Computers in Biology and Medicine 159 (2023) 106939

Contents lists available at ScienceDirect

Computers in Biology and Medicine

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

Clustering ensemble in scRNA-seq data analysis: methods, applications and challenges

Xiner Nie [a,b], Dan Qin [c], Xinyi Zhou [b], Hongrui Duo [b], Youjin Hao [b], Bo Li [b,**], Guizhao Liang [a,*]

[a] Key Laboratory of Biorheological Science and Technology, Ministry of Education, Bioengineering College, Chongqing University, Chongqing, 400044, China
[b] College of Life Sciences, Chongqing Normal University, Chongqing, 400044, PR China
[c] Department of Biology, College of Science, Northeastern University, Boston, MA, 02115, USA

ARTICLE INFO

Keywords:
Single-cell RNA sequencing
Clustering ensemble
Dimensionality reduction
Partitioning-based clustering
Hypergraph-based strategy

ABSTRACT

With the rapid development of single-cell RNA-sequencing techniques, various computational methods and tools were proposed to analyze these high-throughput data, which led to an accelerated reveal of potential biological information. As one of the core steps of single-cell transcriptome data analysis, clustering plays a crucial role in identifying cell types and interpreting cellular heterogeneity. However, the results generated by different clustering methods showed distinguishing, and those unstable partitions can affect the accuracy of the analysis to a certain extent. To overcome this challenge and obtain more accurate results, currently clustering ensemble is frequently applied to cluster analysis of single-cell transcriptome datasets, and the results generated by all clustering ensembles are nearly more reliable than those from most of the single clustering partitions. In this review, we summarize applications and challenges of the clustering ensemble method in single-cell transcriptome data analysis, and provide constructive thoughts and references for researchers in this field.

1. Introduction

Single-cell RNA sequencing (scRNA-seq) is a high-throughput experimental technique that uses RNA sequencing to quantify gene expression profiles of specific cell populations at the single-cell level [1–3]. In contrast to traditional RNA sequencing, scRNA-seq technology can depict the unique gene expression patterns of individual cells in tissues and cell suspensions, reflecting the cellular heterogeneity of the population [4]. Currently scRNA-seq technology is widely used in many life sciences areas, particularly in identifying cell types [5], tumor heterogeneity [6], cellular immune microenvironment [7], cell lineage analysis [8,9] and spatial reprogramming [10]. In recent years, a large amount of single-cell transcriptome data has been accumulated due to the continuous development and popularity of single-cell sequencing technologies. At the same time, computational methods for mining the potential information in single-cell transcriptome data are also attracting more and more attention [11–13]. In 2019, Malte & Fabian summarized the classical single-cell analysis workflow which mainly contains data preprocessing and downstream analysis. Similar to other omics [14,15], the data preprocessing for scRNA-seqing is also subdivided into quality control, normalization [16,17], data correction [18], feature selection [19,20] and dimensionality reduction [21]. Previous studies confirmed the growing role of data analysis techniques in uncovering biological knowledge or information underlying single-cell transcriptional datasets [22]. Currently, many different bioinformatics algorithms, software, or tools are used in various steps of single-cell transcriptome data analysis [23].

As an unsupervised machine learning method, clustering is often applied upstream of single-cell transcriptome data analysis [24]. One of the core analyses of single-cell transcriptome data is to cluster individual cells into different cell clusters for identifying cell types, subtypes (or subgroups), and inferring cell lineages based on the relationship between these cell clusters [25,26]. Clustering methods can be used precisely to help identify cell types by comparing single-cell gene expression profiles without prior knowledge. These data-driven methods have also been utilized in many large bioinformatics projects or atlas [27–29]. Clustering plays a crucial role in the analysis and mining of single-cell transcriptome data, and the results can primarily affect downstream analysis [30].

At present, many traditional clustering algorithms applied to bulk-tissue transcriptome data have been introduced to the field of single-cell transcriptomics, such as hierarchical clustering, *k*-means, self-

* Corresponding author.
** Corresponding author.
E-mail addresses: libcell@cqnu.edu.cn (B. Li), gzliang@cqu.edu.cn (G. Liang).

https://doi.org/10.1016/j.compbiomed.2023.106939
Received 20 January 2023; Received in revised form 31 March 2023; Accepted 14 April 2023
Available online 15 April 2023

organized maps (SOM), graph-based clustering, model-based clustering [31,32] and density-based clustering [33]. Meanwhile, many single-cell clustering algorithms based on machine learning and deep learning have become popular and occur frequently in the field of single-cell transcriptomics analysis [25]. Researchers have combined these clustering algorithms with dimensionality reduction strategies for cell type identification or cell subtype determination, which has extensively promoted the development of single-cell transcriptomics. Nevertheless, problems still need to be solved in this field. In particular, these clustering algorithms still have unstable clustering effects on single-cell transcriptome datasets with high dropouts or noise [34,35]. For instance, as one of the most popular unsupervised clustering algorithms with low complexity, *k*-means is more suitable for single-cell transcriptome data which are concentrated in space. However, *k*-means clustering algorithm fails to give good results when the dataset contains outliers, which makes *k*-means unsuitable for non-convex datasets. Clustering algorithmic choice is driven by factors, such as random initializations, choice of hyper-parameters and features used in clustering, which results in low robustness of each clustering partition. Robustness is also one of the most common pitfalls in clustering single-cell transcriptome data today [35].

Ensemble learning is a powerful machine learning paradigm that has emerged recently [36,37]. It aims to combine the bias and/or variance of multiple "weak learners" to form a "strong learner" (or "ensemble model") to acquire the better predictive performance [37,38]. Nowadays, the ensemble learning strategy has been used in many sections of single-cell transcriptome data analysis, such as dropouts imputation [39–42], dimensionality reduction [43–46], cell annotation [47], trajectory inference [48,49] and cell deconvolution [50,51]. Building a clustering ensemble framework that can generate and integrate different base clustering partitions is considered one of the most popular and effective strategies in single-cell transcriptomics to obtain a more accurate and stable clustering result [52,53].

Herein, we systematically reviewed and summarized the clustering ensemble strategies currently applied to single-cell transcriptome data analysis through a large amount of literature research and conducted a comprehensive introduction from three aspects: principle, application, and challenges, which aimed to provide reference experience and guidelines for researchers in this field.

2. Clustering ensemble

A typical single-cell transcriptome dataset (*i.e.*, single-cell gene expression profile) usually records the expression level data of many genes in hundreds of cells. Assuming that a given single-cell transcription dataset D contains n cells, the expression levels of m genes in each cell are quantified. The matrix below shows the complete single-cell gene expression profile, with each row and column representing a cell and a gene feature, respectively.

$$D = \begin{bmatrix} g_{11} & g_{12} & \cdots & g_{1n} \\ g_{21} & g_{22} & \cdots & g_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ g_{m1} & g_{m2} & \cdots & g_{mn} \end{bmatrix}$$

In scRNA-seq data analysis, all cells are often divided into k non-overlapping clusters using a specific clustering algorithm to determine the types and characteristics of cells (*i.e.*, columns in gene expression matrix).

The clustering ensemble, also known as consensus ensemble or clustering aggregation, aims to recover the natural groups of cells with the knowledge reuse of labels from a set of partitions [54,55]. The issue to be solved by clustering ensemble is to integrate multiple base clustering results (*i.e.*, base partitions) into a final clustering result, as shown in Fig. 1. So far, many different clustering ensemble methods or frameworks have been proposed by researchers and applied to the field of single-cell transcriptomics, such as cola [56], scEFCS [57], SC3 [58] and SHARP [59]. Although these several kinds of frameworks focus on distinct science issues and have different emphases, the key principles and core issues of how to generate and integrate a large number of clustering partitions/models are consistent in order that researchers can get a stable and accurate result eventually.

