clustering algorithm should be designed in an automated framework, such that algorithm either is free from parameters or is calculated dynamically. Here, we use a graph-theoretic approach to find potential solutions for discovering clusters from noisy data, which does not require the number of clusters explicitly. To address all these issues, we propose an algorithm, named **G**raph **A**ttraction **Clust**ering (GAClust) algorithm which shares some features (clique finding) of an existing graph-theoretic approach, CAST Ben-Dor et al. (1999). The clustering result obtained by CAST highly depends upon the fine-tuning of the threshold value. Our proposed method dynamically estimates the parameters based on the dataset used. Moreover, the graph construction method is accomplished, by focusing only on groups of genes or common-neighborhood concept (grounded on proximity measure) rather than absolute measure between two genes.

Most of the researchers tend to use Euclidean distance or Pearson correlation in the traditional clustering process. Even though clustering analysis is an exploratory technique for determining the relationship in gene expression data Pirim et al. (2012), still it does not give the biologically meaningful correlation between genetic co-regulation and affiliation to a common biological process Adryan and Schuh (2004). It is noteworthy that external domain knowledge is the necessary pillar to ensure the relevancy of the discovered clusters. The majority of the existing algorithms ignore external knowledge to get more biologically relevant clusters. Moreover, using only expression values do not give the biological relationships in clusters. Therefore, sufficient attention is given to incorporating the biological knowledge during the search process to ensure that co-expressed gene is highly relevant biologically Nepomuceno et al. (2015). While integrating biological knowledge into gene expression matrix during clustering, it no longer seems to be an unsupervised approach and turns into a semi-supervised clustering. Bryan Bryan (2004) has identified some limitations of gene expression data clustering and pointed out that the lack of natural gene clusters can be overcome using semi- or supervised learning. With this belief, we have been motivated to develop clustering algorithms with biological knowledge in this thesis. Next, we modify GAClust into Semi-supervised GAClust (SGAClust) which holds all the properties of GAClust. The key features of the SGAClust algorithm are as follows: it (i) handles noise efficiently and (ii) discovers clusters automatically. The algorithm gives a nice guideline to estimate parameters that vary from dataset to dataset.

Interestingly, during the last few years, knowledge-driven approaches have been gaining popularity, because statistically significant and homogeneity solutions may not be biologically relevant Henriques and Madeira (2016). Gene Ontology (GO) plays a pivotal role in capturing the relationship among genes and hence give an added advantage if it is incorporated into the clustering process as GO contains the biological classifications of all known genes. This motivates us to investigate external information from GO in this particular domain.

One of the leading causes of death is cancer and has become a serious life-threatening disease for human beings. According to the statistical report, the total number of new cases of cancer has risen to 19.3 million globally with 10.0 million deaths in 2020 Bray et al. (2018). In India, new cases of cancer have been estimated to be 13.9 lakh and it is expected to reach up to 15.7 lakh by 2023[1] Worldwide, the number of breast cancer in women is escalating. Among men cancer, lung cancer is the most frequently occurring cancer, prostate cancer being the second, and colorectum cancer is the third most familiar type of cancer. Patients suffering from advanced stages of cancer result in poor prognosis and also high recurrence rate Lin et al. (2017). Despite having therapeutic advancement of pharmacogenomics and medicine, early cancer detection for increasing the patient's survival rate is still a challenging task. Therefore, gene expression data are used to study the transcriptome of cancer for detecting novel transcripts and alternative splicing with higher accuracy Lin et al. (2017). Gene expression data also help to identify diseased genes by varying the expression value under standard and diseased conditions Hussain and Ramazan (2016). In this study, we target to identify cancer-related genes as potential biomarkers.

According to the definition given by World Health Organization, "A biomarker is any substance, structure, or process that can be measured in the body or its products and

[1] https://www.ndtv.com.

influence or predict the incidence of outcome or disease". Generally, a biomarker can distinguish between healthy and diseased persons. A large variety of biomarkers include proteins, genes, and miRNAs. Clinically, a cancer biomarker may measure the risk factors growing in the specific cell, the possible response over several treatments or cancer progression Goossens et al. (2015). The molecular characterization of gene expression data is considered to have a great role in the early diagnosis of cancer by discovering prognostic biomarkers Zhou and Dickerson (2014); Samee et al. (2012). Identification of cancer risk groups will facilitate better treatment and increase the survival rate of a patient's lifetime. It helps in determining the risk of developing cancer. For instance, a woman having a strong family background of having ovarian cancer can go for genetic testing for the purpose of knowing if she is a carrier for mutation of BRCA1 which may increase the risk factor of cancer Henry and Hayes (2012). Reliable biomarkers are extremely beneficial in understanding the complexity of various diseases Sachnev et al. (2015), reduction of cost, simplifying the experimental setup, and providing a reference to the actual wet laboratory experimental results Martinez-Ledesma et al. (2015).

The identification of cancer-related biomarkers will boost cancer management successfully and help diagnostics in a better way. Though several biomarkers have been identified which are used for diagnosis, still there is a need for improving the process of identifying new biomarkers. Many strategies have made it possible to discover biomarkers and selecting a proper method is a very challenging task Mohammed et al. (2017). In addition to conventional methods Martinez-Ledesma et al. (2015); Li et al. (2013); Chen et al. (2010); Joe and Nam (2016), some frequently used strategies dealing with biomarkers or causal gene identification for cancer include differential expression analysis Stratford et al. (2010); Tusher et al. (2001), network analysis Kulshrestha et al. (2016), co-expression network analysis Liu et al. (2016), top gene ranking Li et al. (2013), statistical analysis Kim et al. (2011), and classification Erbes et al. (2015). The study outlines the biomarker identification method utilizing clustering methods.

## 2 Related work

Cluster analysis can be done in four major steps: (i) data preprocessing, (ii) selection of appropriate proximity measure ((dis)similarity)) Jaskowiak et al. (2013, 2014), (iii) applying clustering algorithm, and finally (iv) evaluation of clustering result. We broadly classify the clustering of gene expression data into two parts, viz., (i) unsupervised and (ii) semi-supervised clustering algorithms. In the literature, there are a rich variety of clustering algorithms for gene expression data Jiang et al. (2004); Kerr et al. (2008); Pirim et al. (2012). Clustering can be broadly classified into partitional, hierarchical, graph-theoretic, and density-based clustering Jiang et al. (2004); Kerr et al. (2008); Pirim et al. (2012); Oyelade et al. (2016).

The most fundamental clustering algorithm is partitional-based clustering algorithm. This type of algorithm partitions the Dataset $ED$ containing $m$ genes into $K$ (used defined) clusters $(K \leq m)\,\{\mathcal{C}_1, \mathcal{C}_2, ..., \mathcal{C}_K\}$, where each datum is residing in only one cluster $\mathcal{C}_i \cap \mathcal{C}_j = \phi$. The simplest partitional-based algorithm is K-means MacQueen (1967) which is widely used in gene expression data. Though the K-means algorithm is very fast and easy to implement, it suffers from several drawbacks. The algorithm suffers from a predefined input parameter determination problem and is unable to find the arbitrary shaped clusters. Another drawback of the K-means algorithm is non-robustness, which is very necessary for analyzing noisy gene expression datasets. There are several variations of the K-means algorithm using soft computing techniques Lu et al. (2004); Lam et al. (2013); Wu (2008); Sheng et al. (2010). Self Organizing Map (SOM) Tamayo et al. (1999) based on an unsupervised artificial neural network is more robust (cluster the huge amount of noisy data), reasonably fast, and easy to implement. Recently, Abu-Jamous and Kelly have proposed a partition-based method, named Clust Abu-Jamous and Kelly (2018). The aim of this algorithm is not to consider the whole set of input data to be partitioned into clusters; instead, it identifies subsets that are assigned to clusters.

Hierarchical clustering (HC) algorithm produces a group of nested clusters forming a tree-like structure called dendrogram rather than forming a set of disjoint clusters like a partitional algorithm. Unlike the partition-based clustering approach, we do not have to assume a fixed number of clusters; rather, we can get any number of clusters by cutting the dendrogram at a proper level. The variation of HC is of two types: agglomerative and divisive. Agglomerative is a bottom–up approach where each data object is considered to be a single cluster. Two data objects are merged based on single linkage, average linkage, centroid linkage, or complete linkage. The process continues until all data points are merged into a single cluster. Whereas the divisive approach is just the opposite of the former one, it is a top–down approach. It starts with all data objects as a single cluster, data points are split to meet some heuristic criteria, until singleton clusters remain. HC finds a similar pattern and displays the result graphically which is easy to interpret for biologists. The problem associated with the hierarchical algorithm is its high computational complexity; splitting or merging of each step takes $\frac{m^2-m}{2}$ times. The total time complexity of the agglomerative clustering algorithm is $O(m^2\ log\ m)$. HC is a greedy approach, where once a decision has been taken, it can never be changed. Another drawback of HC is the lack of robustness. Some of the clustering algorithms that follow

the HC approach are Unweighted Pair Grouping Method (UPGMA) Eisen et al. (1998), Deterministic-Annealing Algorithm (DAA) Rose (1998), and Self Organizing Tree Algorithm (SOTA) Herrero et al. (2001). SOTA is an unsupervised neural network that grows by adopting binary tree topology. As SOTA combines the good features from both neural networks of SOM and hierarchical clustering, that is why, it can easily overcome the problem associated with the classical hierarchical approach.

The key goal of graph-theoretic approach is to partition the data into subgraphs with the help of some geometric property. From the given dataset, we can make a proximity matrix and a weighted graph or proximity graph $G(V, E)$ from the matrix. The nodes $V$ of the graph are genes and edges $E$ are the connection between two genes. The weighting scheme differs from algorithm to algorithm. This approach can easily handle the outliers and does not depend upon the parameter which determines the number of clusters. Ben-Dor et al. (1999) have proposed a graph-theoretic algorithm for gene expression data Cluster Affinity Search Technique (CAST) which introduces an idea of corrupted clique graph model. CAST algorithm has two drawbacks: i) the user-defined affinity threshold value, and ii) the cleaning step is required to define the position of the data points among all the clusters. Bellaachia et al. (2002) overcomes the problem of threshold calculations CAST by proposing Enhanced-CAST (ECAST) which dynamically calculates the threshold value at the starting of every new cluster. CLuster Identification via Connectivity Kernels (CLICK) Sharan and Shamir (2000) does not depend upon the number of clusters and discovers "true" clusters via graph-theoretic method and statistical techniques.

Conventional clustering algorithms find sets of genes depending upon their proximity ((dis)similarity) measure. In contrast, expression-based measures may not find the potential relationships among the genes as these measures are unable to capture the potential functional relationships among genes. Therefore, it is important to adopt ontologies for annotations while comparing entities. Semantic similarity (SS) allows the comparison of GO terms or GO annotated gene products by leveraging the hierarchical structure of the GO graph. SS calculates the closeness between them which in turn reflects numerical value. SS measure is the key technique to incorporate the knowledge of known genes from gene ontology and gene annotation files. A wide variety of SS measures can be found in Pesquita et al. (2009); Pesquita (2017).

According to the Gene Ontology Consortium, 2000, the GO database shows the hierarchical structure of gene annotations reflecting the association among genes and biological terms. GO provides the controlled vocabulary of about 30,000 terms for the three distinct domains *Biological Process* (BP), *Cellular Component* (CC), and *Molecular Function* (MF) to represent the gene properties, gene functionalities, or gene itself. Ontology is actually some set of terms with different hierarchical relationships or parent–child relationships (*is_a*, *part_of*) which is functioning in the previously mentioned domain. Figure 1 depicts the relation in GO Binns et al. (2009). GO is represented as a rooted *Directed Acyclic Graph* (DAG) where each node is represented by a GO term and the edge represents the relationship between the nodes. This graph forms as a hierarchy in such a way that one GO term is related to other GO terms, but the child node may have more than one parent. The knowledge represented in the GO hierarchy may be used to guide the unsupervised clustering process into a semi-supervised clustering which will give functionally enriched clusters.

