Distribution Agreement

In presenting this thesis or dissertation as a partial fulfillment of the requirements for an advanced degree from Emory University, I hereby grant to Emory University and its agents the non-exclusive license to archive, make accessible, and display my thesis or dissertation in whole or in part in all forms of media, now or hereafter known, including display on the world wide web. I understand that I may select some access restrictions as part of the online submission of this thesis or dissertation. I retain all ownership rights to the copyright of the thesis or dissertation. I also retain the right to use in future works (such as articles or books) all or part of this thesis or dissertation.

Signature:

_____ _____

Jiahui Liu Date

A benchmark of rare cell type detection methods for single-cell RNA sequencing data

By

Jiahui Liu

Master of Science in Public Health

Department of Biostatistics and Bioinformatics

_____

Hao Wu, PhD

(Thesis Advisor)

_____

Steve Qin, PhD

(Reader)

A benchmark of rare cell type detection methods for single-cell RNA sequencing data

By

Jiahui Liu

B.S., Huazhong Agricultural University, 2016

Thesis Committee Chair: Hao Wu, PhD

An abstract of
A thesis submitted to the Faculty of the
Rollins School of Public Health of Emory University
in partial fulfillment of the requirements for the degree of
Master of Science in Public Health
in Biostatistics
2023

Abstract

A benchmark of rare cell type detection methods for single-cell RNA sequencing data

By Jiahui Liu

Background: A key task in single-cell RNA-seq (scRNA-seq) data analysis is to detect the rare cell types in the sample, which can be critical for downstream analyses such as differential gene analysis. Various scRNA-seq data detecting rare cell type algorithms have been specifically designed to automatically estimate the rare cell types through define rareness score or optimizing the clustering method. The lack of benchmark studies, however, complicates the choice of the methods.

Results: We conducted a comprehensive evaluation of several widely used algorithms for detecting rare cell types. To assess their accuracy and consistency, we sampled data from European Genome-Phenome Archive (EGA) and evaluated their performance on a range of scRNA-seq datasets with different samples. Additionally, we integrated multiple samples to test the algorithms' population-level performance. Using a set of criteria, including clustering improvement methods and customization of the rareness score, we evaluated the algorithms' performance from various aspects and drew our conclusions based on this benchmarking work. Our evaluation was based on a large number of datasets, providing us with valuable insights into the suitability of these algorithms for identifying rare cell types.

Conclusion: We identified the strengths and weaknesses of each method based on a variety of criteria, including detection accuracy, precision, Cohen's kappa, sensitivity, and specificity at the individual and population levels based on predefined rare cell types, as well as a comparison of runtime and peak memory. We then aggregate these results into multifaceted recommendations for users.

A benchmark of rare cell type detection methods for single-cell RNA sequencing data

By

Jiahui Liu

B.S., Huazhong Agricultural University, 2016

Thesis Committee Chair: Hao Wu, PhD

An abstract of
A thesis submitted to the Faculty of the
Rollins School of Public Health of Emory University
in partial fulfillment of the requirements for the degree of
Master of Science in Public Health
in Biostatistics
2023

Acknowledgement

I would like to express my sincere gratitude to the Rollins School of Public Health at Emory University for their invaluable support throughout my study and research. Over the course of two years, the knowledge and experience I gained in the field of biostatistics greatly facilitated my progress and growth as a researcher and provided me with the confidence to pursue my PhD in biostatistics.

I am deeply grateful to my thesis advisor, Hao Wu, for his invaluable guidance and patient teaching in the areas of statistical computing, machine learning and single-cell RNA sequencing data analysis. His mentorship not only played a key role in shaping my academic journey but also guided my future research directions effectively.

I would also like to extend my thanks to our team members, Luxiao Chen and Wenjing Ma, for their outstanding assistance in my thesis. Their collaboration not only helped me overcome various challenges in my research but also provided me with a platform to discuss results and explore new ideas.

Lastly, I would like to express my heartfelt appreciation to my parents and my girlfriend Yanan Wang for their unwavering support during my study abroad for my master's degree. Their unwavering belief in my ability and unwavering encouragement laid the foundation for my success in this program.

Table of Contents

1. Introduction

Single-cell mRNA sequencing (scRNA-seq) has emerged as a transformative technology that allows for the simultaneous measurement of gene expression at the individual cell level, enabling researchers to capture the complexity and heterogeneity of biological systems[1].This sequencing technology has significantly advanced our understanding of cell-type composition within complex tissues. For example, in human blood tissue, we can now identify and distinguish between various cell types such as B-cells, T-cells, and monocytes with greater precision and resolution than ever before[2]. Moreover, scRNA-seq has facilitated the exploration of relationships between different cell types and has provided new insights into the underlying biological mechanisms that govern cellular differentiation and function, which has further ignited research interest in this field [3]. In recent years, there has been growing interest in detecting rare cell types that exist at low frequencies, particularly those that play crucial roles in human disease and development, such as drug-resistant cells and cancer-initiating cells[4, 5]. However, detecting rare cell types in scRNA-seq data presents significant challenges due to the high dimensionality of the data, with thousands of genes and cells in a single dataset, and the sparsity of the expression matrix where most of the data are zeros. Therefore, developing effective methods for rare cell type detection is critical for advancing our understanding of complex biological systems.

Numerous software methods have been developed to detect rare cell types in scRNA-seq data at the individual level. GiniClust[6], CellSIUS[7], RaceID[8], and scAIDE[9] are among the methods that employ unsupervised clustering algorithms followed by assignment steps to identify rare cell types. FiRE[10], on the other hand, employs an algorithmic approach that directly assigns a rarity score to each cell without clustering.

While these methods have demonstrated effectiveness in detecting rare cell types at the individual level, detecting rare cell types at the population level is a different story and poses several challenges. For instance,

there are batch effects of each sample in different experimental conditions, as well as biological differences between different individuals. Additionally, it remains unclear whether rare cell types identified at the individual level are still considered rare cell types in the population, and distinguishing true rare cell types from outliers at the population level is also a significant challenge.

Currently, Yu et al. compared the number of cell types were estimated by different cluster algorithms[11]. Fa et al. benchmarked on sensitivity of different approaches to cell type identity[12]. However, there is a lack of studies that comprehensively compare the consistency of multiple rare cell type detection algorithms at both the population and individual levels. While publications describing new methods do benchmark against existing approaches, these comparisons often only focus on whether the methods can detect rare cell types, rather than their consistency at the individual and population levels. Moreover, the definition of rare cell types differs between different methods, leading to different interpretations. Therefore, our study aims to comprehensively and objectively evaluate rare cell type detection methods developed for scRNA-seq data, with a focus on their consistency. Specifically, we tested the following methods: GiniClust3, CellSIUS, RaceID, scAIDE and FiRE. To conduct our evaluation, we employed the COVID-19 dataset from European Genome-Phenome Archive (EGA) and selected 10 samples for simulation. We used Combat[13] to remove the batch effect.

2. Methods

2.1 Data collection and preprocessing

We obtained processed read counts for COVID-19 scRNA-seq PBMC studies conducted by Schulte-Schrepping et al. and Su et al. by downloading data from the European Genome-Phenome Archive (EGA) and ArrayExpress database, respectively[14, 15]. The accession numbers provided in the original publications, EGAS00001004571 and E-MTAB-9357, were used to access the data. Then we used Seurat[16] in R to store data from 10 individuals. For each sample, we collected information such as disease stage, sex, and cell type. We merged cell type labels in the Seurat data by combining similar cell types. For

instance, we merged “Classical Monocytes”, “HLA-DR+ CD83+ Monocytes”, “CD163+ Monocytes”, “HLA-DR- S100A+ monocytes” and “Non-classical Monocytes” labels into one cell type called “Monocytes”. We obtained gene expression matrixes for each subject, and we analyzed the clustering results for each individual and combined data. To correct batch effects in the combined data, we used Combat.

2.2 Batch Effect Correction

To correct for batch effects in our scRNA-seq data, we used the ComBat algorithm[17], which has been successfully applied to scRNA-seq data. First, we employed a negative binomial regression model to estimate batch effects using the above count scRNA-seq matrix. Let the expression count value for gene g of sample j from batch i be denoted by y_{gij}. Therefore, we could assume $y_{gij} \sim NB(\mu_{gij}, \phi_{gi})$, where μ_{gij} is the mean and ϕ_{gi} is the dispersion parameter. Then for a certain gene g, in sample j and batch i, we could get a gene-wise model:

$$log\mu_{gij} = \alpha_g + X_j\beta_g + \gamma_{gi} + logN_j$$

$$var(y_{gij}) = \mu_{gij} + \phi_{gi}\mu_{gij}^2$$

Where α_g is the average level for gene g. $X_j\beta_g$ donates the biological condition of sample j. N_j reflects the total counts across all genes in sample j. Next, we used the established methods in edgeR to estimate the batch effect parameter [18, 19].

Following the modeling process, we obtained the estimated batch effect parameters $\hat{\gamma}_{gi}$ and $\hat{\phi}_{gi}$, along with the fitted expectation of the count $\hat{\mu}_{gij}$. Next, we calculated parameters for batch-free distributions as follows: Assuming that the adjusted data $y_{gj}^* \sim NB(\mu_{gj}^*, \phi_g^*)$. Then we could calculate the following formula:

$$log\mu_{gj}^* = log\hat{\mu}_{gij} - \hat{\gamma}_{gi}$$

$$\phi_g^* = \frac{1}{N_{batch}} \sum_i \hat{\phi}_{gi}$$

Finally, we calculated the adjusted data y_{gj}^* by identifying the closest quantile on the batch-free distribution to the quantile of the original data y_{gij} on the empirical distribution. In our benchmarking, we used each subject's data as a specific batch label and processed our data based on this label to correct the batch effect. The resulting adjusted gene expression matrix was used for downstream analysis to detect rare cell types at the population level.

2.3 Rare Cell Types Detecting

Table 1 presents a summary of the key features of the six methods tested for detecting rare cell types. Most of these methods employ unsupervised clustering to identify subtype cells and define rare cell types, with the exception of FiRE. The unsupervised methods do not require any prior information about cell types. FiRE identifies rare cell type cells by computing a rareness score and applying IQR-based thresholding criteria, but it does not use hierarchical or density-based cluster methods to flag outliers. In this project, we utilized FiRE in the without clustering mode, and all other methods in the unsupervised mode without any cell type information. We provide a brief description of each method below.

Table1. Key characteristics of each method.

Tools	Programming Language	Detecting Rare cell type output	Method
GiniClust3	Python	Cluster label	Gini and Fano index clustering
CellSIUS	R	Cluster label	k-means and Markov Cluster
RaceID	R	Cluster label	k-means and regression
scAIDE	Python, R	Cluster label	Embedding and RPH-kmeans
FiRE	R	Rare or normal label	Calculating rareness score

2.3.1 Detecting rare cell type based on Gini and Fano index (GiniClust3)

GiniClust3 is an improved version of GiniClust, designed to be faster and more memory-efficient than previous versions[20]. GiniClust3 employs both Gini index-based features and Fano factor-based features to cluster cells. Initially, GiniClust3 clusters all cells based on Gini index-based features. For each gene, the raw Gini index is computed as twice the area between the Lorenz curve and the diagonal, taking a value between 0 and 1. Subsequently, the raw Gini index values are normalized by eliminating the maximum expression levels of the trends, using a two-step LOESS regression procedure. Genes whose Gini index value is greater than 0.6 and p-value less than 0.0001 are labeled as high Gini genes and used for subsequent analysis. Instead of using DBSCAN as in the previous version, GiniClust3 employs the Leiden algorithm, which is suitable for large datasets, for the clustering step[21].

