OXFORD

Briefings in Bioinformatics, 2024, **25(5)**, bbae411
https://doi.org/10.1093/bib/bbae411
Problem Solving Protocol

CHAI: consensus clustering through similarity matrix integration for cell-type identification

Musaddiq K. Lodi[1], Muzammil Lodi[2], Kezie Osei[3], Vaishnavi Ranganathan[4], Priscilla Hwang[5], Preetam Ghosh[2,*]

[1]Integrative Life Sciences, Virginia Commonwealth University, Richmond, VA 23284, United States
[2]Department of Computer Science, Virginia Commonwealth University, Richmond, VA 23284, United States
[3]Center for Biological Data Science, Virginia Commonwealth University, Richmond, VA 23284, United States
[4]School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213, United States
[5]Department of Biomedical Engineering, Virginia Commonwealth University, Richmond, VA 23284, United States
*Corresponding author. Department of Computer Science, 601 W Main St, Virginia Commonwealth University, Richmond, VA 23284, United States. E-mail: pghosh@vcu.edu

Abstract

Several methods have been developed to computationally predict cell-types for single cell RNA sequencing (scRNAseq) data. As methods are developed, a common problem for investigators has been identifying the best method they should apply to their specific use-case. To address this challenge, we present CHAI (consensus Clustering tHrough similArIty matrix integratIon for single cell-type identification), a wisdom of crowds approach for scRNAseq clustering. CHAI presents two competing methods which aggregate the clustering results from seven state-of-the-art clustering methods: CHAI-AvgSim and CHAI-SNF. CHAI-AvgSim and CHAI-SNF demonstrate superior performance across several benchmarking datasets. Furthermore, both CHAI methods outperform the most recent consensus clustering method, SAME-clustering. We demonstrate CHAI's practical use case by identifying a leader tumor cell cluster enriched with CDH3. CHAI provides a platform for multiomic integration, and we demonstrate CHAI-SNF to have improved performance when including spatial transcriptomics data. CHAI overcomes previous limitations by incorporating the most recent and top performing scRNAseq clustering algorithms into the aggregation framework. It is also an intuitive and easily customizable R package where users may add their own clustering methods to the pipeline, or down-select just the ones they want to use for the clustering aggregation. This ensures that as more advanced clustering algorithms are developed, CHAI will remain useful to the community as a generalized framework. CHAI is available as an open source R package on GitHub: https://github.com/lodimk2/chai.

Keywords: single-cell biology; clustering; cell-type identification; wisdom-of-crowds

Introduction

The advent of single cell RNA sequencing (scRNAseq) has allowed researchers to investigate transcriptional mechanisms at the single cell resolution. Notably, scRNAseq has contributed to the identification of rare cell types, assessing cell heterogeneity, and quantifying cell-cell variation [1]. A common methodology for identifying subpopulations from single cells has been unsupervised clustering [2]. However, the nature of scRNAseq data presents unique challenges in identifying accurate clusters. For example, scRNAseq data is sparse, with frequent gene and cell dropouts. Additionally, scRNAseq data is high dimensional, which leads to data points being similar and therefore unreliable for downstream clustering tasks. Due to these factors, a diverse array of scRNAseq clustering methods have emerged recently [2].

While several clustering methods for scRNAseq data have been published, comprehensive benchmarking studies, such as the one from Yu *et al.*, have indicated that there is no clear 'best method' across all scenarios [3]. Due to the high amount of variability in scRNAseq data, even the most commonly used clustering algorithms have distinct strengths and weaknesses. Take for example Seurat, perhaps the most commonly used scRNAseq clustering platform: while results from Seurat often demonstrate high concordance with ground-truth cell type populations, it also tends to overestimate the number of distinct cell types in a dataset [3, 4]. Seurat, along with other popular scRNAseq clustering workflows such as Spectrum and SC3, use community detection algorithms such as Leiden and Louvain as the primary mechanism for their clustering. Preprocessing steps, such as highly variable gene selection, or dimensionality reduction through Principal Component Analysis (PCA), have also become common place before performing the final clustering [3–6]. Additionally, common unsupervised clustering algorithms, such as k means or hierarchical clustering, are used to create initial clusters before reclustering, such as in CIDR [7]. More recently developed algorithms such as scSHC and CHOIR use a statistical significance testing to determine final cluster assignments and also serve as an evaluation framework outside of the commonly used metrics such as Adjusted Rand Index (ARI) and Normalized Mutual Information (NMI) [8–12].

With the various scRNAseq clustering methodologies currently available, a common question for investigators becomes: Which method should I use? As there is no definite answer for this, an intuitive approach is to integrate the results from the different clustering algorithms, into a 'clustering ensemble' or 'consensus clustering' [13]. This idea extends from the wisdom of crowds

Received: April 9, 2024. **Revised:** June 29, 2024. **Accepted:** August 2, 2024

approach, which states that knowledge from the collective of a group is greater than that of an individual [14–16].

The idea of consensus clustering was introduced by Strehl and Ghosh, who pioneered hypergraph partitioning algorithms for integrating results from individual clustering results [17]. The framework of consensus clustering has been introduced to single cell biology in a variety of ways. A frequently used method, SC3, uses consensus clustering based on Clustering Similarity Partitioning Algorithm (CSPA) by running KMeans clustering several times on a single cell count matrix, taking average similarity across the binary matrix representations, and then performing hierarchical clustering on the average consensus matrix [5]. Another method, scCCESS, performs consensus clustering by combining random low dimensional representations of a single cell count matrix through SIMLR, a clustering kernel specially optimized for single cell clustering. The authors of scCCESS noted that their autoencoder-based ensemble method is highly effective in isolating specific cell types [18]. These methods helped to highlight the effectiveness of wisdom of crowds approach for clustering in single cell biology. However, these consensus clustering methods are self contained, which means that they run the same method several times, and perform consensus clustering on an aggregated matrix. Another method of consensus clustering is to incorporate results from several different methods into one composite result. This approach has also been successfully accomplished and benchmarked for single cell clustering.

A method known as SAFE-Clustering implemented all three of Strehl and Ghosh's algorithms in an application to scRNAseq clustering, which included the clustering methodologies Seurat, SC3, CIDER, t-SNE, and k-means in 2018 [19]. SAFE-Clustering demonstrated robust performance across 12 benchmarking datasets, establishing the premise that consensus clustering is applicable to scRNAseq data. Another ensemble clustering method, SAME-Clustering, uses a Mixture model Ensemble to aggregate results from different scRNAseq clustering methodologies [20]. However, since these methods were created in 2020 and prior, there have been further advancements made to the existing algorithms in their pipeline such as Seurat and SC3, and the other algorithms, such as CIDER and SIMLR, are not as widely used [3]. Additionally, these ensemble clustering approaches are not immediately extendable to multi-omic data integration, which can provide even more insights toward distinct cell types and state. A consensus aggregation approach is only as accurate as the performance of the individual information, and so we identified a need for an updated consensus clustering framework that can also seamlessly allow for multiomic data integration.

Here, we present CHAI (consensus Clustering tHrough similArIty matrIces), a consensus clustering methodology built upon binary similarity matrices. CHAI contains two clustering ensemble approaches, named CHAI-AvgSim and CHAI-SNF. CHAI-AvgSim is performed by aggregating all clustering assignments with an average similarity matrix, and performing Spectral Clustering on the final average matrix. CHAI-SNF extends Similarity Network Fusion (i.e. SNF), which is a network integration algorithm originally designed for multiomic data integration for patient subtyping and classification [21].

Both CHAI methods have demonstrated improved performance across several benchmarking datasets and conditions, showcasing limited variability across runs, and low impact from poor performing algorithms. Additionally, we present a technique to integrate other data modalities into the CHAI framework, such as spatial transcriptomic data or ATAC-Seq data. CHAI contains seven state-of-the-art scRNAseq clustering algorithms (Seurat-Louvain, Seurat-SLC, CHOIR, RACEID, SC3, Spectrum, and scSHC) and is available as an R package [4–6, 8, 9, 22]. We seek to make CHAI a collaborative tool for the community by providing a way for scientists and developers to integrate their own clustering algorithms into the pipeline as well, which may potentially strengthen results as more advanced scRNAseq clustering algorithms emerge in the future.

Overall, CHAI reinforces the importance of the wisdom of crowds approach for scRNAseq clustering. Specifically, this study makes the following contributions: to our knowledge, CHAI is the first method to incorporate average similarity on binary similarity matrices for consensus clustering across various methods on scRNAseq data. Additionally, CHAI is the first method to extend SNF for the purpose of ensemble clustering. This has a wide variety of applications in several fields that require clustering, not just single cell biology. Finally, CHAI is the first method to use SNF for multi-omic integration in single cell biology and highlights the power of simple similarity matrix representation of 'omic' data.

Materials and Methods

The CHAI workflow may be summarized as three majors steps:

(i) Run individual clustering algorithms and compute binary similarity matrix for each.
(ii) Calculate Average Similarity matrix and/or SNF matrix.
(iii) Run Spectral Clustering on either integrated matrix to determine final cell identities.

The package is written in R and is available for installation on GitHub at https://github.com/lodimk2/chai.

Individual clustering algorithms

CHAI incorporates seven algorithms by default when using the package, which are described below. Users may also integrate information from other clustering methods.

Seurat

Seurat begins with dimensionality reduction methods such as PCA, Uniform Manifold Approximation and Projection, and t-distributed stochastic neighbor embedding (tSNE). It then identifies variably expressed genes, then a K nearest neighbor (KNN) graph is computed based upon these. From here, community detection algorithms are used to identify the final clusters. Both Louvain and smart local moving (SLM) rely on the local moving heuristic for modularity optimization. The premise is to continually move individual nodes from one community to another so that each node movement elicits a modularity increase. This is done in a random order. For each node, it is checked whether it is possible to increase the modularity by moving it to a different community. If this is possible, then the node is moved to the community that results in the highest modularity gain. This repeats until it is no longer possible to increase modularity through individual node movements. In CHAI, we used Louvain and SLM. There are two versions of Louvain that are used in the paper: Louvain and Louvain with Multilevel Refinement. Both algorithms follow the same steps, with the difference being that the local moving heuristic is run again at the end of the program to fine-tune the final community structure and to also guarantee that the final community structure can not be further optimized. First, an adjacency matrix of a network and the initial assignments of nodes to communities is inputted. The local moving heuristic is run. If the

number of communities is less than the number of nodes, then a reduced network is created. A recursive call is then performed to identify the community structure of the reduced network. The communities are then merged based off this community structure. Finally, based off which version of Louvain is run, the local moving heuristic can be performed. SLM applies the local moving heuristic differently than Louvain. First, the local moving heuristic is run. Then, if the number of communities is less than the number of nodes, a subnetwork for each community is created and the local moving heuristic is run for each subnetwork. A reduced network is then formed based on the community structure of the subnetworks. A recursive call is performed to identify the community structure of the reduced network, and the communities are merged based on those findings.

