Schneider *et al. J Transl Genet Genom* 2021;5:37-49
DOI: 10.20517/jtgg.2020.48

Journal of Translational Genetics and Genomics

Original Article

Open Access

Check for updates

Use of "default" parameter settings when analyzing single cell RNA sequencing data using Seurat: a biologist's perspective

Isaac Schneider[1], Jason Cepela[1], Mihir Shetty[1], Jinhua Wang[2,3], Andrew C. Nelson[2,4], Boris Winterhoff[1,2], Timothy K. Starr[1,2,5]

[1]Department of Ob-Gyn & Women's Health, University of Minnesota, Minneapolis, MN 55455, USA.
[2]Masonic Cancer Center, University of Minnesota, Minneapolis, MN 55455, USA.
[3]Institute for Health Informatics, University of Minnesota, Minneapolis, MN 55455, USA.
[4]Department of Lab Medicine & Pathology, University of Minnesota, Minneapolis, MN 55455, USA.
[5]Department of Genetics, Cell Biology & Development, University of Minnesota, Minneapolis, MN 55455, USA.

Correspondence to: Dr. Timothy K. Starr, Department of Ob-Gyn & Women's Health, University of Minnesota, 420 Delaware St SE, Minneapolis, MN 55455, USA. E-mail: star0044@umn.edu

How to cite this article: Schneider I, Cepela J, Shetty M, Wang J, Nelson AC, Winterhoff B, Starr TK. Use of "default" parameter settings when analyzing single cell RNA sequencing data using Seurat: a biologist's perspective. *J Transl Genet Genom* 2021;5:37-49. http://dx.doi.org/10.20517/jtgg.2020.48

Received: 29 Sep 2019 **First Decision**: 6 Nov 2020 **Revised**: 13 Nov 2020 **Accepted**: 27 Nov 2020 **Available online**: 1 Jan 2021

Academic Editor: Jinhua Wang **Copy Editor**: Cai-Hong Wang **Production Editor**: Jing Yu

Abstract

Aim: Analysis of large datasets has become integral to biological studies due to the advent of high throughput technologies such as next generation sequencing. Techniques for analyzing these large datasets are normally developed by bioinformaticists and statisticians, with input from biologists. Frequently, the end-user does not have the training or knowledge to make informed decisions on input parameter settings required to implement the analyses pipelines. Instead, the end-user relies on "default" settings present within the software packages, consultations with in-house bioinformaticists, or on methods described in previous publications. The aim of this study was to explore the effects of altering default parameters on the cell clustering solutions generated by a common pipeline implemented in the Seurat R package that is used to cluster cells based on single cell RNA sequencing (scRNAseq) data.

Methods: We systematically assessed the effect of altering input parameters by performing iterative analyses on a single scRNAseq dataset. We compared the clustering solutions using the different input parameters to determine which parameters have a large effect on cell clustering solutions.

Results: We used a range of input parameters for many, but not all, of the input parameters required by the Seurat R pipeline. We found that some input parameters had a very small effect on the clustering solution, while other parameters had a much larger effect.

Conclusion: We conclude that, when implementing the Seurat R package, the "default" parameters should be used with caution. We identified specific parameters that have a significant effect on clustering solutions.

Keywords: Single cell RNA sequencing, cell type annotation, Seurat R package, clustering algorithms

INTRODUCTION

Single cell RNA sequencing (scRNAseq) is a next generation sequencing technology that produces gene expression data on thousands of single cells. This rich dataset can be mined to answer questions such as what types of cells are present in a sample and what is the frequency of the different cell types. Our group is interested in understanding the basis of chemotherapy resistance in ovarian cancer, which is the main cause of mortality in women with this deadly cancer[1]. We are collecting tumor samples from women with ovarian cancer and subjecting them to scRNAseq and other molecular analyses. Our goal is to identify cell types that correlate with chemotherapy resistance. If we can identify these cell types, it may be possible to develop therapies that counteract these cells and/or their effects.

There are multiple inherent problems that make it difficult to perform robust unbiased cell clustering using scRNAseq data. First, gene expression measurements produced by scRNAseq technologies are not exact and spike-in experiments suggest that the technique that we use (10× Genomics 3' Chromium) only measures ~5%-10% of the polyA transcriptome[2]. Second, mRNA content in a cell will vary stochastically at different timepoints in the cell's life cycle, causing the data to be noisy[3]. Third, gold standard gene expression patterns in purified cell types are not yet established, making it difficult to compare gene expression in single cells to biologically well-characterized gene expression datasets[4]. Fourth, cancer cells and immune/stromal cancer associated cells often have different gene expression patterns than their normal counterparts[5,6]. For these reasons, it is difficult to perform unbiased clustering and conclude with certainty that one has identified "true" biological cell types.

To generate scRNAseq datasets, we use the 10× Genomics 3' Chromium platform followed by sequencing on Illumina machines. The data are initially processed using CellRanger software to produce a gene expression matrix based on universal molecular identifier (UMI) counts. A typical dataset will be a matrix of several thousand cells by ~18,000 genes with each cell having UMI counts for 1000-3000 genes. One of the initial steps is to perform unbiased clustering of the cells based on each cell's gene expression. In this paper, we describe our use of the Seurat R package[7] to perform this unbiased clustering. We identify parameters that have a large effect on the clustering solutions generated by the Seurat pipeline and make recommendations for choosing what values to use for these input parameters. We have also used other methods of clustering [e.g., non-negative matrix factorization in ccFindR[8], consensus methods in ClusterExperiment[9] and single-cell consensus clustering (SC3)[10], and imputation based methods like clustering through imputation and dimensionality reduction (CIDR)[11]] and methods of annotation that do not rely on clustering such as SingleR[12] and CellTypeR (manuscript in preparation). We found similar variability in the clustering solutions of these methods. There has been an explosion of methods developed and published to analyze scRNAseq data, which makes it difficult to select the ideal method[13].

Instead of providing "benchmarking" between different methods, the purpose of this paper is to provide guidance to setting parameters using the Seurat R package when analyzing scRNAseq generated from fresh

tumor samples. We recommend doing the same type of comprehensive analysis for any method before relying on "default" parameters.