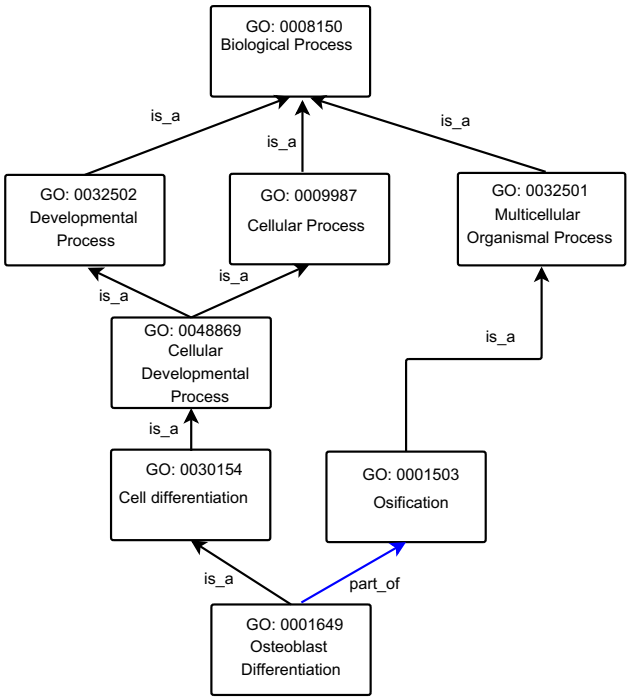



**Fig. 1** The relation in a GO

At first, Lord et al. (2003) have successfully applied SS in biology. Since then, several SS measures have been developed. We present a short survey of SS in the context of GO. To compare GO terms, there are two major approaches: edge-based and node-based. Edge-based approaches are dependent on the number of edges present in between GO terms. *Distance* (average of all paths or shortest path) and *common path* (the lowest common ancestor of two terms to root) are two popularly used techniques to calculate SS. On the other hand, node-based approaches rely on the comparison of the properties of the terms, their ascendants, or their descendants. These semantic similarities are built on the information theory which means how much information

they commonly share. Information content of a term $\mathbb{T}$ is quantified as IC in a specific corpus and is described by negative log-likelihood $IC = -log(P(\mathbb{T}))$, where $P(\mathbb{T})$ represents the probability of occurrence of $\mathbb{T}$ in a specific corpus. Another way to determine IC is to calculate the number of children in GO which is not used commonly. To determine the SS between two terms that is how much information they share, IC can be applied to the common ancestors of both terms. To do this, two main approaches are used: The Most Informative Common Ancestor (MICA) and Disjoint Common Ancestor (DCA). MICA means common ancestor having highest IC Resnik (1995) and DCA represents all common ancestors that do not subsume any other common ancestor Couto et al. (2005). Alternatively, node-based approaches can also be calculated by the number of shared annotations, number of gene-annotated products, number of shared ancestors, node depth, and node-link density, etc. While comparing gene products, often, it can be done pairwise or groups. To quantify the pairwise similarity between two gene products, SS between their terms are combined. In this regard, often maximum, sum, and average are used for combining. Groupwise approaches are directly calculated by set, graph, or vectors which is different from the former one.

Here, we report some of the well-known semantic similarities. Resnik similarity between two terms $\mathbb{T}_i$ and $\mathbb{T}_j$ is calculated by Eq. 1, which is simply IC of their MICA. The lower bound of Resnik measure is 0 and it has no upper limit

$$SS_{Res}(\mathbb{T}_i, \mathbb{T}_j) = IC(MICA). \tag{1}$$

Resnik measure does not consider the distance from both the terms to their lowest common ancestors. Hence, distance is taken into consideration in Lin's, and Jiang and Conrath's. Lin Lin (1998) similarity between two terms say $\mathbb{T}_i$ and $\mathbb{T}_j$ and given by Eq. 2. $SS_{Lin}$ gives the IC between two terms by considering the IC of each term and the IC of MICA. The obtained value of semantic similarity lies between 0 and 1

$$SS_{Lin}(\mathbb{T}_i, \mathbb{T}_j) = \frac{2 \times IC(MICA)}{IC(\mathbb{T}_i) + IC(\mathbb{T}_j)}. \tag{2}$$

Jiang and Cornath's Jiang and Conrath (1997) have proposed an IC-based measure as shown in Eq. 3. The lowest and highest value of this measure is 0 and 1, respectively

$$SS_{JCSS} = 1 - IC(\mathbb{T}_i) + IC(\mathbb{T}_j) - 2 \times IC(MICA). \tag{3}$$

These three node-based measures determine the similarity between two GO terms, which in turn can be extended for comparison of gene products, which have several GO terms. Wang et al. (2007) have proposed an SS as a pairwise measure that is applied as edge-based. Let, a GO term $\mathbb{T}_i$ can be defined by a graph $G_{\mathbb{T}_i} = (\mathbb{T}_i, A_{\mathbb{T}_i}, E_{\mathbb{T}_i})$, where $A_{\mathbb{T}_i}$ is a set of GO terms in $G_{\mathbb{T}_i}$ including $\mathbb{T}_i$ and all ancestors of the term $\mathbb{T}_i$ and $E_{\mathbb{T}_i}$ is the set of edges or semantic relations. To do a quantitative comparison in between two GO terms, GO term is encoded by $\mathbb{T}_i$ as the aggregated contribution of all terms in $G_{\mathbb{T}_i}$. Therefore, S value is used to define the contribution of GO terms $\mathbb{T}_i$. For any term $\mathbb{T}$ in $G_{\mathbb{T}_i}$, the S value of $\mathbb{T}_i$ is represented by Eq. 4

$$\begin{aligned} &S_{\mathbb{T}_i}(\mathbb{T}_i) = 1 \\ &S_{\mathbb{T}_i}(\mathbb{T}) = \max\{w_e \times S_{\mathbb{T}_i}(\mathbb{T}') | \mathbb{T}' \in \text{children of}\,(\mathbb{T}) \text{ if } \mathbb{T} \neq \mathbb{T}_i\}, \end{aligned} \tag{4}$$

where $w_e$ is the contribution factor of the edge between $\mathbb{T}_i$ and its children $\mathbb{T}'$ and $0 < w_e < 1$. After calculating the S values for all the terms present in $G_{\mathbb{T}_i}$, semantic value $SV(\mathbb{T}_i)$ is obtained by Eq. 5

$$SV(\mathbb{T}_i) = \sum_{\mathbb{T} \in A_{\mathbb{T}_i}} S_{\mathbb{T}_i}(\mathbb{T}). \tag{5}$$

Considering the GO hierarchy, $w_e$ for *is*_a is 0.8 and *part*_of is 0.6. Given two graphs say, $G_{\mathbb{T}_i}$ and $G_{\mathbb{T}_j}$ for two GO terms $\mathbb{T}_i$ and $\mathbb{T}_j$, semantic similarity between two terms can be represented by Eq. 6

$$S(\mathbb{T}_i, \mathbb{T}_j) = \frac{\sum_{\mathbb{T} \in A_{\mathbb{T}_i} \cap A_{\mathbb{T}_j}} (S_{\mathbb{T}_i}(\mathbb{T}) + S_{\mathbb{T}_j}(\mathbb{T}))}{SV(\mathbb{T}_i) + SV(\mathbb{T}_j)}. \tag{6}$$

Nowadays, knowledge-based clustering algorithms have become an integral part of the research. However, the number of semi-supervised full-space clustering algorithms is much lesser than the number of unsupervised full-space clustering algorithms. Next, we present a brief survey on semi-supervised algorithms. Adryan and Schuh (2004) have developed a GO-Cluster program that incorporates the hierarchy structure of the GO database as a model for cluster analysis and also gives the visualization of gene expression data at any level of the gene ontology tree. Huang and Pan (2006) have included the gene function in distance metric and showed the advantage of using it over K-medoids (partitional) and hierarchical algorithms. In Ovaska et al. (2008), a fast gene ontology-based clustering has been built which demonstrates hierarchical clustering and a heat map visualization with the help of gene expression data and GO annotations. It helps to identify rapidly the biologically related genes. Verbanck et al. (2013) have incorporated external biological knowledge (GO) to measure the distance between genes and applied it to the K-means algorithm, which gives biologically significant homogeneous co-expressed clusters. Speer et al. (2004); Srivastava et al. (2008); Macintyre et al. (2010); Mitra and Ghosh (2012) have incorporated the GO in clustering process for gene expression data. Hang et al. (2009) have proposed an algorithm using two information such as gene density function and biological knowledge

and the proposed one gave a better result than the standard algorithm.

Zhou et al. (2010) also have developed an algorithm incorporating density of data and gene ontology in the distance-based clustering algorithm. Both the algorithms do not address the issue of identifying the positive and negative co-regulated genes. An algorithm that finds clusters comprised of co-regulated genes is being proposed by Ji and Tan (2004). To identify interesting partial negative–positive co-regulated gene clusters, Xu et al. (2006) have proposed an algorithm that also discovers overlapping clusters. We have also proposed a semi-supervised density-based clustering (SDC) Mandal and Sarmah (2018) incorporating GO information which can effectively identify both positively and negatively co-expressed genes.

## 3 Proposed methods

In this section, we describe a simple heuristic clustering algorithm in detail. Let us denote the gene expression data by a two-dimensional matrix, $ED_{m\times n}$ organized in terms of $m$ rows, $G = \{g_1, g_2, \ldots, g_m\}$ and $n$ columns, $C = \{c_1, c_2, \ldots, c_n\}$. Rows represent genes and columns denote experimental conditions or samples. Each entry $ge_{ij} \in ED_{m\times n}$ of the matrix corresponds to the value of a gene $g_i$ under a specific condition $c_j$, where $i = \{1, 2, \ldots, m\}$ and $j = \{1, 2, \ldots, n\}$.

Given neighborhood distance threshold $\Upsilon$ (user-specified parameter), GAClust proceeds in three steps, producing $K$ number of clusters $\{\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_K\}$ from input gene expression data $ED_{m\times n}$. The number of clusters and their size is highly influenced by the parameter $\Upsilon$. GAClust is a graph-theoretic clustering algorithm, based on the clique graph and divisive approach. The divisive approach follows a top–down analysis. It initiates with a large cluster and gradually splits into small clusters until each cluster contains a single piece of data. The fundamental assumption of this model is a true biological partition of genes which relies on certain functionality of the genes.

The similarity between the expression patterns can be represented by a similarity matrix $Sim$, where $Sim_{x,y}$ denotes the similarity in between gene $g_x$ and $g_y$. This can be easily computed by proximity measure (similarity or dissimilarity). Furthermore, the similarity matrix can be represented by a weighted graph $\mathcal{G}^*(V, E)$, where vertices $V$ denote genes and $E$ represents the edges (similar expression pattern) between two genes. The weight of the similarity graph is defined by the similarity between two genes. A graph is said to be a clique graph if it consists of a disjoint complete graph Bellaachia et al. (2002). If two genes are similar, then there exists an edge else; no edge is present between them. In this context, the clique graph is composed of clusters of genes where the similarity of each gene within the clique is higher than the genes belonging to other cliques. A clique graph $\mathcal{H}$ is formed by genes (vertices) $G = \{g_1, g_2, \ldots, g_m\}$, such that each clique $cq_i \in \mathcal{H}$ contains an edge between every two genes $g_i, g_p \in cq_i$. Additionally, there are no edges between genes $g_i$ and $g_k$ where $g_i \in cq_i$ and $g_k \in G \setminus cq_i$. Mathematically, a clique $\mathcal{H}$ graph for a given graph $\mathcal{G}^*(V, E)$ is defined in such a way that (i) each vertex of $\mathcal{H}$ presents a maximal clique of $\mathcal{G}^*$ and (ii) two two distinct vertices of $\mathcal{H}$ are adjacent. Gene expression data are noisy; hence, an ideal clique graph is never possible. In expression data, contamination errors are introduced resulting in similarity graph $C(\mathcal{H})$ which is not a clique graph. Therefore, the clustering problem can be modelled as restoring clique graph $\mathcal{H}$ using edge modification problem from corrupted clique graph where the error is introduced. The implementation of GAClust is described next stepwise and the algorithm is shown in Algorithm 1.

***Graph construction:*** We compute an $m \times m$, $R_{m\times m}$ (similarity matrix) matrix from expression data to construct a graph. The edge between two genes $g_x$ and $g_y$ has been given a weight using Eq. 7 defined as similarity $R$ where $\mathcal{N}(g_x)$ is neighbors of gene $g_x$ and $\mathcal{CN}(g_x, g_y)$ is common neighborhood of $g_x$ and $g_y$. The $R$ is stated as the similarity between two genes

$$R(g_x, g_y) = \begin{cases} 1, & \text{if} \quad (g_x = g_y) \\ \frac{c}{a+b-c}, & \text{if} \quad (|\mathcal{CN}(g_x, g_y)| \neq 0) \\ 0, & \text{if} \quad (|\mathcal{CN}(g_x, g_y)| == 0), \end{cases} \tag{7}$$

where

$$a = |\mathcal{N}(g_x)| \tag{8}$$

$$b = |\mathcal{N}(g_y)| \tag{9}$$

$$c = |\mathcal{CN}(g_x, g_y)|. \tag{10}$$

**Definition 3.1** Neighborhood of a gene, $\mathcal{N}(g_x)$, is described by the genes $g_z$, residing within its user-defined radius $\Upsilon$

$$\mathcal{N}(g_x) = \{g_z | z \in G, Dist_{Euc}(g_x, g_z) \leq \Upsilon\}. \tag{11}$$

$\mathcal{N}(g_x)$ is defined in Eq. 11 where $Dist(g_x, g_z)$ is determined by Euclidean distance shown in Eq. 12

$$Dist_{Euc}(g_x, g_z) = \sqrt{\left(\sum_{j=1}^{n} (g_{xj}, g_{zj})^2\right)}. \tag{12}$$

**Definition 3.2** Common neighborhood between two genes $g_x$ and $g_y$ are the genes $\{g_1, g_2, \ldots, g_q\}$ which belong to the

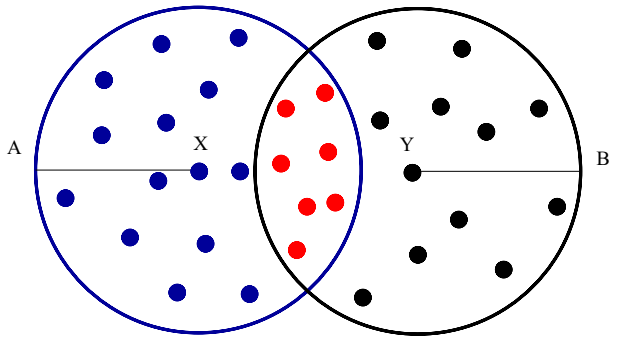



**Fig. 2** Schematic diagram of common neighborhood between two objects. Blue and black colored circles represent the neighborhood of $X$ and $Y$ objects, respectively, within its $\Upsilon$ distance. Red colored solid circles represent the common neighbor objects of both $X$ and $Y$ within $\Upsilon$ distance

neighborhood of both genes, $g_x$ and $g_y$ with respect to $\Upsilon$ and are given by Eq. 13

$$\mathcal{CN}(g_x, g_y) = \{g_k \in \mathcal{N}(g_x) \cap \mathcal{N}(g_y)\}, k = 1, 2, \ldots, q. \quad (13)$$

The concept of common neighborhood is shown in Fig. 2.