CHOIR

CHOIR constructs a hierarchical clustering tree. Using all cells, it identifies a set of features that have variable levels of expression. Then, dimensionality reduction is applied using either PCA, latent semantic indexing (LSI), or iterative LSI, with PCA being the default method. A nearest neighbor adjacency matrix is computed, and to generate the layers of the clustering tree, Louvain and Leiden clustering is used. MRtree is used to reshape the clustering trees into a hierarchical tree [8].

RaceID

RaceID uses K-means clustering. First, a similarity matrix is constructed, which contains Pearson's correlation coefficients for all pairs of cells. K-means clustering is then applied to it, and the number of clusters used for k-means clustering is decided on by the difference of the average within cluster dispersion in the data. It also computes Jaccard's similarity to check if fewer clusters should have been produced [22].

SC3

SC3 uses a gene filter to remove any genes or transcripts that are in less than X% of cells (X being commonly set to 6). After calculating the distance between the cells, using Euclidean, Pearson, and Spearman metrics, all distance matrices are then transformed. K-means clustering is then applied. A consensus matrix is computed using CSPA (Cluster-based Similarity Partitioning). For each individual cluster result, a binary similarity matrix is made. If two cells belong to one cluster, their similarity is 1; otherwise, it is 0. The consensus matrix is created by averaging all similarity matrices of the individual clustering. [5].

Spectrum

Spectrum uses an adaptive density-aware kernel (based on the Zelnik–Manor self-tuning kernel and the Zhang density-aware kernel) to construct the similarity matrices. These matrices are combined using tensor product graph (TPG) diffusion. Then, the spectral clustering method is applied to the similarity matrix [6].

scSHC

scSHC used hierarchical clustering as a part of their algorithm. The first step is to compute the distance between each cell, but since scRNA-seq data have small counts and high dimensionality, finding the Euclidean distance is unreliable. Therefore, Euclidean distance on the latent variables is computed instead. To identify the clusters, a desired family-wise error rate is decided upon (0.05 in simulated data and 0.25 on real data applications). The method goes down the tree to decide which splits should be kept. This decision is made using hypothesis testing: a test statistic is formed using the average silhouette, which is then compared to the desired family-wise error rate. If it is greater or equal to the desired family-wise error rate, then it failed to reject the null hypothesis, and all data should belong to one cluster. Otherwise, the data is split into the two proposed clusters and the method continues down the tree [9].

CHAI-AvgSim

Once the individual clustering assignment algorithms are run, they will each be represented as a table containing the Cell ID in one column, and the Clustering Assignment as the other column. From here, we convert this table to a binary similarity matrix. We represent a cell to clustering assignment vector as a binary similarity matrix using the following rules:

(i) If two cells have the same clustering assignment, assign a value of 1 to a binary similarity matrix corresponding to the two cells.
(ii) If two cells do not have the same clustering assignment, assign a value of 0 to the binary similarity matrix corresponding to the two cells.

Through this method, each clustering assignment is converted into a binary similarity matrix. Each binary similarity matrix per algorithm is then aggregated into an Average Similarity matrix, which simply put is a cell to cell correlation matrix containing the per element average rank across all individual clustering algorithm matrices.

Consider a dataset with m cells. Therefore, each binary similarity matrix per algorithm will be of dimension $m \times m$. To construct an Average Similarity matrix of $m \times m$ dimension, we calculate the average per cell using the following formula:

$$\bar{M}_{ij} = \frac{1}{n}\sum_{k=1}^{n}(M^{(k)})_{ij},$$

Where:

n = Total number of samples
n_{ij} = Number of pairs of samples that are in the same cluster in both the first and second clustering
a_i = Number of samples in theith cluster according to the first clustering
b_j = Number of samples in thejth cluster according to the second clustering.

This formula is repeated across each cell in the matrix until a final $m \times m$ matrix is created.

Once the Average Similarity matrix is computed, we use Spectral Clustering to determine the final cell clusters [23]. If the true number of clusters is known to the user, they can use this k value as the number of partitions to make on the Average Similarity matrix. If the true number of k in the dataset is not known to the user, we recommend calculating the k value for which the silhouette score is the highest. For all evaluations conducted in our benchmarking, we conduct a silhouette score evaluation in range 2 to $k + 1$, with k being the true number of clusters present in the dataset. Despite the true number of clusters being known in the benchmarking dataset, we choose a value of k computationally in order to simulate working with unknown data.

CHAI-SNF

The CHAI-SNF method begins similarly to CHAI-AvgSim, where a clustering table containing Cell ID and Clustering Assignment is converted into a binary similarity matrix for each clustering algorithm. However, rather than taking an average vote across cell to assignment similarities, we apply the SNF algorithm across all binary similarity matrices [21].

The SNF algorithm was created for multiomic data integration in bulk RNA sequencing data. It was used to integrate patient to patient similarity matrices across three data modalities: mRNA expression, DNA methylation, and microRNA (miRNA) expression. Once the matrices were integrated, the final matrix was used for downstream tasks such as cancer subtyping and survival analysis [21]. In brief, SNF performs similarity matrix fusion by converting a pairwise patient similarity matrix to a graph, where nodes are the patients and edges are the relationships between the patients. From here, SNF uses a network fusion step based on message passing theory that iteratively updates each network, which makes it more similar to the other networks until all networks are the same. SNF has been demonstrated to remove low edge weights, also known as 'weak edges', from the final network, and include only relationships that are more likely to be in concordance with the ground-truth [21].

Ultimately, since we have cell to cell similarity matrices for each clustering algorithm, applying SNF to the individual algorithm's binary similarity matrix representation was straightforward. We implemented SNF using the SNFtool package in R, available on CRAN, using the default parameters. For more detailed information on SNF, please refer to Wang *et al.* [21].

Similar to CHAI-AvgSim, we infer the final clusters by running Spectral Clustering on the final SNF combined matrix, either by knowing the true k value or by calculating the best k by silhouette score optimization.

GraphST binary matrix representation for spatial transcriptomics

GraphST is a method that integrates spatial coordinates with scRNAseq data. One step in their process is to represent the distance between cells as a binary matrix [24]. We incorporate that logic here into CHAI in order to integrate spatial transcriptomics into CHAI-AvgSim and CHAI-SNF.

GraphST creates an undirected neighborhood graph represented as a binary adjacency matrix, where the number of neighbors to any one cell is set to be a predefined number k. The neighbors of a spot $s \in S$, where each spot is represented as a vertex of the graph, represent the k spatially closest spots to s. Enumerating S, the adjacency matrix $M \in \mathbb{R}^{n \times n}$, where n is the number of spots, is constructed such that $a_{ij} = 1$ if $i, j \in S$ are neighbors and 0 otherwise [24].

A neighborhood matrix created utilizing the same logic is incorporated into CHAI-AvgSim as another clustering assignment in the average matrix. Additionally, after applying CHAI-SNF on the various clustering assignments to produce a preliminary clustering assignment matrix, SNF is applied once again on this resultant matrix and the created neighborhood matrix to obtain the final clustering matrix that incorporates spatial data.

Evaluation metrics

Adjusted rand index

ARI is a frequently used evaluation metric for clustering data, particularly in single cell genomics clustering [19]. ARI measures the concordance between a predicted set of clusters and the true set of clusters, scaled between −1 and 1. The higher the ARI, the better the performance, with 1 indicating a perfect overlap between the predicted and true clusters [25].

ARI may be calculated using the following formula:

$$ARI = \frac{\sum_{ij}\binom{n_{ij}}{2} - [\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}/\binom{n}{2}]}{\frac{1}{2}[\sum_i \binom{a_i}{2} + \sum_j \binom{b_j}{2}] - [\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}/\binom{n}{2}]},$$

Where:

n = Total number of samples
n_{ij} = Number of pairs of samples that are in the same cluster in both the first and second clustering
a_i = Number of samples in the ith cluster according to the first clustering
b_j = Number of samples in the jth cluster according to the second clustering.

Normalized mutual information

Normalized Mutual Information (NMI) is a measure used to quantify the similarity between predicted clusters and the true clusters. It stems from the concept of mutual information, which measures the amount of information obtained about one random variable through the observation of another random variable. NMI ranges from 0 to 1, where 0 indicates no mutual information between the predicted and true clusters, and 1 indicates perfect agreement between the predicted and true clusters [26].

The mutual information between the predicted and true clusters, C and K, is given by

$$NMI(C, K) = \frac{2 \cdot I(C, K)}{H(C) + H(K)},$$

Where:

$NMI(C, K)$ = Normalized Mutual Information between clustering C and K
$I(C, K)$ = Mutual Information between clustering C and K
$H(C)$ = Entropy of clustering C
$H(K)$ = Entropy of clustering K.

Silhouette score

To evaluate the best k for Spectral Clustering on either the CHAI-AvgSim or CHAI-SNF matrix, we calculate the best average Silhouette Score. Silhouette score measures how close each sample in one cluster is to the samples in neighboring clusters, which helps to assess the quality of clustering. This metric ranges from −1 to 1, with a high score indicating a cell is matched closely to its labeled cluster. Silhouette Score is calculated using the following formula:

$$\text{Silhouette score} = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{b(i) - a(i)}{\max\{a(i), b(i)\}}\right),$$

Where:

n is the total number of samples
$a(i)$ is the average distance from sample$i$ to other points in the same cluster
$b(i)$ is the smallest average distance from sample$i$ to points in a different cluster.