The basic clustering algorithm used in Seurat is a shared nearest neighbor (SNN) graph-based clustering method[14]. The pipeline for performing unbiased cell clustering within the Seurat pipeline is: (1) filter the dataset based on minimum/maximum cut-offs for genes/cell, cells/gene, and optional parameters such as mitochondrial gene UMI count as a percentage of total; (2) normalize the data; (3) find variable genes; (4) scale the data; (5) identify principle components; and (6) identify cell "neighbors" and cell clusters. Each of these steps requires multiple user-defined input parameters. It should be noted that the developers of the Seurat pipeline are constantly upgrading and improving the methods integrated into the package based on their work and input from others. In this paper, we report our findings on how similar/different the clustering solutions are when using a range of input parameters for each of these steps when analyzing single cells dissociated from a fresh ovarian cancer tumor sample.

METHODS

Sample processing

A fresh tumor sample was collected from a patient enrolled in our ovarian cancer precision medicine initiative (OCPMI) following our IRB approved protocol (#2018NTLS170). Fat, fibrous, and necrotic areas were removed from the tumor sample and a 0.9-g sample was used for scRNAseq. A single cell suspension was created using the Miltenyi Biotech GentleMACs Tissue Dissociation Kit following protocol 2.2.1. Briefly, the tumor sample was minced and placed in a specialized conical tube containing a mixture of dissociation enzymes in media. The tube was placed on a mechanical rotator for 30 s followed by incubation on a rotator at 37 °C for 30 min. The process was repeated, and the cell solution was poured through a 70-micron filter to remove cell clumps and debris. Cells were treated with red cell lysis buffer for 5-10 min at room temperature, centrifuged, and resuspended in hypothermosol. An aliquot was removed for measuring cell viability using the Cell Countess (Life Sciences). The single cell solution was diluted to a concentration of 1000 viable cells/μL and transported to the sequencing facility on ice.

scRNAseq sequencing

Samples were sequenced at the University of Minnesota's Genomics Center using the 10× Genomics Single Cell 3' Protocol utilizing the ChromiumTM Single Cell 3' Library & Gel Bead Kit and Chromium™ Single Cell A Chip Kit following the manufacturer's protocol (Protocol document CG000183 Rev C). Approximately 20,000 cells were partitioned into nanoliter-scale Gel Bead-In-EMulsions (GEMs) with one cell per GEM. Within each GEM, cells were lysed, and then primers were released and mixed with cell lysate. Incubation of the GEMs produced barcoded, full-length cDNA from mRNA. The full-length, barcoded cDNA was then amplified by PCR prior to library construction. Sequencing was performed using an Illumina HiSeq 2500 or NovaSeq to a depth of at least 100 thousand paired-end reads per cell.

Sequence processing

Illumina raw sequencing output files were processed using Cell Ranger™ software (v. 3.02) to produce a filtered gene × cell matrix of UMI counts. The matrix consists of three output files that define a sparse matrix (Supplementary Materials, Files barcodes.tsv, features.tsv and matrix.mtx). The output statistics generated by CellRanger are listed in Supplementary Table 1. The filtered matrix was used as input for the Seurat R software package to create the Seurat R object. When we initiated this project, we were using Seurat v 3.0.0.9000, but we have tested the outputs using Seurat v 3.2.

Seurat analysis

We established a baseline analysis of our dataset based on our analysis of over 75 scRNAseq datasets from ovarian cancer patients. The baseline was established using preliminary attempts on many of the Seurat

parameters as well as iterative analyses testing multiple parameter variations. For this study, we demonstrate the effect of changing these parameters by comparing clustering results to our baseline analysis.

Cell filtering parameters

Initial cell filtering parameters for selecting cells based on number of genes/cell, UMI counts/cell, and percent mitochondrial genes were established based on manual visualization of graphic outputs for these metrics (Supplementary Figure 1A-C and seurat_filter_1.R). Images were analyzed for bi-modal distributions to determine if there were "outlier" populations. In Supplementary Figure 1A-C, there are no obvious bimodal distributions. We analyzed 75 similar samples over the course of our OCPMI program and identified an outlier population based on percent mitochondrial genes [Supplementary Figure 1C] and genes/cell [Supplementary Figure 1A]. Based on these multiple datasets, we set a minimum requirement of genes/cells at 100 and maximum percent mitochondrial genes at 0.75%. We also arbitrarily set the upper level of genes/cell at 10,000 and UMI counts/cell at 200,000 based on identification of bimodal peaks in other samples. The minimum number of cells/gene was arbitrarily set at 20, which eliminated ~7000 genes from the analysis [Supplementary Figure 1D]. All of these cut-offs were chosen after manual inspection of ~30-40 samples. For biologists with one or few datasets, these will all be arbitrarily set.

Normalization parameters

Normalization is important for scRNAseq datasets due to the sparse data, bias and noise inherent in this technique and multiple methods have been proposed for scRNAseq[15]. The Seurat package offers three methods of normalization: LogNormalize, centered log ratio transformation (CLR), and relative counts (RC). Our baseline analysis uses the LogNormalize method and we compared this to the RC method. Furthermore, we compared the results when using different scale factors (100,000 and 1000), which is a required parameter in these normalization methods (baseline = 10,000).

Variable gene parameters

Variable gene parameters were tested by comparing the three selection methods (vst, mean.var.plot, and dispersion). Comparisons were also conducted for different Loess Spans (0.3, 0.1, and 0.5), number of bins (1, 10, 20, and 100), binning method (equal_width or equal_frequency), and number of features based on percentage of total genes (20%, 1%, and 6.7%). Our baseline analysis used the vst method, a Loess span of 0.3, 20 bins using equal width, and a target of 6.7% of features [Supplementary Figure 1E].

Scale data parameters

When scaling the data, we compared the following four combinations of variables to regress: UMI count, percent mitochondrial genes, both UMI count and percent mitochondrial genes, or no variables regressed out. We also tested three values for the scale.max parameter (10, 50, and 100). For the baseline, we regressed out UMI count and percent mitochondrial genes with a maximum scale of 50. Note that we did not interrogate the different models or block sizes and minimum cells to block.

Clustering parameters

The Seurat implementation of SNN requires the following parameter inputs when running the FindNeighbors function: reduction type, number of dimensions, a k parameter, a prune parameter, a nearest neighbor method (rann or annoy), an annoy distance metric, and a nearest neighbor error boundary. After identifying cell neighbors, the FindClusters function identifies clusters and also requires the following input parameters: algorithm choice (Louvain, refined Louvain, SLM, or Leiden), a resolution parameter, and other parameters that we left at default. We limited our analysis to a principle components reduction.