3. Generation of multiple clustering partitions

How to generate a large number of different clustering partitions/models is the first core issue for all clustering ensemble strategies. From this perspective, the clustering ensemble methods and frameworks currently applied in single-cell transcriptomics could be grouped into three categories: gene-oriented methods, cell-oriented methods, and different algorithm-oriented strategies (as shown in Fig. 2). Meanwhile, both advantages and disadvantages of them are presented in Table 1.

3.1. Gene-oriented approaches

One way to generate multiple clustering results is to form multiple subsets of data by disassembling or combining gene features of the dataset and then performing clustering on this basis to obtain multiple clustering results. These subsets contain the same cells as the original dataset, but the difference is that only a part of the gene features is included in a given sub-dataset. Gene-oriented (*i.e.*, feature-oriented) approach consists of two main categories, random sampling and gene ranking.

3.1.1. Random sampling

The most commonly used method to generate base clustering partitions is to randomly select a certain number of gene features from the gene expression matrix, aiming at forming a subset of data containing part of the original gene information. By repeatedly sampling many times, a series of sub-datasets will be generated, and on this basis, subsequent cluster analysis will be performed. Considering the effect on the subsequent analysis, here, the gene expression matrix preprocessed by dimensionality reduction is usually used so that the essential genes remain in the dataset. For the Cola method, it obtains preprocessed data by feature selection before generating multiple clustering partitions [56], and in the sampling process, it provides an option to choose whether to sample genes or cells according to the user's needs. Similar to a random sampling of genes, sampling of cells is also a significant strategy, which will be discussed in section 3.2.

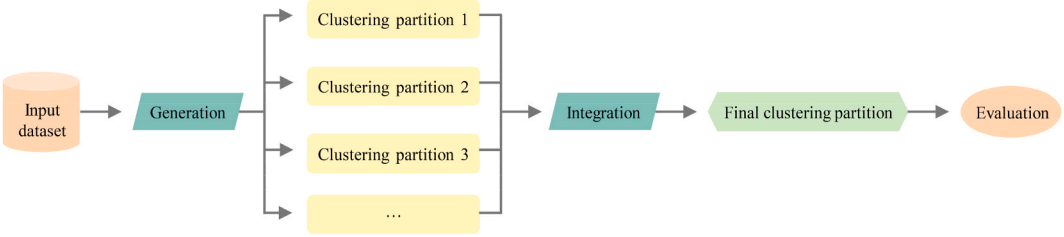

Fig. 1. The generic framework for clustering ensemble.

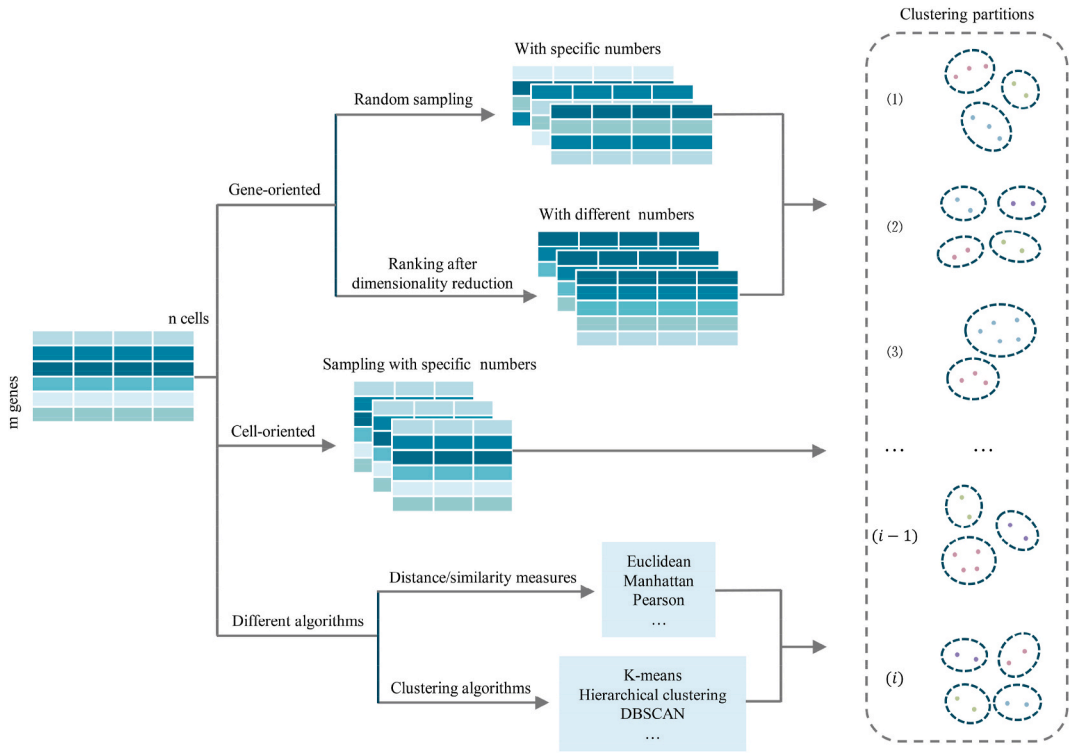

Fig. 2. Three strategies for generating multiple clustering partitions.

Table 1
Advantages and disadvantages of three strategies used to generate multiple clustering partitions.

Strategies	Advantages	Disadvantages
Gene-oriented approaches	For high-dimensional datasets, it is useful to retain a subset of genes and eliminate a large number of unnecessary features, which can significantly reduce computational consumption and facilitate the visualization of results.	Only a part of genes observed in each base clustering partition which may leads to quite important information loss, so that the accuracy of the final clustering result is reduced.
Cell-oriented approaches	For individual cells, all gene features are involved in the subsequent clustering analysis with no information missing.	Unlike gene-oriented approaches, sub-datasets produced by cell-oriented approaches are more different from the original data. To obtain a reliable and accurate result, it should increase the iterations which may take up a lot of computational resources.
Algorithm-oriented approaches	Compared with the original dataset, the input datasets have the same number of cells and genes. It is often used to investigate the influence of factors related to clustering, which is irrelevant to the original data itself.	Compared with gene-oriented approaches, each input dataset has more dimensions and thus needs to consume more computational resources.

3.1.2. *Gene ranking*

According to a pre-defined criterion, sorting gene features to obtain a subset of single-cell expression profile data is another commonly used strategy. The subset of data containing partial gene features can also be considered as the product of the raw data after dimensionality reduction. After the raw data's dimensionality reduction, the remaining or generated features are sorted in descending order of variability. Since each feature contributes unequally to the final result, the top-ranked features with high values often tend to be reserved as low-dimensional input data for subsequent analysis. For high-dimensional single-cell transcriptome data, dimensionality reduction allows information masked in the high dimension to be revealed in the lower dimension. However, dimensionality reduction also brings some new troubles. Due to the deletion of partial information structure, dimensionality reduction will also increase the risk of losing important information from datasets. To fill this gap, combining multiple dimensionality reduction methods is often recommended as an optimized solution [33]. In practice, using different dimensionality reduction methods can generate multiple sub-datasets. The clustering results obtained from these sub-datasets are eventually integrated into a final clustering result, compensating for the drawback of using a particular dimensionality reduction method alone. Next, the description will be expanded from different dimensionality reduction methods.

As one of the distinctive dimensionality reduction methods, feature selection eliminates the less variable genes and retaining the more variable genes for clustering analysis. It firstly calculates values that reflect the degree of variation for each feature gene, such as standard deviation, variance, and coefficient of variation. Then, the genes are sorted in descending order based on these index values. The top-ranked ones are continuously taken from the sorted genes according to a specific ratio to obtain multiple sub-datasets. Cui et al. quantified the standard deviation for each gene and ranked the genes from largest to smallest according to the variance value. For the sorted matrix, a set of top-ranked gene subsets were taken to form the sub-matrix for subsequent analysis [60]. On the choice of the number of highly variable genes, the author also puts forward corresponding suggestions: 50, 100, 150, 200 and 250 are sequentially recommended as the numbers of highly variable genes when the dimension of the dataset is less than 8000, while 200, 400, 600, 800 and 1000 are sequentially recommended as the numbers of highly variable genes when the data dimension exceeds 12000. Bian et al. screened 5000 significant genes in advance by establishing a non-negative kernel autoencoder. Subsequently, four unsupervised feature selection methods were added to further remove less variable genes, including Low-Variance, Laplacian Score, SPEC, and

MCFS. Therefore, in the scEFCS framework, a total of 5 feature subsets are used for subsequent analysis [57].