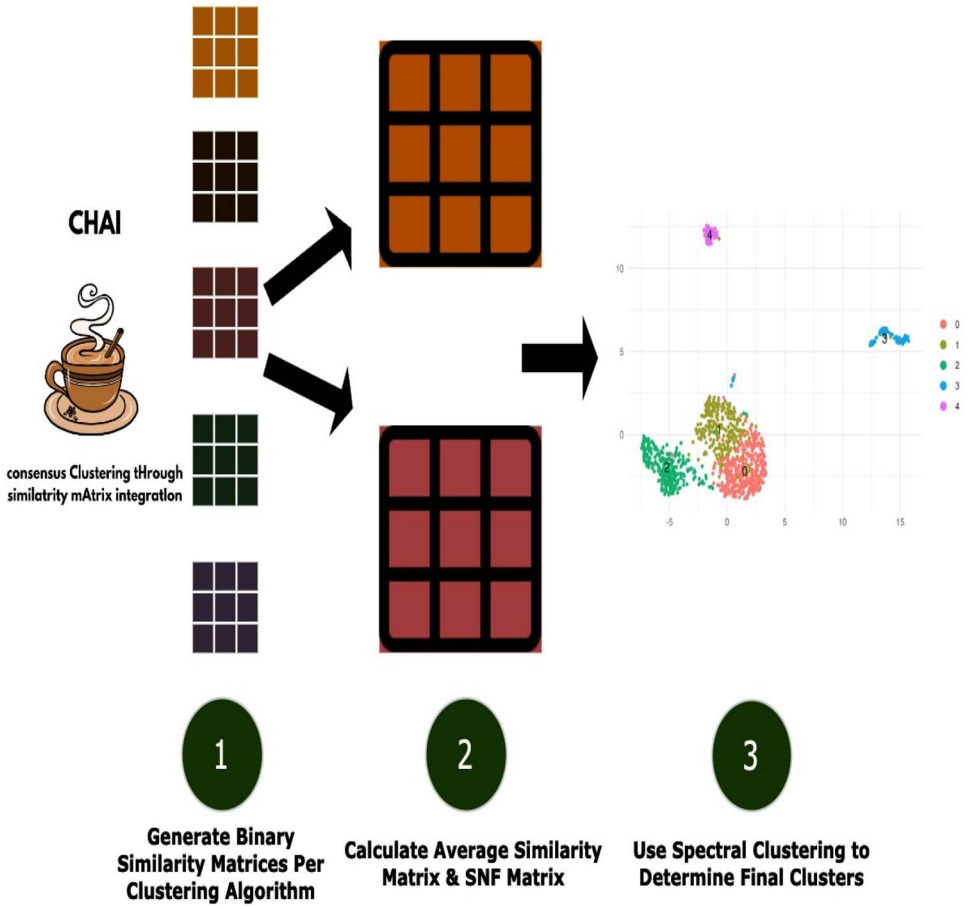


Figure 1. Flowchart depicting the CHAI workflow.

Results

CHAI workflow

CHAI is a consensus clustering method that presents two different approaches for the integration of individual clustering results: Average Similarity and SNF [21]. For a more detailed description of each method, please refer to Methods section.

All CHAI-related methods (CHAI-AvgSim, CHAI-SNF, and CHAI-ST) operate under binary matrices. For clustering algorithms, these matrices are calculated by determining if two cells are predicted to be in the same cluster. If they are, we assign 1 to the matrix entry to designate that these two cells are related. If not, we assign 0.

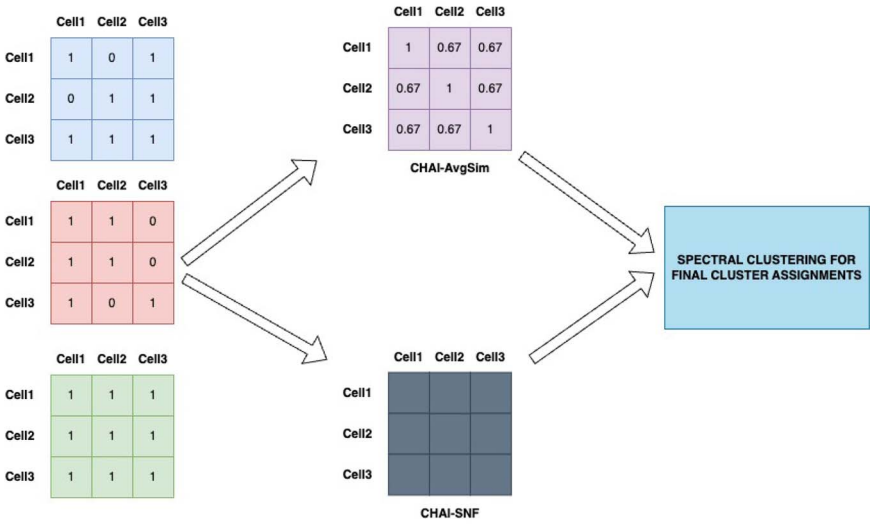
For the spatial coordinates binary matrix representation, we use the methodology from GraphST [24]. First we calculate a pairwise distance between cells based on the spatial coordinates. Then, we run a KNN graph, with K being 3. If two cells are neighbors based on this KNN graph, we assign a value of 1 to this cell–cell relationship. If not, we assign 0.

To further illustrate this concept, consider a toy example with three clustering algorithms and three cells. For all CHAI methods, we first calculate the binary matrices. Fig. 1 depicts the overall workflow, while Fig. 2a and b show the example runs of the CHAI methodology.

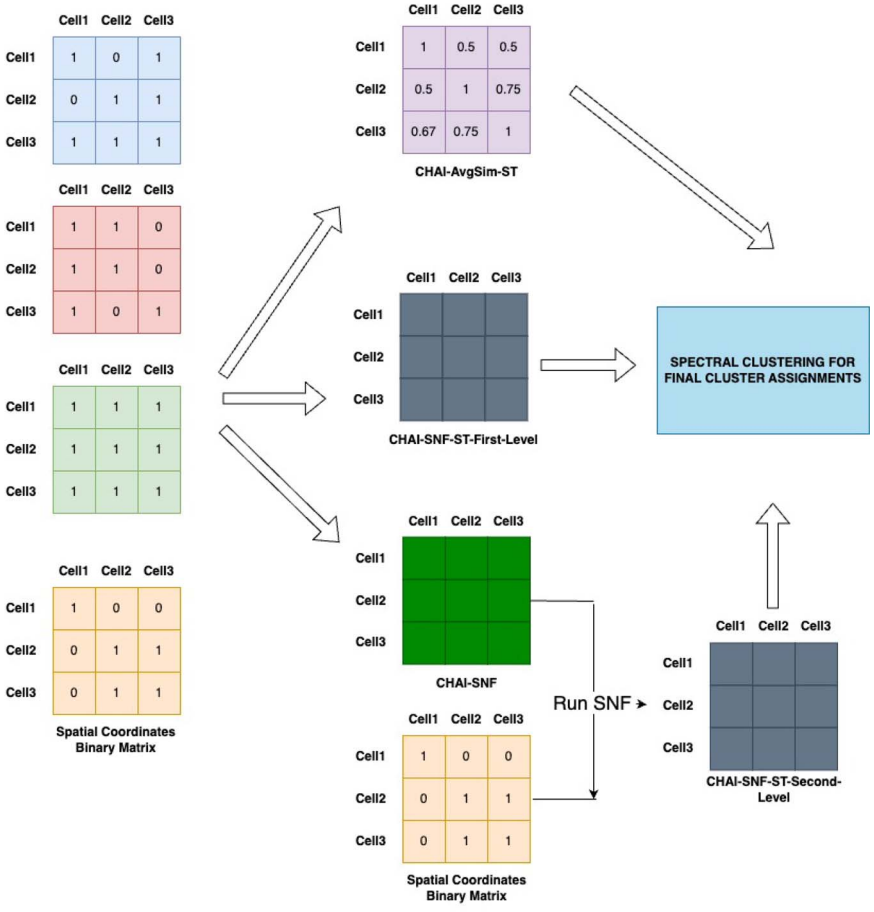
Figure 2a shows how CHAI-AvgSim and CHAI-SNF are run. For CHAI-AvgSim, we calculate an average of all the binary matrices from the different clustering algorithms. Then, we run Spectral Clustering on the resultant matrix to determine the final clusters. For CHAI-SNF, we run SNF with default parameters on the binary matrices from the clustering algorithms. Then, we perform Spectral Clustering on the resultant CHAI-SNF matrix to determine the final clustering assignments.

We present three different ways to integrate spatial transcriptomic data into CHAI (Fig. 2b). For CHAI-AvgSim-ST, we simply include the binary matrix representation of the spatial coordinate data as another matrix to be included into the AvgSim calculation. We then run Spectral Clustering on the resultant matrix to determine the final clusters. For CHAI-SNF-First-Level, we run SNF on all binary matrices, including the spatial coordinates binary matrix representation. For CHAI-SNF-Second-Level, we first run SNF on just the clustering assignment matrices and keep the spatial coordinates binary matrix separate. Once the SNF matrix for the clustering assignment binary matrices are calculated, we run SNF again, this time with the clustering assignment matrix from the first level SNF and the spatial coordinates matrix. For both CHAI-SNF-First-Level and CHAI-SNF-Second-Level, we run Spectral Clustering on the resulting matrix to determine the final clusters. The main difference between CHAI-SNF-First-Level and CHAI-SNF-Second-Level is that the latter gives more weight to the spatial coordinate data, since it is included separately as an 'omic' rather than just another clustering assignment as considered in CHAI-SNF-First-Level. Users may make the decision to run CHAI-SNF-First-Level or CHAI-SNF-Second-Level based on their prior biological knowledge of their datasets.

We benchmarked the performance of both CHAI methods on several datasets. We used 10 publicly available scRNAseq datasets for our main performance evaluation. Additionally, we



(a) Example of running CHAI-AvgSim and CHAI-SNF



(b) Example of running CHAI-ST-AvgSim, CHAI-ST-SNF-First-Level, and CHAI-ST-SNF-Second-Level

Figure 2. CHAI Workflow Examples.

took advantage of the size and complexity in the Zheng68K PBMC dataset to create subsampled datasets to evaluate the performance of CHAI on various dataset conditions, such as the number of cells and the number of cell types. In brief, we find that CHAI is a more consistent and accurate performer in diverse dataset conditions when compared with baseline algorithms.

We chose to evaluate using ARI and NMI as they each measure the overlap between predicted and ground truth clustering assignments, and their value decreases as disagreements between subpopulations increase [27]. We display the ARI evaluation in the main text, and the NMI evaluation in the supplementary materials.

CHAI outperforms existing clustering methods on benchmarking datasets

To assess the performance of CHAI-AvgSim and CHAI-SNF, we compared them to seven individual algorithms that form the consensus method. We ran each algorithm on 10 commonly used benchmarking datasets with varying tissue source, the number of cells and the number of cell types. We evaluated the performance using ARI.

We see in Fig. 3a that both CHAI-AvgSim and CHAI-SNF demonstrate robust and consistent performance across benchmarked datasets. Notably, CHAI-AvgSim was a top three performer in 8 out of 10 datasets. We show the frequency of top three performers in each dataset in a heatmap, depicted in Fig. 3b. CHAI-AvgSim and CHAI-SNF have the highest frequency of being the top three performing algorithms, with scores of 80% and 60%, respectively.