A major input parameter required by the Seurat pipeline is the number of reduction dimensions to use. The Seurat pipeline recommends using an Elbow plot of PC standard deviations [Supplementary Figure 1F and G]

or a Jackstraw plot using iterative analysis, which assigns a *P*-value to each PC [Supplementary Figure 1H]. We found that the selection of this value has a large effect on the clustering solution, and neither the PC Elbow plot nor the Jackstraw plot provided an easily determined value to use.

To obtain a robust clustering solution, we performed iterative analyses using a range of values for several of these parameters and compared the clustering solutions to find the number of clusters that is most frequently obtained. To automate this iterative analysis, we wrote an R script (seurat_pkpr_loop.R) that tests ranges of the four most important clustering parameters: number of reduction dimensions (n = 8), k parameter (n = 8), prune parameter (n = 8), and resolution parameter (n = 10). For the range of PC dimensions to test, we selected the PC value with a Jackstraw calculated p-value closest to, but still below 1×10^{-10}, and the previous seven PC values (e.g., in Supplementary Figure 1H, we would select PC values from 42 to 49). The range of k-parameters was based on the cell count [Supplementary Table 2]. The range of prune parameters was from 0.2 to 1.6, in 0.2 increments, and the range of resolution parameters was from 0.4 to 2.2, in 0.2 increments. The R script runs the Seurat pipeline 5,120 times ($8 \times 8 \times 8 \times 10$) and returns graphs and tables, including summary tables. The summary tables are used to identify the most frequent clustering solution. The four parameters are then selected to produce that clustering solution (Supplementary Figure 2 and seurat_pkpr_loop.R_output_files).

Cell type assignment

A final custom script was used to assign cell types to clusters based on percentage overlap of upregulated genes in that cluster compared to gene lists generated for known cell types based on literature. We named this annotation method cluster annotation by differential gene expression (CADGE). The custom R script (seurat_final.R) and the gene lists (annotated_gene_lists) used to perform CADGE are provided in the Supplementary Materials.

Comparisons of cell type similarities and calculation of adjusted rand index

The percent cell type similarity was calculated by dividing the number of cells with equivalent cell type annotation by the total number of cells. The adjusted rand index (ARI) value was calculated by applying the adjustedRandIndex function from mClust[16] to the cluster assignment lists for each comparison analyses.

RESULTS

The Seurat R package is a popular scRNAseq analysis pipeline. The R package has incorporated functions to normalize and scale data, identify variable genes, perform dimensional reduction, and identify cell clusters based on gene expression similarities. The pipeline consists of loading UMI count data, performing normalization, identifying variable genes, scaling the data with an option to regress out variables, finding cell neighbors, and finding cell clusters [Figure 1]. The pipeline is agnostic to specific methods within these computational tasks and allows the user to specify methods and set parameters. For many of these methods and parameters, however, there are no established biological guidelines to use for selecting the appropriate method or parameter value.

To analyze the effects of the different methods and parameters on clustering results, we established a "baseline" analysis of an scRNAseq dataset we generated from an ovarian cancer tumor specimen. The dataset was produced using the 10× Genomics 3' Chromium platform followed by sequencing on an Illumina NovaSeq. To establish the baseline analysis, we filtered out cells and genes based on an analysis of more than 50 similar datasets (Supplementary Figure 1A-D and Methods).

Find neighbors and find clusters parameters

Next, we iteratively ran the Seurat pipeline 5,120 times testing all possible combinations of 8 PC dimensions, 8 *K* values, 8 prune values, and 10 resolution values (Supplementary Figure 2 and Methods).

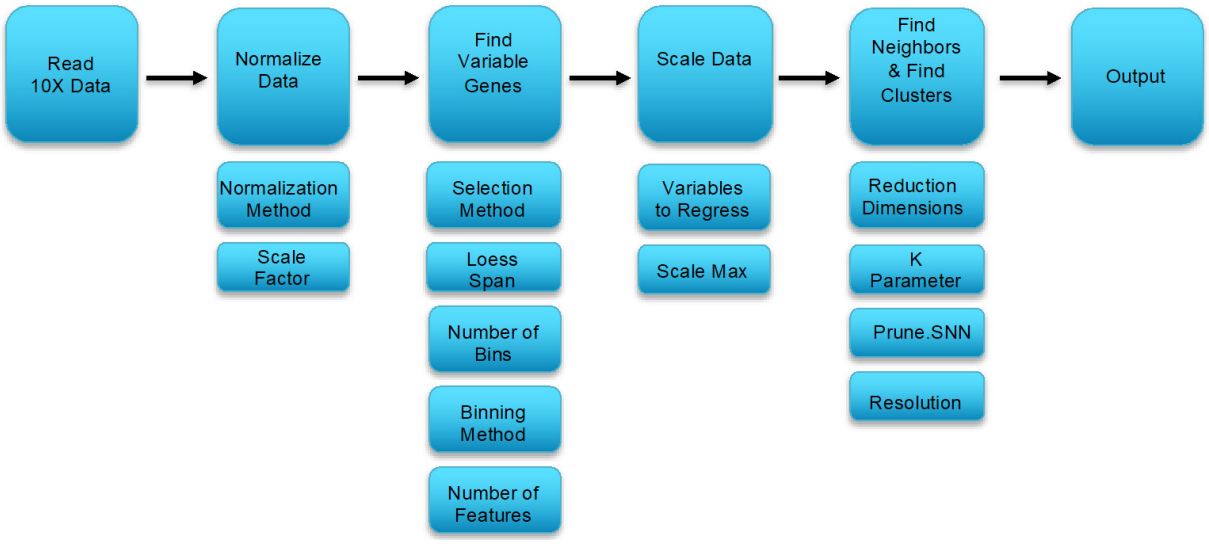

Figure 1. Seurat clustering pipeline with parameters that must be set at each step