Feature extraction is one of the primary dimensionality reduction methods for high-dimensional single-cell expression profiles. Unlike feature selection, classical feature extraction techniques (*e.g.*, principal component analysis) generates "new components" that combine multiple gene features. The biological meaning underlying these "new components" (*i.e.*, "principal components") has yet to discover. However, this doesn't mean these "new principal components" are meaningless absolutely. In practical research, to fully use all of the variation information, researchers usually choose different sub-dateset composed of several principal components to represent the original dataset. The ANMF-CE method proposed by Zhu et al. generates multiple base clustering partitions by selecting a set of new dimensions after feature extraction [61]. It is noted that, the feature extraction method used here is the adaptive total-variation for non-negative matrix factorization algorithm (ATV-NMF), which can handle missing values, noise, and arbitrarily shaped clusters.

Drastically different from other dimensionality reduction methods, random projection (RP) does not need to compute the distance or similarity between cells or "new components", which reduces the running time and cost and ensures with high probability that the low-dimensional data has similar variation information as the high-dimensional data. SHARP is a representative clustering ensemble framework based on the random RP method. It continuously performs RP on the matrix to obtain multiple low dimensional datasets, which will replace the original dataset for hierarchical clustering to obtain multiple different clustering partitions [59].

Autoencoder is one of the most prevalent neural network models, and it works by extracting linear or nonlinear features from raw data. Many types of autoencoders (AEs) have been effectively used to capture low-dimensional information from high-dimensional scRNA-Seq data. For example, the scIAE method proposed by Yin et al. divided the single-cell expression profile into the training set and test set, and then performed RP individually to yield multiple subsets of the original data [62].

3.2. Cell-oriented approaches

Another strategy is to disassemble or combine single-cell data to form multiple sub-datasets (*i.e.*, subspace) and analyze them to generate multi-clustering partitions. It should be emphasized that the sub-datasets produced by the strategy described above differ from the original data matrix significantly. The gene features in these sub-matrices are the same, and the sole distinction among them is cell samples (Fig. 2). The clustering ensemble workflow proposed by Risso et al., allows users to choose to sample the cells randomly [63]. The clustering ensemble framework Cola proposed by Gu et al. is to continuously repeat the process of random sampling and re-clustering of genes or cells finally forms a stable clustering result [56]. Compared with the method of sampling genes, the clustering results of sub-datasets obtained by sampling cells are more credible [56].

Although the random sampling method can quickly generate a large number of sub-datasets, it is difficult to build a sub-dataset completely standing for the original dataset. In order to obtain a stable result and increase the number of sampling iterations as much as possible, a random stratified sampling strategy for the original single-cell expression profile data set can also be used. Unfortunately, the above methods often lead to excessive computational consumption when processing large scRNA-seq datasets [64]. Subsequently, Ringeling and Canzar proposed a new approach (*i.e.*, RC approach) to choose cells. It first extracts a portion of cells uniformly at random, and then clustering those selected cells using *k*-means to obtain cells that can fully represent the entire original dataset [64]. Hu et al. also proposed a similar strategy. The difference from the RC approach is that the center of each cluster by *k*-means is regarded as the representative cell in Hu's method [65].

For the sub-datasets formed by randomly sampling a part of cells from the original dataset, it exists a risk of partial information overlap. To address this, the SHARP algorithm provides an alternative way of sampling cells by dividing the large dataset into multiple blocks equally in advance, which can maximize the use of computational resources, avoids memory overflow, and minimizes the impact of sample imbalance [59].

3.3. Algorithm-oriented approaches

Both gene-oriented and cell-oriented approaches to generate sub-datasets of single-cell expression profiles could lose part of the original data information to varying degrees. In contrast to the first two approaches, using different options including distance metrics and clustering algorithms for the original dataset can incorporate all the matrix's information into the calculation, but also generate a set of clustering results in the context of different algorithms. When performing cluster analysis of single-cell transcriptome data, there are two non-unique requirements for algorithms: one is the algorithm for measuring the distance/similarity between cells, and the other is the algorithm for grouping cells.

3.3.1. Distance/similarity measures

The partitioning clustering algorithm is crucial to measure the distance or similarity between cells [66]. There are many common indicators to measure the distance between cells, such as Euclidean distance, Manhattan distance, Mahalanobis distance and Minkowski distance. In addition, the distance between cells can also be indirectly quantified by the similarity between cells. The higher the similarity between cells, the closer the distance is. Therefore, some commonly used similarity measures (*e.g.*, Pearson and Spearman correlation coefficient) are also taken into account. Multiple cell-to-cell covariance matrices can be generated by employing a series of different distance or similarity metrics. Correspondingly, the same number of multiple clustering partitions are also produced using a specific clustering algorithm (Fig. 2). Since each distance or similarity metric focuses on different characteristics and emphasis of the input data matrix, the final clustering result by integrating multiple partitions generated by different algorithms should be more robust and credible.

Based on this strategy, Kiselev et al. proposed a consensus clustering method for single-cell data analysis named single-cell consensus clustering (SC3) [58]. For SC3 framework, it provides three popular distance and similarity indicators including Euclidean distance, Pearson and Spearman correlation coefficient, to obtain more stable clustering partitions [58]. In addition, the framework proposed by Wang et al. introduced four distance or similarity metrics. Apart from the three ones mentioned above, consensus distance was brought in Ref. [62].

3.3.2. Clustering algorithms

In the process of clustering, it has always been a rigorous challenge for researchers to accurately group cells into distinct clusters. Using various clustering algorithms for single-cell transcriptome data may usually lead to inconsistent clustering partitions, the so-called preference of clustering algorithms. Application of different clustering algorithms on a given dataset can generate a large number of base clustering partitions, on which the integration of these partitions will produce a more reliable clustering result.

Taking scEFCS as an example, it offered nine built-in clustering algorithms or software packages commonly used in single-cell transcriptomics, including SC3, Monocle, CIDR, pcaReduce, Rphenograph, Seurat, SHARP, SINCERA, and RaceID [57]. For the ECBN framework, four popular clustering algorithms or packages suitable for normalized datasets are embedded, containing CIDR, Seurat, SC3 and t-SNE + *k*-means [67]. The GeoWaVe proposed by Burton et al. supports five popular clustering algorithms [68], including FlowSOM [69], PHATE with *k*-means clustering [70], SPADE [71], Phenograph [72] and PARC

[73].

3.4. Other approaches

In addition to the three methods mentioned above, there are other strategies applied in integrated clustering, such as using different parameter choices. Also, the Landmark-based spectral clustering (LCS) ensemble algorithm is one of the typical representatives, which produces different clustering partitions by varying the number of selected cells and kernel bandwidths to achieve clustering ensemble [64].

4. Aggregation of multiple clustering partitions

4.1. Voting strategy

A widely used voting strategy is considered the most straightforward and effective solution to obtain a final result from many base clustering partitions, as shown in Fig. 3. As an easy-to-use method, the principle of majority rule (*i.e.*, voting) makes the final clustering result stable and representative of most base clustering partitions. It is worth noting that a set of base clustering partitions are required as input to produce a more accurate clustering result, which may lead to high computational costs. Compared to other strategies, the advantages and disadvantages of voting strategy is exhibited in Table 2.

4.2. Hypergraph-based strategy

The graph is a diagram representing a system of connections and a topological structure demonstrating the relationship between data points. Generally, a graph is usually represented by *G (V, E)*, where *V* and *E* represent the included vertices and edges, respectively. A hypergraph is a generalization of a graph in which edges can connect any number of vertices and is usually represented by *H (V, E)*. Unlike a graph, a hyperedge represents the relationship connecting more than two vertices.