The similarity $R \in [0, 1]$ is symmetrical, i.e., $R(g_x, g_y) = R(g_y, g_x)$. The value lies between 0 and 1, $0 \leq R \leq 1$. 0 means genes are not connected, and 1 means the neighbors of $g_x$ is overlapped with neighbors of $g_y$. Higher $R(g_x, g_y)$, i.e, values closer to one, indicates that the two neighbors are closely connected.

***Node addition:*** The key step of GAClust is to compute the average attraction ($\mathcal{A}$) between unclustered data to its present cluster to make further decisions. Clusters are generated one at a time.

**Definition 3.3** The attraction ($\mathcal{A}$) of a gene $g_x$ with respect to a cluster $\mathcal{C}_{now}$ is the sum of similarity between $g_x$ and all genes in $\mathcal{C}_{now}$

$$\mathcal{A}(g_x) = \sum_{g_y \in \mathcal{C}_{now}} R(g_x, g_y). \quad (14)$$

We initiate the current cluster by denoting $\mathcal{C}_{now} = \phi$. Clusters are formed by adding high connectivity genes one at a time to $\mathcal{C}_{now}$ until no changes have been found.

**Definition 3.4** A gene $g_x$ is said to have high connectivity to be included in the current cluster $\mathcal{C}_{now}$ if it satisfies the condition $\mathcal{A}(g_x) \geq \eta|\mathcal{C}_{now}|$ where $\eta$ is an attraction threshold, where as a gene $g_x$ is said to be low connectivity if it satisfies the following condition: $\mathcal{A}(g_x) < \eta|\mathcal{C}_{now}|$.

The crucial task of the algorithm is parameter estimation of $\eta$ and $\Upsilon$. Unlike CAST Ben-Dor et al. (1999), GAClust calculates threshold $\eta$ dynamically with the help of Eqs. 15 and 16 where $deg(g_x)$ indicates the degree of a vertex $V$ or gene $g_x$ and $R(g_x, g_y)$ must be greater than 0.5

$$deg(g_x) = \begin{cases} \sum_{y=1}^{m} 1, & \text{if } R(g_x, g_y) \geq 0.5 \; and \; x \neq y \\ 0, & \text{otherwise} \end{cases} \quad (15)$$

$$\eta = 0.5 \times \frac{\sum_{x=1}^{m} \sum_{y=1, x \neq y}^{m} R(g_x, g_y)}{\sum_{i=1}^{m} deg(g_i)}. \quad (16)$$

***Node deletion:*** After the addition step, low connectivity genes are removed from the current cluster $\mathcal{C}_{now}$. We keep on removing genes from $\mathcal{C}_{now}$ until it gets stabilized to form a single cluster.

Repeating the node addition and removal steps further, we get $K$ number of clusters. All singleton clusters are considered as noise.

We have chosen the neighborhood distance $\Upsilon$ of a gene as sufficiently large (+0.5) from the graph of sorted K-Nearest Neighbor (KNN) distance from each gene. This $K$ value is determined by taking the square root of the total number of genes present in an input dataset.

**Algorithm 1:** GAClust algorithm

**Input :** $ED_{m \times n}$ with a set of genes $G = \{g_1, g_2, \ldots, g_m\}$ and a set of samples $C = \{c_1, c_2, \ldots, c_n\}$, $\Upsilon$, $\eta$

**Output:** $\mathcal{C} = \{\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_K\}$

```
C = φ
Compute R matrix with the help of Υ using Equation 7
while (G ≠ φ) do
    C_now = φ, A(G) = 0
    Select g_x ∈ G such that R(g_x, g_y) = max{R(g_w, g_y)|g_w, g_y ∈ G}
    C_now = C_now ∪ g_x
    G = G \ g_x
    ∀g_y ∈ G, A(g_y) = A(g_y) + R(g_y, g_x)
    while (Changes in C_now) do
        while (max{A(g_z)|g_z ∈ G} ≥ η|C_now|) do
            Select g_a ∈ G with maximum attraction such that
              A(g_a) = max{A(g_w)|g_w ∈ G}
            C_now = C_now ∪ {g_a}
            G = G \ {g_a}
            ∀g_b ∈ G ∪ C_now, A(g_b) = A(g_b) + R(g_b, g_a)
        end
        while (min{A(g_z)|g_z ∈ C_now} < η|C_now|) do
            Select g_a ∈ G with minimum attraction such that
              A(g_a) = min{A(g_w)|g_w ∈ G}
            C_now = C_now \ {g_a}
            G = G ∪ {g_a}
            ∀g_b ∈ G ∪ C_now, A(g_b) = A(g_b) − R(g_b, g_a)
        end
    end
    C = C ∪ C_now
end
```

SGAClust is the extended version of the GAClust algorithm. SGAClust takes four input parameters neighborhood similarity threshold $\Upsilon'$, attraction threshold $\eta'$, $w_1$, and $w_2$. Here, we incorporate combined similarity (*Com_sim*) as shown in Eq. 17 instead of only an expression-based distance measure to find the neighborhood of a gene

$$Com\_sim = w1 * Sim + w2 * SS. \tag{17}$$

We combine similarity measures (*Sim*) and semantic similarity (*SS*) to improve clustering results. where, $w1 + w2 = 1$ and $0 \leq w2 \leq 1$ Lee and Lin (2016). Weight parameters $w1$ and $w2$ control the weights to two similarity measures. Most commonly used proximity measure is Euclidean distance which gives the dissimilarity between gene $g_i$, $g_j$ as Eq. 12Jiang et al. (2004). We first convert $Dist_{Euc}$ into a similarity measure as, $Sim = \frac{1}{1+Dist_{Euc}}$. For SS, Wang's measure is taken under consideration. Wang's measure distinguishes the two relations (is_a and part_of) in GO hierarchy structure, whereas Lin's measure does not differentiate between both the relations. We redefine the definition of neighborhood of a gene.

**Definition 3.5** Neighborhood of a gene $\mathcal{N}(g_i)$ is described by the genes $g_x$, residing within its user-defined radius $\Upsilon'$

$$\mathcal{N}(g_i) = \{g_x | x \in G, Com\_sim(g_i, g_x) \geq \Upsilon'\}. \tag{18}$$

SGAClust algorithm considers $\Upsilon'$ and $\eta'$ instead of $\Upsilon$ and $\eta$ in GAClust. The parameters are calculated by Eqs. 19 and 20, where $\Upsilon$ and $\eta$ are estimated as mentioned in GAClust algorithm

$$\Upsilon' = \frac{1}{1+\Upsilon} \tag{19}$$

$$\eta' = \frac{1}{1+\eta}. \tag{20}$$

The main algorithmic approach is similar to GAClust. It consists of three major steps, i.e., graph construction, node addition, and node deletion. We compute $R_{m\times m}$ matrix and then construct a graph similar to the GAClust algorithm except for finding neighborhood of genes. Each cluster is generated by adding high connectivity genes and removing low connectivity genes from the cluster.

## 4 Time complexity

GAClust algorithm takes $O(m^2)$ operations to compute $R_{m\times m}$ matrix. Node addition and node deletion take much much lesser time than to construct $R_{m\times m}$. Therefore, the overall running time of GAClust algorithm is $O(m^2)$. For SGAClust, we are computing the similarity as well as semantic similarity. Hence, the running time of SGAClust algorithms is $O(2 \times m^2)$. With this, we have also shown the running time of CAST and GAClust in seconds, which is reported in Table 1.

**Table 1** Running time (in seconds) of CAST and GAClust on real datasets

| Dataset | CAST | GAClust |
|---|---|---|
| Armstrong-v2 | 2.939 | 1566.3558 |
| Bhattacharjee | 2.965 | 723.1718 |
| Laiho | 1.574 | 1403.6556 |
| Ramaswamy | 1.886 | 906.0788 |
| Singh | 0.108 | 40.4585 |

## 5 Experimental results and discussion

To provide a comparison of proposed algorithms, we select a suite of clustering algorithms K-means, HC, CAST Ben-Dor et al. (1999), SOTA, and CLICK Sharan and Shamir (2000) which are applied on synthetic data as well as real gene expression datasets. The performances of algorithms are established by the means of internal criteria and biological assessment. Internal measure is a pure indication of how many groups are present in a dataset, i.e., how well the partition solution is produced by a clustering algorithm which captures the separation of data among different clusters. It is useful when we do not know the true clustering solutions. Each clustering result produced by different algorithms on several datasets is assessed with four commonly used cluster validation indices to judge the quality of clusters.

To generate clusters by CAST and SOTA, we have used the MultiExperiment Viewer (MeV) available at http://mev.tm4.org/ algorithms with default parameter settings. CLICK algorithm is executed as a part of Expander software version 7.0 (http://acgt.cs.tau.ac.il/expander/) with default homogeneity value as mentioned in the software. K-means and HC average linkage are executed in MATLAB. Our methods are also implemented in MATLAB environment.

### 5.1 Clustering performance metrices

In this study, we adopt commonly used cluster validation indices to judge the quality of clusters. They are Mean Squared Error (MSE) Abu-Jamous and Kelly (2018), Davies–Bouldin (DB) Davies and Bouldin (1979), Ball and Hall index (BH) Ball and Hall (1965), and C index (CI) Hubert and Schultz (1976). Lower MSE, DB, BH, and CI values suggest the good clustering result.

Within-cluster dispersion is measured by MSE Abu-Jamous and Kelly (2018). Let us consider the clusters of

any algorithm is $\mathcal{C} = \{\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_K\}$. If a cluster $\mathcal{C}_i$ has $P$ number of genes and $N$ number columns of each gene, then MSE of a cluster is calculated using Eq. 21

$$MSE(\mathcal{C}_i) = \frac{1}{N \times P} \sum_{p=1}^{P} ||\overrightarrow{g_p} - \vec{z}||; \tag{21}$$

$\overrightarrow{g_p}$ is the expression profile of $p^{th}$ gene in this cluster, $\vec{z}$ is the average expression profiles of all genes belonging to that cluster, and $||\overrightarrow{g_i} - \vec{z}||$ is the euclidean distance between these two vectors. We compute MSE for each cluster and take the average to report the value.