These parameters are required to execute the FindNeighbors and FindClusters functions in Seurat. Large differences in clustering solutions occur when these parameters are altered, with clustering solutions ranging from 6 to 27 clusters [Supplementary Figure 2A]. From this iterative analysis, we identified a cluster solution of 14 clusters that was the most frequent cluster number solution in the 5,120 runs [Supplementary Figure 2A]. The 14 clusters are visualized using a uniform manifold approximation and projection (UMAP) projection [Figure 2A] and consists of clusters ranging in size from 44 to 608 cells [Figure 2B]. Clusters were annotated with the predicted cell type by calculating the percentage overlap between genes upregulated in each cluster (compared to other clusters) and genes in a set of annotated gene lists for purified cell types. The baseline analysis categorized clusters as immune cells, endothelial cells, epithelial cells, or fibroblasts [Figure 2C]. It is important to note that the UMAP and t-distributed stochastic neighbor embedding (TSNE) clustering dimensional reductions used to visualize the cells are not incorporated into the Seurat SNN graph-based clustering. The UMAP and TSNE clusters tend to be similar to the SNN graph-based clustering, but there are many cells that are clustered discordantly between the methods, which can be detected as mis-matched colored cells within the groupings produced by UMAP and TSNE. The UMAP and TSNE plots are helpful for visualization, but they are not used for the actual clustering. The parameter settings used to produce the baseline analysis depicted in Figure 2 are listed in Supplementary Table 3 and the script is included in the Supplementary Materials (File seurat_final.R).

Normalization parameters

The Seurat package gives three options for normalizing data: natural log transformation using log1p (LogNormalize), relative counts (RC), and a centered log ratio transformation (CLR). Our baseline analysis used the natural log transformation. Each of these methods produces slightly different clustering solutions. Both the RC and CLR methods produced fewer clusters compared to the LogNormalize method (11 and 10 *vs.* 14, respectively) [Figure 3A]. Cells are annotated similarly in all three methods [Figure 3B], but actual cluster placement varies considerably between the three methods, as is evident by their ARI comparison values in the 0.6 range [Figure 3C]. The ARI value is a measure of similarity/dissimilarity between two clustering solutions with a value of 1 representing complete similarity and a value of 0 representing complete dissimilarity.

In addition to a normalization method, Seurat requires users to select a scale factor. Compared to our baseline scale factor value of 10,000, increasing the scale factor by a factor of 10 when normalizing the

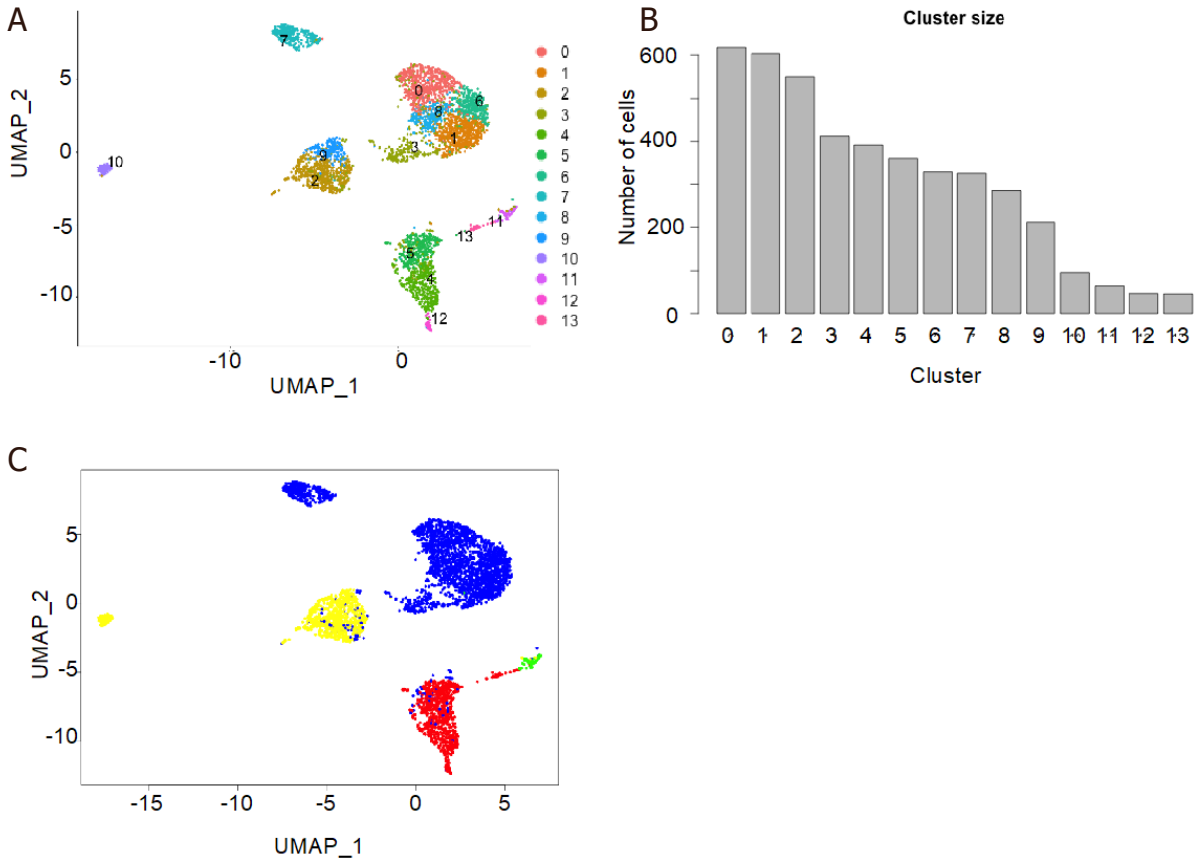

Figure 2. UMAP projection of 4,344 ells from an ovarian cancer tumor specimen colored by cluster number (A); number of cells per cluster (B); and cells colored by cell type determined by analysis of upregulated genes in each cluster compared to gene lists of known cell types (C) (see Methods). Blue: cancer epithelial cells; red: fibroblasts; yellow: immune cells; green: endothelial cells. UMAP: uniform manifold approximation and projection

data resulted in loss of a cluster, while decreasing the scale factor by a factor of 10 generated an additional cluster when compared to the baseline [Figure 3D]. Cells were generally annotated similarly [Figure 3E], and the clustering solutions were more similar, based on ARI values in the 0.8 range [Figure 3F].