Considering the characteristics of hypergraphs, data clustering partitions are more suitable to be represented by hypergraphs (the advantages of Hypergraph-based method as shown in Table 2), and the cluster labels can be converted into appropriate hyperedges. The clustering partition p_i constructs a binary matrix with rows and columns representing cells (vertices) and clusters (hyperedges). The entries in the matrix are denoted by v_{jk}, representing the value of the *j*-th row under the *k*-th hypergraph. The labels of the cells should satisfy the following rules:

$$v_{ik} = \begin{cases} 1\ , \text{ the } i^{th}\text{cell} \in \text{the } k^{th} cluster \\ 0\ , \text{ the } i^{th}\text{cell} \notin \text{the } k^{th} cluster \end{cases}$$

In the binary matrix, Hyperedge elements are assigned the value of 1 if the cell belongs to the specific cluster and a value of 0 otherwise. Based on this, each cluster of the clustering result is mapped to a hyperedge, and the entire clustering result is mapped to a hypergraph. Finally, the hypergraphs transformed from a set of clustering partitions can be combined into a large hypergraph to realize the integration of the clustering partitions. There are three classic graph-based integration

Table 2
Advantages and disadvantages of two strategies used to combine multiple base partitions.

Strategies	Advantages	Disadvantages
Voting	Among all the base clustering partitions, voting selects the one that can best represent the whole base clustering partitions as the final result, which is the most direct and effective approach.	The more base clustering partitions that participate in voting, the more accurate the final clustering result will be. Therefore, it needs to consume a large amount of computational resources.
Hypergraph-based method	Hypergraph-based strategy allows for a more flexible representation of relationships between data points, which is useful for integrating multiple base clustering partitions.	As the number of vertices and edges increase, Hypergraph-based strategy can become too complex to calculate. Therefore, it is difficult to apply hypergraph-based clustering in a large-scale data set due to the computational complexity.

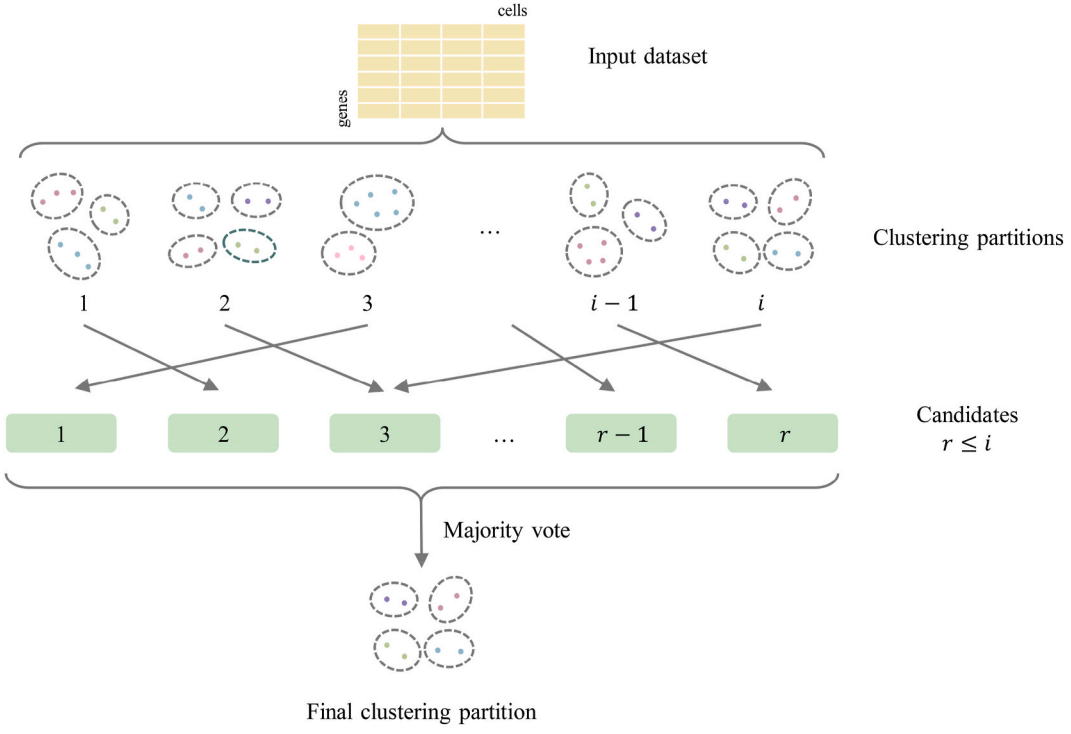

Fig. 3. The voting-based aggregation strategy.

methods, namely Cluster-based Similarity Partitioning Algorithm (CSPA, Fig. 4), HyperGraph-Partitioning Algorithm (HGPA, Fig. 5) and Meta-CLustering Algorithm (MCLA, Fig. 5) [54]. These three methods are all the practicalization of the concept of hypergraph, but the specific usage is different.

4.2.1. Cluster-based similarity partitioning algorithm (CSPA)

In the CSPA method, for each single clustering result, the similarity is defined as 1 if two cells belong to the same cluster (*i.e.*, same cell label) and 0 if the opposite is true. Therefore, each base clustering partition can be converted into a cell-to-cell binary similarity matrix. When the cells in the clustering partition are different, the cells in the binary similarity matrix are the union of all cells involved. If a cell is not included in a clustering result, all elements related to the cell are defined as 0. Then, these similarity matrices are averaged to form a consistency matrix C, where C_{ij} represents the probability that cell i and cell j are clustered in a cluster. Finally, an appropriate clustering algorithm is selected for matrix C and then clustering analysis is performed to obtain a stable and accurate clustering result (Fig. 4). Cui et al. integrated the cell-to-cell similarity matrix, and identified accurate cell subgroups consequently using the idea of CSPA [60]. In order to strengthen the intra-cluster similarity and weaken the interference between clusters, the framework SCENA adopted the local affinity network method to achieve significant distinction between clusters by enhancing the affinities of cells sharing KNN [60]. The SCENA approach only requires the integration of multiple cell-to-cell similarity matrices to obtain the consistency matrix and then generates the final clustering result without generating multiple base clustering results.

4.2.2. Hypergraph-Partitioning Algorithm (HGPA)

Different from the CSPA method, the HGPA method chooses to repartition the clusters of each clustering result directly. By cutting the minimum number of hyperedges, the large hypergraph is divided into multiple regions of equal size. HGPA needs to traverse all the cuts as much as possible. The partition formed by a cut can be judged as the final partitioning result when it satisfies the following two conditions: (i) the number of cut hyperedges is minimum. (ii) The clusters formed by the cuts should have approximately the same size. All hyperedges and vertices have the same weight here. It is worth noting that since the final partitioning requires the partitions to be as comparable in size as possible, HGPA may not be suitable when the classes within the dataset are highly imbalanced.

The relationship between cells and different clusters only includes two cases, 0 and 1, which are idealized for the actual data structure. On the contrary, the relationship between cells and different clusters is measurable and may not be 0 with a high probability. As a result, too many 0 in the hypergraph structure brings hidden dangers to the subsequent analysis. To solve this problem, Lu et al. proposed graph-based linking measure (GLM) which redefined the relationship between clusters based on hypergraph and performs k-means on the newly generated graph-based linking matrix to obtain a final clustering result [34].