The Fano factor, which is the variance of the mean expression value for each gene, is used in the analysis. The highly variable genes for the subsequent analysis were identified using Scanpy by default [22]. The gene expression data was then dimensionally reduced using principal component analysis (PCA), followed by Leiden or Louvain clustering. Instead of using cell-level analysis, a consensus matrix was generated based on the cluster level of the Gini and Fano clustering methods. If two cells were clustered in the same group, the connectivity was assigned a value of 1; otherwise, it was assigned a value of 0. The consensus matrix was subjected to k-means clustering, and the resulting clusters were converted back to single-cell level clustering. Finally, clusters with a cell population less than 1% were defined as rare cell clusters.

2.3.2 Detecting rare cell types based on correlated gene with MCL (CellSIUS)

CellSIUS initially divides N cells into m clusters C_1, …, C_m, and then identifies cell subpopulations and their characteristics as follows. The first step is to identify genes with bimodal expression: for each gene, one-dimensional k-means clustering is used to divide the expression level of cells in each cluster C_j into high and low groups. Candidate marker genes are selected based on three criteria, including an average

expression fold greater than 2 between the two groups, all cells in the high-level group being larger than the user-defined percentage, and a significant difference between the two groups of gene expression values (t test and Benjamini-Hochberg correction). For the list of candidate marker genes, the method assesses whether the subpopulation of cells expressing them is specific to cluster C_j based on the significant difference in the expression value of gene ith in high expression cells compared to cells not in cluster C_j (t test and FDR correction).

For each cluster C_j, the correlation matrix of all candidate gene expression for all cells in the cluster C_j is converted into a graph where the genes correspond to nodes and the edges are weighted by the correlation between them. MCL[23] is then used to identify correlated gene sets. A one-dimensional k-means method is used to the mean expression of each gene set for each cluster. Cells are assigned to a new cluster when they fall into the high mode. Finally, cells assigned to the final cluster combine all subgroups to which they belong. The minimum number of genes for a cluster to be considered is 3.

2.3.3 Detecting rare cell type based on calculating rareness score (FiRE)

FiRE is a rapid method for estimating the density around each related multidimensional data point. It utilizes the sketching technique[24] as the primary algorithm. Unlike most existing techniques, FiRE calculates a rareness score for each individual expression profile, allowing users to focus more attention on the small set of potentially rare cells. It includes two phases:

In the first phase, the Sketching process is repeated L times. Hash codes are generated for the entire set of expression profiles at each pass iteration. Each hash code can be thought of as a bucket. The sketching process needs to ensure that the cells sharing the same bucket are close to each other in the original high-dimensional space. The density estimate for the i-th cell in the l-th pass is calculated as follows:

$$p_{il} = \frac{Number\ of\ cells\ in\ the\ bucket\ consist\ of\ cell\ i}{Overall\ number\ of\ cells}$$

In the second phase, FiRE reduces the variance of density estimates for individual cells by combining them. The FiRE score is defined as follows:

$$FiREscore_i = -2\sum_{l=1}^{L} log(pil)$$

Then, an IQR-based threshold criteria was used to determine the rare cell type.

2.3.4 Detecting rare cell type based on embedding and RPH-kmeans (scAIDE)

scAIDE is a fully unsupervised deep learning clustering analysis framework consisting of two main components: AIDE for dimensionality reduction and RPH-kmeans for clustering. AIDE includes an imputation module and a dimensionality reduction module. The imputation module uses an autoencoder (AE) to correct biological noise in the gene expression vector and can recover estimated expression vectors since AE captures important latent structure of the data in hidden layers and learns to regenerate the data. In the dimensionality reduction module, a fully connected network called multidimensional scaling (MDS) encoder is used to transform the data into a space that is suitable for Euclidean-based clustering methods (e.g., k-means). RPH-kmeans is a random projection hashing-based clustering algorithm that matches with the MDS encoder for clustering.

One major challenge for k-means is its sensitivity to initial cluster centroids. When the size of the underlying cluster group is highly imbalanced, as is often the case for scRNA-seq data, the resulting clusters can be biased towards larger cell populations. To address this issue, RPH-kmeans was proposed, which utilizes a Locality Sensitive Hashing (LSH)[25] technique to initialize the cluster centers. The pipeline of RPH-kmeans can be summarized into two steps. In the first stage, the number of data points is iteratively reduced using LSH. In each iteration, data points that hash into the same bucket will be merged into one weighted point. In the end, a data skeleton with a much smaller number of points is generated. In the second stage, weighted k-means is applied to the skeleton to produce initial centers for RPH-kmeans. To evaluate the

performance of the algorithms, pre-determined group labels or rare cell type labels can be used at the individual level.

2.3.5 Detecting rare cell types based on screened for outliers (RaceID)

RaceID is an unsupervised method that can identify rare cell types in a population, even those represented by a single cell. The method comprises three steps. In the first step, larger clusters are identified using k-means clustering. The number of clusters used for k-means clustering is determined using the gap statistic[26], which measures the difference between the uniform distribution and within-cluster dispersion in the actual data. By default, the cluster number is determined as the first local maximum of the gap statistic, where the maximum exceeds its neighbors by more than 25% of its standard deviation. If the gap statistic does not show a clear maximum, then the number of clusters dividing the point where the gap statistic starts to saturate should be used as input to the k-means clustering. The algorithm uses Jaccard's similarity to quantify cluster reproducibility. If the Jaccard similarity of multiple clusters is below 0.5, the clusters should be repeated with fewer clusters. The outlier identification step of the algorithm corrects underestimation of the actual number of clusters. Therefore, it is recommended to start with a conservative estimate of the number of clusters.

In the second step, RaceID identifies outlier cells within each cluster by evaluating the transcript count variability of every gene across all cells in the cluster. The expected baseline level of expression variability, which is quantified by the transcript count variance, is inferred from the ensemble of all cells. A second-order polynomial is fitted to the transcript count variance as a function of the average transcript count in logarithmic space. For a given cell, if the multiple testing corrected transcript count probability of a specified number of genes (two for the data) is below a defined probability threshold (<10e−4 for the data), the cell is considered an outlier.

After identifying outlier cells in the second step, the last step of the RaceID algorithm involves inferring the final cluster of different cell types or states. Outlier cells are first merged into the outlier cluster if their transcriptome correlation exceeds the 75th percentile of the distribution of correlations between cells within the original cluster after outlier removal. New cluster centers are then calculated for the remaining original and new outlier clusters by averaging transcript counts within these clusters. Each cell is then reassigned to the most highly correlated cluster center, resulting in the final cluster assignments.

2.4 Evaluation metrics

To help guide the assessment of detecting rare cell type algorithm efficiency, we employed four different metrics, accuracy, precision, consistency, sensitivity and specificity values in binary classification. After obtaining the batch-corrected outputs and detecting rare cell types at both individual and population level, we computed the accuracy, precision, consistency, sensitivity and specificity scores based on the original rare cell types we defined. For accuracy score, it is the proportion of correct predictions (both true positives and true negatives) among the total number of cases examined[27]. The formula for accuracy score can be written as:

$$ACC = \frac{TP + TN}{TP + FP + TN + FN}$$

Where TP is the true positives, FP is the false positives, TN is the true negatives, and FN is are the false negatives. For precision, also as Positive Predictive Value, it measures how many observations predicted as positive are in fact positive. The formula for precision is following:

$$PPV = \frac{TP}{TP + FP}$$

For consistency part, we would like to use Cohen's kappa coefficient, it is generally thought to be a more robust measure than simple percent agreement calculation[28]. We used both individual and population level result to calculate this coefficient. The formula for Cohen's kappa coefficient is following:

$$\kappa = \frac{2 \times (TP \times TN - FN \times FP)}{(TP + FP) \times (FP + TN) + (TP + FN) \times (FN + TN)}$$

Where TP is the true positives, FP is the false positives, TN is the true negatives, and FN is are the false negatives. Sensitivity (true positive rate) is the probability of a positive test result, conditioned on the individual truly being positive. Specificity (true negative rate) is the probability of a negative test result, conditioned on the individual truly being negative. The formula for calculating sensitivity and specificity is following:

$$TPR = \frac{TP}{TP + FN}$$

$$TNR = \frac{TN}{TN + FP}$$

Where TP is the true positives, FP is the false positives, TN is the true negatives, and FN is are the false negatives.

2.5 Uniform manifold approximation and projection (UMAP) visualization

We used the Seurat and ggplot2 package in the R environment to visualize the raw data, batch-corrected output and rare cell type detecting result. To achieve this, we employed the UMAP algorithm[29] with the default number of neighbors, which allowed us to obtain a clear visualization of our detecting results.

2.6 Computation evaluation of runtime

We captured the runtime of each method using the time function available in R and Python environments. We did not take into account the pre-filtering steps, and only measured the runtime of the main function in each method. All jobs were run on a Linux server in RSPH HPC cluster.

3. Results

3.1 Data cleaning and batch effect correction result

After cleaning the 10 people COVID-19 scRNA-seq dataset, we removed the cells that were labeled as "mixed" and "undefined" based on the original labels. This resulted in a dataset with 9 cell types. We

constructed a table (Table 2) to summarize the number of cells in each label (main cluster). Figure 1 shows the visualization for the 10 people scRNA-seq dataset using UMAP plot with before and after the data cleaning process.

Table 2. Summarize the number of cells in each label for each sample.