Visualization of cell clusters using UMAP or TSNE allows for manually associating clusters with one another. The validity of this manual grouping of clusters is supported by similar cell type annotations, which generally indicates that clusters within a group of clusters are all assigned the same cell type annotation [Figure 4]. However, subtle but important differences arise between the clustering solutions produced by the three normalization methods. Some clusters are identified by one method but not the others (Clusters 3 and 12 in Figure 4A and Cluster 10 in Figure 4C). Another difference appears when one method defines a group of cells as a single cluster, while another method splits the group. For example, the fibroblast cells are split into three groups in the LogNormalize method [Figure 4A and D], but only two groups in the RC and CLR methods [Figure 4B, C, E, F], and the two groups produced by the RC and CLR methods are not the same.

An analysis of the clustering solutions produced using different scale factors also reveals subtle differences between them. For example, the additional cluster identified when the scale factor was reduced (Cluster 14, Figure 5A and B) was a small group of cells originally identified as macrophages in the baseline analysis (Cluster 2, Figure 5A and D). With the new clustering solution, the cells switched annotation to B cells [Figure 5B and E]. The cluster that was lost when the scale factor was increased (Cluster 12, Figure 5A and C)

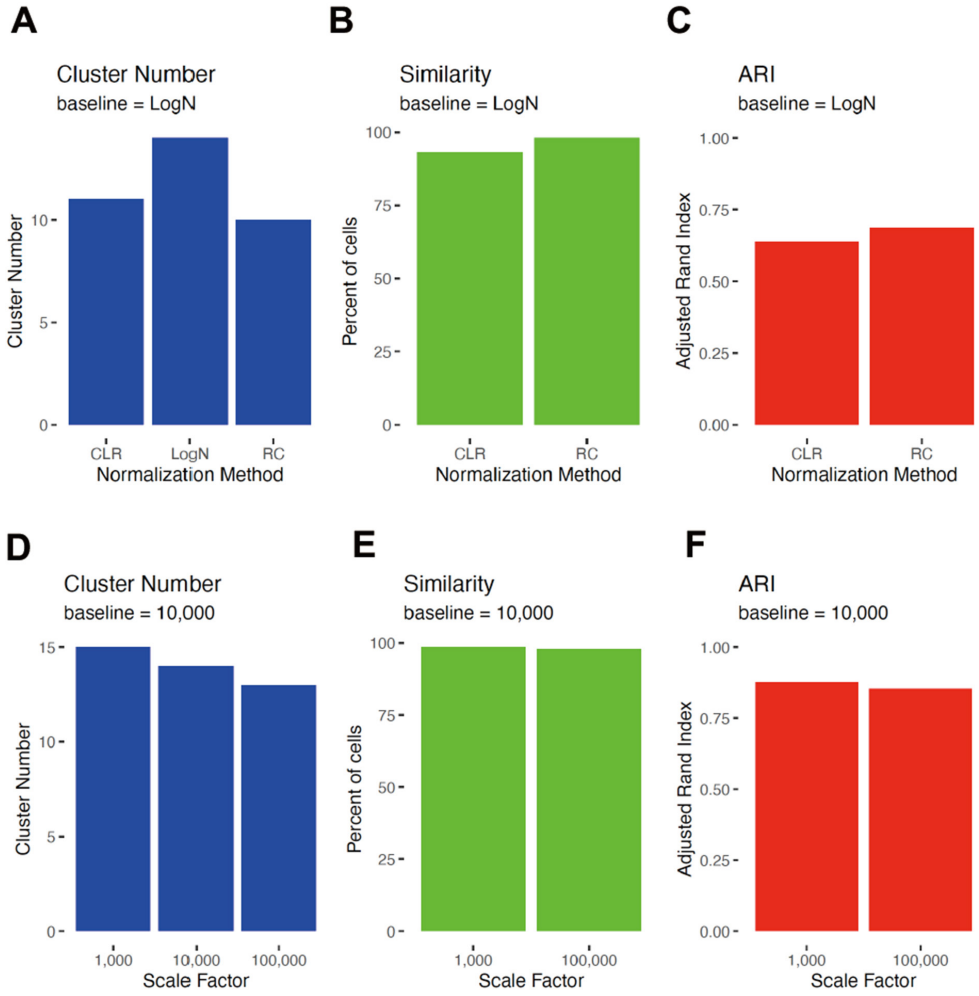

Figure 3. Cluster number (A, D); similarity by cell type (B, E); and ARI (C, F) when comparing baseline method (LogNormalize, Scale Factor 10,000) to the RC or CLR methods (A-C) or to scale factors of 1000 or 100,000 (D-F). ARI: adjusted rand index

was re-classified as endothelial cells in the new clustering solution (Cluster 10, Figure 5C and F). This change is reflected in the small percentage of cells that have dissimilar cell calls and in the reduced ARI comparing the clustering solutions to baseline [Figure 3E and F].

Variable gene parameters

Three methods are available for selecting variable genes in the Seurat pipeline (vst, mean.var.plot, and dispersion). Running our baseline analysis using the three different methods resulted in the following number of variable genes per method: vst (baseline) = 1,125, mean.var.plot = 818, and dispersion = 1,125. Of the total number of unique genes selected by all three selection methods (n = 1,750), only 439 (25%) were selected by all three methods [Figure 6A]. Half of the genes identified as variable were only identified by a single method (871/1750 = 50%). Somewhat surprisingly, even with these different sets of variable genes, the clusters detected were similar [Figure 6B], and only 2% of cells were annotated discordantly [Figure 6C]. Cell assignments to clusters, however, did vary, as evidenced by the low ARI values [Figure 6D]. Based on TSNE visualization, we manually assigned clusters into superclusters based on their visual proximity to each other [Figure 7A-C]. Over 96% of cells were placed into concordant superclusters.