The variability of performance in other baseline algorithms is very noticeable in this analysis. Widely used algorithms such as SC3 and RaceID demonstrate very strong performances in some datasets, like the Zeisel mouse brain dataset, but very poorly in others, such as the SC-Mixology-Dropseq dataset [5, 22, 28, 29]. The primary benefit of the CHAI consensus algorithms is that they reduce this variability in performance. We visualize this variability by plotting the distribution of ARI values as a boxplot, seen in Fig. 3c. Both CHAI-AvgSim and CHAI-SNF have higher median ARI than any of the baseline clustering methods. This analysis also helps to highlight the difference in performance between the two CHAI methods. CHAI-SNF has a higher median ARI, a higher third quartile threshold, and a higher maximum ARI than CHAI-AvgSim, demonstrating its potential for high accuracy. However, it has a much larger interquartile range, which suggests higher variability in performance. CHAI-AvgSim, on the other hand, has a comparable median ARI with other baseline methods, such as Seurat-Louvain and Seurat-SLC. The primary advantage of CHAI-AvgSim lies in its low interquartile range, as it has the lowest interquartile range when compared with any other baseline algorithm. This shows that CHAI-AvgSim is a much more consistent performer across various datasets than any other algorithm including CHAI-SNF, making it a robust choice.

We also calculated the rank of each algorithm across the benchmarking datasets, as shown in Fig. 3. This was done as another metric to measure top performance. Algorithms with a lower rank are higher performers (1 being the best rank, and so on). The median rank of CHAI-AvgSim and CHAI-SNF are quite low at ∼3 making them a safe choice for accurate clustering across diverse datasets. Additionally, we see that the minimum for CHAI-SNF and CHAI-AvgSim is ∼1 and 2, respectively, showing that it is more likely to be a top performing algorithm than the other baseline algorithms.

Here, we also compare CHAI to a previous consensus clustering method, SAME Clustering [20]. CHAI incorporates more algorithms than SAME clustering and also runs the latest version of Seurat [4]. We demonstrate that at least one of the two CHAI methods outperforms SAME clustering in 8 of the 10 datasets. SAME clustering and CHAI have similar median ARI's and distributions. CHAI-SNF has the highest upper quartile cutoff value and the highest median across all algorithms. It also demonstrated the highest ARI for any of the benchmarking datasets. Despite the similarities in ARI distribution, we see that both CHAI methods have a lower distribution of rank when compared with SAME clustering. CHAI-AvgSim and CHAI-SNF have a median rank of 3 and 2, respectively, compared with SAME clustering's median rank of 5. Additionally, CHAI-AvgSim is the most consistent performer in terms of rank, with its lowest rank across datasets being 5, compared with CHAI-SNF's lowest rank of 6 and SAME-Clustering's lowest rank of 7.

CHAI outperforms existing clustering methods across varying dataset sizes and complexity

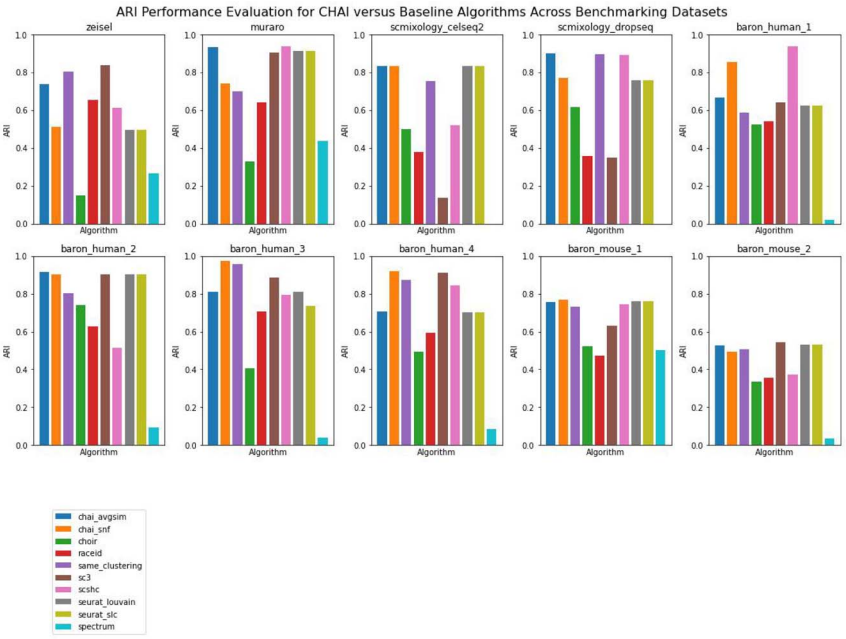
In order to evaluate CHAI on varying datasets in terms of complexity and size, we took advantage of the varying cell types and large number of cells in the Zheng68K PBMC dataset [30]. We created six different datasets, with three different sizes and number of cell types. We refer to the datasets with five equally sized groups as 'simple' cases and randomly selected groups as 'challenging'.

CHAI-AvgSim and CHAI-SNF are robust performers across dataset conditions, as seen in Fig. 4a. Both methods are top three performers in all six of the subsampled datasets; additionally, CHAI-AvgSim is the top performer in three of the six datasets. Either CHAI method has a better ARI than SAME-Clustering, the other consensus clustering method, in all six of the subsampled sets. In Fig. 4c, we note that CHAI-AvgSim has the highest median ARI, while CHAI-SNF has the lowest interquartile range. This suggests that CHAI-AvgSim calculates a higher ARI more frequently, but CHAI-SNF is more consistent in performance.

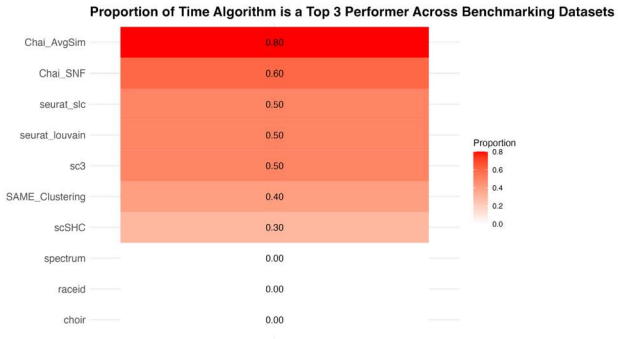
We also sought to evaluate how well each method performs when faced with a simple or challenging dataset. Figure 4b displays the percent difference between simple and challenging datasets for each algorithm across dataset sizes. Most algorithms decrease in performance in terms of ARI when evaluated on a dataset with randomly selected groups, across dataset size. Notably, CHAI-SNF seems to actually increase in performance on challenging datasets, even as the size of the dataset increases. We consider that a consistent algorithm would perform well when dataset sizes are the same, but the topologies of clusters are different. Therefore, we examine the absolute value of percent difference across dataset sizes, but between the simple and challenging datasets, depicted in Fig. 4d. CHAI-SNF has very little difference between simple and challenging datasets; this is in contrast to CHAI-AvgSim, which has the highest median ARI and a low interquartile range, but displays a larger percent difference between its simple and challenging cases. Both methods ultimately outperform the other consensus method, SAME-Clustering, in terms of median ARI, consistent performance by ARI distribution, and low percent difference between simple and challenging cases.

CHAI derives validated biological insights in a breast cancer dataset: case study

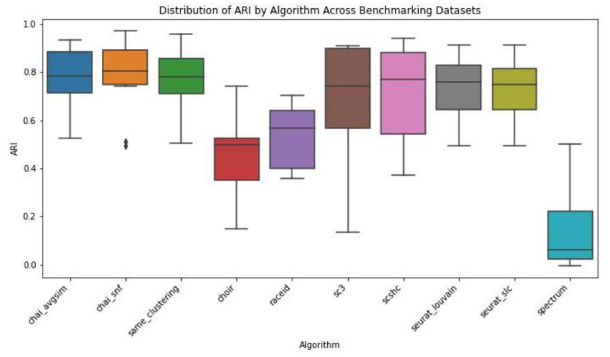
A potential concern surrounding consensus clustering methods is that the features of certain methods may be overshadowed by the results from all other methods. scRNAseq clustering methods use a variety of different techniques to determine the final cell to cluster assignments, which involve a varying degree of biological information [3]. Many methods, such as Seurat and CHOIR, filter the initial expression matrix through PCA and identify the highly variable genes within the dataset [4, 8]. Other methods, such as tSNE + KMeans Clustering, do not use any prederived biological insight prior to clustering [20]. There are also clustering methods, such as CIDER, which recluster cells based on differentially expressed gene (DEG) signature [31]. With this diversity in clustering in mind, we tested if CHAI can reliably derive biological conclusions as a standalone method. We decided to use CHAI-AvgSim for this analysis, as it demonstrated better consistency across dataset conditions than CHAI-SNF in our benchmarking.



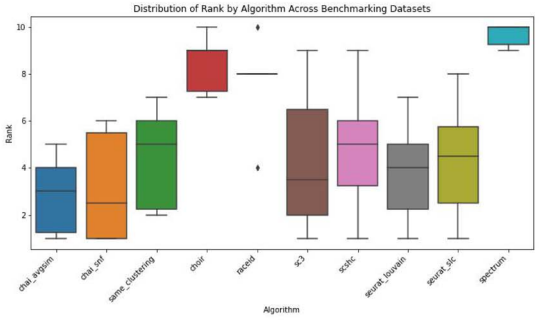
(a) ARI Evaluation for 10 Benchmark Datasets.