The DB index Davies and Bouldin (1979); Desgraupes (2013) is the mean value of all clusters is defined as in Eq. 22 where $\delta_i$ is intracluster distance, whereas $\Delta_{ij}$ is the intercluster distance between clusters $\mathcal{C}_i$ and $\mathcal{C}_j$

$$DB(\mathcal{C}) = \frac{1}{K} \sum_{i=1}^{K} max_{j \neq i} \left( \frac{\delta_i + \delta_j}{\Delta_{ij}} \right). \tag{22}$$

The intracluster distance is computed by the data which belongs to a cluster $\mathcal{C}_j$ to their barycenter $\mathcal{O}^{\{j\}}$ which is a row vector as given below

$$\delta_j = \frac{1}{|\mathcal{C}_j|} \sum_{a \in I_j} ||M_a^{\{j\}} - \mathcal{O}^{\{j\}}||, \tag{23}$$

where cluster $\mathcal{C}_j$ can be represented by a submatrix $M^{\{j\}}$ and $I_j$ is set of indices of all genes present in this cluster. The submatrix $M^{\{j\}}$ can also be denoted as $M_{\{I_j\}}$. The intercluster distance $\Delta_{ij}$ is the distance between the centroids $\mathcal{O}_i$ and $\mathcal{O}_j$ of clusters $\mathcal{C}_i$ and $\mathcal{C}_j$ as presented in Eq. 24

$$\Delta_{ij} = d(\mathcal{O}^{\{i\}}, \mathcal{O}^{\{j\}}) = ||\mathcal{O}^{\{j\}} - \mathcal{O}^{\{i\}}||. \tag{24}$$

The BH index measures the mean dispersion of a cluster which formally means the squared distance of the data residing in a cluster to their centroid Ball and Hall (1965); Desgraupes (2013). The mathematical formula of BH for mean through all clusters is indicated by Eq. 25 where $n_a$ is the cardinality of a cluster

$$BH(\mathcal{C}) = \frac{1}{K} \sum_{k=1}^{K} \frac{1}{n_k} \sum_{b \in I_k} ||M_b^{\{k\}} - \mathcal{O}^{\{k\}}||^2. \tag{25}$$

Let us consider the total number of distinct pairs in the dataset be $N_T = \frac{N(N-1)}{2}$ where $N = \sum_{k=1}^{K} n_k$. In a cluster $\mathcal{C}_k$, the number of distinct pairs is $\frac{n_k(n_k-1)}{2}$. Therefore, the total number of such pairs is $N_W$ shown in Eq. 26

$$N_W = \sum_{k=1}^{K} \frac{n_k(n_k - 1)}{2}. \tag{26}$$

The CI is defined as mentioned in Eq. 27Hubert and Schultz (1976); Desgraupes (2013)

$$CI = \frac{S_W - S_{min}}{S_{max} - S_{min}}. \tag{27}$$

Inside a cluster, the sum of $N_W$ distances between all pairs of points is denoted by $S_W$. Among all $N_T$ pairs in the entire dataset, $S_{min}$ takes the sum of $N_W$ smallest distances, whereas $S_{max}$ takes the sum of $N_W$ largest distances.

## 5.2 Synthetic data generation

For a better explanation and to establish the effectiveness of GAClust, we first generate five synthetic datasets which can be visualized in Fig. 3. Each dataset is comprised of 400 genes where we implant four clusters of 100 genes in each of them. At first, we create a background matrix of size 400 rows and 10 columns from a normal distribution of the mean ($\mu$) 0 and standard deviation ($\sigma$) 1. We implant four different types of clusters where the first cluster has up-regulated patterns (Cluster 1), the second cluster has down-regulated patterns (Cluster 2), the third cluster has up-regulated and then down-regulated patterns (Cluster 3), and the fourth one has down-regulated and then up-regulated patterns (Cluster 4), as shown in Fig. 3. To create an up-regulated cluster, we randomly select one gene expression profile and sort the expression values in ascending order. Then, we replicate the same expression profile for randomly other 99 genes. Similarly, we create down-regulated patterns except for the expression values which are necessarily in descending order. For Cluster 3 first half of the expression, values are up-regulated for the first five columns and then down-regulated for the next five conditions and vice versa for cluster 4. Here, one point we need to keep in mind is that we create the clusters in a non-overlapping manner. Thus, we create a matrix say D1. To make the datasets more realistic, next, we add random noise from a normal distribution with $\mu$ 0 and varying $\sigma$ 0.25, 0.5, 0.75, and 1 with each of the cells of D1 to get matrices D2, D3, D4, and D5, respectively. Before applying clustering algorithms, we normalized the datasets by z scores.

Running K-means and HC, a user-specified number of clusters $K$ is required. Therefore, we use implanted true number of clusters as $K$ for synthetic datasets to obtain $K$ number of clusters. CLICK algorithm returns partitions leaving some data unclustered. We have considered those data as a single cluster to compute all internal validation indices Di Gesú et al. (2005). To determine the parameter $\Upsilon$ for each synthetic dataset, we plot the graph of sorted KNN distance from each gene in Fig. 4. The parameter $\Upsilon$ for GAClust is kept relatively large which is given in Table 2. The attraction threshold $\eta$ is computed dynamically according to Eqs. 15 and 16 except for dataset D1, because it has a replication

of data in four clusters. Hence, we consider $K = \sqrt{4} = 2$ of KNN for dataset D1. The $\eta$ is calculated using Eq. 16 where multiplication factor is 1 instead of 0.5, as well as $\Upsilon$, is increased with 1. It is important to mention that in this experiment, we have not considered semi-supervised algorithms for synthetic datasets as there is no GO information for synthetic data.

### 5.3 Performance on synthetic datasets

We now analyze the performance of GAClust with all other algorithms under consideration for internal measures using MSE, DB, BH, and CI. Figure 5 shows the histograms of four cluster validation indices for comparing the performance of six clustering algorithms on synthetic datasets. In the diagram, the x-axis denotes the clustering algorithms, while the y-axis denotes the metric values. Different colors are being used to different clustering algorithms. The graph demonstrates that GAClust can identify clusters in presence of a higher amount of noise. The internal metric increases with the increasing noise. Now, if we look closely at the figure, then it can be understood that GAClust outperforms all the algorithms (For Dataset D3 CAST performs better than GAClust) in terms of DB score where a lower metric signifies better performance. While comparing GAClust with all other methods, it performs similar to CAST and is sometimes inferior for some datasets based on MSE and BH values. On the other hand, GAClust performs slightly lesser than CAST, K-means, and HC for datasets D4 and D5 based on CI. For better understanding, we summarize the values of metrics for all clustering algorithms on all five datasets in Table 3. MSE and BH score gives a similar rank for all the algorithms. CAST performs best followed by GAClust, SOTA, K-means, and HC, while the CLICK algorithm performs the worst. From the Table, it is not too hard to recognize that CLICK holds the last position for DB, MSE, and BH, and the second last position followed by SOTA for CI score. K-means is the second-best algorithm and SOTA and CLICK both are not performing well according to CI and DB scores. As for all the measures, K-means and HC are quite close in many circumstances. Based on Table 3, it appears that K-means is slightly superior to HC. It appears that the CAST algorithm has very good predictive power and GAClust is competitive in comparison with CAST as well as other state-of-the-art methods. It is important to note that CI is greatly influenced by the fact of producing optimal index values for different number of clusters. Therefore, CI does not perform well to evaluate the clusters generated by different algorithms.

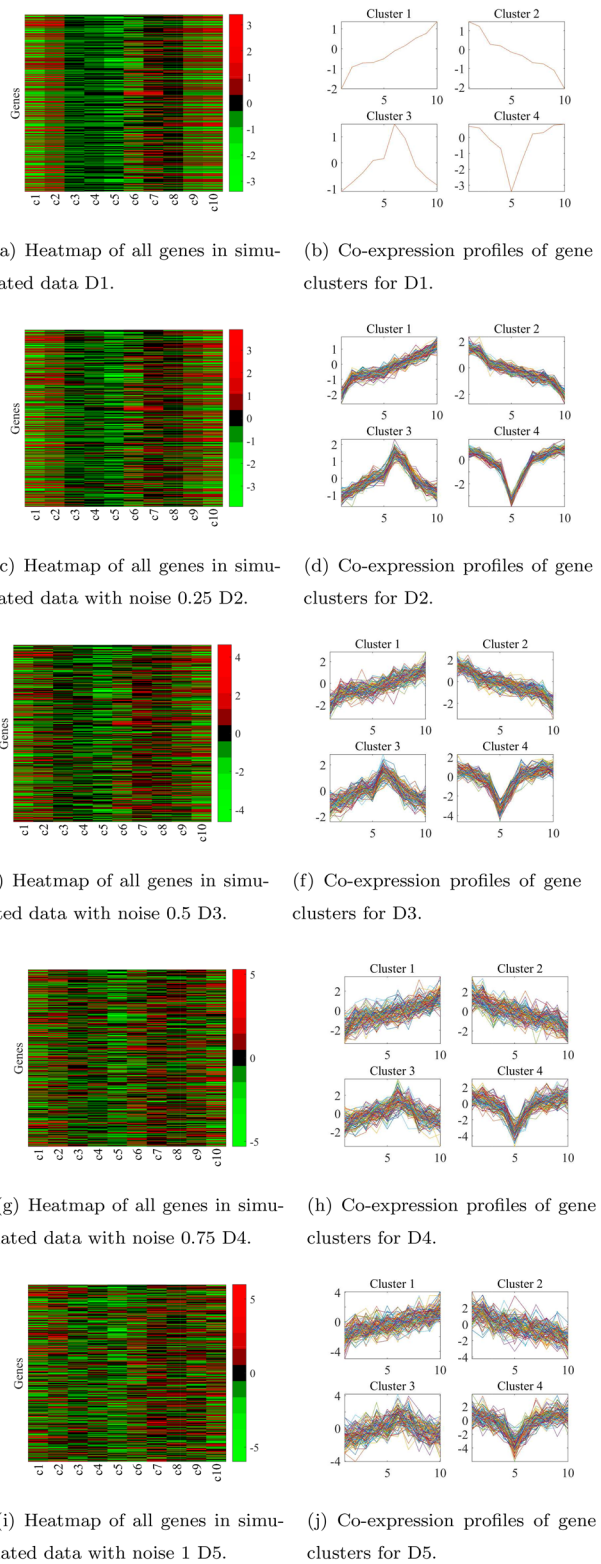
### 5.4 Real dataset description

To examine the capability of clustering algorithms, we test all algorithms on five different Affymetrix cancer gene expression datasets. The description of the gene expression datasets is summarized in Table 4. The reported datasets are obtained from Affymetrix chips. The description of the table consists of the name of the dataset (first column), type of tissue (second column), number of genes (third column), number of samples (fourth column), and number of classes (fifth column). The microarray gene expression datasets which were already preprocessed by De Souto et al. de Souto et al. (2008) are taken from a website https://schlieplab.org/Static/Supplements/CompCancer/. Before applying clustering algorithms, we normalized all the datasets using z-score to $\mu$ 0 and $\sigma$ 1. Next, we describe the datasets in detail.

### 5.5 Performance on real datasets

To investigate the comparative performance of GAClust, and SGAClust on cancer gene expression datasets, we execute K-means, HC, SOTA, CAST, CLICK, and SDC as the competing methods. To obtain the optimal number of clusters for two widely used traditional clustering algorithms, K-means and HC, we execute these algorithms on real data with $K$ values ranging from 2 to 50. Afterwards, the $K$ value is chosen in such a way, where the DB clustering index is minimized Abu-Jamous and Kelly (2018). In practice, we can cut the dendrogram at any level to get the desired number of clusters. However, for a fair comparison, we have done this exhaustive experimentation. We plot DB scores for each of the clustering algorithms generated by K-means and HC in Figs. 6 and 7, respectively. We apply CAST, SOTA, and CLICK with default parameter settings on real datasets.

As mentioned previously for synthetic datasets, we proceed similarly for deciding the input parameter $\Upsilon$ of GAClust for real datasets. In this case, we again plot sorted KNN graph for every dataset depicted in Fig. 8 Ester et al. (1996). With the help of this figure, visually, we can predict the $\Upsilon$ value for GAClust. Here, $K$ value of KNN graph is considered to be $M_p$ and $\frac{1}{1+\Upsilon}$ as $\varepsilon$ for SDC algorithm. We keep the value of $\delta$ as minimum as possible. The default value of $\delta$ is 3 for the SDC algorithm. We use MATLAB and R implementation for Lin's Yang et al. (2012) and Wang semantic similarity measure Yu et al. (2010), respectively. For Lin's measure, we download the gene ontology file (released on 2016-09-10) and annotation file of *Homo Sapience* from www.geneontology.org. We keep the values $w1 = 0.6$ and $w2 = 0.4$ for both SDC and SGAClust algorithms, as we want to give more weightage on proximity measure than semantic similarity measure. In SGAclust, $\varepsilon$ is used as $\Upsilon'$. Additionally, it is important to note that $\eta$ is calculated




(a) Heatmap of all genes in simulated data D1.

(b) Co-expression profiles of gene clusters for D1.

(c) Heatmap of all genes in simulated data with noise 0.25 D2.

(d) Co-expression profiles of gene clusters for D2.

(e) Heatmap of all genes in simulated data with noise 0.5 D3.

(f) Co-expression profiles of gene clusters for D3.

(g) Heatmap of all genes in simulated data with noise 0.75 D4.

(h) Co-expression profiles of gene clusters for D4.

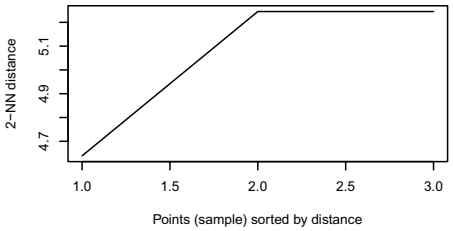
(i) Heatmap of all genes in simulated data with noise 1 D5.

(j) Co-expression profiles of gene clusters for D5.