Sample ID	B Cell	CD4	CD8	DCs	Megakaryocyte	Monocytes	Neutrophils	NK cell	Plasmablasts	Overall
C19-CB-0001	491	733	501	59	7	1660	6	148	7	3612
C19-CB-0002	453	100	1343	78	5	930	4	216	0	3129
C19-CB-0003	256	16	449	56	10	1677	6	110	0	2580
C19-CB-0005	46	23	48	6	11	1141	10	36	0	1321
C19-CB-0008	171	1252	208	22	23	804	1236	176	138	4030
C19-CB-0009	140	1227	432	52	31	1257	681	365	101	4286
C19-CB-0011	23	287	189	2	13	118	223	71	5	931
C19-CB-0012	111	1379	714	14	24	639	255	474	54	3664
C19-CB-0013	52	829	370	9	19	619	418	357	31	2704
C19-CB-0016	42	351	632	11	70	830	394	130	56	2516

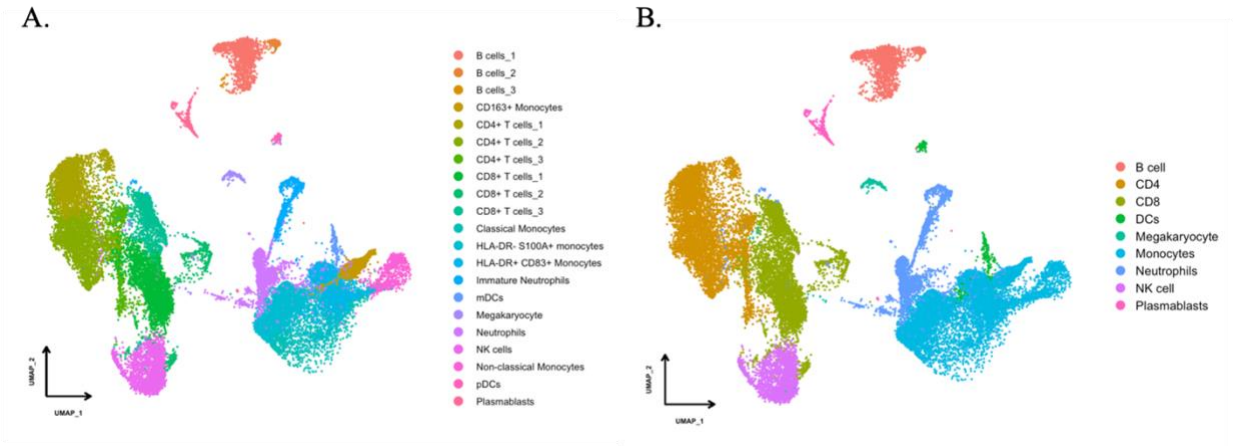

Figure 1. UMAP plot for the population level. A is the UMAP plot before data cleaning, B is the UMAP plot after data cleaning.

Based on the data selection, we defined *Megakaryocyte* and *Plasmablasts* as the true rare cell types. Inspired by GiniClust3's definition of rare cell types, we also considered a cell type as rare if it had less than 1% of the total population at the individual level. To test the authenticity and accuracy of rare cell types obtained

by different methods, we used the existing rare cell types, which were mostly derived from *DCs*, *Megakaryocyte*, and *Plasmablasts* cells, as a benchmark. Thus, we defined *DCs, Megakaryocyte*, and *Plasmablasts* cells as the true rare cell types at the population level. Table 3 summarizes the rare cell types defined under each sample and population level. Prioritizing *Megakaryocyte* and *Plasmablasts* as the true rare cell types was based on the original cell labels and our data selection criteria.

Table 3. The true rare cell for both individual and population level.

Sample ID	True Rare Cell Types	Number of Cells
C19-CB-0001	Megakaryocyte, Neutrophils, Plasmablasts	20
C19-CB-0002	Megakaryocyte, Neutrophils	9
C19-CB-0003	CD4, Megakaryocyte, Neutrophils	32
C19-CB-0005	DCs, Megakaryocyte, Neutrophils	27
C19-CB-0008	DCs, Megakaryocyte, Plasmablasts	183
C19-CB-0009	Megakaryocyte, Plasmablasts	132
C19-CB-0011	DCs, Megakaryocyte, Plasmablasts	20
C19-CB-0012	DCs, Megakaryocyte, Plasmablasts	92
C19-CB-0013	DCs, Megakaryocyte, Plasmablasts	59
C19-CB-0016	DCs, Megakaryocyte, Plasmablasts	137
Population	DCs, Megakaryocyte, Plasmablasts	914

3.2 Detecting rare cell types using GiniClust3

To evaluate the performance of GiniClust3 in detecting rare cell types in the COVID-19 scRNA-seq dataset, we applied the method at both the individual and population level after filtering out lowly expressed genes and poor-quality cells. At the individual level, we identified 2368 cells (8.23% of the total) and at the population level, we identified 3254 cells (11.31% of the total) as rare cells (Table 4). Figure 2 shows the UMAP plot for the detecting result with both individual and population level. Furthermore, we found 519 cells (1.80% of the total) that were identified as rare cells in both the individual and population levels. We identified a total of 7 common and 256 rare cell clusters (with a cell population < 1%) at the population

level, with the smallest cluster containing only 1 cell (in 55 clusters). Figure 3 shows the overlaps within individual level and population level using UMAP plot. The process of rare cell type identification for each individual level took approximately 2 minutes and for population level took approximately 6 minutes, indicating that GiniClust3 is a really fast method and suitable for analyzing large datasets.

Table 4. Detecting rare cell type using GiniClust3.

	# Rare cells at individual level (%)	# Rare cells at population level (%)	# Common rare cells (%)
C19-CB-0001	290 (8.02%)	374 (10.35%)	66 (1.83%)
C19-CB-0002	276 (8.82%)	292 (9.33%)	66 (2.11%)
C19-CB-0003	196 (7.60%)	205 (7.95%)	22 (0.85%)
C19-CB-0005	69 (5.22%)	83 (6.28%)	2 (0.15%)
C19-CB-0008	277 (6.87%)	591 (14.67%)	66 (1.64)
C19-CB-0009	718 (16.75)	536 (12.51%)	168 (3.92)
C19-CB-0011	15 (1.61%)	94 (10.10%)	5 (0.53%)
C19-CB-0012	256 (6.99%)	494 (13.48%)	56 (1.53%)
C19-CB-0013	158 5.84%)	324 (11.98%)	41 (1.52%)
C19-CB-0016	113 (4.49%)	261 (10.37%)	27 (1.07%)
Overall	2368 (8.23%)	3254 (11.31%)	519 (1.80%)

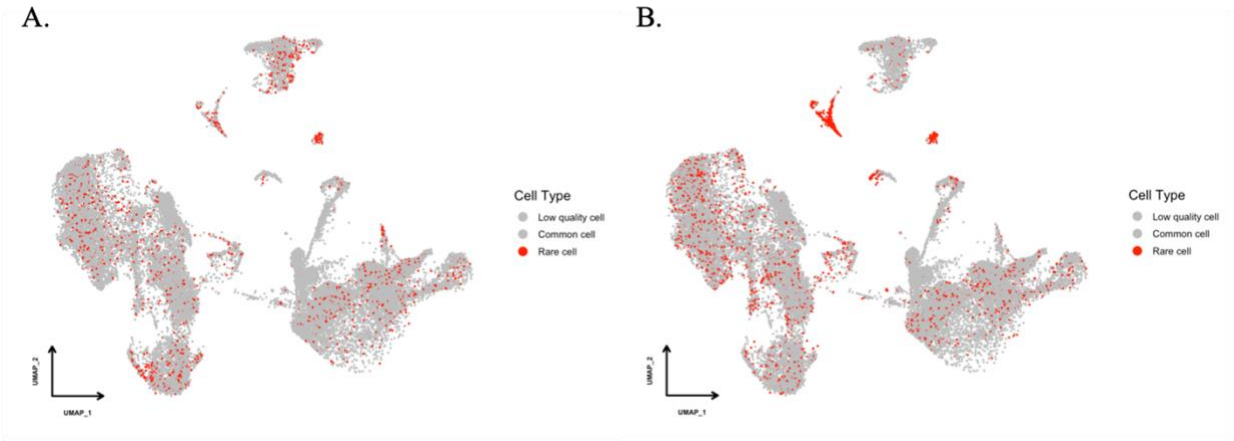

Figure 2. UMAP plot for detecting rare cell types at individual and population using GiniClust3. A is the detecting result at individual level, B is the detecting result at population level.

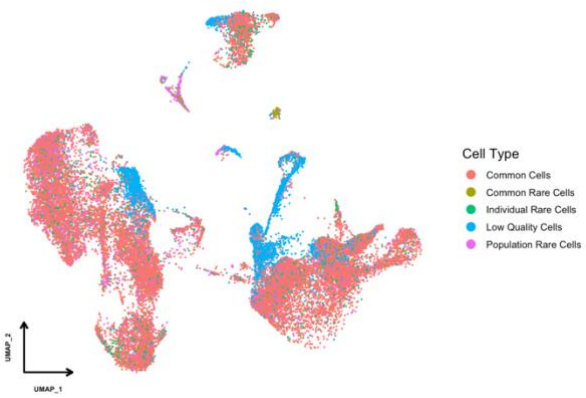

Figure 3. UMAP plot for detecting rare cell types in comparing overall labels using GiniClust3.

Once rare cell types have been detected at both the individual and population levels, it is important to test their consistency and accuracy. Figure 4 displays the distribution of rare cell types at both levels, indicating that GiniClust3 may still detect some common cells as rare cell types according to our original label and definition. Figure 5 shows the detection results based on the original rare cell labels. From this figure, we observed that the number of true rare cell types at the population level is greater than at the individual level. Furthermore, some rare cell types can be found at both the individual and population levels, particularly in Sample 5, Sample 6, Sample 7, Sample 8, Sample 9, and Sample 10 at the individual level. In the population data, we found 195 rare cells at both individual and population levels. However, based on the original cell labels, some cells remain unrecognized at both levels and some cells cannot be filtered through the quality control, especially in first 4 samples.

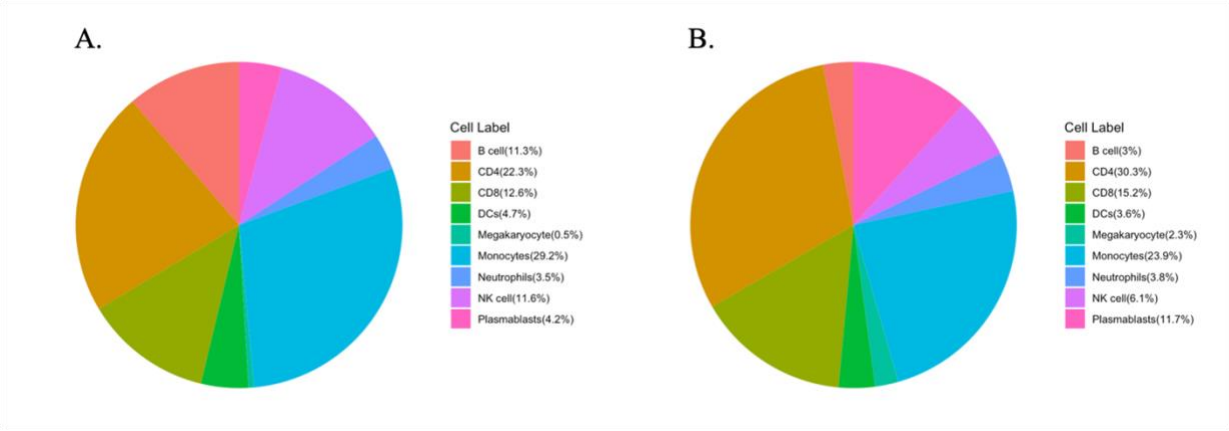

Figure 4. Detecting rare cell types result with original label using GiniClust3. A is the detecting result at individual level, B is the detecting result at population level.

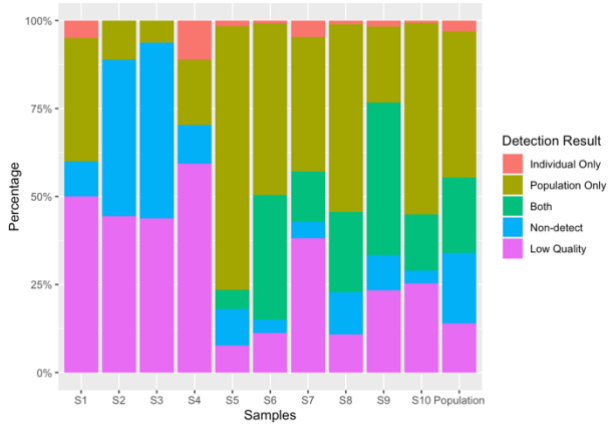

Figure 5. Visualization the consistency and accuracy result using GiniClust3.