(b) Percentage of occurrences where algorithm was a top 3 performer in benchmarking datasets based on ARI



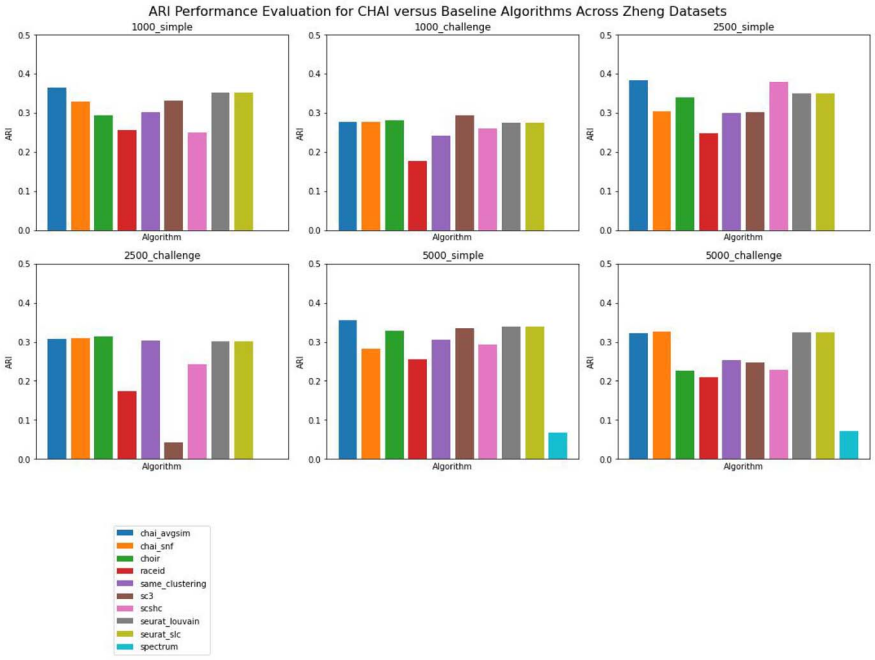
(c) Distribution of ARI by algorithm across benchmarking datasets. CHAI-AvgSim and CHAI-SNF are the first two boxes respectively.



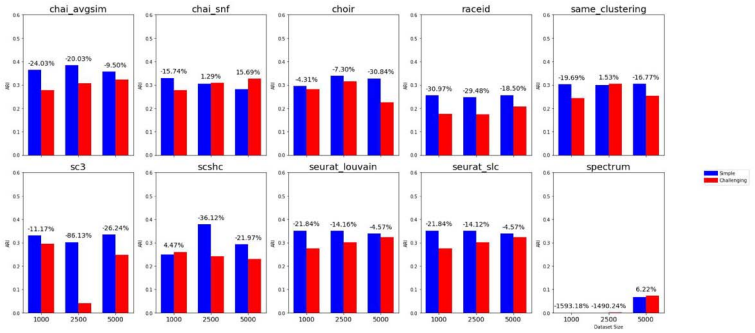
(d) Distribution of rank by algorithm across benchmarking datasets. CHAI-AvgSim and CHAI-SNF are the first two boxes respectively. A lower rank corresponds to better performance.

Figure 3. CHAI evaluation on benchmarking datasets.

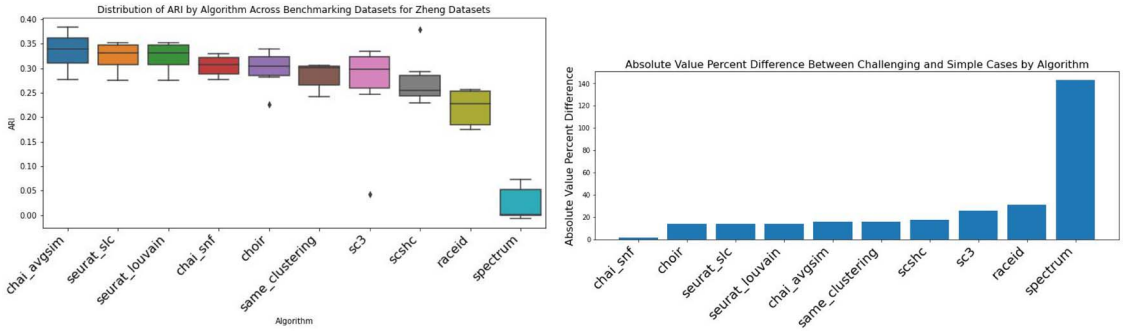
Here, we perform clustering on a dataset from Hwang *et al.*, which studies collective cell migration of breast cancer [32]. During collective migration *in vivo*, breast cancer cells move as a cluster and prior work suggests that cells within the clusters can be heterogeneous [33]. Thus, Hwang *et al.* used single cell sequencing to identify different cell populations within collectively migrating clusters, with the ultimate goal to understand how cells at the front, known as leader cells, may have unique gene signatures that allow them to lead migration. To induce migration, Hwang *et al.* used biochemical and biomechanical



(a) ARI evaluation for Zheng subsampled datasets.



(b) Comparison of performance for each algorithm between simple and challenging datasets. The percent difference between simple and challenging cases for each dataset size is annotated on the graph.



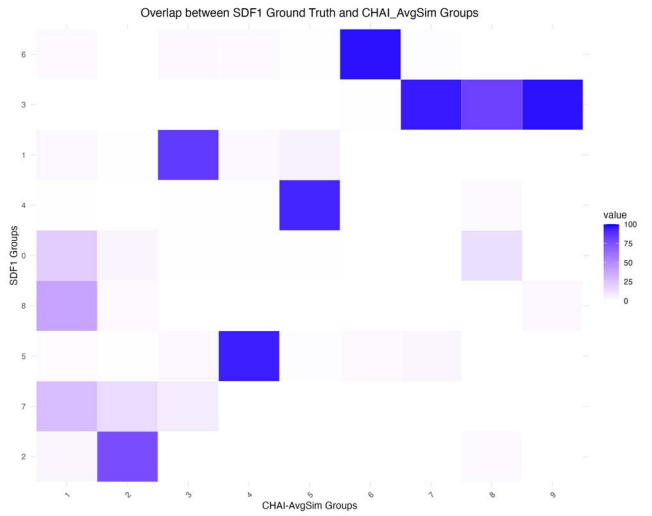
(c) Distribution of ARI for each algorithm across Zheng datasets

(d) Absolute value percent difference between simple and challenging cases for each algorithm.

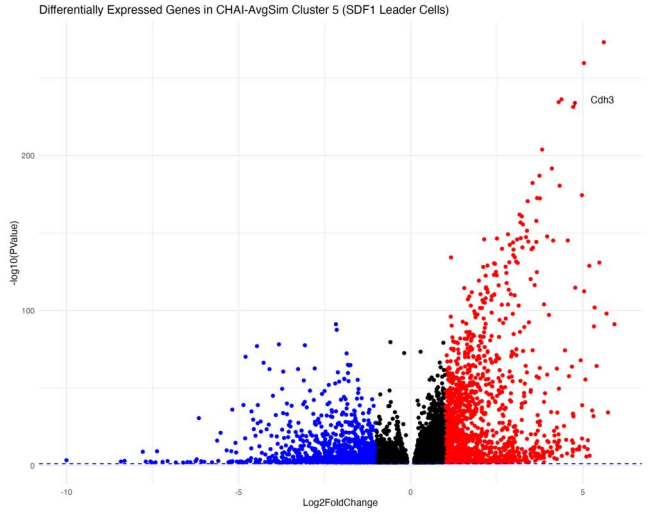
Figure 4. CHAI evaluation on Zheng subsampled datasets.

gradients and performed single cell sequencing analysis after migration had occured (GEO Accession number: GSE171203) [32]. After induction of biochemical gradient stromal-derived factor 1 (SDF1), single cell sequencing analysis of tumor clusters revealed 9 different cell population types and 1 primary cluster of leader cells with differential expression of Cadherin-3 (CDH3) [32].

In our data validation, we analyzed the dataset for the cell clusters migrating in response to the biochemical gradient stromal derived factor 1 (SDF1) and refer to this dataset as 'SDF1'. First, we performed consensus clustering using CHAI-AvgSim on the SDF1 dataset, which also revealed 9 different clusters. To determine how accurately CHAI was able to identify leader cells in the SDF1



(a) Percentage of overlapping cells between Ground Truth and CHAI-AvgSim predicted clusters. The leader cell population in the ground truth is Cluster 4.



(b) Volcano plot of differentially expressed genes in CHAI-AvgSim Cluster 5, which has the highest overlap to the leader cell population in the SDF1 dataset. The CDH3 gene is labeled.

Figure 5. CHAI-AvgSim analysis of CDH3 leader cell population in SDF1-induced migration dataset.

dataset, we compared percentage of shared cells between the ground truth clusters and the clusters predicted by CHAI-AvgSim. In Hwang *et al.*'s single cell analysis, cluster 4 contained the leader cells, and we see in Fig. 5a that cluster 4 has greater than 90% cell overlap with CHAI-AvgSim Cluster 5. In other words, over 90% of the cells predicted to be in Cluster 5 from CHAI-AvgSim are in fact experimentally validated leader cells.

To validate biological relevance of our approach, we calculated DEGs and visualized them using a volcano plot in Fig. 5b. We calculated the DEGs by running the FindAllMarkers function in Seurat [4]. The primary goal behind this analysis was to determine whether CHAI cluster 5 cell population was enriched for CDH3, a demonstrated leader cell marker in the original study [32], as a way to validate cluster 5 is indeed the leader cell population. Our analysis demonstrates that CDH3 is significantly upregulated in the CHAI-AvgSim leader cell cluster, when compared with other clusters. Thus, CHAI-AvgSim was able to accurately identify the leader cell subpopulation distinctly. This study demonstrates the accuracy of CHAI and validates its ability as a method to derive biological insights.

Integration of spatial transcriptomics data with CHAI: CHAI-ST

As CHAI relies on binary matrices to represent cell to cell relationships, we evaluated if other modalities may be integrated into the CHAI framework, provided that they can be represented as binary matrices. Spatial transcriptomics is an emerging sequencing technology that quantifies the location of a cell at the time of sequencing [34]. A recently published method, GraphST, is able to represent the relationship between cells based on their spatial coordinate distance as a binary matrix [24]. We extend this approach from GraphST and easily integrate it into the proposed CHAI framework. The main purpose of this experiment was to quantify if the incorporation of other data modalities to CHAI will improve the overall clustering accuracy.

We present several options to integrate spatial transcriptomics into the CHAI package. For CHAI-AvgSim, we integrated the spatial transcriptomics data by simply including it in the average matrix calculation as another modality. For CHAI-SNF, we first ran SNF on the clustering algorithm binary matrices. We then ran SNF again on the clustering algorithm SNF matrix and the binary matrix from the spatial transcriptomics data, therefore running two levels of SNF. Finally. we run CHAI-SNF-First-Level, in which we incorporate the spatial transcriptomics binary matrix alongside the binary matrices of the other clustering algorithms, and run SNF just once to determine the final clustering assignments [21].