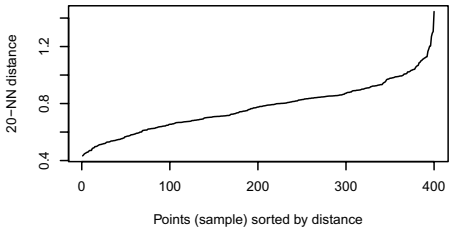
**Fig. 3** **a** Heatmap of all genes in simulated data D1. **b** Co-expression profiles of gene clusters for D1. **c** Heatmap of all genes in simulated data with noise 0.25 D2. **d** Co-expression profiles of gene clusters for D2. **e** Heatmap of all genes in simulated data with noise 0.5 D3. **f** Co-expression profiles of gene clusters for D3. **g** Heatmap of all genes in simulated data with noise 0.75 D4. **h** Co-expression profiles of gene clusters for D4. **i** Heatmap of all genes in simulated data with noise 1 D5. **j** Co-expression profiles of gene clusters for D5. Synthetic gene expression data with 400 genes and 10 samples with and without noise shown in the left column. The right column denotes the corresponding profiles of four gene clusters. The x direction shows the samples or conditions and y direction denotes the genes

dynamically for GAClust from where $\eta'$ is estimated. The parameter settings of all three algorithms can be found in Table 5.

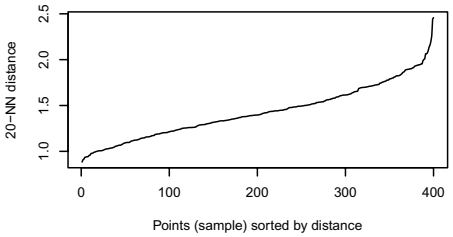
Figure 9 shows the results of these competing algorithms under various evaluation criteria on five cancer gene expression datasets. In addition to this, we summarize the results by taking an average across all datasets and reported in Table 6. From Table 6, we can see that unsupervised clustering algorithm CAST achieves the best performance among all other methods for all five datasets together across two validation indices, i.e., CI and DB. The possible reason of performing the best result is to identify more singleton clusters as outliers. On the other hand, SGAClust is considered to be the best performer for MSE and BH indices. Although SDC provides the best result, still we have not considered it as the best one because of the 'NAN' value for the Bhattacharjee dataset. If we closely observe the figure, we can see that individually for each dataset, SDC gives the lowest values for MSE and BH indices. GAClust is the second-best performer across all datasets for CI and DB. With the help of CI, semi-supervised clustering algorithms do not perform well contrasting with all unsupervised methods. Our proposed algorithm GAClust always take the immediate next position after the CAST algorithm for all four measures. In comparison to K-means and HC, both the algorithms perform very similarly mainly for MSE and BH, as can be observed from the table. In some of the datasets, K-means and HC perform very closely which can be easily observed from Fig. 9. Regarding the clustering algorithms SOTA and CLICK, it also provides similar values for 2 indices (MSE and BH) out of 4 indices. Overall, we can say that CLICK and SOTA give similar types of results for all the datasets. We note that these two algorithms are not good for analyzing gene expression data as they degrade the cluster quality. In summary, this output suggests that GAClust is more advantageous than K-means, HC, CLICK, and SOTA and as good as the CAST algorithm. Semi-supervised algorithms SDC and SGAClust are better than any other unsupervised algorithms with reference to MSE and BH. For the other two measures, i.e., CI and DB, the semi-supervised algorithms perform differently. It gives poor performance for CI and GAClust shows better results than semi-supervised algorithms for the DB index. It can be observed that CAST algorithm generates huge number of clusters, whereas GAClust has less number of clusters than CAST. SGAClust also identifies less number of clusters than GAClust. Another important observation is



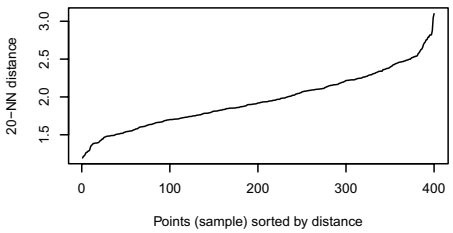

(a) Graph of sorted KNN distance for D1



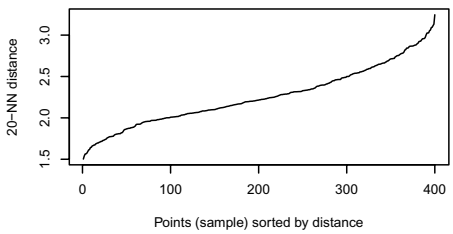

(b) Graph of sorted KNN distance for D2




(c) Graph of sorted KNN distance for D3




(d) Graph of sorted KNN distance for D4




(e) Graph of sorted KNN distance for D5

**Fig. 4** **a** Graph of sorted KNN distance for D1; **b** Graph of sorted KNN distance for D2; **c** Graph of sorted KNN distance for D3; **d** Graph of sorted KNN distance for D4; **e** Graph of sorted KNN distance for D5 Determination of $\Upsilon$ of GAClust for synthetic data by the graphs of sorted K-Nearest Neighbor (KNN) distance

**Table 2** Parameter settings of GAClust for synthetic datasets

| Dataset | Attraction threshold ($\eta$) | Neighborhood distance ($\Upsilon$) |
|---|---|---|
| D1 | 0.6241 | 5.2 + 1 |
| D2 | 0.4853 | 1.3 + 0.5 |
| D3 | 0.4604 | 2.3 + 0.5 |
| D4 | 0.3687 | 3 + 0.5 |
| D5 | 0.3278 | 3.1 + 0.5 |

to detect more number of singleton clusters in GAClust and SGAClust. This creates difference in CI for CAST, GAClust, and SGAClust. It is noteworthy that the SGAClust algorithm is better than the SDC algorithm.

*Enrichment analysis:* Due to the biological complexity, enrichment analysis for co-expressed genes in real expression data is one of the most commonly used techniques for biological validation rather than statistical analysis. The best analytical decision can be made with the aid of biological knowledge, annotation database, resulting clusters, and p value acquired from statistical methods. In contrast, co-expressed genes in a cluster are expected to be enriched related to the biological role. More importantly, enrichment analysis is helpful to determine the over-representation of given input gene lists over the background set of genes. The background gene set is compiled from the GO database. The three categories of GO are BP, MF, and CC. A cluster is considered to be enriched if the p values of all the annotation terms are less than the significance cut-off value. Moreover, if one of the annotation terms is from any one of the GO categories, for instance, BP, it is said to be enriched.

Proposed algorithms discover several genes as outliers; now, it is time to investigate whether reduction of genes affects the enrichment analysis or not. Among the many

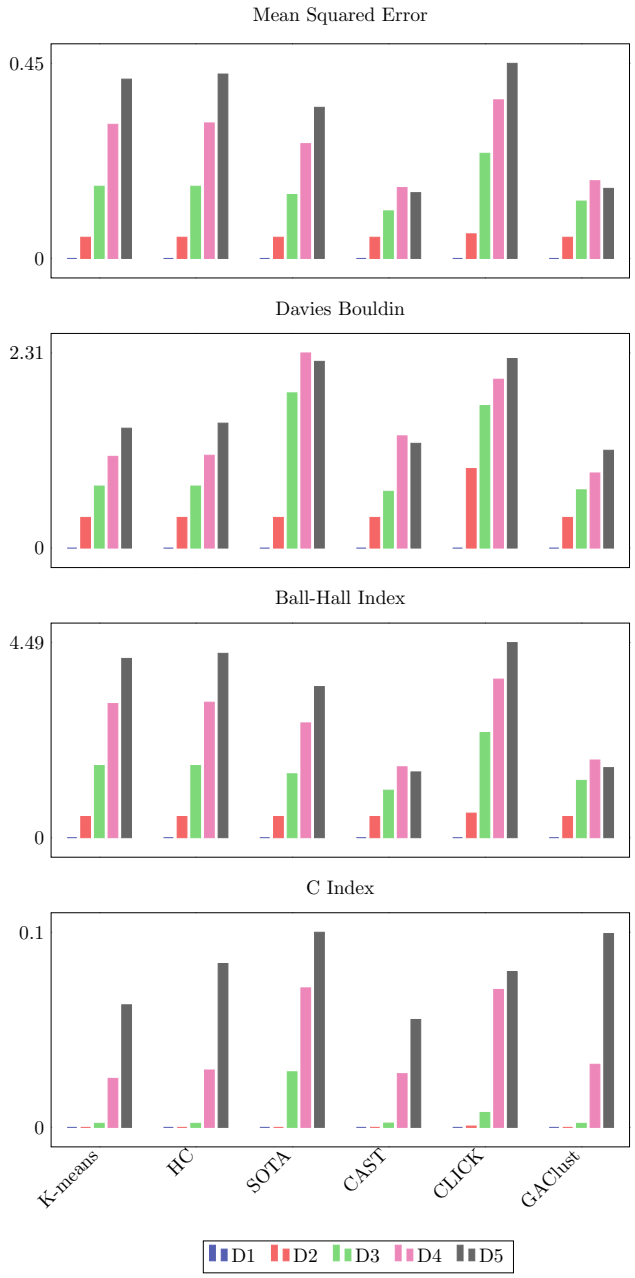



**Fig. 5** Histogram of different cluster validation indices on five synthetic datasets

**Table 3** Average internal measure on five synthetic datasets

| Algorithm | MSE | DB | BH | CI |
|---|---|---|---|---|
| K-means | 0.1874 | 0.7211 | 1.8739 | 0.0182 |
| HC | 0.1904 | 0.7356 | 1.9036 | 0.0233 |
| SOTA | 0.1617 | 1.3454 | 1.6169 | 0.0404 |
| CAST | 0.0946 | 0.7212 | 0.9462 | 0.0172 |
| CLICK | 0.2227 | 1.3757 | 2.2270 | 0.0321 |
| GAClust | 0.1043 | 0.6204 | 1.0423 | 0.0270 |

available enrichment tools, we have used FuncAssociate Berriz et al. (2003) for calculating p values. FuncAssociate uses Fisher's exact test to compute the hypergeometric functional score and adjusts the score for multiple testing using another method, named Westfall and Young procedure Berriz et al. (2003). It is necessary to convert the gene list into Official IDs from Affymetrix id using web-based tool Database for Annotation, Visualization and Integrated Discovery (DAVID) Huang et al. (2007). The resulting clusters discovered from each of the methods are submitted into FuncAssociate 3.0 Berriz et al. (2003) one by one as a gene query list. Each of the methods is evaluated by their potentiality of identifying the total number of enriched GO terms with a 5% significant cut-off for each dataset. Figure 10 depicts the number of significant GO terms for each method on each dataset. Considering all the datasets, we see different methods giving the enrichment result in different numbers of significant GO terms ranging from 3 to 1032.

Considering only unsupervised algorithms, we can observe that HC detects maximum numbers of enriched GO terms on the Bhattacharjee dataset and the K-means algorithm discovered the highest number of GO terms on the Singh dataset. It can also be noted that the CLICK algorithm outperforms among all unsupervised methods for the Armstrong-v2 dataset. Moreover, the CAST algorithm wins over Laiho and Ramaswamy datasets in this experiment. Although there is variation in the performance of different datasets for unsupervised algorithms, overall, taking all the five datasets together, GAClust identifies 1553 GO terms which is definitely higher than any other unsupervised testing method. Based on unsupervised algorithms, SOTA is considered to be the worst performer by yielding 1229 numbers of GO terms. While comparing between semi-supervised and unsupervised methods, it can be found that semi-supervised algorithm, i.e., SGAClust outperforms all other methods including GAClust on every dataset, whereas the SDC algorithm shows the worst performance in this case. SDC algorithm can only identify 384 significant GO terms which is much much lesser than GO terms identified by the SOTA algorithm.

Next, the central point of our discussion is p values. For that, we summarize the lowest p value corresponding to a GO term among all resulting clusters of each method on every dataset as given in Table 7. Lower p values signify better cluster. SGAClust outperforms all other clustering algorithms by giving lowest p values for all datasets, whereas SDC gives higher p values for all datasets. Interestingly, SOTA yields second-best *p* value for two datasets, Bhattacharjee and Ramaswamy, but it shows worst performance in the previous experiment. GAClust also provides lower *p* values for datasets Laiho and Singh among all unsupervised ones and second best among all methods. CLICK algorithm also seems to achieve lower p value of 1.23E-39

**Table 4** A brief description of cancer gene expression datasets

| Dataset | Tissue type | Genes | Samples | Class | Ref |
|---|---|---|---|---|---|
| Armstrong-v2 | Blood | 2194 | 72 | 3 | Armstrong et al. (2002) |
| Bhattacharjee | Lung | 1543 | 203 | 5 | Bhattacharjee et al. (2001) |
| Laiho | Colon | 2202 | 37 | 2 | Laiho et al. (2007) |
| Ramaswamy | Multi-tissue | 1363 | 190 | 14 | Ramaswamy et al. (2001) |
| Singh | Prostate | 339 | 102 | 2 | Singh et al. (2002) |

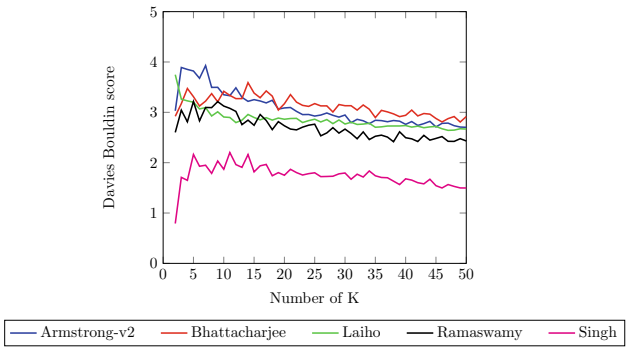



**Fig. 6** Selection of K for K-means algorithm with respect to Davies–Bouldin score for cancer gene expression datasets

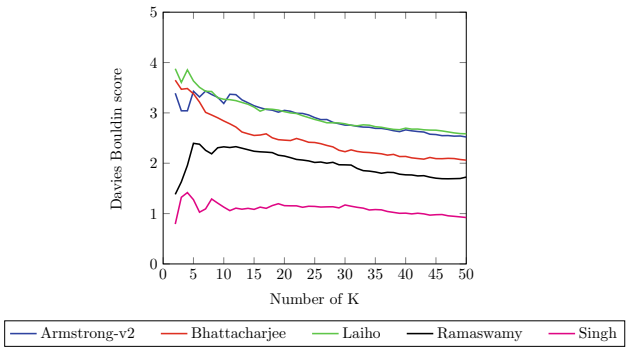



**Fig. 7** Selection of number of clusters for hierarchical clustering with respect to Davies–Bouldin score for real datasets

for Laiho while comparing with unsupervised methods and is the second-best performer after SGAClust. Moreover, K-means algorithms give second-best result in this regard for Armstrong dataset. Surprisingly, with the reference to the previous discussion, we can see that the methods, which provides the good number of GO terms, may not give the lowest p value, for instance GAClust algorithm.