Table 5 presents the evaluation metrics results at both individual and population levels for GiniClust3. While the reported accuracies are high, they are predominantly driven by high specificities rather than sensitivities. What's more, the precisions in both population level and individual level are less than 0.2, indicating there are a lot of false-positive result from GiniClust3. Notably, the reported sensitivities at population level are better than they are at individual level. All of the specificity are high indicating low false negative. Additionally, Cohen's Kappa values are also low, with values less than 0.15 indicating only slight consistency between individual and population levels. In summary, GiniClust3's sensitivity in detecting rare cell types at the population level is better than individual level, but the precision for both individual and population level still need to be improved GiniClust3's detecting common rare cell type ability is good. Improvements in consistency between these levels are necessary for further development of GiniClust3.

Table 5. Evaluation metrics result using GiniClust3.

	Accuracy	Precision	Cohen's Kappa	Sensitivity	Specificity
C19-CB-0001	0.901	0.003	0.101	0.100	0.904
C19-CB-0002	0.886	0.000	0.133	0.000	0.888
C19-CB-0003	0.904	0.000	0.022	0.000	0.911
C19-CB-0005	0.943	0.072	-0.037	0.455	0.948
C19-CB-0008	0.856	0.047	0.026	0.077	0.904
C19-CB-0009	0.797	0.065	0.120	0.402	0.810
C19-CB-0011	0.969	0.200	0.056	0.250	0.982
C19-CB-0012	0.913	0.086	0.055	0.268	0.929
C19-CB-0013	0.936	0.165	0.088	0.578	0.943
C19-CB-0016	0.919	0.195	0.076	0.216	0.955
Population	0.880	0.176	0.080	0.730	0.885

3.3 Detecting rare cell types in CellSIUS

To evaluate and compare various feature selection and clustering techniques for scRNA-seq data, we utilized a scRNA-seq dataset consisting of a mixture of 10 people cell lines with known cellular composition. After removing cells that did not pass quality control or could not be accurately assigned to a cell line, we applied the graph-based clustering algorithm MCL to identify gene sets with correlated expression patterns from the remaining cluster-specific candidate marker genes. MCL does not require a predetermined number of clusters and operates on the gene correlation network obtained from single-cell RNAseq data, detecting communities within this network that contain co-expressed genes.

Our analysis included 3203 (11.13%) individual cells and 2683 (9.32%) cells at the population level (cell population < 1%) (table 5). Figure 6 shows the UMAP plot for the detecting result with both individual and population level. Additionally, we identified 2209 (7.68%) rare cells that were common to both individual and population levels. Specifically, we observed only a small number of rare cells in the first 4 subjects at the population level, and no common rare cells were detected. However, in the last six subjects, we

identified a high proportion of rare cells and a significant number of common rare cells at both individual and population levels, with high consistency among the subjects. Figure 7 shows the overlaps within individual level and population level using UMAP plot. Overall, the rare cell type identification process for each individual took 20 minutes and population levels took approximately 1 hour, indicating that CellSIUS is capable of analyzing large datasets efficiently.

Table 6. Detecting rare cell type using CellSIUS.

	# Rare cells at individual level (%)	# Rare cells at population level (%)	# Common rare cells (%)
C19-CB-0001	272 (7.53%)	6 (0.17%)	0 (0%)
C19-CB-0002	27 (0.86%)	7 (0.22%)	0 (0%)
C19-CB-0003	190 (7.36)	5 (0.19%)	0 (0%)
C19-CB-0005	60 (4.54%)	12 (0.91%)	0 (0%)
C19-CB-0008	1213 (30.10%)	1094 (27.15%)	1070 (26.56%)
C19-CB-0009	615 (14.35%)	567 (13.23%)	511 (11.92%)
C19-CB-0011	223 (23.95%)	190 (20.41%)	184 (19.76%)
C19-CB-0012	243 (6.63%)	205 (5.59%)	173 (4.72%)
C19-CB-0013	235 (8.69%)	293 (10.84%)	191 (7.06%)
C19-CB-0016	125 (4.97%)	304 (12.08%)	80 (3.17%)
Overall	3203 (11.13%)	2683 (9.32%)	2209 (7.68%)

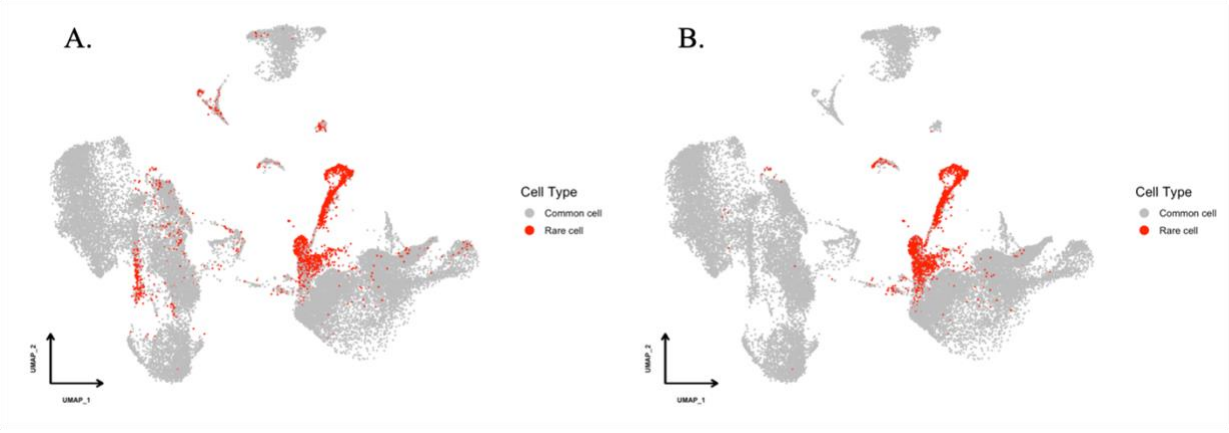

Figure 6. UMAP plot for detecting rare cell types at individual and population using CellSIUS. A is the detecting result at individual level, B is the detecting result at population level.

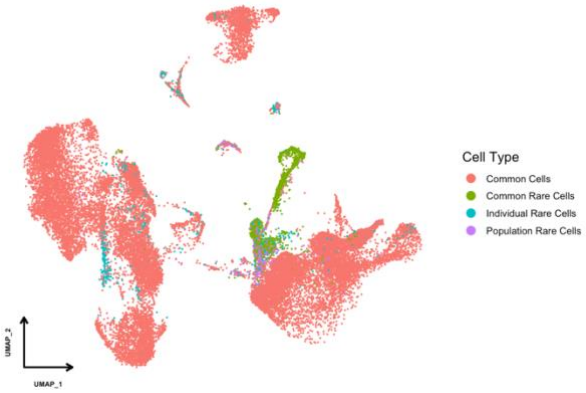

Figure 7. UMAP plot for detecting rare cell types in comparing overall labels using CellSIUS.

After identifying rare cell types at both the individual and population levels, as well as using GiniClust3, it is necessary to evaluate the consistency and accuracy of CellSIUS in both aspects. Figure 8 depicts the distribution of rare cell types at both the individual and population levels, indicating that CellSIUS can still detect some common cell types, such as Neutrophils, as rare cell types based on our original labeling and definition. Figure 9 illustrates the detection results based on the original rare cell labels, indicating that most of the true rare cell types cannot be identified using CellSIUS. Additionally, only in Samples 6, 8, and 10 for individual data, some rare cell types can be detected at both the individual and population levels. However, at the population level, although CellSIUS detected 2209 rare cells, only 27 overlapped with the original label.

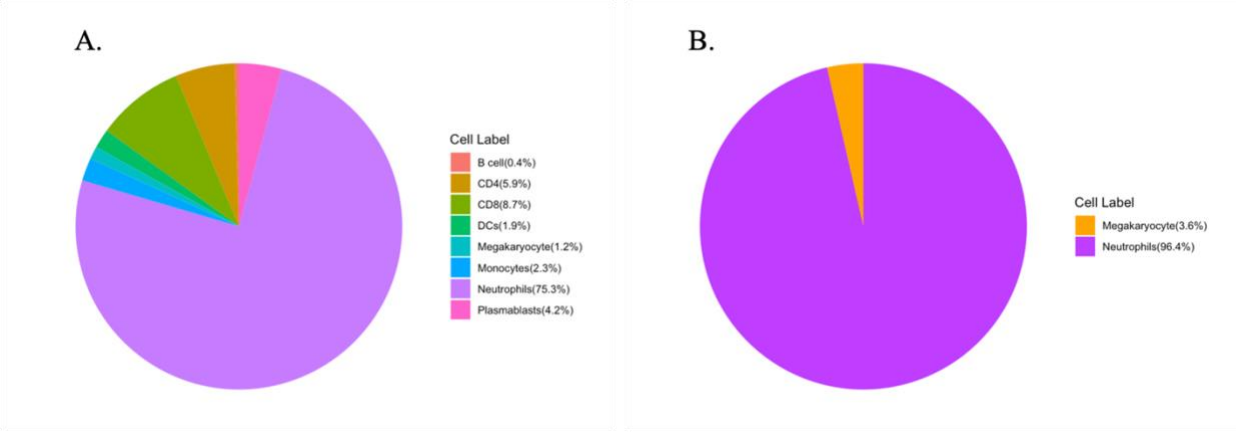

Figure 8. Detecting rare cell types result with original label using CellSIUS. A is the detecting result at individual level, B is the detecting result at population level.

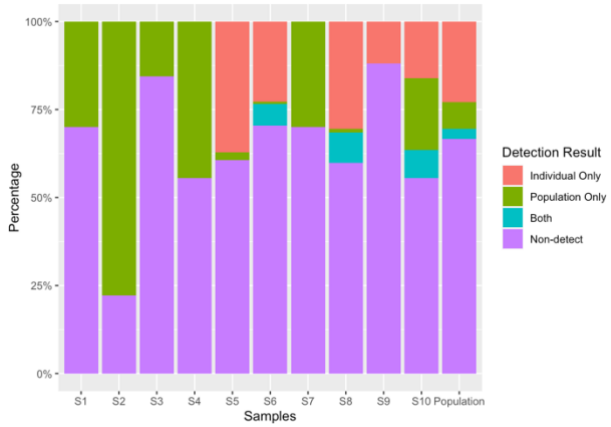

Figure 9. Visualization the consistency and accuracy result using CellSIUS.

Table 7 shows the evaluation metrics results at both individual and population levels for CellSIUS. As same as GiniClust3, even though the accuracy is high, the precision is really low, indicating a lot of false positive occurred. What's more, the sensitivity in both population level and individual level is also low. Additionally, Cohen's Kappa values is the first four subjects are less than 0 indicating there is no consistency between individual level and population level in the first four subjects. In other subjects, 5 subjects' metrics are greater than 0.69, and the population level is also greater than 0.7, which indicated there is a strong consistency between the last 6 subjects and population level. We also could see the detecting common cells' ability for CellSIUS is good since all of the specificity are high indicating low false negative. In summary, CellSIUS's precision to detect rare cell types is not good since a lot of false positive result, but the consistency for the last 6 sample and the population level is good. Therefore, based on the significant overlap within the last six samples and population level, we can conclude that CellSIUS may primarily focus on detecting the sub-type of *Neutrophils* as its rare cell types.