Because the vst method uses a local polynomial regression (loess), an input parameter required is the loess span. We compared our baseline analysis (loess span = 0.3) to loess span values of 0.1 and 0.5. There were

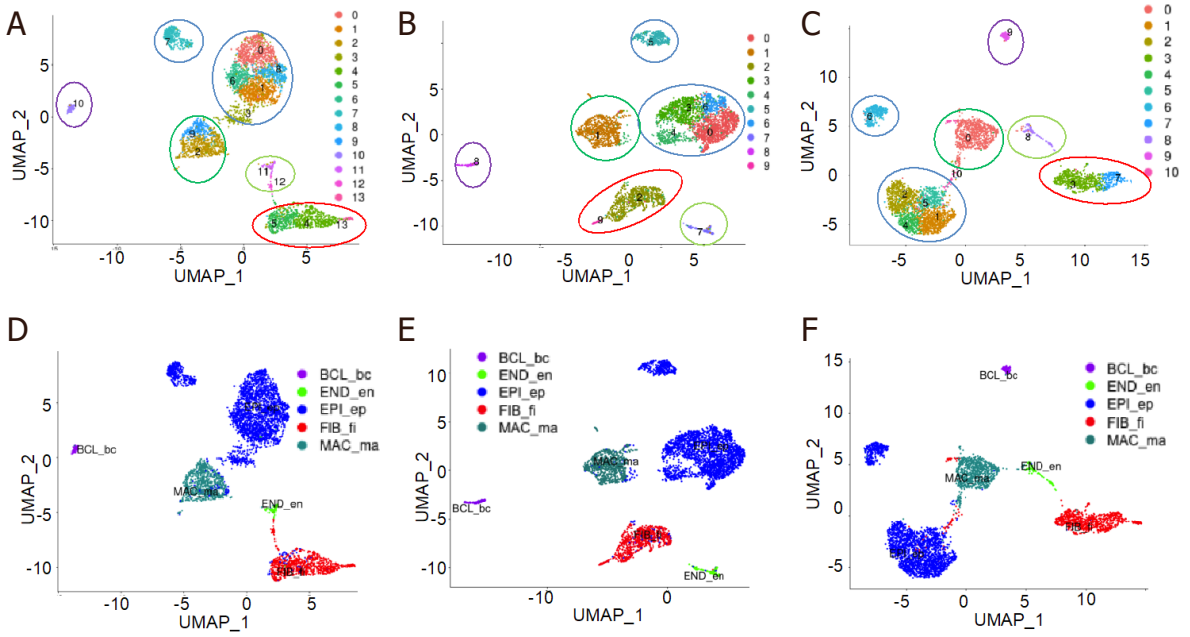

Figure 4. UMAP plots of clustering solutions using different methods of normalization. LogNormalize (A, D baseline) was compared to RC (B, E) and CLR (C, F) normalization methods. Colored circles are manually placed around clusters that were annotated similarly. BCL_bc: B cells; END_en: endothelial cells; EPI_ep: epithelial cancer cells; FIB_fi: fibroblasts; MAC_ma: macrophages. UMAP: uniform manifold approximation and projection

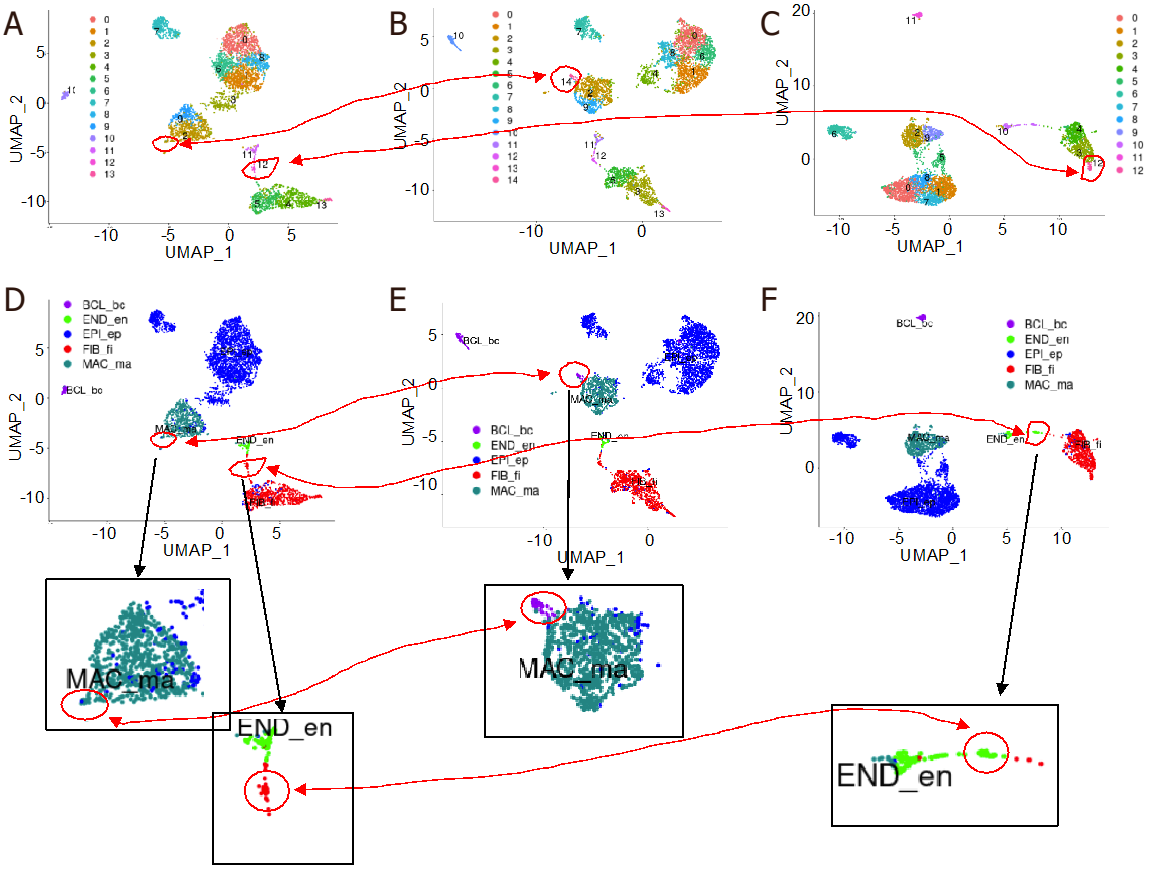

Figure 5. UMAP plots of clustering solutions using scale factors of: 10,000 (baseline) (A, D); 1000 (B, E); and 100,000 (C, F). The cells are colored by: cluster (A-C); and cell type (D-F). Red circles and arrows indicate cluster gained when scale factor was reduced to 1000 (B, E) and cluster lost when increasing scale factor to 100,000 (C, E). Boxes below (D-F) show enlarged versions of the clusters within the red circles. UMAP: uniform manifold approximation and projection