4.2.3. Meta-CLustering algorithm (MCLA)

The MCLA method employs reclustering of clusters from multiple base clustering partitions to achieve the final determination of cell labels. Each hyperedge represents a cluster in the large hypergraph. When using the MCLA strategy for clustering ensemble, the hyperedges are grouped and then refolded, and eventually, the cells can be assigned to the closest cluster. As shown in Fig. 5, it can be divided into four steps:

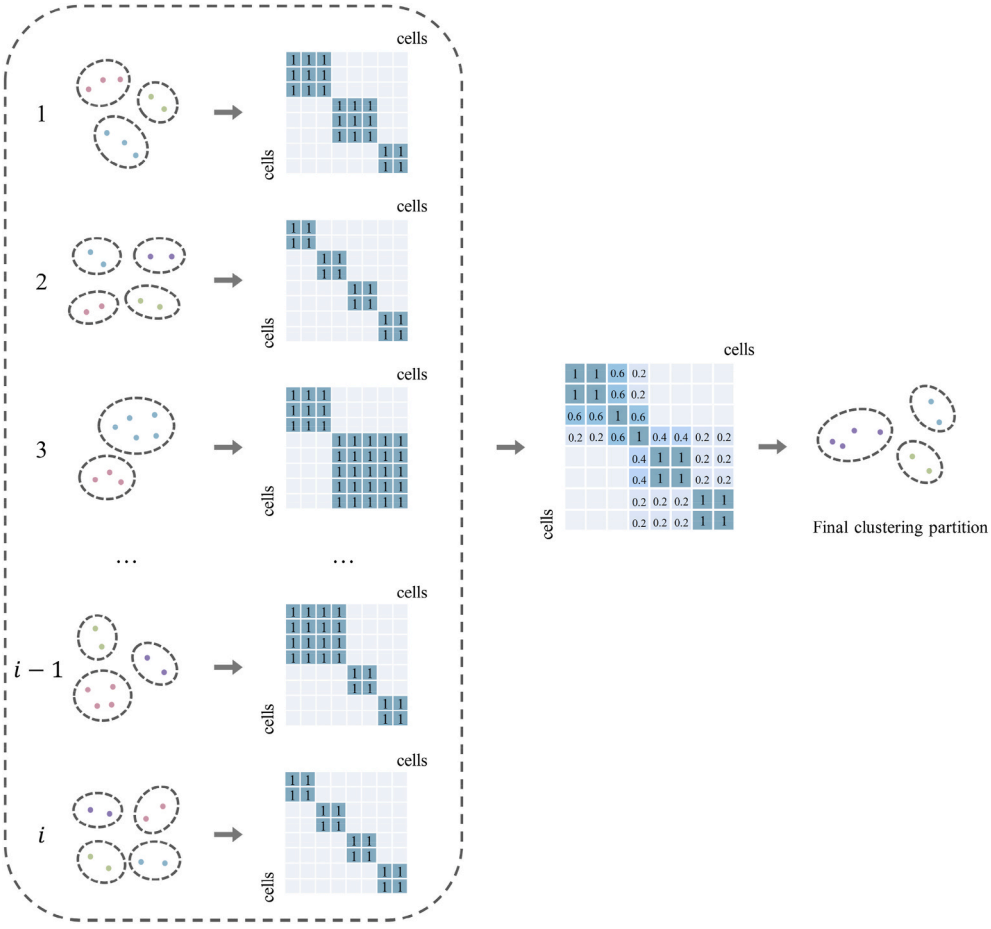

Fig. 4. The CSPA-based aggregation strategy.

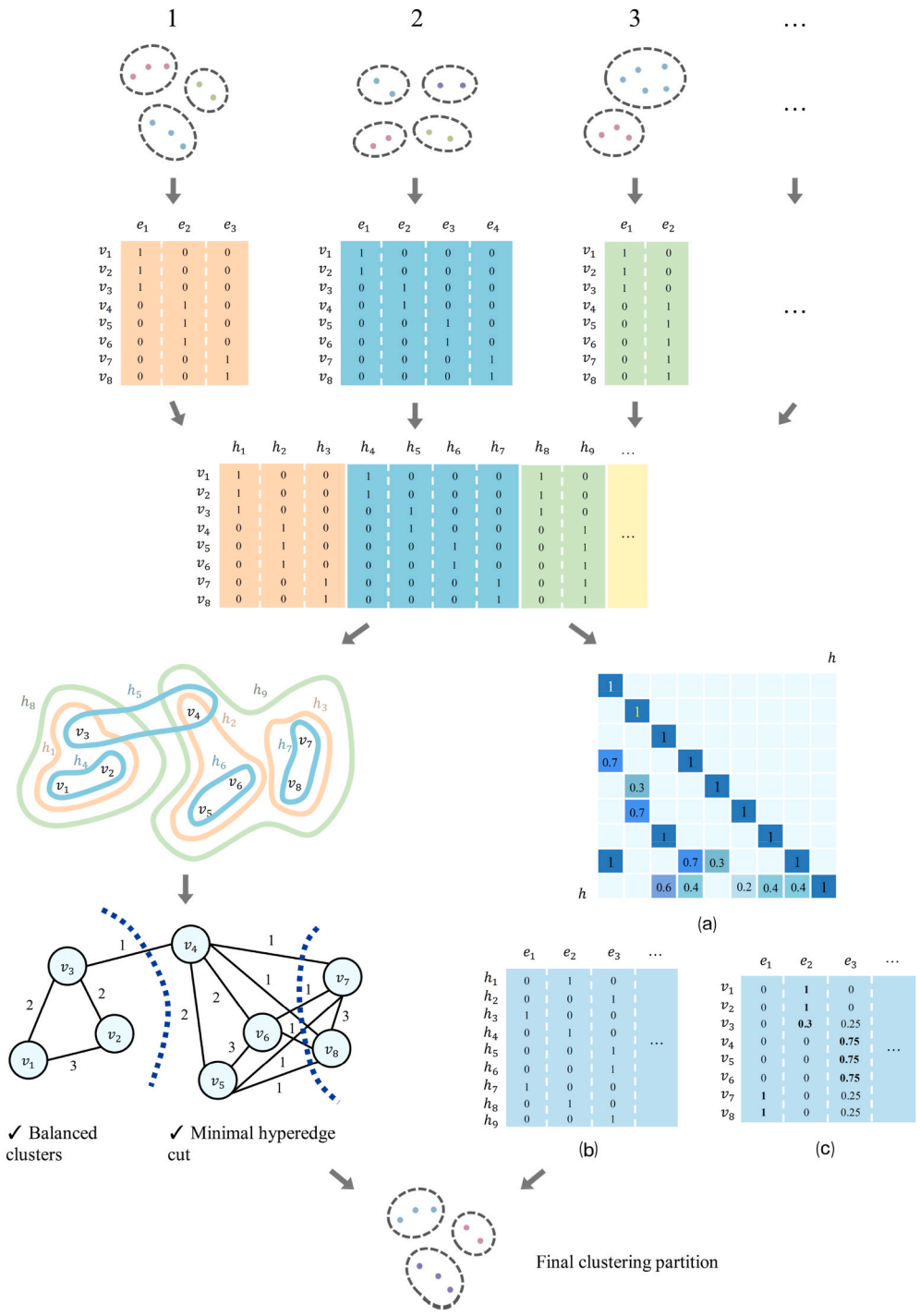

Fig. 5. The HGPA-based (left) and MCLA (right) aggregation strategy.

(i) Construction of meta-graph. All hyperedges of large hypergraph are regarded as vertices of the meta-graph, and the weights of edges are proportional to the similarity between two vertices. Therefore, the binary Jaccard measure is introduced to calculate the weight, as follows:

$$\omega_{a,b} = \frac{h_a \cap h_b}{h_a \cup h_b}$$

where $\omega_{a,b}$ stands for the weights of the edges between vertex a and vertex b, and h_a and h_b refers to the indicator vectors corresponding to vertex a and vertex b, respectively. (ii) They grouped the meta-graph into k balanced meta-clusters through a graph-based clustering method. Each meta-cluster contains a set of hyperedges. (iii) By averaging the hyperedges in each meta-cluster, the meta-clusters are folded into meta-hyperedges. The larger the value, the stronger the association. (iv) The cells are divided into the meta-cluster with the most significant value.