Table 7. Evaluation metrics result using CellSIUS.

	Accuracy	Precision	Cohen's Kappa	Sensitivity	Specificity
C19-CB-0001	0.919	0.000	-0.003	0.000	0.924
C19-CB-0002	0.988	0.000	-0.004	0.000	0.991
C19-CB-0003	0.914	0.000	-0.004	0.000	0.925
C19-CB-0005	0.934	0.000	-0.015	0.000	0.954
C19-CB-0008	0.687	0.056	0.899	0.372	0.702
C19-CB-0009	0.843	0.062	0.843	0.288	0.861
C19-CB-0011	0.739	0.000	0.860	0.000	0.755
C19-CB-0012	0.928	0.148	0.758	0.391	0.942
C19-CB-0013	0.891	0.000	0.694	0.000	0.911
C19-CB-0016	0.922	0.264	0.325	0.241	0.961
Population	0.882	0.036	0.722	0.105	0.907

3.4 Detecting rare cell types in scAIDE

To evaluate scAIDE's general performance and consistency, we compared it to individual and population-level analyses using the aforementioned dataset. De novo clustering analysis has the potential to provide valuable biological insights into the identification of rare cell types. Two critical factors for accurately separating different cell types and identifying rare subpopulations are ensuring that cells are well-represented in low dimensions and that clustering algorithms can identify small groups of cells. Through simulation experiments, we demonstrated that the AIDE embedding can successfully separate different cell types, and that RPH-kmeans is well-suited for detecting rare cell types. Not only did we identify different subpopulations within each dataset, but we also detected primed differentiation development of cell types. In total, we identified 422 (1.47%) individual rare cells and 328 (1.14%) population rare cells (cell population $< 1\%$) (see Table 8). Additionally, we detected 315 (1.09%) common rare cells at both the individual and population levels. Specifically, we identified a very small number of rare cells in the first four subjects and the "C19-CB-0011" subject at the population level (less than 10 cells), and we could not

detect common rare cells in “C19-CB-0003” and “C19-CB-0005.” Figure 10 shows the UMAP plot for the detecting result with both individual and population level.

One interesting finding from scAIDE is the high level of consistency in non-rare cell subjects. Rare cell types from seven subjects at the population level were found 100% at the individual level. In addition, for subject “C19-CB-0008,” we found 115 rare type cells at the population level and 103 rare type cells at the individual level, with 102 cells being common rare cells. While the number of rare cell types identified may not be as large as in previous methods, scAIDE demonstrates excellent consistency. Figure 11 shows the overlaps within individual level and population level using UMAP plot. In terms of running time, scAIDE consists of three parts, with rare cell type identification taking 2 hours for each individual level and 12 hours for the population level. This indicates that scAIDE is still suitable for analyzing very large datasets and is consistent in its performance.

Table 8. Detecting rare cell type using scAIDE.

	# Rare cells at individual level (%)	# Rare cells at population level (%)	# Common rare cells (%)
C19-CB-0001	30 (0.83%)	4 (0.11%)	4 (0.11%)
C19-CB-0002	46 (1.47%)	1 (0.03%)	1 (0.03%)
C19-CB-0003	0 (0%)	0 (0%)	0 (0%)
C19-CB-0005	8 (0.61%)	0 (0%)	0 (0%)
C19-CB-0008	103 (2.56%)	115 (2.85%)	102 (2.53%)
C19-CB-0009	96 (2.24%)	88 (2.05%)	88 (2.05%)
C19-CB-0011	6 (0.64%)	3 (0.32%)	3 (0.32%)
C19-CB-0012	52 (1.42%)	43 (1.17%)	43 (1.17%)
C19-CB-0013	26 (0.96%)	26 (0.96%)	26 (0.96%)
C19-CB-0016	55 (2.19%)	48 (1.91%)	48 (1.91%)
Overall	422 (1.47%)	328 (1.14%)	315 (1.09%)

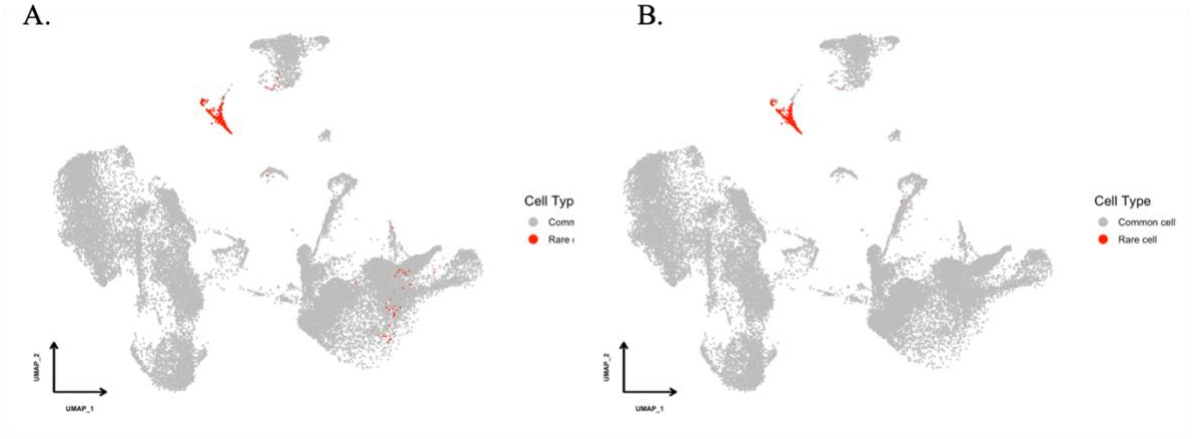

Figure 10. UMAP plot for detecting rare cell types at individual and population using scAIDE. A is the detecting result at individual level, B is the detecting result at population level.

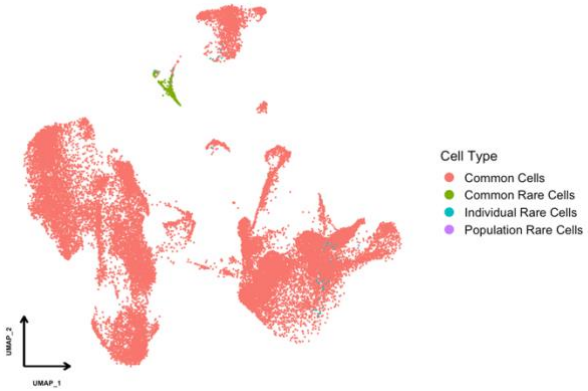

Figure 11. UMAP plot for detecting rare cell types in comparing overall labels using scAIDE.

After detecting rare cell types at both individual and population levels using scAIDE, it is important to test the consistency and accuracy of the method. Figure 12 illustrates the distribution of rare cell types at both levels, and it shows that scAIDE detected a significant number of *Plasmablasts* cells as its rare cell type at both levels. While *Plasmablasts* are one of the truly rare cell types according to our definition, scAIDE needs to improve its ability to detect other rare cell types such as *DCs* and *Megakaryocytes*. Figure 13 presents the detection results based on the original rare cell labels, revealing that more than 50% of the true rare cell types cannot be detected using scAIDE. Since both individual and population-level methods can identify *Plasmablasts*, the overlap between the two groups is significant. However, it cannot be denied that scAIDE's ability to identify other rare cell types needs improvement.

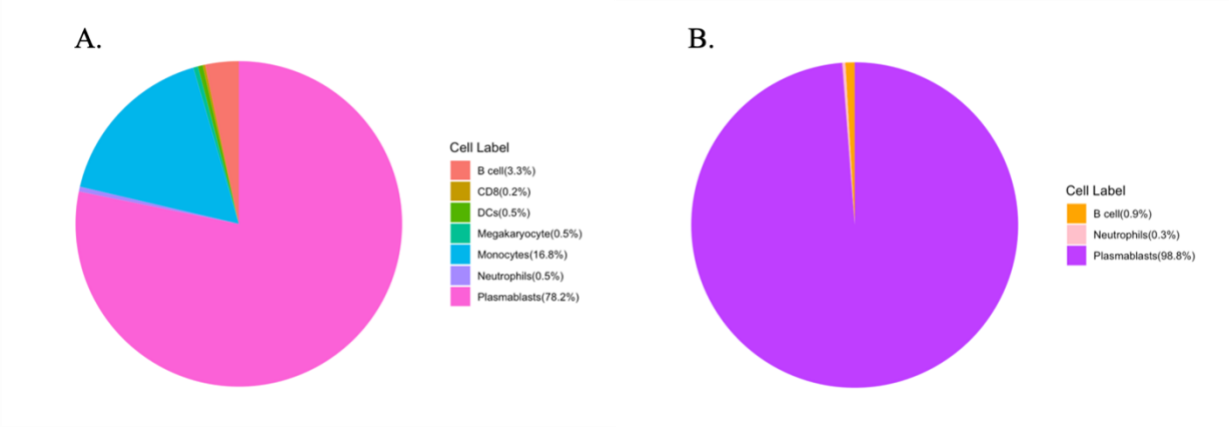

Figure 12. Detecting rare cell types result with original label using scAIDE. A is the detecting result at individual level, B is the detecting result at population level.

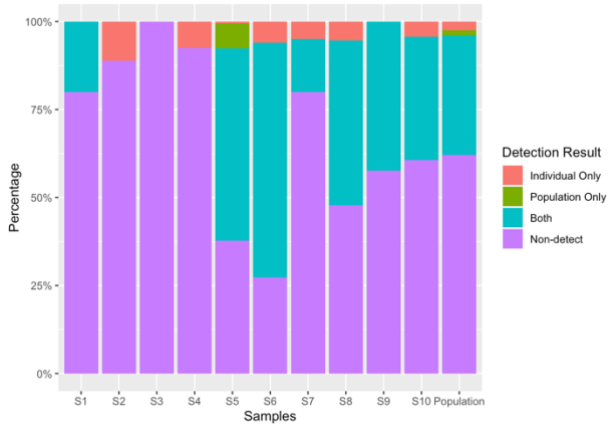

Figure 13. Visualization the consistency and accuracy result using scAIDE.

Table 9 shows the evaluation metrics results at both individual and population levels for scAIDE. The accuracy and specificity for scAIDE is good which indicate a high correction in detecting common cells. What's more, for the first four subjects, the precision and sensitivity is not good since one of individual and population level cannot detect much rare cell types. For the last 6 subjects and population level, the precision is really good indicating the most detecting results for scAIDE are true rare cell types even though it still has some false negative result. Additionally, Cohen's Kappa values is the first four subjects are less than 0.2 indicating there is a small consistency between individual level and population level in the first four subjects. In other 6 subjects, both individual metrics and population metrics are greater than 0.3, which indicated there is a moderate consistency between the last 6 subjects and population level. Therefore, based

on the numerous overlaps within the seven samples, we can conclude that scAIDE may mainly focus on detecting *Plasmablasts* as its rare cell type in this simulation setting. The detecting rare cell type's ability for the first four subjects still need to be improved. Although scAIDE has a high precision for the last 6 subjects and population level, it still needs to improve to reduce false negative result.