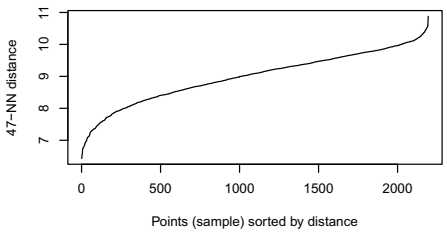
## 6 Potential biomarker identification

In this section, we identify biomarkers with the help of network-based biomarker identification techniques Mandal et al. (2018). The foundation of this technique is based on the clustering results. This technique uses two user-defined thresholds, i.e., $\varphi$ for the number of biomarkers and $\psi$ for the number of clusters.

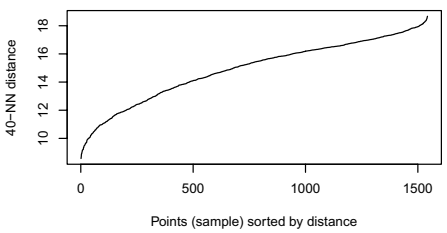
Among the potential biomarkers, Yu et al. (2018) have reported that *APP* is highly expressed in AML. Therefore, clinically it has a significant impact on blood cancer. *SETBP1* is considered as oncogene and it defines the molecular characteristics of Leukemia Coccaro et al. (2017). The mutation of *SETBP1* gene is an important factor in cancer development. The gene *ILS1* plays an important key factor in many cancers including lung tumor Li et al. (2018, 2014) which have evaluated the function of gene *PBX1* in the proliferation of non-small-cell lung cancer. Lung cancer is a widely spread oncological disease. The study in Yu et al. (2018) has reported that *COL1A2* is treated as a tumor suppressor in the colorectal cancer cell and also provided a therapeutic approach to treat this disease. For colorectal cancer, *FBN1* gene may be consider as a promising biomarker Li et al. (2015). The gene *MAGI2* is altered in 0.81% of all types of cancers such as lung, colon, and breast cancer[2]. The study Fane et al. (2017) focuses on the vital role of gene *NFIA* in multiple cancer types. The gene *FN1* is found to be dysregulated in multiple cancers such as colon cancer Li et al. (2019). According to Cancer Genetics Web, *TCF4* plays an useful oncogene role in ovarian cancer The gene *BMP2* is highly overexpressed in lung cancer tissue compared to normal tissue Bach et al. (2018). In the study Pan et al. (2018), it has been investigated that *RPS16* gene is useful on tumorigenesis and development of prostate cancer (Table 8).

Zhu et al. have suggested that the gene *EPB42* is a suitable biomarker for therapeutic strategies for AML patients Zhu et al. (2017). *GZMA* can be considered as a useful early biomarker for ALL therapy Myoumoto et al. (2007). The study in Juurikka et al. (2019) has mentioned that *MMP8* is treated as a prognostic factor in cancer treatment. Han et al. have predicted that the gene *MMP2* can be a potential biomarker for prognosis and diagnosis of lung cancer Han et al. (2020). It has been found in the study of Lin et al. (2020)

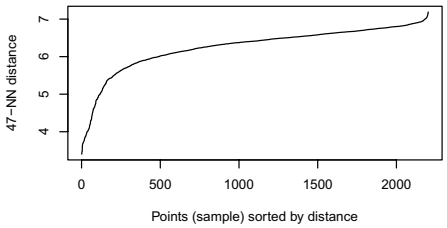
[2] https://www.mycancergenome.org.



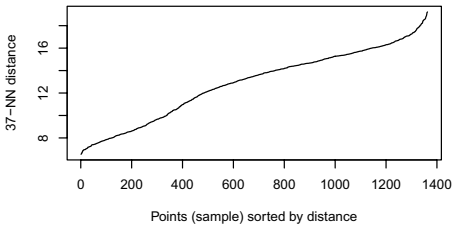

(a) Graph of sorted KNN distance for Armstrong-v2 data



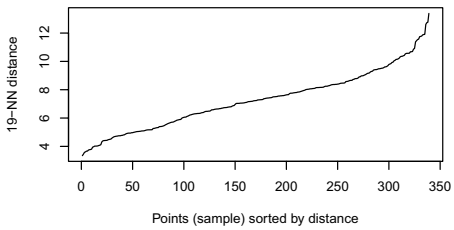

(b) Graph of sorted KNN distance for Bhattacharjee dataset




(c) Graph of sorted KNN distance for Laiho dataset




(d) Graph of sorted KNN distance for Ramaswamy dataset




(e) Graph of sorted KNN distance for Singh dataset

**Fig. 8** Determination of $\Upsilon$ of GAClust for real data by the graphs of sorted KNN distance

**Table 5** Parameter setting of GAClust for cancer gene expression datasets

| Dataset | GAClust | | SDC | | SGAClust | |
|---|---|---|---|---|---|---|
| | $\eta$ | $\Upsilon$ | $M_p$ | $\varepsilon$ | $\eta'$ | $\Upsilon'$ |
| Armstrong-v2 | 0.2911 | 10.5 + 0.5 | 47 | 0.08 | 0.8362 | 0.08 |
| Bhattacharjee | 0.3429 | 18 + 0.5 | 40 | 0.05 | 0.8794 | 0.05 |
| Laiho | 0.2945 | 6.8 + 0.5 | 47 | 0.12 | 0.7067 | 0.12 |
| Ramaswamy | 0.3523 | 18 + 0.5 | 37 | 0.05 | 0.7166 | 0.05 |
| Singh | 0.3622 | 12 + 0.5 | 19 | 0.07 | 0.8711 | 0.07 |

that the gene *TRIM2* is highly expressed in lung adenocarcinoma tissues.

In this study, we have identified several potential biomarkers from the clustering results identified by GAClust, SGAClust, and CAST which are validated through literature. Comparing these three algorithms, we have noticed that the total number of predicted biomarkers by the GAClust, SGAClust, and CAST algorithms is 10, 11, and 18, respectively. GAClust detects nine valid biomarkers whereas SGAClust and CAST detect 9 and 8, respectively. Therefore, based on this experiment, we can easily comment that GAClust identifies the maximum percentage (90%) of valid biomarkers and CAST performs worst in this case. Some of the biomarkers are common in both GAClust and SGAClust as they follow the similar approach to find gene clusters.

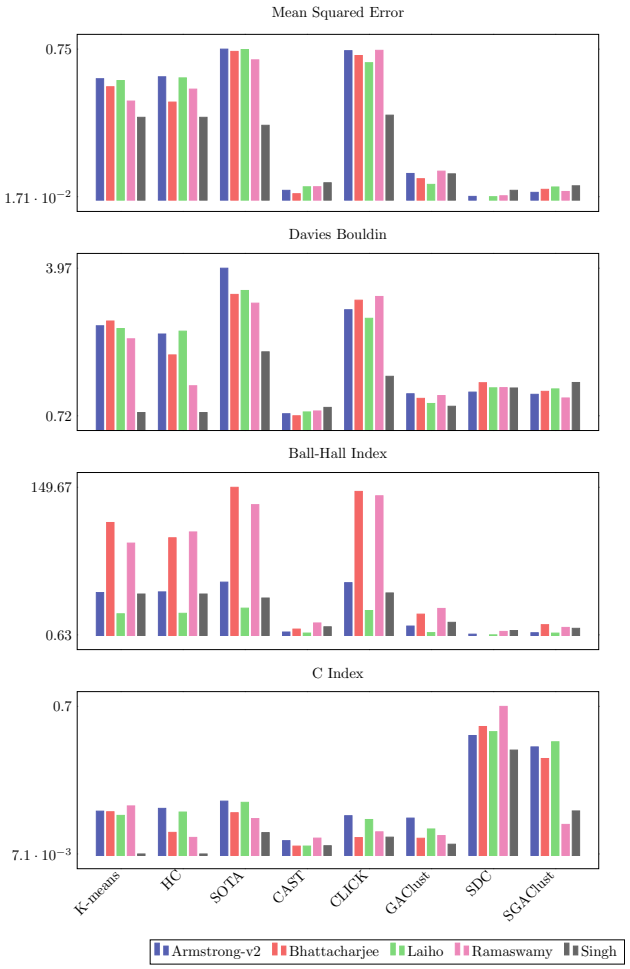



**Fig. 9** Different cluster validation indexes for cancer datasets

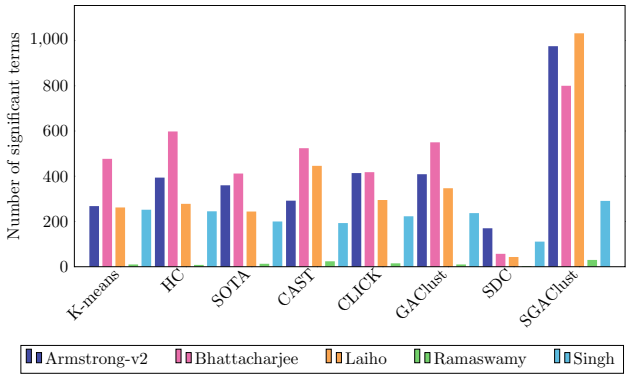



**Fig. 10** Number of enriched terms (shown in y-axis) by six different methods (shown in x-axis) for different datasets

**Table 6** Average internal measure on five cancer gene expression datasets

| Algorithm | MSE | DB | BH | CI |
|---|---|---|---|---|
| K-means | 0.5317 | 2.2720 | 62.9152 | 0.1693 |
| HC | 0.5329 | 1.8682 | 62.3143 | 0.1259 |
| SOTA | 0.6597 | 3.2350 | 80.2281 | 0.1984 |
| CAST | 0.0599 | 0.8048 | 6.7760 | 0.0573 |
| CLICK | 0.6605 | 2.8258 | 81.6232 | 0.1281 |
| GAClust | 0.1182 | 1.0797 | 14.7863 | 0.1061 |
| SDC | 0.0266 | 1.3402 | 2.7707 | 0.5926 |
| SGAClust | 0.0544 | 1.2665 | 6.3043 | 0.3731 |

## 7 Conclusion

The proposed unsupervised GAClust algorithm is based on a graph-theoretic clustering algorithm. The main focus of the GAClust algorithm is to develop a parameterless clustering and which makes the algorithm distinct from CAST. Moreover, the GAClust algorithm decides the threshold dynamically depending upon the individual dataset whereas the CAST algorithm does not follow any guideline. Unlike CAST, GAClust obviates the need for a cleaning step due to this dynamic threshold as proposed in the original algorithm. Our algorithm is advantageous as it does not require the number of clusters a priori. We have also provided a guideline for the input parameters. The performance of GAClust is compared with five state-of-the-art methods for synthetic and cancer gene expression datasets using both internal and external measures as a validation criterion.

The first striking conclusion we can draw is that no algorithm is superior throughout all the measures overall datasets synthetic as well as real. Indeed, in many cases, we have observed that one algorithm may give the best result for some metric and may also be worst considering another metric. From the study, we can say that GAClust is a well-suited algorithm in comparison with all other methods for synthetic datasets. Following the real datasets, the GAClust algorithm outperforms all other comparing methods for biological significance. Hence, GAClust is biologically more significant than other algorithms. Additionally, GAClust outperforms all algorithms except for CAST.