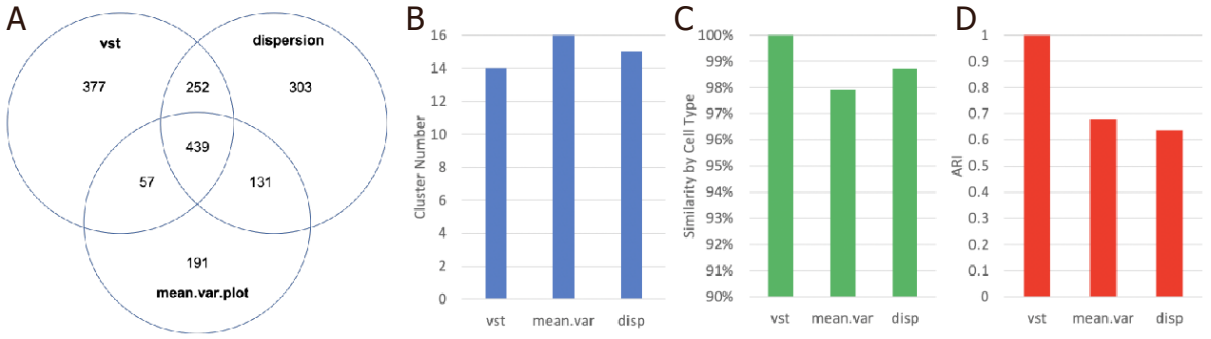

Figure 6. Results of using different methods of determining variable genes. Venn diagram illustrating overlap of variable genes detected by three different methods (A) (vst, mean.var.plot, and dispersion). Bar graphs showing differences in: cluster number (B); percent of cells called concordantly (C); and the ARI measurement (D). Mean.var and dispersion methods are compared to the baseline method, vst in (C, D). ARI: adjusted rand index

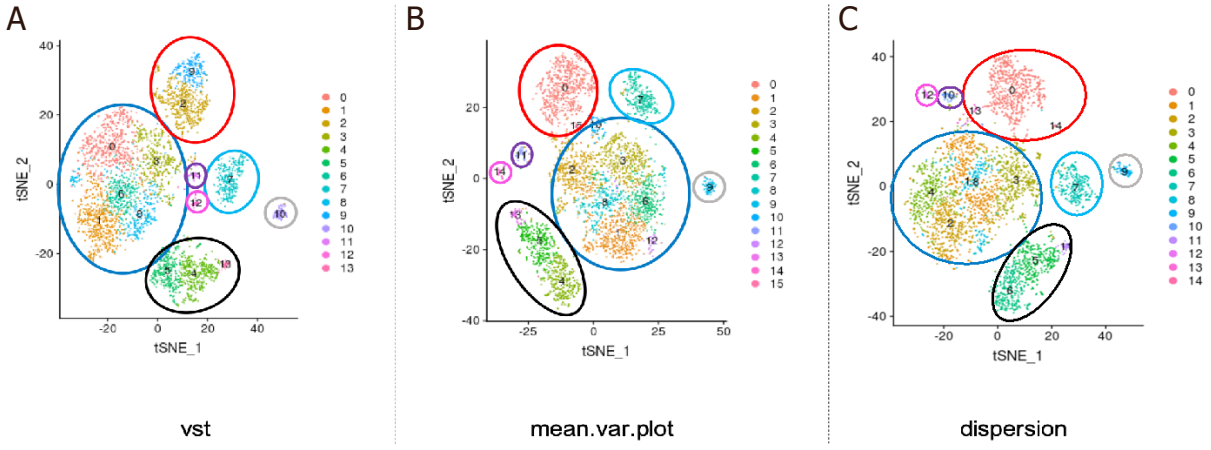

Figure 7. Clusters generated using three different lists of variable genes selected by the three methods (vst, mean.var.plot, and dispersion). Clusters were manually assigned to "superclusters" based on proximity in the TSNE plots. TSNE: t-distributed stochastic neighbor embedding

almost no differences in cluster numbers, cell annotations, and ARI values when varying the loess span [Supplementary Figure 3A-C], bin widths [Supplementary Figure 3D-F], or binning method [Supplementary Figure 3G-I].

The number of variable genes used will affect clustering. For our baseline analysis, we set the number of variable genes to be 6.7% of all genes detected. This number was chosen because it produced a list of ~1000 variable genes if 15,000 total genes are detected. There is no strong biological rationale, however, to choose this cut-off. We compared our baseline analysis (6.7%) to analyses using 1% and 20% of total genes. Our baseline analysis resulted in 1,125 variable genes, while the 1% and 20% cut-offs resulted in 168 and 3,358 variable genes, respectively. Reducing the variable gene list to 1% of total genes had a strong effect on clustering, with a large number of cells being placed in different clusters based on the ARI value [Figure 8C]. Increasing the variable gene list to 20% of all genes produced a similar clustering solution to the baseline analysis using 6.7% of genes [Figure 8].

Scale data parameters

Before scaling and centering data, it is often recommended to "regress out" variables that could affect data analysis. The Seurat pipeline allows users to regress out any variables. Our baseline analysis regressed out

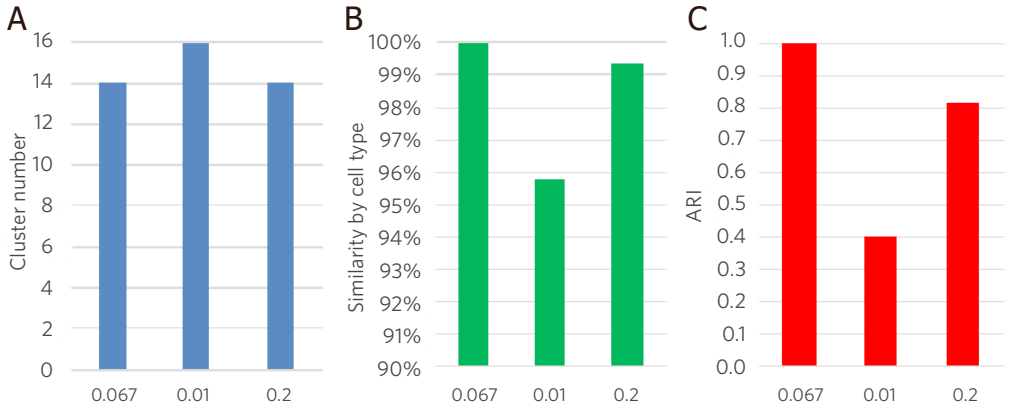

Figure 8. Bar graphs showing differences when using variable gene lists of 1,125 (0.067 baseline) compared to 168 genes (0.01) and 3,358 genes (0.2) in: (A) cluster number; (B) percent of cells called concordantly; and (C) the ARI measurement. ARI: adjusted rand index