Traditional clustering ensemble methods usually treat each base clustering partition equally. Theoretically, the equal-weight treatment may need to be consistent with the actual situation due to various factors, including datasets containing different cells and genes and algorithms used. Different weights are introduced to examine the contribution of each clustering partition to make the final results more stable and accurate. Wan et al. proposed Weighted-based meta clustering (wMetaC) for integrating clustering partitions generated by

random projection, and Similarity-based meta clustering (sMetaC) is used to integrate clustering partitions from different data blocks [59]. Besides, the scEFCS framework also adopts the wMetaC strategy when integrating clustering partitions generated by a combination of five feature selection methods and nine clustering algorithms [57]. Thus, assigning different weights to each base clustering partition in the design of future clustering ensemble frameworks is becoming a significant trend and initiative to improve the stability of clustering further.

5. Summary and conclusions

Clustering ensemble utilizes ensemble learning techniques to obtain a more robust clustering result by merging multiple clustering partitions of a dataset. Moreover, for clustering ensemble there is no strict restriction on the value of k (*i.e.*, the number of the cluster) for each base clustering partition. Currently, ensemble clustering is considered to surpass the performance of a single clustering algorithm in many aspects, such as robustness, stability, and consistency (definitions of these indicators presented in Table S1). However, like a coin has two sides, there are still some challenges in applications of Clustering ensemble. For example, each base clustering result interprets the data from different perspectives, and the heterogeneity between the base clustering results needs further investigation. In addition, the consensus function needs further optimization, and the performance of clustering ensemble algorithms, particularly the running speed, also needs to be enhanced.

This review systematically reviewed and summarized the current status and progress of clustering ensemble frameworks. The clustering ensemble method is applied to single-cell transcriptome data analysis from both generation and integration of base clustering partitions, intending to provide guidelines for researchers in this field (*e.g.*, algorithm developers). On the one hand, a series of data subsets can usually be generated using gene-oriented approaches, cell-oriented approaches or different algorithms to generate multiple base clustering results. On the other hand, to change the clustering results from many to one, the voting-based strategy or the graph-based strategy is usually used. Strictly speaking, there are both advantages and disadvantages for these strategies (as shown in Table 1 and Table 2), and researchers should choose an appropriate strategy for analysis according to the actual situation.

For a particular single-cell transcriptome dataset, a single clustering produces only a clustering partition, while clustering ensemble can generate multiple clustering partitions, which are equivalent to performing multiple single clustering) and then integrate them. Although the clustering ensemble algorithms are more computationally time-consuming than the single clustering algorithms, it is able to obtain significantly better clustering partitions, which is crucial for further identification of cell type, cell heterogeneity and cellular immune microenvironment, especially cell lineage analysis. It is foreseeable that the continuous development and optimization of integrated clustering will contribute greatly to the improvement of single-cell transcriptome data analysis methods and further promote the prosperity of single-cell theory and technologies. In general, one of the primary tasks is to appropriately partition the entire cell population into various subpopulations and accurately determine these cell types in single-cell transcriptomics analysis. To acquire more precise cell type information, obtaining stable single-cell clustering outcomes from scRNA-seq datasets is extremely crucial. Therefore, this review will provide a modern outlook on the reliable clustering of single-cell transcriptomics data, and deliver some benchmark examples and useful advice for researchers in related fields.

Author contributions

Conceptualization & Data Curation: Xiner Nie, Guizhao Liang and Bo Li; Writing-Original Draft: Xiner Nie and Dan Qin; Writing-Reviewing and Editing: Youjin Hao, Hongrui Duo, Xinyi Zhou; Resources, & Visualization: Xiner Nie, Guizhao Liang; Supervision & Funding Acquisition: Guizhao Liang and Bo Li. All authors reviewed the manuscript.

Declaration of competing interest

None Declared.

Acknowledgement

This project was supported by National Natural Science Foundation of China (*No*. 32172196 and 31771975), Science and Technology Research Program of Chongqing Municipal Education Commission (*No*. KJQN202100538), Chongqing Primary and Secondary School Innovative Talent Training Project (*No*. CY220506), and Open Fund of Yunnan Key Laboratory of Plant Reproductive Adaptation and Evolutionary Ecology, Yunnan University.

List of Abbreviations

Abbreviation Terms

scRNA-seq	Single-cell RNA sequencing
SOM	Self-organized maps
PCA	Principal component analysis
RP	Random projection
SC3	Single-cell consensus clustering
LCS	Landmark-based spectral clustering
CSPA	Cluster-based similarity partitioning algorithm
HGPA	HyperGraph-Partitioning algorithm
MCLA	Meta-CLustering algorithm
GLM	Graph-based linking measure
wMetaC	Weighted-based meta clustering
sMetaC	Similarity-based meta clustering

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.compbiomed.2023.106939.

References

[1] E. Mereu, A. Lafzi, C. Moutinho, C. Ziegenhain, D.J. McCarthy, A. Alvarez-Varela, E. Batlle, Sagar, D. Grun, J.K. Lau, S.C. Boutet, C. Sanada, A. Ooi, R.C. Jones, K. Kaihara, C. Brampton, Y. Talaga, Y. Sasagawa, K. Tanaka, T. Hayashi, C. Braeuning, C. Fischer, S. Sauer, T. Trefzer, C. Conrad, X. Adiconis, L.T. Nguyen, A. Regev, J.Z. Levin, S. Parekh, A. Janjic, L.E. Wange, J.W. Bagnoli, W. Enard, M. Gut, R. Sandberg, I. Nikaido, I. Gut, O. Stegle, H. Heyn, Benchmarking single-cell RNA-sequencing protocols for cell atlas projects, Nat. Biotechnol. 38 (2020) 747–755.

[2] D. Ofengeim, N. Giagtzoglou, D. Huh, C. Zou, J. Yuan, Single-cell RNA sequencing: unraveling the brain one cell at a time, Trends Mol. Med. 23 (2017) 563–576.

[3] Aleksandra A. Kolodziejczyk, J.K. Kim, V. Svensson, John C. Marioni, Sarah A. Teichmann, The technology and biology of single-cell RNA sequencing, Mol. Cell 58 (2015) 610–620.

[4] E. Papalexi, R. Satija, Single-cell RNA sequencing to explore immune cell heterogeneity, Nat. Rev. Immunol. 18 (2018) 35–45.

[5] X. Sun, X. Lin, Z. Li, H. Wu, A comprehensive comparison of supervised and unsupervised methods for cell type identification in single-cell RNA-seq, Briefings Bioinf. 23 (2022).

[6] Y. Zhang, D. Wang, M. Peng, L. Tang, J. Ouyang, F. Xiong, C. Guo, Y. Tang, Y. Zhou, Q. Liao, Single-cell RNA sequencing in cancer research, J. Exp. Clin. Cancer Res. 40 (2021) 1–17.

[7] A.S. Venteicher, I. Tirosh, C. Hebert, K. Yizhak, C. Neftel, M.G. Filbin, V. Hovestadt, L.E. Escalante, M.L. Shaw, C. Rodman, S.M. Gillespie, D. Dionne, C.C. Luo, H. Ravichandran, R. Mylvaganam, C. Mount, M.L. Onozato, B.V. Nahed, H. Wakimoto, W.T. Curry, A.J. Iafrate, M.N. Rivera, M.P. Frosch, T.R. Golub, P. K. Brastianos, G. Getz, A.P. Patel, M. Monje, D.P. Cahill, O. Rozenblatt-Rosen, D. N. Louis, B.E. Bernstein, A. Regev, M.L. Suva, Decoupling genetics, lineages, and microenvironment in IDH-mutant gliomas by single-cell RNA-seq, Science 355 (2017).

[8] B. Treutlein, D.G. Brownfield, A.R. Wu, N.F. Neff, G.L. Mantalas, F.H. Espinoza, T. J. Desai, M.A. Krasnow, S.R. Quake, Reconstructing lineage hierarchies of the distal lung epithelium using single-cell RNA-seq, Nature 509 (2014) 371–375.