We evaluated CHAI with the integration of spatial transcriptomics coordinates on four datasets using ARI. From this analysis, we find that the integration of spatial transcriptomics with CHAI-SNF improves the ARI in all four datasets. Additionally, we see that the integration of spatial transcriptomics causes either CHAI method to be the top performing algorithm in three out of the four datasets. The ARI for CHAI-AvgSim stays relatively the same when including spatial transcriptomics in most datasets, except for the Vandenbom Liver Cancer dataset, where the integration of the additional data significantly aids its performance. From this analysis, we conclude that it is best to include spatial transcriptomics with CHAI-SNF. We see that incorporating the spatial coordinates separately and running two levels of SNF leads to better ARI in three of the four datasets. There is also no downside to including spatial transcriptomics data with CHAI-SNF or CHAI-AvgSim if available; even if the results do not significantly improve, we see that adding the additional information will still keep the ARI approximately the same.

To evaluate the effectiveness of CHAI-ST as a standalone method, we compared it to GraphST and STGNNKs, two methods for clustering of spatial transcriptomics data [24, 35]. Since the benchmarking results in Fig. 6 show that integrating the spatial transcriptomic results into CHAI-ST-SNF at the second level yielded the best results, we chose to use this method for our evaluation, in addition to CHAI-AvgSim-ST. Sicnce STGNNks relies on 10X Genomics Visium datasets as input, we compared both CHAI-ST methods to the baseline methods on three human DLPFC 10X Visium datasets [35, 36]. These datasets are frequently used for benchmarking of spatial transcriptomic clustering methods, including GraphST since they have experimentally annotated ground truth cluster labels [24]. We chose to evaluate on datasets 151507, 151508, and 151509 [36].

From the results in Fig. 7a. we found that GraphST outperforms both CHAI-ST methods as well as STGNNks in terms of ARI on the Human DLPFC 10X Visium datasets, with a median ARI of 0.43. Additionally, we found that CHAI outperforms STGNNks across the three 10X datasets. We hypothesize that the superior performance of GraphST is due to the fact that it

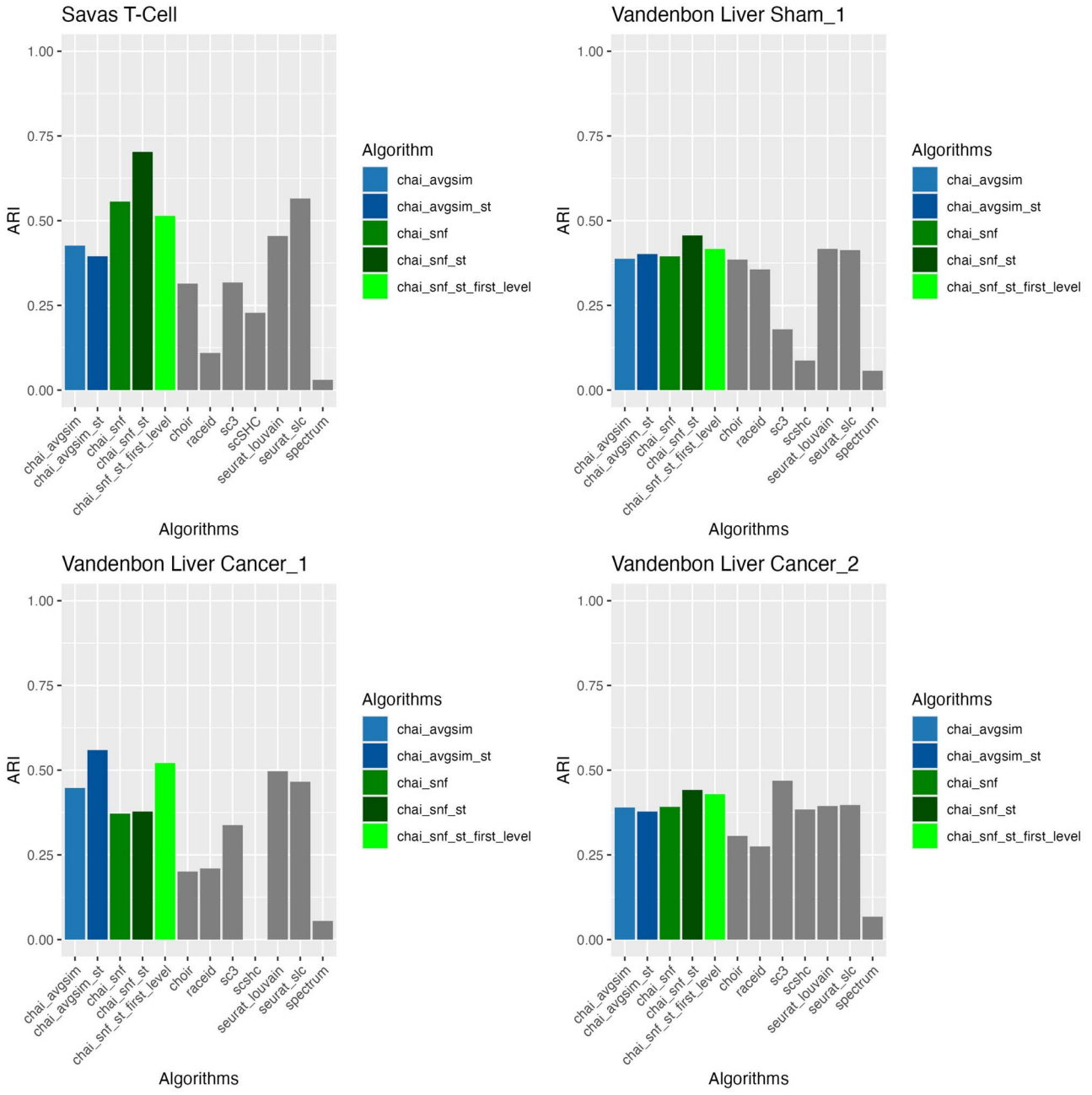


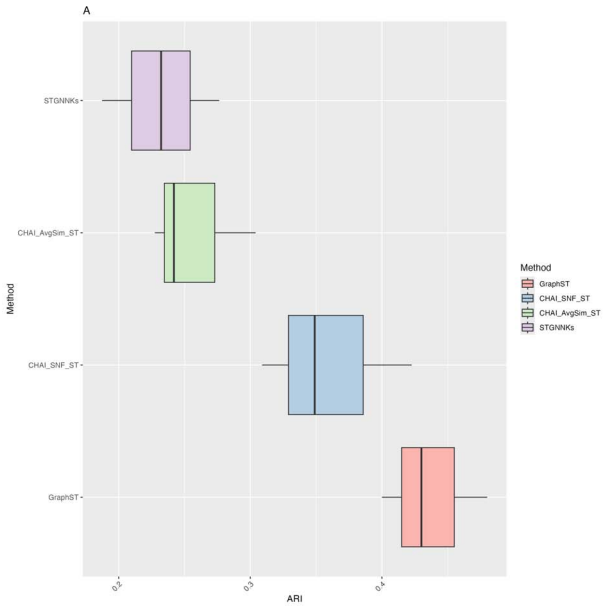
Figure 6. ARI evaluation for CHAI spatial transcriptomic integration; all CHAI spatial transcriptomics integrated are suffixed with '-st' in the bar labels.

incorporates image data into their clustering pipeline, while CHAI and STGNNks do not. When comparing a dataset without images, we demonstrate that in the Savas Breast Cancer dataset, CHAI-SNF-ST outperforms GraphST. We unfortunately were not able to compare STGNNks with this dataset since it is not in the 10X Visium format, which is what that software requires. A further extension of CHAI-ST would be to include image data into the consensus pipeline.

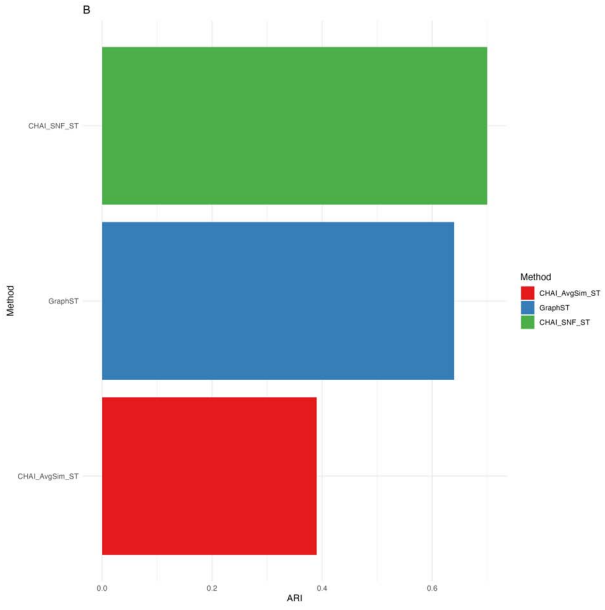
Discussion

Clustering for scRNAseq data is a common task that has a variety of approaches. Each method has their own individual strengths and weaknesses, and there is currently no one best method that works with definitive superiority in all situations. This conclusion has been drawn from several benchmarking studies, including the one we put forward in this study [3]. Other ensemble clustering methods have been applied for scRNAseq data, but these are based on older versions of scRNAseq clustering methods and have not been updated or maintained frequently [19, 20]. With CHAI-AvgSim and CHAI-SNF, we present two distinct consensus clustering methods that each have their own advantages. Both methods demonstrate improved performance on several dataset conditions and complexities.

First, we chose 10 benchmarking datasets to evaluate both CHAI-AvgSim and CHAI-SNF on and compared them with the individual clustering algorithms that made up the consensus pipeline. We found that CHAI-SNF has the highest median ARI across all of the dataset runs, and the highest maximum ARI as well. However, CHAI-AvgSim demonstrates comparable median ARI while also having the lowest interquartile range out of all of the other algorithms. This, combined with the fact that



(a) Comparison of both CHAI-ST methods to GraphST and STGNNks based on ARI on three Human DLPFC 10X Visium Datasets.



(b) Comparison of CHAI-ST to GraphST on the Savas Breast Cancer dataset, which contains spatial coordinates but not cell image data. Evaluation was performed based on ARI.

Figure 7. CHAI-ST benchmarking on human DLPFC 10X visium datasets and Savas breast cancer dataset.

CHAI-AvgSim is a top three performer in 80% of all benchmarking datasets, suggests that it is a more consistent and safer choice to use when the exact structure of a dataset is not known. We note the variation across all of the datasets in most of the algorithms. The previous consensus clustering method we chose to compare to, SAME clustering, has a similar median ARI and interquartile range when compared with both CHAI-AvgSim and CHAI-SNF. However, it has a much lower median rank and does not feature as regularly in the list of top 3 performers across datasets. When evaluated on simple and challenging cases, both CHAI-AvgSim and CHAI-SNF show consistency between the two cases. We note that CHAI-SNF has a significant percent difference between its simple and challenging cases, across all dataset sizes. From this analysis, we are able to conclude that CHAI-SNF is least susceptible to varying performance as dataset complexities increase.