Table 9. Evaluation metrics result using scAIDE.

	Accuracy	Precision	Cohen's Kappa	Sensitivity	Specificity
C19-CB-0001	0.988	0.133	0.234	0.200	0.993
C19-CB-0002	0.983	0.022	0.042	0.111	0.986
C19-CB-0003	0.988	0.000	0.000	0.000	1.000
C19-CB-0005	0.977	0.250	0.994	0.074	0.995
C19-CB-0008	0.979	0.981	0.934	0.552	0.999
C19-CB-0009	0.992	1.000	0.956	0.727	1.000
C19-CB-0011	0.981	0.667	0.665	0.200	0.998
C19-CB-0012	0.987	0.923	0.904	0.522	0.999
C19-CB-0013	0.987	0.962	1.000	0.424	1.000
C19-CB-0016	0.967	0.982	0.931	0.394	1.000
Population	0.882	0.988	0.838	0.354	1.000

3.5 Detecting rare cell types in FiRE

All of the methods employed unsupervised clustering as an intermediate step for detecting rare cells, but clustering has its limitations. It can be sensitive to parameters and inefficient when density varies across data points. Additionally, the resolution of group identities can be challenging, especially with minor clusters that get overlooked during the first pass due to the influence of major cell types on expression variance. To address these limitations, we used FiRE, a monolithic algorithm that bypasses clustering to estimate cell rareness directly. The algorithm leverages Sketching, a powerful technique for low-dimensional encoding of a large volume of data points. FiRE assigns a continuous score to each cell, such that outlier cells and cells from minor populations receive higher scores than those representing major

subpopulations. While a continuous score is useful, binary annotation of cell rarity can be more straightforward for analysis. To this end, we introduced a thresholding scheme using score distribution properties (Methods). Using this method, we detected 818 (2.84%) rare cells at the individual level and 2551 (8.87%) rare cells at the population level, as shown in Table 10. However, we found that "C19-CB-0003" and "C19-CB-0011" had no rare cell types detected at the individual level. For "C19-CB-0001" and "C19-CB-0002," over half of the rare cells detected were common across the samples, either at the individual level or population level. Figure 14 shows the UMAP plot for the detecting result with both individual and population level. Except for these four samples, in the remaining six samples, the vast majority of rare cells detected at the level with fewer rare cells were common across samples. Figure 15 shows the overlaps within individual level and population level using UMAP plot. The process of rare cell type identification for each individual level took approximately 3 minutes and for population level took approximately 10 minutes, indicating that FiRE is a really fast method and suitable for analyzing large datasets.

Table 10 Detecting rare cell type using FiRE

	# Rare cells at individual level (%)	# Rare cells at population level (%)	# Common rare cells (%)
C19-CB-0001	13 (0.36%)	31 (0.86%)	8 (0.22%)
C19-CB-0002	55 (1.76%)	53 (1.69%)	23 (0.74%)
C19-CB-0003	0 (0%)	35 (1.36%)	0 (0%)
C19-CB-0005	181 (13.70%)	32 (2.42%)	31 (2.35%)
C19-CB-0008	19 (0.47%)	906 (22.48%)	19 (0.47%)
C19-CB-0009	236 (5.51%)	403 (9.40%)	232 (5.41%)
C19-CB-0011	0 (0%)	215 (23.09%)	0 (0%)
C19-CB-0012	10 (0.27%)	177 (4.83%)	10 (0.27%)
C19-CB-0013	33 (1.22%)	349 (12.91%)	33 (1.22%)
C19-CB-0016	271 (10.77%)	350 (13.91%)	270 (10.73%)
Overall	818 (2.84%)	2551 (8.87%)	626 (2.18%)

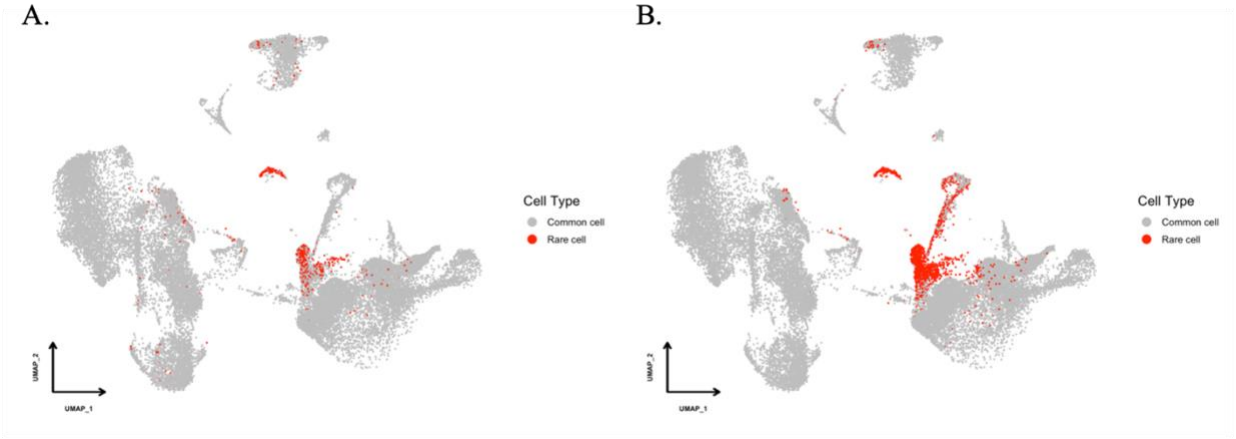

Figure 14 UMAP plot for detecting rare cell types at individual and population using FiRE. A is the detecting result at individual level, B is the detecting result at population level.

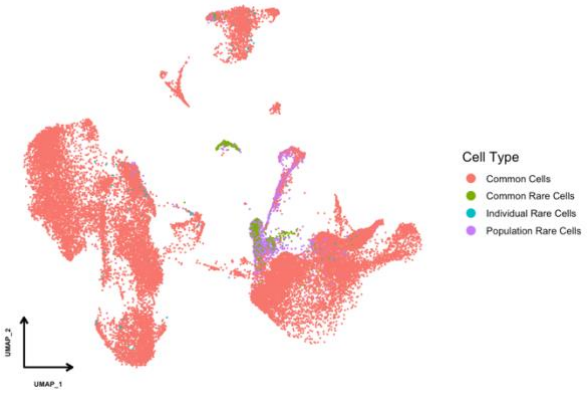

Figure 15 UMAP plot for detecting rare cell types in comparing overall labels using FiRE.

After detecting rare cell types using FiRE, we evaluated the consistency and accuracy of our results at both the individual and population levels. Figure 16 illustrates the distribution of rare cell types detected by FiRE at both levels. We observed that while FiRE identified Neutrophil cells as rare cell types, these cells were not rare in our simulation dataset. Conversely, FiRE correctly detected Megakaryocyte cells as rare cell types at the population level, but we still need to improve the accuracy of this method. To evaluate the accuracy of FiRE, we examined the detection results based on the original rare cell labels. As shown in Figure 17, we detected rare cell types in seven out of ten samples at both individual and population levels.

For these seven samples, more than half of the rare cell types detected by FiRE were consistent across both levels, except for sample No. 8. We also found that FiRE performed well in detecting Megakaryocyte cells at the population level, and there were 142 rare cell types detected consistently at both levels. However, the accuracy of FiRE needs improvement, as many Neutrophil cells were detected despite not being rare in our simulation dataset. Nevertheless, when it comes to consistency, FiRE performed better than GiniClust3 and CellSIUS in this simulation. Overall, based on the significant overlap in our results, we conclude that FiRE has good consistency performance but lower accuracy.

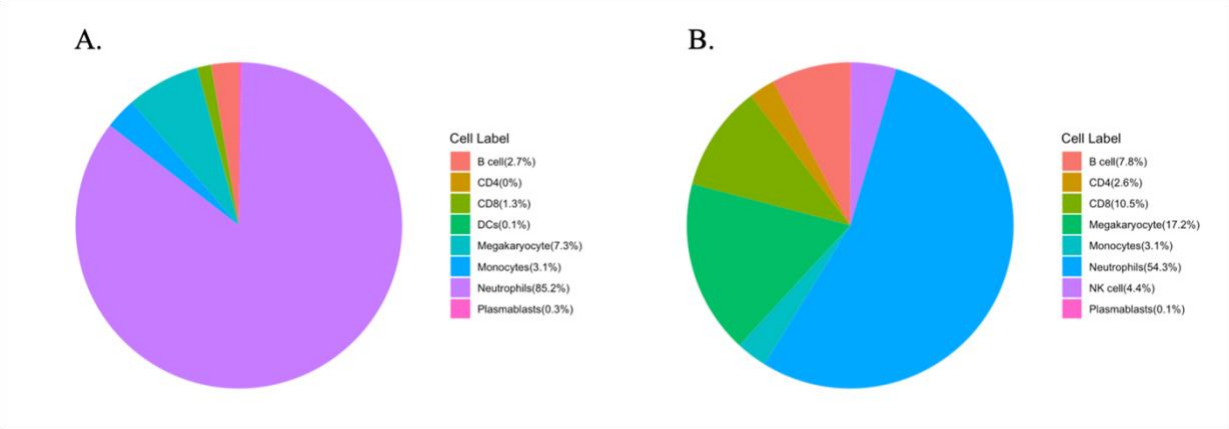

Figure 16. Detecting rare cell types result with original label using FiRE. A is the detecting result at individual level, B is the detecting result at population level.

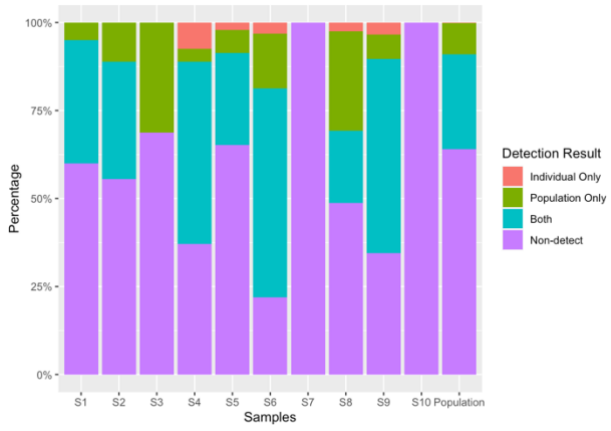

Figure 17. Visualization the consistency and accuracy result using FiRE.

Table 11 represents the evaluation metrics results at both individual and population levels for FiRE. As same as the above methods, the accuracy and specificity for scAIDE is good which indicate a high correction in detecting common cells. What's more, for five of ten subjects, the precision and sensitivity is pretty fine indicating a moderate detecting ability in rare cell types for these five subjects. For other five subjects and population level, FiRE's detecting ability still need to be improved. Additionally, at individual level, Cohen's Kappa values is fine except "C19-CB-0008" subject, indicating it is a substantial or perfect consistency in FiRE. At the same time, at the population level, it shows a fair consistency in FiRE. Overall, based on the significant overlap in our results, we conclude that FiRE has good consistency performance but lower precision.