[9] Z. Wang, Z. Li, K. Zhou, C. Wang, L. Jiang, L. Zhang, Y. Yang, W. Luo, W. Qiao, G. Wang, Y. Ni, S. Dai, T. Guo, G. Ji, M. Xu, Y. Liu, Z. Su, G. Che, W. Li, Deciphering cell lineage specification of human lung adenocarcinoma with single-cell RNA sequencing, Nat. Commun. 12 (2021) 6500.
[10] R.B. Burl, E.A. Rondini, H. Wei, R. Pique-Regi, J.G. Granneman, Deconstructing cold-induced brown adipocyte neogenesis in mice, Elife 11 (2022).
[11] Y. Chen, H. Zhang, X. Sun, Improving the Performance of Single-Cell RNA-Seq Data Mining Based on Relative Expression Orderings, Brief Bioinform, 2022.
[12] B. Hie, J. Peters, S.K. Nyquist, A.K. Shalek, B. Berger, B.D. Bryson, Computational methods for single-cell RNA sequencing, Annual Review of Biomedical Data Science 3 (2020) 339–364.
[13] T.S. Andrews, V.Y. Kiselev, D. McCarthy, M. Hemberg, Tutorial: guidelines for the computational analysis of single-cell RNA sequencing data, Nat. Protoc. 16 (2021) 1–9.
[14] J. Fu, Y. Zhang, Y. Wang, H. Zhang, J. Liu, J. Tang, Q. Yang, H. Sun, W. Qiu, Y. Ma, Z. Li, M. Zheng, F. Zhu, Optimization of metabolomic data processing using NOREVA, Nat. Protoc. 17 (2022) 129–151.
[15] Q. Yang, J. Hong, Y. Li, W. Xue, S. Li, H. Yang, F. Zhu, A novel bioinformatics approach to identify the consistently well-performing normalization strategy for current metabolomic studies, Briefings Bioinf. 21 (2020) 2142–2152.
[16] Q. Yang, Y. Wang, Y. Zhang, F. Li, W. Xia, Y. Zhou, Y. Qiu, H. Li, F. Zhu, NOREVA: enhanced normalization and evaluation of time-course and multi-class metabolomic data, Nucleic Acids Res. 48 (2020) W436–W448.
[17] B. Li, J. Tang, Q. Yang, S. Li, X. Cui, Y. Li, Y. Chen, W. Xue, X. Li, F. Zhu, NOREVA: normalization and evaluation of MS-based metabolomics data, Nucleic Acids Res. 45 (2017) W162–W170.
[18] Q. Yang, B. Li, S. Chen, J. Tang, Y. Li, Y. Li, S. Zhang, C. Shi, Y. Zhang, M. Mou, W. Xue, F. Zhu, MMEASE: online meta-analysis of metabolomic data by enhanced metabolite annotation, marker selection and enrichment analysis, J. Proteonomics 232 (2021), 104023.
[19] J. Tang, M. Mou, Y. Wang, Y. Luo, F. Zhu, MetaFS: performance assessment of biomarker discovery in metaproteomics, Briefings Bioinf. 22 (2021).
[20] Q. Yang, B. Li, J. Tang, X. Cui, Y. Wang, X. Li, J. Hu, Y. Chen, W. Xue, Y. Lou, Y. Qiu, F. Zhu, Consistent gene signature of schizophrenia identified by a novel feature selection strategy from comprehensive sets of transcriptomic data, Briefings Bioinf. 21 (2020) 1058–1068.
[21] M.D. Luecken, F.J. Theis, Current best practices in single-cell RNA-seq analysis: a tutorial, Mol. Syst. Biol. 15 (2019), e8746.
[22] G. Chen, B. Ning, T. Shi, Single-cell RNA-seq technologies and related computational data analysis, Front. Genet. 10 (2019) 317.
[23] R. Hong, Y. Koga, S. Bandyadka, A. Leshchyk, Y. Wang, V. Akavoor, X. Cao, I. Sarfraz, Z. Wang, S. Alabdullatif, F. Jansen, M. Yajima, W.E. Johnson, J. D. Campbell, Comprehensive generation, visualization, and reporting of quality control metrics for single-cell RNA sequencing data, Nat. Commun. 13 (2022) 1688.
[24] L. Yu, Y. Cao, J.Y.H. Yang, P. Yang, Benchmarking clustering algorithms on estimating the number of cell types from single-cell RNA-sequencing data, Genome Biol. 23 (2022) 49.
[25] R. Petegrosso, Z. Li, R. Kuang, Machine learning and statistical methods for clustering single-cell RNA-sequencing data, Briefings Bioinf. 21 (2020) 1209–1223.
[26] P.G. Camara, Methods and challenges in the analysis of single-cell RNA-sequencing data, Curr. Opin. Struct. Biol. 7 (2018) 47–53.
[27] T.M. Consortium, Single-cell transcriptomics of 20 mouse organs creates a Tabula Muris, Nature 562 (2018) 367–372.
[28] X. Han, R. Wang, Y. Zhou, L. Fei, H. Sun, S. Lai, A. Saadatpour, Z. Zhou, H. Chen, F. Ye, Mapping the mouse cell atlas by microwell-seq, Cell 172 (2018) 1091–1107, e1017.
[29] O. Rozenblatt-Rosen, M.J. Stubbington, A. Regev, S.A. Teichmann, The human cell atlas: from vision to reality, Nature 550 (2017) 451–453.
[30] Z. Zhang, F. Cui, C. Lin, L. Zhao, C. Wang, Q. Zou, Critical downstream analysis steps for single-cell RNA sequencing data, Briefings Bioinf. 22 (2021).
[31] T. Tian, J. Wan, Q. Song, Z. Wei, Clustering single-cell RNA-seq data with a model-based deep learning approach, Nat. Mach. Intell. 1 (2019) 191–198.
[32] T. Tian, J. Zhang, X. Lin, Z. Wei, H. Hakonarson, Model-based deep embedding for constrained clustering analysis of single cell RNA-seq data, Nat. Commun. 12 (2021) 1873.
[33] T. Ronan, Z. Qi, K.M. Naegle, Avoiding common pitfalls when clustering biological data, Sci. Signal. 9 (2016) re6-re6.
[34] Y. Lu, Z. Yu, Y. Wang, Z. Ma, K.-C. Wong, X. Li, GMHCC: high-throughput analysis of biomolecular data using graph-based multiple hierarchical consensus clustering, Bioinformatics 38 (2022) 3020–3028.
[35] V.Y. Kiselev, T.S. Andrews, M. Hemberg, Challenges in unsupervised clustering of single-cell RNA-seq data, Nat. Rev. Genet. 20 (2019) 273–282.
[36] Z.-H. Zhou, Ensemble learning, in: Z.-H. Zhou (Ed.), Machine Learning, Springer Singapore, Singapore, 2021, pp. 181–210.
[37] H. Yin, J. Tao, Y. Peng, Y. Xiong, B. Li, S. Li, H. Yang, MSPJ: discovering potential biomarkers in small gene expression datasets via ensemble learning, Comput. Struct. Biotechnol. J. 20 (2022) 3783–3795.
[38] O. Sagi, L. Rokach, Ensemble learning: a survey, Wiley Interdisciplinary Reviews: Data Min. Knowl. Discov. 8 (2018) e1249.
[39] L. Gan, G. Vinci, G.I. Allen, Correlation imputation in single cell RNA-seq using auxiliary information and ensemble learning, in: Proceedings of the 11th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics, 2020, pp. 1–6.
[40] W. Gong, I.-Y. Kwak, P. Pota, N. Koyano-Nakagawa, D.J. Garry, DrImpute: imputing dropout events in single cell RNA sequencing data, BMC Bioinf. 19 (2018) 1–10.
[41] P. Ye, W. Ye, C. Ye, S. Li, L. Ye, G. Ji, X. Wu, scHinter: imputing dropout events for single-cell RNA-seq data with limited sample size, Bioinformatics 36 (2020) 789–797.
[42] X.-F. Zhang, L. Ou-Yang, S. Yang, X.-M. Zhao, X. Hu, H. Yan, EnImpute, Imputing dropout events in single-cell RNA-sequencing data via ensemble learning, Bioinformatics 35 (2019) 4827–4829.