When comparing both CHAI methodologies to SAME-Clustering, it is important to note that we used the current version of SAME-Clustering available, where SC3 does not run in its package due to a bug (see: https://github.com/yycunc/SAMEclustering/issues/4. Therefore, SC3 is included in our pipeline, while not being included in SAME-Clustering's in all of the evaluations we conducted [5, 20]. Despite this fact, we are still confident of CHAI's performance as it incorporates several other algorithms that are not included in SAME-Clustering. Users may also notice Spectrum's poor performance, often displaying subzero and negative ARI [6]. We included Spectrum anyways to demonstrate that CHAI's performance is overall unaffected by a singular poor performing algorithm, provided that the rest of the algorithms demonstrate a reasonable accuracy. As more clustering algorithms are added and the community continues to see variable performances, CHAI will remain to be a stable choice unlikely to be influenced by one singular extremely poor performing algorithm.

When gold standard cell types are not available, we sought to demonstrate CHAI's practical usability for identifying important clusters and biomarkers in a real-world application. We found that CHAI was able to identify a CDH3-enriched cell population which has been linked to leading cell migration in breast cancer [32]. This demonstrates that not only does CHAI have a better performance in terms of accuracy it is also able to derive biologically meaningful results.

As multiomic data for single cell genomics increase, the need to integrate this information will continue to arise [37]. In this study, we choose spatial transcriptomic coordinate data as an example for multiomic integration with CHAI. Using a binary similarity matrix method developed from GraphST, we show that adding this additional omic to CHAI-AvgSim increases it significantly in one benchmarking dataset and keeps performance relatively the same in the other datasets [24]. For CHAI-SNF on the other hand, the integration of spatial transcriptomic data increases the performance in all cases. As the original purpose of SNF was to integrate disparate modes of data for the same sample, this makes CHAI-SNF a logical choice for this purpose [21]. The nature of CHAI allows for it to accommodate other forms of data, so long as they can be represented as a binary similarity matrix between cells. This makes it a generalized method for not only standard clustering, but multiomic clustering as well. The flexibility of the binary matrix architecture will lend CHAI usable in a variety of different purposes going forward.

We have found that both CHAI methods outperform existing baseline methods on a variety of datasets in terms of size, complexity, and number of cell-types. Additionally, both CHAI methods demonstrate the least percent change between simple and challenging dataset subsamples from the Zheng 68k dataset [30]. In fact, we found that CHAI-SNF actually improves its performance for challenging datasets. CHAI also shows a performance improvement when integrated with other 'omics' of data, in this case spatial transcriptomics coordinates. For these advancements, CHAI provides value as a software package that can be used as is by the community and will continue to be useful in the future as more advanced clustering algorithms and 'omics' representations develop.

An important consideration is deciding which CHAI method to use; based on our evaluation, we make the recommendation to users to use CHAI-AvgSim for the majority of datasets

and conditions. This is due to CHAI-AvgSim's superior performance in terms of median ARI and smaller variation across several diverse benchmarking datasets. However, CHAI-SNF is the superior method for multi-omic integration, as it demonstrated improved performance against CHAI-AvgSim when integrating spatial transcriptomics data.

Further evaluation remains to be done on the best algorithms to use in the consensus pipeline for a particular dataset conditions. An immediate limitation of CHAI is that it is not currently possible to select an ideal set of algorithms to be used in the final consensus, as the individual algorithms demonstrate large variation in performance. Even in very obvious cases of poor performance, such as Spectrum on the Baron dataset evaluations in Fig. 3a, dropping Spectrum led to very negligible changes in performance. As more robust and consensus algorithms are created, CHAI will maintain its success as an integration method, and this will alleviate concerns regarding the performance of individual algorithms. In these instances, we aim for CHAI to be customizable, where several algorithms can be added or removed based on user preference. Ideally, these choices will be informed by community best practices. However, based on current evaluations, it is our recommendation to include as many algorithms as possible.

Conclusion

We present CHAI, a consensus clustering method demonstrating robust and superior performance in a wide variety of dataset conditions for scRNA-seq data. CHAI is able to detect key biomarkers in cancer tumor cells; additionally, CHAI provides a platform for multiomic integration. We hope that CHAI is a tool for the community, where new algorithms may be integrated seamlessly and other omics are built into the pipeline.

Data

Baron pancreas data

Baron *et al.* addresses the limitations of previous gene expression profiling in the pancreas by using a droplet-based, single-cell RNA sequencing method to analyze over 12 000 individual pancreatic cells from four human donors and two mouse strains [38]. The analysis demonstrated 15 distinct clusters of cells, including subpopulations which were validated through immunohistochemistry. Additionally, heterogeneity was observed within human beta-cells, highlighting differences in gene regulation related to functional maturation and endoplasmic reticulum stress. Leveraging single-cell data, the researchers detected disease-associated differential expression and identified novel cell type-specific transcription factors and signaling receptors [38]. Over the years, the Baron dataset has served as a resource for validating and comparing findings in single-cell RNA sequencing studies because it is a large dataset with a view of gene expression patterns across distinct cell types [39]. You may download the data through GEO with accession number GSE84133.

Muraro pancreas data

Few proteins uniquely distinguish cells within the pancreas, creating a challenge because traditional techniques such as immunohistochemistry rely on specific markers and may not sufficiently distinguish various cell populations. Muraro *et al.* describes using an automated platform that combines Fluorescence-Activated Cell Sorting (FACS), robotics, and the CEL-Seq2 sequencing protocol [40]. This approach allowed them to obtain transcriptomes from thousands of single pancreatic cells from deceased organ donors. As a result, they were able to identify cell type-specific transcription factors, discover a subpopulation of REG3A-positive acinar cells, and establish CD24 and TM4SF4 as markers for sorting alpha and beta cells. (GEO accession number: GSE85241).

SC-Mixology data

The SC-Mixology dataset involves three human lung adenocarcinoma cell lines: HCC827, H1975, and H2228. Single cells from each cell line were processed using CEL-seq2, Drop-seq, and 10X Chromium library preparation methods then sorted into 384-well plates. Additionally, bulk RNA from each cell line was mixed in different ratios, diluted to single-cell equivalents, and sequenced [29]. The data are downloadable from the authors' Github: https://github.com/LuyiTian/sc_mixology.

Zeisel mouse brain

Zeisel *et al.* utilized single-cell RNA sequencing to analyze 3436 mouse brain and 1504 lung cell transcriptomes, aiming to understand vascular diseases. They identified 15 distinct cell clusters in the brain cortex and hippocampus and 17 in the lung, providing insight on tissue cellular diversity and organization [41] (GEO accession number: GSE103840).

Zheng 68K PBMC data

The Zheng68K dataset by 10X CHROMIUM is a large dataset consisting of 68 450 blood mononuclear cells. The dataset was developed using an adaption of GemCode single-cell technology. There are eleven subtypes of cells within this dataset, those being CD8+ cytotoxic T cells (30.3%), CD8+/CD45RA+ naive cytotoxic cells (24.3%), CD56+ NK cells (12.8%), CD4+/CD25 T Reg cells (9.0%), CD19+ B cells (8.6%), CD4+/CD45RO+ memory cells (4.5%), CD14+ monocyte cells (4.2%), dendritic cells (3.1%), CD4+/CD45RA+/CD25- naive T cells (2.7%), CD34+ cells (0.4%), and CD4+ T Helper2 cells (0.1%). For CHAI benchmarking, we took advantage of the diversity contained in the Zheng68K dataset by subsampling it into six smaller datasets, those being:

1. 1000 cells with 5 equal populations
2. 1000 cells with random populations
3. 2500 cells with 5 equal populations
4. 2500 cells with random populations
5. 5000 cells with 5 equal populations
6. 5000 cells with random populations.

From this subsampling analysis, we were able to benchmark CHAI against varying dataset conditions and controls [30]. We consider the datasets with equal populations to be 'simple' datasets and with random groups to be 'challenging' datasets.

Savas breast cancer T Cell Data

Savas *et al.* [42] studied the characteristics of T cells in breast cancer tumor-infiltrating lymphocytes (TILs). Multi-parameter flow cytometry was utilized to analyze breast cancers for their TIL content. Data were obtained from 84 individuals with primary breast cancers and 45 individuals with metastatic breast cancers. The findings revealed significant heterogeneity in the infiltrating T cell population and suggested that CD8+ tissue resident memory T (TRM) cells contribute to breast cancer immunosurveillance and are primarily modulated by immune checkpoint inhibition.

The dataset used in this paper was obtained by performing single cell RNA sequencing on 5759 purified CD3+ single T cells passing quality control from two primary triple negative breast

cancer (TNBC) patients, encompassing a total of 15 623 genes and 11 different gene expression annotations. The spatial coordinates of the cells obtained from the tissue are also recorded. Data used can be downloaded from Broad Institute's Single Cell Portal with accession number SCP2331.

Vandenbon mouse liver cancer visium data

Zonation refers to the spatial organization of gene expression within the liver such that hepatocyte functions are specified by relative distance to the bloodstream. In [43], Vandenbon *et al.* utilized spatial transcriptomics in order to investigate the quantity and zonation of hepatic genes in mice with cancer with the intention of determining whether liver zonation is influenced by solid cancers. This study found that liver zonation was influenced by breast cancers, exemplified by affected xenobiotic catabolic process genes, zonally elicited acute phase response, and zonally activated innate immune cells in the liver. Breast cancers zonally influencing liver gene expression profiles results in zonal liver functions also being affected. Data for this study were obtained from wild-type female mice. Four mouse liver samples consisting of two 4T1 cancer-bearing mice samples, Cancer1 and Cancer2, and two sham samples, Sham1 and Sham2, were processed with 10x Genomics Visium spatial transcriptomics, culminating in a dataset with a total of 7758 spots and 32 285 genes clustered into 13 cell type categories.

For this case study, the Cancer1 (2110 spots), Cancer2 (1438 spots), and Sham1 (1952 spots) samples were utilized. The data used can be downloaded from Broad Institute's Single Cell Portal with accession number SCP2046.

Key Points

- Several clustering methods have emerged for scRNAseq data; however, there is no consensus on the true 'best' method to use in all cases.
- We present CHAI, a clustering algorithm that uses a wisdom of crowds approach to integrate the results from several different clustering algorithms into one composite clustering assignment.
- CHAI demonstrates improved performance on several benchmarking datasets, including outperforming previous consensus clustering methods. CHAI also provides a platform for the integration of multi-omic data, which we demonstrate using spatial transcriptomics.