Table 11. Evaluation metrics result using FiRE.

	Accuracy	Precision	Cohen's Kappa	Sensitivity	Specificity
C19-CB-0001	0.995	0.538	0.360	0.350	0.993
C19-CB-0002	0.981	0.055	0.416	0.333	0.983
C19-CB-0003	0.988	0.000	0.988	0.000	1.000
C19-CB-0005	0.867	0.088	0.261	0.593	0.872
C19-CB-0008	0.956	0.632	0.032	0.066	0.998
C19-CB-0009	0.923	0.085	0.706	0.152	0.948
C19-CB-0011	0.979	0.000	0.978	0.000	1.000
C19-CB-0012	0.977	0.800	0.102	0.087	0.999
C19-CB-0013	0.978	0.485	0.154	0.271	0.994
C19-CB-0016	0.890	0.240	0.852	0.474	0.913
Population	0.893	0.076	0.343	0.213	0.915

3.6 Detecting rare cell types in RaceID

To evaluate the accuracy and consistency of RaceID, we performed individual and population-level analyses on the above dataset using k-means clustering and gap statistics for rare cell type detection. After clustering and outlier detection, we obtained the final cluster inference. Table 12 shows that we detected 6758 (23.49%) rare cells at the individual level and 1548 (5.38%) rare cells at the population level. Notably,

the number of rare cells detected at the individual level was much higher than that at the population level. For nine out of ten subjects, although most of the rare cell types detected at the population level were also detected at the individual level, they were still different from those identified at the individual level. This difference may be due to the parameter settings that allow for deeper subtyping at the individual level in RaceID. To visualize the results, we created UMAP plots for both individual and population-level analyses, as shown in Figure 18. Additionally, Figure 19 shows the overlaps between individual-level and population-level analyses based on the UMAP plot. The process of rare cell type identification for each individual level took approximately 3 hours and for population level took approximately 20 hours, indicating that RaceID is slower than other methods.

Table 12 Detecting rare cell type using RaceID

	# Rare cells at individual level (%)	# Rare cells at population level (%)	# Common rare cells (%)
C19-CB-0001	947 (26.22%)	216 (5.98%)	94 (2.60%)
C19-CB-0002	211 (6.74%)	161 (5.15%)	29 (0.93%)
C19-CB-0003	637 (24.68%)	200 (7.75%)	121 (4.69%)
C19-CB-0005	275 (20.82%)	56 (4.23%)	41 (3.10%)
C19-CB-0008	1282 (31.81%)	280 (6.95%)	271 (6.72%)
C19-CB-0009	1166 (27.20%)	260 (6.07%)	234 (5.46%)
C19-CB-0011	100 (10.74%)	29 (3.11%)	22 (2.36%)
C19-CB-0012	948 (25.87%)	124 (3.38%)	120 (3.28%)
C19-CB-0013	666 (24.63%)	97 (3.59%)	77 (2.85%)
C19-CB-0016	526 (20.91%)	125 (4.97%)	73 (2.90%)
Overall	6758 (23.49%)	1548 (5.38%)	1082 (3.76%)

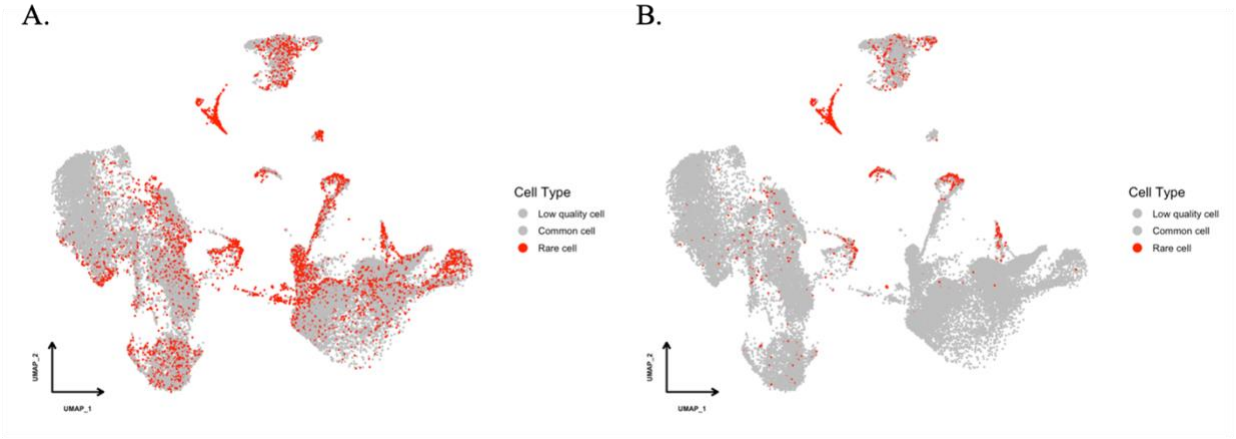

Figure 18. UMAP plot for detecting rare cell types at individual and population using RaceID. A is the detecting result at individual level, B is the detecting result at population level.

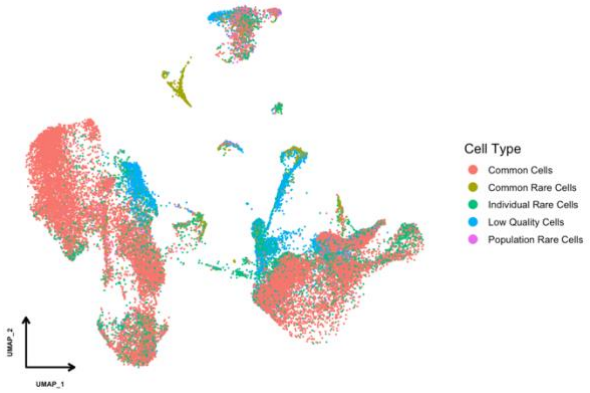

Figure 19 UMAP plot for detecting rare cell types in comparing overall labels using RaceID.

After detecting rare cell types using RaceID, we evaluated the consistency and accuracy of our results at both the individual and population levels. To evaluate the accuracy of RaceID, we examined the detection results based on the original rare cell labels. Figure 20 illustrates the distribution of rare cell types detected by RaceID at both levels. We observed that at individual level, while RaceID can identify *DCs*, Megakaryocyte and *Plasmablasts* as its rare cell type, the rare cell at individual level also included *CD8* (14.8%), *Monocytes* (27%) and *Neutrophils* (22.9%) as its rare cell types. It can be seen that in addition to being able to identify the three rare cell types we defined, RaceID will also mine potential common cell subtypes at the individual level. At the population level, as same condition as individual level, it could

detect the truly rare cell types but still include some outliers. As shown in Figure 21, we detected common rare cell types in all of the samples at both individual and population levels. Except for the NO.2 and NO.7 samples, we found that the proportion of non-detected rare cells of other samples is very small. Especially at the population level, only 137 (15.01%) rare cell types were not identified, and 66 (7.23%) rare cell types failed to pass the filter. It can be seen that RaceID is very accurate at the population level. Compared with the above four methods, the consistency of RaceID is much better than the above software. In terms of accuracy, although it can identify the main rare cell type at the population level, it still identifies a lot of outliers. We need further inspection.

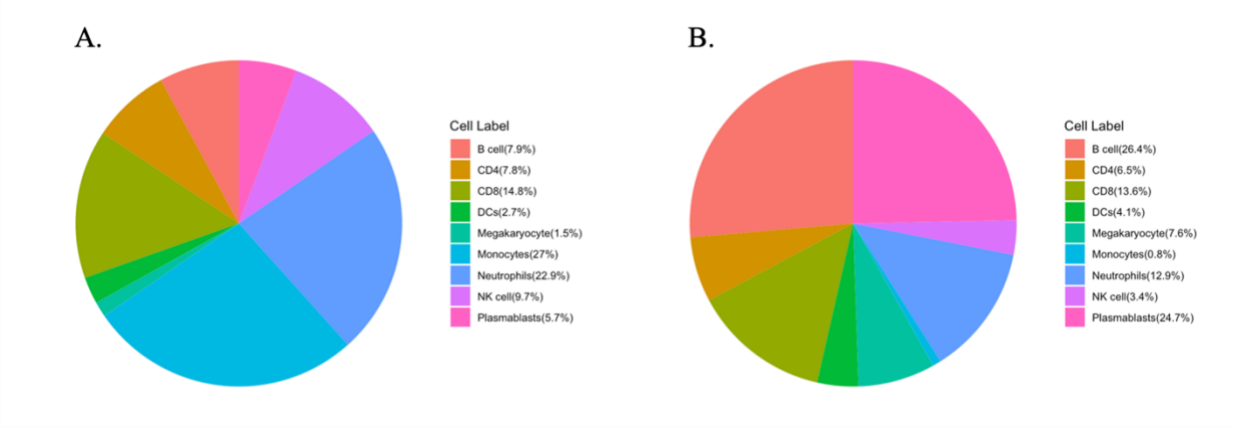

Figure 20. Detecting rare cell types result with original label using RaceID. A is the detecting result at individual level, B is the detecting result at population level.

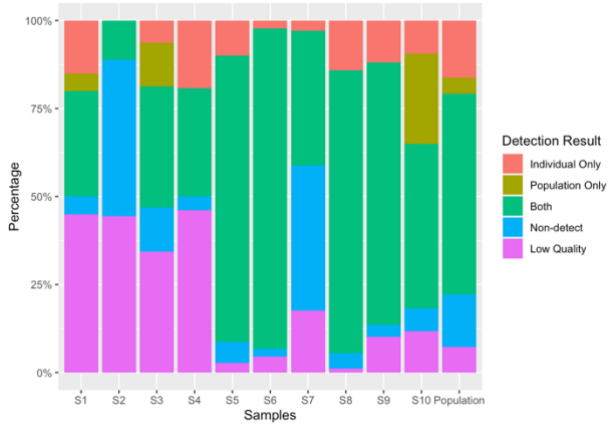

Figure 21. Visualization the consistency and accuracy result using RaceID.

Based on the evaluation metrics at both individual and population level in table 13, we found RaceID has a lower accuracy and precision at individual level comparing other above methods indicating a lot of false positive results happened. In addition, the specificity in RaceID performs also worse at both individual and population level which indicated RaceID will generate more false-positive result than other methods at individual level even though the number of overlaps is great. What's more, for most of the subjects, the sensitivity is really high indicating low false negative result. Comparing individual and population level, population level generates less false-positive result in RaceID. Additionally, Cohen's Kappa values is fine, indicating it is a slight or fair consistency in RaceID. Compared with the above four methods, the sensitivity of RaceID is much better than the above software. However, RaceID also meet more false-positive results comparing with the above four methods. We need further inspection to reduce the false positive to improve RaceID's ability.

Table 13. Evaluation metrics result using RaceID.