We have also proposed SGAClust which integrates GO with GAClust. The main advantage of both algorithms is that we do not have to give the number of clusters as an input. Among these two clusters, SGAClust outperforms all

**Table 7** Comparison of p values among all datasets on various datasets

| Datasets | Algorithms | GO ID | GO Name | p value |
|---|---|---|---|---|
| Armstrong-v2 | K-means | GO:0000786 | Nucleosome | 4.96E-24 |
| | HC | GO:0043299 | Leukocyte degranulation | 1.95E-23 |
| | SOTA | GO:0031325 | Positive regulation of cellular metabolic process | 1.05E-14 |
| | CAST | GO:0000786 | Nucleosome | 3.42E-21 |
| | CLICK | GO:0007165 | Signal transduction | 6.27E-23 |
| | GAClust | GO:0007165 | Signal transduction | 2.39E-20 |
| | SDC | GO:0032502 | Developmental process | 8.18E-14 |
| | SGAClust | GO:0007165 | Signal transduction | 2.91E-91 |
| Bhattacharjee | K-means | GO:0070268 | Cornification | 1.59E-20 |
| | HC | GO:0002376 | Immune system process | 4.79E-22 |
| | SOTA | GO:0002376 | Immune system process | 1.29E-26 |
| | CAST | GO:0000786 | Nucleosome | 1.3E-16 |
| | CLICK | GO:0030198 | Extracellular matrix organization | 7.88E-22 |
| | GAClust | GO:0070268 | Cornification | 2.16E-19 |
| | SDC | GO:0044421 | Extracellular region part | 6.40E-08 |
| | SGAClust | GO:0007165 | Signal transduction | 2.52E-70 |
| Laiho | K-means | GO:0031012 | Extracellular matrix | 4.20E-32 |
| | HC | GO:0031012 | Extracellular matrix | 6.10E-33 |
| | SOTA | GO:0031012 | Extracellular matrix | 1.85E-37 |
| | CAST | GO:0031012 | Extracellular matrix | 2.49E-29 |
| | CLICK | GO:0031012 | Extracellular matrix | 1.24E-39 |
| | GAClust | GO:0031012 | Extracellular matrix | 4.14E-39 |
| | SDC | GO:0032963 | Collagen metabolic process | 1.69E-11 |
| | SGAClust | GO:0065007 | Biological regulation | 3.15E-72 |
| Ramaswamy | K-means | GO:0002376 | Immune system process | 6.48E-07 |
| | HC | GO:0097458 | Neuron part | 2.96E-07 |
| | SOTA | GO:0043005 | Neuron projection | 7.82E-09 |
| | CAST | GO:0043005 | Neuron projection | 5.89E-07 |
| | CLICK | GO:0043005 | Neuron projection | 4.69E-08 |
| | GAClust | GO:0097458 | Neuron part | 1.11E-08 |
| | SDC | GO:0030425 | Dendrite | 1.11E-06 |
| | SGAClust | GO:0043005 | Neuron projection | 2.14E-09 |
| Singh | K-means | GO:0006614 | SRP-dependent cotranslational protein targeting to membrane | 1.80E-75 |
| | HC | GO:0006614 | SRP-dependent cotranslational protein targeting to membrane | 1.80E-75 |
| | SOTA | GO:0006614 | SRP-dependent cotranslational protein targeting to membrane | 2.34E-63 |
| | CAST | GO:0006614 | SRP-dependent cotranslational protein targeting to membrane | 7.48E-69 |
| | CLICK | GO:0006614 | SRP-dependent cotranslational protein targeting to membrane | 3.97E-67 |
| | GAClust | GO:0006614 | SRP-dependent cotranslational protein targeting to membrane | 1.80E-76 |
| | SDC | GO:0006413 | Translational initiation | 1.97E-15 |
| | SGAClust | GO:0043005 | Neuron projection | 5.88E-80 |

other competing methods. It is being observed that external domain knowledge such as GO gives reliable clusters and what makes it best performer among all algorithms. Additionally, all the proposed algorithms are equally effective to identify potential cancer biomarkers. We have validated the found biomarkers from the literature.

We conclude that a semi-supervised algorithm provides significant clusters. Biologically, it is proven that one gene may participate in many biological pathways, and this allows it to belong to multiple clusters. The drawback of the proposed full-space clustering algorithms is that it finds disjoint clusters and cannot find overlapping clusters. Detecting overlapping clusters is a crucial task and can be exploited using it as future work.

**Table 8** Potential biomarker identification of different proposed full-space clustering algorithms using network-based methods

| Algorithms | Datasets | Potential biomarkers |
|---|---|---|
| GAClust | Armstrong-v2 | *APP*, *SETBP1* |
| | Bhattacharjee | *ISL1*, *PBX1* |
| | Laiho | *COL1A2*, *FBN1* |
| | Ramaswamy | *MAGI2*, *NFIA* |
| | Singh | *RPS23*, *RPS16* |
| SGAClust | Armstrong-v2 | *FN1*, *APP* |
| | Bhattacharjee | *ISL1*, *BMP2* |
| | Laiho | *FN1*, *COL1A2* |
| | Ramaswamy | *MAGI2*, *TCF4* |
| | Singh | *RPS23*, *RPS16*, *RPS3A* |
| CAST | Armstrong-v2 | *EPB42*, *GZMA*, *MMP8*, *SLC4A1*, *CDC20*, *CEACAM8*, *DEFA4*, *GZMB*, *KIF2C* |
| | Bhattacharjee | *SNAP25*, *SYT1* |
| | Laiho | *COL1A2*, *MMP2* |
| | Ramaswamy | *MAGI2*, *TRIM2* |
| | Singh | *RPS16*, *RPS23*, *RPS13* |

## References


Abu-Jamous B, Kelly S (2018) Clust: automatic extraction of optimal co-expressed gene clusters from gene expression data. Genome Biol 19(1):1–11

Armstrong SA, Staunton JE, Silverman LB, Pieters R, den Boer ML, Minden MD, Sallan SE, Lander ES, Golub TR, Korsmeyer SJ (2002) Mll translocations specify a distinct gene expression profile that distinguishes a unique leukemia. Nat Genet 30(1):41–47

Bach D-H, Park HJ, Lee SK (2018) The dual role of bone morphogenetic proteins in cancer. Mol Therapy-Oncolytics 8:1–13

Ben-Dor A, Shamir R, Yakhini Z (1999) Clustering gene expression patterns. J Comput Biol 6(3–4):281–297

Berriz GF, King OD, Bryant B, Sander C, Roth FP (2003) Characterizing gene sets with funcassociate. Bioinformatics 19(18):2502–2504

Bhattacharjee A, Richards WG, Staunton J, Li C, Monti S, Vasa P, Ladd C, Beheshti J, Bueno R, Gillette M et al (2001) Classification of human lung carcinomas by mrna expression profiling reveals distinct adenocarcinoma subclasses. Proc Natl Acad Sci 98(24):13790–13795

Bryan J (2004) Problems in gene clustering based on gene expression data. J Multivariate Anal 90(1):44–66

Chen AH, Tsau Y-W, Lin C-H (2010) Novel methods to identify biologically relevant genes for leukemia and prostate cancer from gene expression profiles. BMC genomics 11(1):274

Coccaro N, Tota G, Zagaria A, Anelli L, Specchia G, Albano F (2017) Setbp1 dysregulation in congenital disorders and myeloid neoplasms. Oncotarget 8(31):51920

de Souto MC, Costa IG, de Araujo DS, Ludermir TB, Schliep A (2008) Clustering cancer gene expression data: a comparative study. BMC Bioinformatics 9(1):497

Desgraupes B (2013) Clustering indices. University of Paris Ouest-Lab Modal'X 1:34

Di Gesú V, Giancarlo R, Bosco GL, Raimondi A, Scaturro D (2005) Genclust: a genetic algorithm for clustering gene expression data. BMC Bioinformatics 6(1):289

Eisen MB, Spellman PT, Brown PO, Botstein D (1998) Cluster analysis and display of genome-wide expression patterns. Proc Natl Acad Sci 95(25):14863–14868

Erbes T, Hirschfeld M, Rücker G, Jaeger M, Boas J, Iborra S, Mayer S, Gitsch G, Stickeler E (2015) Feasibility of urinary microrna detection in breast cancer patients and its potential as an innovative non-invasive biomarker. BMC Cancer 15(1):193

Fane M, Harris L, Smith AG, Piper M (2017) Nuclear factor one transcription factors as epigenetic regulators in cancer. Int J Cancer 140(12):2634–2641

Goossens N, Nakagawa S, Sun X, Hoshida Y (2015) Cancer biomarker discovery and validation. Trans Cancer Res 4(3):256

Han L, Sheng B, Zeng Q, Yao W, Jiang Q (2020) Correlation between mmp2 expression in lung cancer tissues and clinical parameters: a retrospective clinical analysis. BMC Pulmonary Medi 20(1):1–9

Henriques R, Madeira SC (2016) Bic2pam: constraint-guided biclustering for biological data analysis with domain knowledge. Algorithms Mol Biol 11(1):23

Henry NL, Hayes DF (2012) Cancer biomarkers. Mol Oncol 6(2):140–146

Herrero J, Valencia A, Dopazo J (2001) A hierarchical unsupervised growing neural network for clustering gene expression patterns. Bioinformatics 17(2):126–136

Huang D, Pan W (2006) Incorporating biological knowledge into distance-based clustering analysis of microarray gene expression data. Bioinformatics 22(10):1259–1268

Huang DW, Sherman BT, Tan Q, Collins JR, Alvord WG, Roayaei J, Stephens R, Baseler MW, Lane HC, Lempicki RA (2007) The david gene functional classification tool: a novel biological module-centric algorithm to functionally analyze large gene lists. Genome Biol 8(9):R183

Hubert L, Schultz J (1976) Quadratic assignment as a general data analysis strategy. Br J Math Stat Psychol 29(2):190–241

Hussain SF, Ramazan M (2016) Biclustering of human cancer microarray data using co-similarity based co-clustering. Expert Syst Appl 55:520–531

Jaskowiak PA, Campello RJ, Costa IG (2014) On the selection of appropriate distances for gene expression data clustering. BMC Bioinform 15(Suppl 2):S2

Ji L, Tan K-L (2004) Mining gene expression data for positive and negative co-regulated gene clusters. Bioinformatics 20(16):2711–2718

Jiang D, Tang C, Zhang A (2004) Cluster analysis for gene expression data: a survey. Knowl Data Eng IEEE Trans 16(11):1370–1386

Joe S, Nam H (2016) Prognostic factor analysis for breast cancer using gene expression profiles. BMC Med Inform Decision Making 16(1):56

Juurikka K, Butler GS, Salo T, Nyberg P, Åström P (2019) The role of mmp8 in cancer: a systematic review. Int J Mol Sci 20(18):4506

Kerr G, Ruskin HJ, Crane M, Doolan P (2008) Techniques for clustering gene expression data. Comput Biol Med 38(3):283–293

Kim H, Watkinson J, Anastassiou D (2011) Biomarker discovery using statistically significant gene sets. J Comput Biol 18(10):1329–1338

Kulshrestha A, Suman S, Ranjan R (2016) Network analysis reveals potential markers for pediatric adrenocortical carcinoma. OncoTargets Therapy 9:4569

Laiho P, Kokko A, Vanharanta S, Salovaara R, Sammalkorpi H, Järvinen H, Mecklin J, Karttunen T, Tuppurainen K, Davalos V et al (2007) Serrated carcinomas form a subclass of colorectal cancer with distinct molecular basis. Oncogene 26(2):312–320

Lee W-P, Lin C-H (2016) Combining expression data and knowledge ontology for gene clustering and network reconstruction. Cognit Comput 8(2):217–227

Li Z, Herold T, He C, Valk PJ, Chen P, Jurinovic V, Mansmann U, Radmacher MD, Maharry KS, Sun M et al (2013) Identification of a 24-gene prognostic signature that improves the european leukemianet risk classification of acute myeloid leukemia: an international collaborative study. J Clinical Oncol 31(9):1172–1181

Li W, Huang K, Guo H, Cui G, Zhao S (2014) Inhibition of non-small-cell lung cancer cell proliferation by pbx1. Chin J Cancer Res 26(5):573

Li L, Sun F, Chen X, Zhang M (2018) Isl1 is upregulated in breast cancer and promotes cell proliferation, invasion, and angiogenesis. OncoTargets Therapy 11:781

Li J, Ma S, Lin T, Li Y, Yang S, Zhang W, Zhang R, Wang Y (2019) Comprehensive analysis of therapy-related messenger rnas and long noncoding rnas as novel biomarkers for advanced colorectal cancer. Front Genet 10:803

Lin W, Feng M, Li X, Zhong P, Guo A, Chen G, Xu Q, Ye Y (2017) Transcriptome profiling of cancer and normal tissues from cervical squamous cancer patients by deep sequencing. Mol Med Rep 16(2):2075–2088

Lin Z, Lin X, Zhu L, Huang J, Huang Y (2020) Trim2 directly deubiquitinates and stabilizes snail1 protein, mediating proliferation and metastasis of lung adenocarcinoma. Cancer Cell Int 20(1):1–14