[43] H. Wang, X. Ma, Learning deep features and topological structure of cells for clustering of scRNA-sequencing data, Briefings Bioinf. 23 (2022), bbac068.
[44] W. Wu, X. Ma, Network-based structural learning nonnegative matrix factorization algorithm for clustering of scRNA-seq data, IEEE ACM Trans. Comput. Biol. Bioinf 20 (2022) 566–575.
[45] T. Kim, K. Lo, T.A. Geddes, H.J. Kim, J.Y.H. Yang, P. Yang, scReClassify: post hoc cell type classification of single-cell rNA-seq data, BMC Genom. 20 (2019) 1–10.
[46] C. Fang, Y. Li, SCDF: a novel single-cell classification method based on dimension-reduced data fusion, in: International Conference on Intelligent Computing, Springer, 2022, pp. 196–206.
[47] T. Jiang, W. Zhou, Q. Sheng, J. Yu, Y. Xie, N. Ding, Y. Zhang, J. Xu, Y. Li, ImmCluster: an Ensemble Resource for Immunology Cell Type Clustering and Annotations in Normal and Cancerous Tissues, Nucleic Acids Research, 2022.
[48] J. Smolander, S. Junttila, M.S. Venäläinen, L.L. Elo, scShaper: an ensemble method for fast and accurate linear trajectory inference from single-cell RNA-seq data, Bioinformatics 38 (2022) 1328–1335.
[49] Y. Gan, N. Li, C. Guo, G. Zou, J. Guan, S. Zhou, TiC2D: trajectory inference from single-cell RNA-seq data based on consensus clustering, IEEE ACM Trans. Comput. Biol. Bioinf (2021).
[50] M. Dong, A. Thennavan, E. Urrutia, Y. Li, C.M. Perou, F. Zou, Y. Jiang, SCDC: bulk gene expression deconvolution by multiple single-cell RNA sequencing references, Briefings Bioinf. 22 (2021) 416–427.
[51] M. Cai, M. Yue, T. Chen, J. Liu, E. Forno, X. Lu, T. Billiar, J. Celedón, C. McKennan, W. Chen, Robust and accurate estimation of cellular fraction from tissue omics data via ensemble deconvolution, Bioinformatics 38 (2022) 3004–3010.
[52] T. Alqurashi, W. Wang, Clustering ensemble method, International Journal of Machine Learning and Cybernetics 10 (2019) 1227–1246.
[53] K.-H. Pho, H. Akbarzadeh, H. Parvin, S. Nejatian, H. Alinejad-Rokny, A multi-level consensus function clustering ensemble, Soft Comput. 25 (2021) 13147–13165.
[54] A. Strehl, J. Ghosh, Cluster ensembles—a knowledge reuse framework for combining multiple partitions, J. Mach. Learn. Res. 3 (2002) 583–617.
[55] K. Golalipour, E. Akbari, S.S. Hamidi, M. Lee, R. Enayatifar, From clustering to clustering ensemble selection: a review, Eng. Appl. Artif. Intell. 104 (2021), 104388.
[56] Z. Gu, M. Schlesner, D. Hübschmann, cola: an R/Bioconductor package for consensus partitioning through a general framework, Nucleic Acids Res. 49 (2021) e15.
[57] C. Bian, X. Wang, Y. Su, Y. Wang, K.-c. Wong, X. Li, scEFSC: accurate single-cell RNA-seq data analysis via ensemble consensus clustering based on multiple feature selections, Comput. Struct. Biotechnol. J. 20 (2022) 2181–2197.
[58] V.Y. Kiselev, K. Kirschner, M.T. Schaub, T. Andrews, A. Yiu, T. Chandra, K. N. Natarajan, W. Reik, M. Barahona, A.R. Green, SC3: consensus clustering of single-cell RNA-seq data, Nat. Methods 14 (2017) 483–486.
[59] S. Wan, J. Kim, K.J. Won, SHARP: hyperfast and accurate processing of single-cell RNA-seq data via ensemble random projection, Genome Res. 30 (2020) 205–213.
[60] Y. Cui, S. Zhang, Y. Liang, X. Wang, T.N. Ferraro, Y. Chen, Consensus clustering of single-cell RNA-seq data by enhancing network affinity, Briefings Bioinf. 22 (2021) bbab236.
[61] Y.-L. Zhu, Y.-L. Gao, J.-X. Liu, R. Zhu, X.-Z. Kong, Ensemble adaptive total variation graph regularized NMF for singlecell RNA-seq data analysis, Curr. Bioinf. 16 (2021) 1014–1023.
[62] Q. Yin, Y. Wang, J. Guan, G. Ji, scIAE: an integrative autoencoder-based ensemble classification framework for single-cell RNA-seq data, Briefings Bioinf. 23 (2022) bbab508.
[63] D. Risso, L. Purvis, R.B. Fletcher, D. Das, J. Ngai, S. Dudoit, E. Purdom, clusterExperiment, RSEC, A Bioconductor package and framework for clustering of single-cell and other large gene expression datasets, PLoS Comput. Biol. 14 (2018), e1006378.
[64] F.R. Ringeling, S. Canzar, Linear-time cluster ensembles of large-scale single-cell RNA-seq and multimodal data, Genome Res. 31 (2021) 677–688.
[65] L. Hu, J. Zhou, Y. Qiu, X. Li, An ultra-scalable ensemble clustering method for cell type recognition based on scRNA-seq data of alzheimer's disease, in: Proceedings of the 3rd Asia-Pacific Conference on Image Processing, Electronics and Computers, 2022, pp. 275–280.
[66] M.C. Thrun, Distance-based clustering challenges for unbiased benchmarking studies, Sci. Rep. 11 (2021), 18988.
[67] D. Zhang, Y. Zhu, ECBN: Ensemble Clustering Based on Bayesian Network Inference for Single-Cell RNA-Seq Data, 2020 39th Chinese Control Conference (CCC), IEEE, 2020, pp. 5884–5888.
[68] R.J. Burton, S.M. Cuff, M.P. Morgan, A. Artemiou, M. Eberl, GeoWaVe: Geometric Median Clustering with Weighted Voting for Ensemble Clustering of Cytometry Data, bioRxiv, 2022.
[69] K. Quintelier, A. Couckuyt, A. Emmaneel, J. Aerts, Y. Saeys, S. Van Gassen, Analyzing high-dimensional cytometry data using FlowSOM, Nat. Protoc. 16 (2021) 3775–3801.
[70] K.R. Moon, D. van Dijk, Z. Wang, S. Gigante, D.B. Burkhardt, W.S. Chen, K. Yim, A. v.d. Elzen, M.J. Hirn, R.R. Coifman, N.B. Ivanova, G. Wolf, S. Krishnaswamy,

Visualizing structure and transitions in high-dimensional biological data, Nat. Biotechnol. 37 (2019) 1482–1492.
[71] P. Qiu, E.F. Simonds, S.C. Bendall, K.D. Gibbs, R.V. Bruggner, M.D. Linderman, K. Sachs, G.P. Nolan, S.K. Plevritis, Extracting a cellular hierarchy from high-dimensional cytometry data with SPADE, Nat. Biotechnol. 29 (2011) 886–891.
[72] Jacob H. Levine, Erin F. Simonds, Sean C. Bendall, Kara L. Davis, E.-ad D. Amir, Michelle D. Tadmor, O. Litvin, Harris G. Fienberg, A. Jager, Eli R. Zunder, R. Finck, Amanda L. Gedman, I. Radtke, James R. Downing, D. Pe'er, Garry P. Nolan, Data-driven phenotypic dissection of AML reveals progenitor-like cells that correlate with prognosis, Cell 162 (2015) 184–197.
[73] S.V. Stassen, D.M.D. Siu, K.C.M. Lee, J.W.K. Ho, H.K.H. So, K.K. Tsia, PARC: ultrafast and accurate clustering of phenotypic data of millions of single cells, Bioinformatics 36 (2020) 2778–2786.