Conflict of interest: None declared.

Funding

This work was partially supported by 5R21MH128562-02 (PI: Roberson-Nay), 5R21AA029492-02 (PI: Roberson-Nay), CHRB-2360623 (PI: Das), NSF-2316003 (PI: Cano), VCU Quest (PI: Das), and VCU Breakthroughs (PI: Ghosh) funds awarded to P.G.

Code availability

CHAI is available as an R package here: https://github.com/lodimk2/chai.

References

1. Xiaojun W, Yang B, Udo-Inyang I. et al. Research techniques made simple: single-cell RNA sequencing and its applications in dermatology. *J Invest Dermatol* 2018;**138**:1004–9.
2. Zhang S, Li X, Lin J. et al. Review of single-cell RNA-seq data clustering for cell-type identification and characterization. *RNA* 2023; **29**:517–30. https://doi.org/10.1261/rna.078965.121.
3. Lijia Y, Cao Y, Yang JYH. et al. Benchmarking clustering algorithms on estimating the number of cell types from single-cell RNA-sequencing data. *Genome Biol* 2022; **23**:49. https://doi.org/10.1186/s13059-022-02622-0.
4. Butler A, Hoffman P, Smibert P. et al. Integrating single-cell transcriptomic data across different conditions, technologies, and species. *Nat Biotechnol* 2018; **36**:411–20. https://doi.org/10.1038/nbt.4096.
5. Kiselev VY, Kirschner K, Schaub MT. et al. Sc3: consensus clustering of single-cell rna-seq data. *Nat Methods* 2017; **14**:483–6. https://doi.org/10.1038/nmeth.4236.
6. John CR, Watson D, Barnes MR. et al. Spectrum: fast density-aware spectral clustering for single and multi-omic data. *Bioinformatics* 2020; **36**:1159–66. https://doi.org/10.1093/bioinformatics/btz704.
7. Lin P, Troup M, Ho JWK. CIDR: ultrafast and accurate clustering through imputation for single-cell RNA-seq data. *Genome Biol* 2017; **18**:59.
8. Petersen C, Mucke L, Ryan Corces M. Choir improves significance-based detection of cell types and states from single-cell data. Biorxiv. 2024.
9. Grabski IN, Street K, Irizarry RA. Significance analysis for clustering with single-cell RNA-sequencing data. *Nat Methods* 2023; **20**:1196–202. https://doi.org/10.1038/s41592-023-01933-9.
10. Steinley D. Properties of the hubert-arable adjusted rand index. *Psychol Meth* 2004; **9**:386–96. https://doi.org/10.1037/1082-989X.9.3.386.
11. Chaitankar V, Ghosh P, Perkins E. et al. A novel gene network inference algorithm using predictive minimum description length approach. *BMC Syst Biol* 2010; **4**:S7. https://doi.org/10.1186/1752-0509-4-S1-S7.
12. Chaitankar V, Ghosh P, Perkins E. et al. Time lagged information-theoretic approaches to the reverse engineering of gene regulatory networks. *BMC Bioinformatics* 2010; **11**:S19. https://doi.org/10.1186/1471-2105-11-S6-S19.
13. Vega-Pons S, Ruiz-Shulcloper J. A survey of clustering ensemble algorithms. *Int J Pattern Recognit Artif Intell* 2011; **25**:337–72. https://doi.org/10.1142/S0218001411008683.
14. Hamada D, Nakayama M, Saiki J. Wisdom of crowds and collective decision-making in a survival situation with complex information integration. *Cogn Res: Princ Implic* 2020; **5**:48. https://doi.org/10.1186/s41235-020-00248-z.
15. Nalluri JJ, Barh D, Azevedo V. et al. Mirsig: a consensus-based network inference methodology to identify pan-cancer mirna-mirna interaction signatures. *Sci Rep* 2017; **7**:39684. https://doi.org/10.1038/srep39684.
16. Nalluri J, Rana P, Barh D. et al. Determining causal mirnas and their signaling cascade in diseases using an influence diffusion model. *Sci Rep* 2017; **7**:8133. https://doi.org/10.1038/s41598-017-08125-4.
17. Strehl A, Ghosh J. Cluster-ensembles: a knowledge reuse framework for combining multiple partitions. *J Mach Learn Res* 2002; **3**:583–617.

18. Geddes TA, Kim T, Nan L. et al. Autoencoder-based cluster ensembles for single-cell rna-seq data analysis. *BMC Bioinformatics* 2019; **20**:660. https://doi.org/10.1186/s12859-019-3179-5.
19. Yang Y, Huh R, Culpepper HW. et al. SAFE-clustering: single-cell aggregated (from ensemble) clustering for single-cell RNA-seq data. *Bioinformatics* 2019; **35**:1269–77. https://doi.org/10.1093/bioinformatics/bty793.
20. Huh R, Yang Y, Jiang Y. et al. SAME-clustering: single-cell aggregated clustering via mixture model ensemble. *Nucleic Acids Res* 2020; **48**:86–95. https://doi.org/10.1093/nar/gkz959.
21. Wang B, Mezlini AM, Demir F. et al. Similarity network fusion for aggregating data types on a genomic scale. *Nat Methods* 2014; **11**:333–7. https://doi.org/10.1038/nmeth.2810.
22. Grün D, Lyubimova A, Kester L. et al. Single-cell messenger rna sequencing reveals rare intestinal cell types. *Nature* 2015; **525**:251–5. https://doi.org/10.1038/nature14966.
23. Ng AY, Jordan MI, Weiss Y. On spectral clustering: analysis and an algorithm. In: Leen TK, Dietterich TG, Tresp V, editors. *Advances in Neural Information Processing Systems*. Cambridge, MA: MIT Press, 2001;849–56.
24. Long Y, Ang KS, Li M. et al. Spatially informed clustering, integration, and deconvolution of spatial transcriptomics with GraphST. *Nat Commun* 2023; **14**:1155. https://doi.org/10.1038/s41467-023-36796-3.
25. Warrens MJ, van der Hoef H. Understanding the adjusted rand index and other partition comparison indices based on counting object pairs. *J Classif* 2022; **39**:487–509. https://doi.org/10.1007/s00357-022-09413-z.
26. Zhang P. Evaluating accuracy of community detection using the relative normalized mutual information. *J Stat Mech Theor Exp* 2015; **2015**:P11006. https://doi.org/10.1088/1742-5468/2015/11/P11006.
27. Pouyan MB, Kostka D. Random forest based similarity learning for single cell rna sequencing data. *Bioinformatics* 2018; **34**:i79–88. https://doi.org/10.1093/bioinformatics/bty260.
28. Zeisel A, Hochgerner H, Lönnerberg P. et al. Molecular architecture of the mouse nervous system. *Cell* 2018; **174**:999–1014.e22. https://doi.org/10.1016/j.cell.2018.06.021.
29. Tian L, Dong X, Freytag S. et al. Benchmarking single cell rna-sequencing analysis pipelines using mixture control experiments. *Nat Methods* 2019; **16**:479–87. https://doi.org/10.1038/s41592-019-0425-8.
30. Zheng GXY, Terry JM, Belgrader P. et al. Massively parallel digital transcriptional profiling of single cells. *Nat Commun* 2017; **8**:14049.
31. Zhiyuan H, Ahmed AA, Yau C. CIDER: an interpretable meta-clustering framework for single-cell RNA-seq data integration and evaluation. *Genome Biol* 2021; **22**:337.
32. Hwang PY, Mathur J, Cao Y. et al. A cdh3-β-catenin-laminin signaling axis in a subset of breast tumor leader cells control leader cell polarization and directional collective migration. *Dev Cell* 2023; **58**:34–50.e9. https://doi.org/10.1016/j.devcel.2022.12.005.
33. Hwang PY, Brenot A, King AC. et al. Randomly distributed k14+ breast tumor cells polarize to the leading edge and guide collective migration in response to chemical and mechanical environmental cues. *Cancer Res* 2019; **79**:1899–912. https://doi.org/10.1158/0008-5472.CAN-18-2828.
34. Williams CG, Lee HJ, Asatsuma T. et al. An introduction to spatial transcriptomics for biomedical research. *Genome Med* 2022; **14**:68. https://doi.org/10.1186/s13073-022-01075-1.
35. Peng L, He X, Peng X. et al. Stgnnks: identifying cell types in spatial transcriptomics data based on graph neural network, denoising auto-encoder, and. *Comput Biol Med* 2023; **166**:107440. https://doi.org/10.1016/j.compbiomed.2023.107440.
36. Maynard KR, Collado-Torres L, Weber LM. et al. Transcriptome-scale spatial gene expression in the human dorsolateral prefrontal cortex. *Nat Neurosci* 2021; **24**:425–36. https://doi.org/10.1038/s41593-020-00787-0.
37. Cao Z-J, Gao G. Multi-omics single-cell data integration and regulatory inference with graph-linked embedding. *Nat Biotechnol* 2022; **40**:1458–66. https://doi.org/10.1038/s41587-022-01284-4.
38. Baron M, Veres A, Wolock SL. et al. A single-cell transcriptomic map of the human and mouse pancreas reveals inter- and intra-cell population structure. *Cell Systems* 2016; **3**:346–360.e4. https://doi.org/10.1016/j.cels.2016.08.011.
39. Cheng Y, Fan X, Zhang J. et al. A scalable sparse neural network framework for rare cell type annotation of single-cell transcriptome data. *Commun Biol* 2023; **6**:545. https://doi.org/10.1038/s42003-023-04928-6.
40. Muraro MÂJ, Dharmadhikari G, Grün D. et al. A single-cell transcriptome atlas of the human pancreas. *Cell Syst* 2016; **3**:385–394.e3. https://doi.org/10.1016/j.cels.2016.09.002.
41. Zeisel A, Muñoz-Manchado AB, Codeluppi S. et al. Cell types in the mouse cortex and hippocampus revealed by single-cell rna-seq. *Science* 2015; **347**:1138–42. https://doi.org/10.1126/science.aaa1934.
42. Savas P, Virassamy B, Ye C. et al. Single-cell profiling of breast cancer t cells reveals a tissue-resident memory subset associated with improved prognosis. *Nat Med* 2018; **24**:1941. https://doi.org/10.1038/s41591-018-0176-6.
43. Vandenbon A, Mizuno R, Konishi R. *et al*. Murine breast cancers disorganize the liver transcriptome in a zonated manner. *Commun Biol* 2023; **6**:97. https://doi.org/10.1038/s42003-023-04479-w.