Liu J, Jing L, Tu X (2016) Weighted gene co-expression network analysis identifies specific modules and hub genes related to coronary artery disease. BMC Cardiovascular Disorders 16(1):54

Lord PW, Stevens RD, Brass A, Goble CA (2003) Investigating semantic similarity measures across the gene ontology: the relationship between sequence and annotation. Bioinformatics 19(10):1275–1283

Macintyre G, Bailey J, Gustafsson D, Haviv I, Kowalczyk A (2010) Using gene ontology annotations in exploratory microarray clustering to understand cancer etiology. Pattern Recognit Lett 31(14):2138–2146

Mandal K, Sarmah R, Bhattacharyya DK (2018) Biomarker identification for cancer disease using biclustering approach: an empirical study. IEEE/ACM Trans Comput Biol Bioinform 16(2):490–509

Martinez-Ledesma E, Verhaak RG, Treviño V (2015) Identification of a multi-cancer gene expression biomarker for cancer clinical outcomes using a network-based algorithm. Sci Rep 5:11966

Mitra S, Banka H (2006) Multi-objective evolutionary biclustering of gene expression data. Pattern Recognit 39(12):2464–2477

Mitra S, Ghosh S (2012) Feature selection and clustering of gene expression profiles using biological knowledge, IEEE Transactions on Systems, Man, and Cybernetics. Part C (Applications and Reviews) 42(6):1590–1599

Mohammed A, Biegert G, Adamec J, Helikar T (2017) Identification of potential tissue-specific cancer biomarkers and development of cancer versus normal genomic classifiers. Oncotarget 8(49):85692

Myoumoto A, Nakatani K, Koshimizu T-A, Matsubara H, Adachi S, Tsujimoto G (2007) Glucocorticoid-induced granzyme a expression can be used as a marker of glucocorticoid sensitivity for acute lymphoblastic leukemia therapy. J Human Genet 52(4):328–333

Nepomuceno JA, Troncoso A, Nepomuceno-Chamorro IA, Aguilar-Ruiz JS (2015) Integrating biological knowledge based on functional annotations for biclustering of gene expression data. Comput Methods Programs Biomed 119(3):163–180

Pan Y-L, Jun Q, Zhou L, Zhang T-T, Qiang L (2018) Ribosomal protein 16 overexpresses in prostate cancer and promotes tumor progression. J Shanghai Jiaotong Univ (Med Sci) 38(4):394–399

Pesquita C, Faria D, Falcao AO, Lord P, Couto FM (2009) Semantic similarity in biomedical ontologies. PLoS Comput Biol 5(7):e1000443

Pirim H, Ekşioğlu B, Perkins AD, Yüceer Ç (2012) Clustering of high throughput gene expression data. Comput Oper Res 39(12):3046–3061

Ramaswamy S, Tamayo P, Rifkin R, Mukherjee S, Yeang C-H, Angelo M, Ladd C, Reich M, Latulippe E, Mesirov JP et al (2001) Multiclass cancer diagnosis using tumor gene expression signatures. Proc Natl Acad Sci 98(26):15149–15154

Rose K (1998) Deterministic annealing for clustering, compression, classification, regression, and related optimization problems. Proc IEEE 86(11):2210–2239

Sachnev V, Saraswathi S, Niaz R, Kloczkowski A, Suresh S (2015) Multi-class bcga-elm based classifier that identifies biomarkers associated with hallmarks of cancer. BMC Bioinform 16(1):166

Samee NMA, Solouma NH, Kadah YM (2012) Detection of biomarkers for hepatocellular carcinoma using a hybrid univariate gene selection methods. Theoretical Biol Med Modell 9(1):34

Singh D, Febbo PG, Ross K, Jackson DG, Manola J, Ladd C, Tamayo P, Renshaw AA, D'Amico AV, Richie JP et al (2002) Gene expression correlates of clinical prostate cancer behavior. Cancer cell 1(2):203–209

Stratford JK, Bentrem DJ, Anderson JM, Fan C, Volmar KA, Marron J, Routh ED, Caskey LS, Samuel JC, Der CJ et al (2010) A six-gene signature predicts survival of patients with localized pancreatic ductal adenocarcinoma. PLoS Med 7(7):e1000307

Tamayo P, Slonim D, Mesirov J, Zhu Q, Kitareewan S, Dmitrovsky E, Lander ES, Golub TR (1999) Interpreting patterns of gene expression with self-organizing maps: methods and application to hematopoietic differentiation. Proc Natl Acad Sci 96(6):2907–2912

Tellaroli P, Bazzi M, Donato M, Brazzale AR, Drăghici S (2016) Cross-clustering: a partial clustering algorithm with automatic estimation of the number of clusters. PloS one 11(3):e0152333

Tusher VG, Tibshirani R, Chu G (2001) Significance analysis of microarrays applied to the ionizing radiation response. Proc Natl Acad Sci 98(9):5116–5121

Wang JZ, Du Z, Payattakool R, Yu PS, Chen C-F (2007) A new method to measure the semantic similarity of go terms. Bioinformatics 23(10):1274–1281

Yang H, Nepusz T, Paccanaro A (2012) Improving go semantic similarity measures by exploring the ontology beneath the terms and modelling uncertainty. Bioinformatics 28(10):1383–1389

Yu G, Li F, Qin Y, Bo X, Wu Y, Wang S (2010) Gosemsim: an r package for measuring semantic similarity among go terms and gene products. Bioinformatics 26(7):976–978

Yu G, Yin C, Jiang L, Xu D, Zheng Z, Wang Z, Wang C, Zhou H, Jiang X, Liu Q et al (2018) Amyloid precursor protein has clinical and prognostic significance in aml1-eto-positive acute myeloid leukemia. Oncol Lett 15(1):917–925

Yu Y, Liu D, Liu Z, Li S, Ge Y, Sun W, Liu B (2018) The inhibitory effects of col1a2 on colorectal cancer cell proliferation, migration, and invasion. J Cancer 9(16):2953

Zhou W, Dickerson JA (2014) A novel class dependent feature selection method for cancer biomarker discovery. Comput Biol Med 47:66–75

Zhu G-Z, Yang Y-L, Zhang Y-J, Liu W, Li M-P, Zeng W-J, Zhao X-L, Chen X-P (2017) High expression of ahsp, epb42, gypc and hemgn predicts favorable prognosis in flt3-itd-negative acute myeloid leukemia. Cell Physiol Biochem 42(5):1973–1984

Adryan B, Schuh R (2004) Gene-ontology-based clustering of gene expression data, Bioinformatics 20(16):2851–2852. http://dx.doi.org/10.1093/bioinformatics/bth289

Ball GH, Hall DJ (1965) Isodata, a novel method of data analysis and pattern classification. Tech. rep, Stanford research inst Menlo Park CA

Bellaachia A, Portnoy D, Chen Y, Elkahloun AG (2002) E-cast: a data mining algorithm for gene expression data., in: BIOKDD, pp. 49–54

Binns D, Dimmer E, Huntley R, Barrell D, O'donovan C, Apweiler R (2009) Quickgo: a web-based tool for gene ontology searching, Bioinformatics 25(22):3045–3046

Bray F, Ferlay J, Soerjomataram I, Siegel RL, Torre LA, Jemal A (2018) Global cancer statistics 2018: Globocan estimates of incidence and mortality worldwide for 36 cancers in 185 countries, CA: a cancer journal for clinicians 68(6):394–424

Couto FM, Silva MJ, Coutinho PM (2005) Semantic similarity over the gene ontology: family correlation and selecting disjunctive ancestors, in: Proceedings of the 14th ACM international conference on Information and knowledge management, pp. 343–344

Davies DL, Bouldin DW (1979) A cluster separation measure, IEEE transactions on pattern analysis and machine intelligence (2):224–227

Ester M, Kriegel H-P, Sander J, Xu X (1996) A density-based algorithm for discovering clusters in large spatial databases with noise., in: Kdd, Vol. 96, pp. 226–231

Hang S, You Z, Chun LY (2009) Incorporating biological knowledge into density-based clustering analysis of gene expression data, in: Fuzzy Systems and Knowledge Discovery, 2009. FSKD'09. Sixth International Conference on, Vol. 5, IEEE, pp. 52–56

Jaskowiak PA, Campello RJ, Costa Filho IG (2013) Proximity measures for clustering gene expression microarray data: a validation methodology and a comparative analysis, IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB) 10(4):845–857

Jiang JJ, Conrath DW, Semantic similarity based on corpus statistics and lexical taxonomy, arXiv preprint cmp-lg/9709008

Jiang D, Pei J, Zhang A (2003) Dhc: a density-based hierarchical clustering method for time series gene expression data, in: Bioinformatics and Bioengineering, 2003. Proceedings. Third IEEE Symposium on, IEEE, pp. 393–400

Jiang D, Pei J, Zhang A (2004) Gpx: interactive mining of gene expression data, in: Proceedings of the Thirtieth international conference on Very large data bases-Volume 30, VLDB Endowment, pp. 1249–1252

Lam Y, Tsang PW, Leung C (2013) Pso-based k-means clustering with enhanced cluster matching for gene expression data, Neural Comput Appl 22(7-8):1349–1355. http://dx.doi.org/10.1007/s00521-012-0959-5

Li W-h, Zhang H, Guo Q, Wu X-d, Xu Z-s, Dang C-x, Xia P, Song Y-c (2015) Detection of snca and fbn1 methylation in the stool as a biomarker for colorectal cancer, Disease markers

Lin D (1998) An information-theoretic definition of similarity, in: J. W. Shavlik (Ed.), Proceedings of the Fifteenth International Conference on Machine Learning (ICML 1998), Madison, Wisconsin, USA, Morgan Kaufmann, 1998, pp. 296–304

Liu J, Wang W, Yang J (2004) Gene ontology friendly biclustering of expression profiles, in: Computational Systems Bioinformatics Conference. CSB 2004. Proceedings. 2004 IEEE, IEEE, 2004, pp. 436–447

Lu Y, Lu S, Fotouhi F, Deng Y, Brown SJ (2004) Incremental genetic k-means algorithm and its application in gene expression data analysis, BMC Bioinformatics 5:172. http://dx.doi.org/10.1186/1471-2105-5-172

MacQueen J et al. (1967) Some methods for classification and analysis of multivariate observations. In: Proceedings of the fifth Berkeley symposium on mathematical statistics and probability, Vol. 1, Oakland, CA, USA, pp. 281–297

Mandal K, Sarmah R (2018) A density-based clustering for gene expression data using gene ontology, in: Proceedings of the International Conference on Computing and Communication Systems, Springer, pp. 757–765

Ovaska K, Laakso M, Hautaniemi S (2008) Fast gene ontology based clustering for microarray experiments, BioData Mining 1. http://dx.doi.org/10.1186/1756-0381-1-11

Oyelade J, Isewon I, Oladipupo F, Aromolaran O, Uwoghiren E, Ameh F, Achas M, Adebiyi E (2016) Clustering algorithms: their application to gene expression data, Bioinformatics and Biology insights 10:BBI–S38316

Pesquita C (2017) Semantic similarity in the gene ontology, in: The gene ontology handbook, Humana Press, New York, NY, pp. 161–173

Resnik P, Using information content to evaluate semantic similarity in a taxonomy, arXiv preprint cmp-lg/9511007

Sharan R, Shamir R (2000) Click: a clustering algorithm with applications to gene expression analysis. In: Proc Int Conf Intell Syst Mol Biol, Vol. 8:16

Sheng W, Tucker A, Liu X (2010) A niching genetic *k*-means algorithm and its applications to gene expression data, Soft Comput. 14(1):9–19. http://dx.doi.org/10.1007/s00500-008-0386-9

Speer N, Spieth C, Zell A (2004) A memetic clustering algorithm for the functional partition of genes based on the gene ontology, in: Proceedings of the 2004 IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology, CIBCB 2004, La Jolla, CA, USA, pp. 252–259. http://dx.doi.org/10.1109/CIBCB.2004.1393961

Srivastava S, Zhang L, Jin R, Chan C, A novel method incorporating gene ontology information for unsupervised clustering and feature selection, PloS one 3(12)

Verbanck M, Lê S, Pagès J (2013) A new unsupervised gene clustering algorithm based on the integration of biological knowledge into expression data, BMC Bioinformatics 14:42. http://dx.doi.org/10.1186/1471-2105-14-42

Wu F (2008) Genetic weighted k-means algorithm for clustering large-scale gene expression data, BMC Bioinformatics 9(S-6). http://dx.doi.org/10.1186/1471-2105-9-S6-S12

Xu X, Lu Y, Tung AK, Wang W (2006) Mining shifting-and-scaling co-regulation patterns on gene expression profiles, in: 22nd International Conference on Data Engineering (ICDE'06), IEEE, pp. 89–89

Zhou X, Sun H, Wang D-P, Zhang Y, Zhou Y (2010) Analysis of gene expression data based on density and biological knowledge, in: 2010 Fifth International Conference on Frontier of Computer Science and Technology, IEEE, pp. 448–453