	Accuracy	Precision	Cohen's Kappa	Sensitivity	Specificity
C19-CB-0001	0.693	0.010	0.059	0.818	0.693
C19-CB-0002	0.916	0.005	0.116	0.200	0.917
C19-CB-0003	0.717	0.020	0.178	0.619	0.718
C19-CB-0005	0.789	0.047	0.193	0.929	0.788
C19-CB-0008	0.661	0.129	0.289	0.929	0.645
C19-CB-0009	0.732	0.105	0.273	0.938	0.724
C19-CB-0011	0.876	0.140	0.331	0.976	0.874
C19-CB-0012	0.752	0.092	0.173	1.000	0.746
C19-CB-0013	0.753	0.077	0.151	0.956	0.749
C19-CB-0016	0.791	0.148	0.180	0.962	0.801
Population	0.949	0.364	0.196	0.614	0.959

3.7 Computing time benchmarks

We conducted a benchmark of the computation performance for the five analysis methods mentioned above in both individual and population-level simulation scenarios. All simulations were run on a Linux PC with a 2.80 GHz CPU and 8GB RAM. GiniClust3 demonstrated the fastest performance, taking only 2 minutes at the individual level and 6 minutes at the population level due to its optimized clustering method and use of Python. FiRE was the second fastest method, taking approximately 3 minutes at the individual level and approximately 10 minutes at the population level since it does not include the clustering process. CellSIUS was slower than both GiniClust3 and FiRE, requiring 20 minutes at the individual level and approximately 1 hour at the population level. In comparison, scAIDE and RaceID were the slowest methods, taking approximately 3 hours for each individual level and half a day to a full day for the population level.

4. Discussion

4.1 Rare Cell type detecting methods

In this work, we tested the rare cell type detecting methods' abilities at both individual and population scenarios. For each dataset and scenario tested, different methods emerged top. Table 16 shows the comparing evaluation metrics between these five methods. At individual level, RaceID and scAIDE ranked top but RaceID generated more false-positive result, scAIDE generated more false-negative result. For RaceID, 8 of 10 subjects got most overlaps within the five detecting methods. However, the accuracy and precision for RaceID are the lowest within the five detecting methods because of plenty of false positive result. For FiRE, the number of overlaps in 4 of 10 samples is also high, and the precision ranked the second stage. However, the sensitivity ranked fourth stage because of more false negative result. For scAIDE, as same as FiRE, the precision of 6 of 10 samples is relatively high, but the detecting ability for the first 4 samples still need to be improved. The last part is GiniClust3 and CellSIUS, and their precision and sensitivity are the worst among the 5 methods at individual level. However, only at individual level, we cannot judge their expressiveness solely by judging their precision. There are many other factors that will

affect the above five methods. The most important thing is that the parameters used by each method are different. For example, GiniClust3 and RaceID, we filter out low-expression cells by expressing at least 1000 genes per cell. In the results section, we can see that some real rare cell types are filtered out because of this parameter. Another example is FiRE, we calculated its rareness score based on IQR, which may also filter out some rare cell types. In addition, 4 of the above 5 methods obtain more sub-types by improving the clustering algorithm, but due to the different emphases of each clustering method, the results will also be biased.

At population level, scAIDE ranked top among the 5 methods. RaceID ranked second, CellSIUS, FiRE and GiniClust3 have lower precision and sencitivity at population level, even lower than 30%. In addition to the influencing factors mentioned by individual, the method selection of batch effect correction may also affect the results of the accuracy of the above methods. In addition to ComBat, there are still more than 10 batch effect correction methods available.

In addition to evaluating the 5 methods by precision, we also needed to evaluate the 5 methods by consistency, since all methods treated other cells as considered rare cell types by themselves. Through the result section based on the original label, we can see that scAIDE is also the best performer in terms of consistency at both individual and population level. Followed by FiRE and CellSIUS. RaceID came in fourth. The consistency of GiniClust3 is poor. If we don't look at the original label but simply look at the consistency results of these five methods. RaceID, scAIDE, and CellSIUS have a lot of overlaps, but RaceID varies greatly in the number of rare cell types identified at the individual and population levels. Therefore, the consistency of CellSIUS and scAIDE is relatively good (both a similar number of rare cell types and a large number of overlap).

Table 14. Comparing evaluation metrics within 5 methods.

	Accuracy		Precision		Cohen's Kappa		Sensitivity		Specificity	
	IND	POP	IND	POP	IND	POP	IND	POP	IND	POP
GiniClust3	0.902	0.880	0.083	0.176	0.064	0.080	0.235	0.730	0.917	0.885
CellSIUS	0.877	0.882	0.053	0.036	0.435	0.722	0.129	0.105	0.893	0.907
scAIDE	0.983	0.882	0.592	0.998	0.666	0.838	0.320	0.354	0.997	1.000
FiRE	0.953	0.893	0.292	0.076	0.485	0.343	0.233	0.213	0.970	0.915
RaceID	0.768	0.949	0.077	0.364	0.194	0.196	0.833	0.614	0.766	0.959

*IND means at individual level, POP means at population level.

4.2 Runtime and memory evaluation

Although RaceID has the highest overlaps within these 5 methods, it does have the longest running time. The running time of CellSIUS and scAIDE is second, and the memory they need is also greater than the other two methods. GiniClust3 and FiRE are indeed the fastest in terms of speed. Therefore, to select one software to use, we still need to choose according to our needs.

5. Conclusion:

In this study, we conducted two testing scenarios using ten datasets to address the challenge of rare cell type detection. Our findings suggest that when detecting rare cell types at an individual level, it is advisable to use either scAIDE or RaceID, depending on the specific needs. While scAIDE may produce more false negatives, RaceID may generate more false positives. Thus, it is crucial to clarify the objectives in advance at the population level, scAIDE, GiniClust3, and RaceID demonstrated excellent performance. However, considering their running times, we recommend using scAIDE and GiniClust3 for population-level analysis, as they offered both precise detection results and fast execution speed. In terms of consistency, scAIDE is the best, so we recommend using scAIDE to compare individual-level and population-level results.

Reference:

1. Shapiro, E., T. Biezuner, and S. Linnarsson, *Single-cell sequencing-based technologies will revolutionize whole-organism science.* Nat Rev Genet, 2013. **14**(9): p. 618-30.
2. Tsoucas, D. and G.C. Yuan, *Recent progress in single-cell cancer genomics.* Curr Opin Genet Dev, 2017. **42**: p. 22-32.
3. Liang, S.H., et al., *Single-cell manifold-preserving feature selection for detecting rare cell populations.* Nature Computational Science, 2021. **1**(5): p. 374-384.
4. Stegle, O., S.A. Teichmann, and J.C. Marioni, *Computational and analytical challenges in single-cell transcriptomics.* Nat Rev Genet, 2015. **16**(3): p. 133-45.
5. Repana, D., et al., *The Network of Cancer Genes (NCG): a comprehensive catalogue of known and candidate cancer genes from cancer sequencing screens.* Genome Biol, 2019. **20**(1): p. 1.
6. Jiang, L., et al., *GiniClust: detecting rare cell types from single-cell gene expression data with Gini index.* Genome Biol, 2016. **17**(1): p. 144.
7. Wegmann, R., et al., *CellSIUS provides sensitive and specific detection of rare cell populations from complex single-cell RNA-seq data.* Genome Biol, 2019. **20**(1): p. 142.
8. Grun, D., et al., *Single-cell messenger RNA sequencing reveals rare intestinal cell types.* Nature, 2015. **525**(7568): p. 251-5.
9. Xie, K., et al., *scAIDE: clustering of large-scale single-cell RNA-seq data reveals putative and rare cell types.* NAR Genom Bioinform, 2020. **2**(4): p. lqaa082.
10. Jindal, A., et al., *Discovery of rare cells from voluminous single cell expression data.* Nat Commun, 2018. **9**(1): p. 4719.
11. Yu, L., et al., *Benchmarking clustering algorithms on estimating the number of cell types from single-cell RNA-sequencing data.* Genome Biol, 2022. **23**(1): p. 49.
12. Fa, B.T., et al., *GapClust is a light-weight approach distinguishing rare cells from voluminous single cell expression profiles.* Nature Communications, 2021. **12**(1).
13. Zhang, Y., G. Parmigiani, and W.E. Johnson, *ComBat-seq: batch effect adjustment for RNA-seq count data.* NAR Genom Bioinform, 2020. **2**(3): p. lqaa078.
14. Schulte-Schrepping, J., et al., *Severe COVID-19 Is Marked by a Dysregulated Myeloid Cell Compartment.* Cell, 2020. **182**(6): p. 1419-+.
15. Su, Y.P., et al., *Multi-Omics Resolves a Sharp Disease-State Shift between Mild and Moderate COVID-19.* Cell, 2020. **183**(6): p. 1479-+.
16. Hao, Y., et al., *Integrated analysis of multimodal single-cell data.* Cell, 2021. **184**(13): p. 3573-3587 e29.
17. Johnson, W.E., C. Li, and A. Rabinovic, *Adjusting batch effects in microarray expression data using empirical Bayes methods.* Biostatistics, 2007. **8**(1): p. 118-27.
18. Robinson, M.D., D.J. McCarthy, and G.K. Smyth, *edgeR: a Bioconductor package for differential expression analysis of digital gene expression data.* Bioinformatics, 2010. **26**(1): p. 139-140.
19. Chen, Y.S., A.T.L. Lun, and G.K. Smyth, *Differential Expression Analysis of Complex RNA-seq Experiments Using edgeR.* Statistical Analysis of Next Generation Sequencing Data, 2014: p. 51-74.
20. Dong, R. and G.C. Yuan, *GiniClust3: a fast and memory-efficient tool for rare cell type identification.* BMC Bioinformatics, 2020. **21**(1): p. 158.
21. Tsoucas, D. and G.C. Yuan, *GiniClust2: a cluster-aware, weighted ensemble clustering method for cell-type detection.* Genome Biol, 2018. **19**(1): p. 58.
22. Wolf, F.A., P. Angerer, and F.J. Theis, *SCANPY: large-scale single-cell gene expression data analysis.* Genome Biol, 2018. **19**(1): p. 15.
23. Enright, A.J., S. Van Dongen, and C.A. Ouzounis, *An efficient algorithm for large-scale detection of protein families.* Nucleic Acids Res, 2002. **30**(7): p. 1575-84.

24. Wang, Z., et al., *Sizing Sketches: A Rank-Based Analysis for Similarity Search.* Sigmetrics'07: Proceedings of the 2007 International Conference on Measurement & Modeling of Computer Systems, 2007. **35**(1): p. 157-168.
25. Gionis, A., P. Indyk, and R. Motwani, *Similarity search in high dimensions via hashing.* Proceedings of the Twenty-Fifth International Conference on Very Large Data Bases, 1999: p. 518-529.
26. Grun, D., L. Kester, and A. van Oudenaarden, *Validation of noise models for single-cell transcriptomics.* Nat Methods, 2014. **11**(6): p. 637-40.
27. Metz, C.E., *Basic principles of ROC analysis.* Semin Nucl Med, 1978. **8**(4): p. 283-98.
28. Pontius, R.G. and M. Millones, *Death to Kappa: birth of quantity disagreement and allocation disagreement for accuracy assessment.* International Journal of Remote Sensing, 2011. **32**(15): p. 4407-4429.
29. McInnes, L., J. Healy, and J. Melville *UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction*. 2018. arXiv:1802.03426 DOI: 10.48550/arXiv.1802.03426.