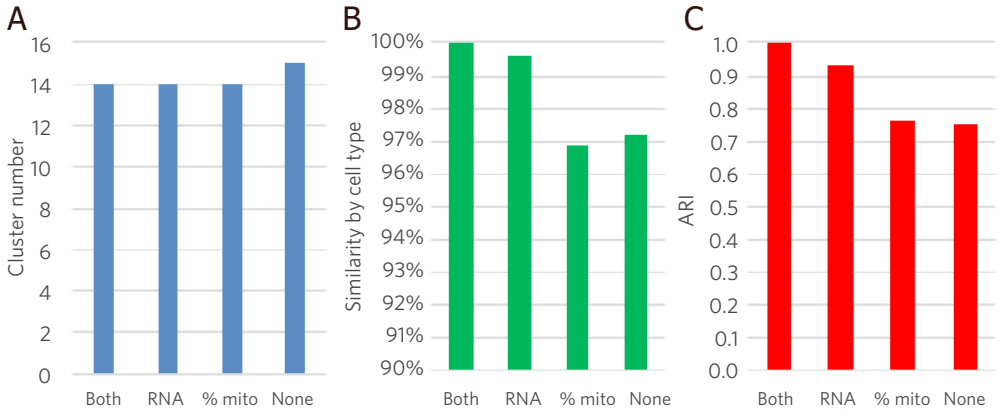

Figure 9. Bar graphs showing differences when regressing out total UMI count (RNA) and percent of UMI count attributed to mitochondrial genes (% mito) in: (A) cluster number; (B) percent of cells called concordantly; and (C) the ARI measurement. Baseline analysis regressed both RNA and %mito (both). UMI: universal molecular identifier; ARI: adjusted rand index

both the total UMI count and the percentage of total genes attributed to the 13 mitochondrial genes. When we compared our baseline to analyses that regressed out these variables individually, or did not regress out any variables, we found that cluster solutions were very similar and 97% of cells were assigned the same cell type [Figure 9A and B]. There was, however, a significant number of cells that were not placed in similar clusters, as evidenced by ARI values below 0.8 [Figure 9C]. The cluster solutions, cell assignments, and ARI value were most similar comparing the baseline (Both) to regressing out only the total UMI counts, suggesting the addition of regressing out percent mitochondrial genes does not change clustering solutions greatly.

The ScaleData function also requires a maximum scale value, which defaults to 10 or 50 depending on the method used. For the baseline analysis, we used a maximum scale value of 50 and then compared it to maximum values of 10 or 100. We found almost equivalent clustering solutions using all three values [Supplementary Figure 4].

DISCUSSION

Identifying cell clusters is the main output of the Seurat pipeline. A large portion of subsequent downstream analysis will use these clusters to categorize cells and use their aggregated gene expression to make phenotype hypotheses and to make comparisons to other samples. A frequent observation noted

in papers presenting their analysis of scRNAseq datasets is that they identified "X" number of cell types based on this clustering. We found, however, that the number of clusters identified can easily be changed by altering the parameters used for clustering. In this study, we interrogated the effects of altering many of these parameters and report which changes had a strong effect on the clustering solutions.

The Seurat R package, at its core, uses an SNN graph-based algorithm to identify cell clusters based on UMI count data. The most important parameters affecting the clustering solutions include the number of dimensional reductions to use, the k-parameter, a prune parameter, and a resolution parameter. Altering these values generated clustering solutions ranging from 6 to 27 clusters [Supplementary Figure 2A]. As there is no biological rationale for choosing the exact values of these parameters, we recommend running an iterative analysis over a range of values and select the clustering solution that is produced at the highest frequency [Supplementary Figure 2A].

Altering other parameters will also have subtle effects on the clustering solutions produced. We found that the normalization method [Figure 3A-C] has a stronger effect than changing the scale factor during normalization [Figure 3D-F], with both of them changing cluster solutions [Figures 4 and 5]. Changing the method used to select variable genes had a significant effect on which genes were chosen [Figure 6A] and resulted in somewhat dissimilar cluster placement [Figure 6D]. Surprisingly, however, the cell type assignments were highly similar [Figures 6C and 7].

Increasing the number of variable genes to use, from ~1000 to ~3000, did not affect the clustering solution as much as lowering the number of variable genes from ~1000 to ~125 [Figure 8]. The parameters for loess span, bin widths, or binning method used when finding variable genes had negligible effects on the clustering solutions [Supplementary Figure 3]. When scaling the data, in our dataset, regressing out UMI count had a strong effect, while regressing out mitochondrial gene percentages had a much smaller effect [Figure 9]. Changing the scale maximum when scaling data had essentially no effect on the clustering solutions [Supplementary Figure 4].

In conclusion, biologists will always be dependent on statisticians and bioinformaticians to analyze the large datasets being generated by rapidly advancing technologies. Relying on default parameters built into the analysis packages, however, could result in analyses that do not reveal the true biological attributes of the sample being studied. An analogy could be made to performing a cell culture experiment and altering variables such as the timing of measurements, temperature, or the media being used before making conclusions about the cell properties. We recommend that computational "replicates" be conducted when using R packages such as Seurat to analyze biological datasets by varying the input parameters with each replicate. This is especially important when there is not a strong biological rationale for setting a given parameter.

DECLARATIONS

Acknowledgments

The authors would like to acknowledge the assistance received from Joshua Baller, Ying Zhang, Christine Henzler and Marissa Macchiatto at the Minnesota Supercomputing Institute at the University of Minnesota. The authors also are grateful to John Garbe, Emma Stanley and Jerry Daniels at the University of Minnesota Genomics Center for their assistance in generating the scRNAseq data.

Authors' contributions

Made substantial contributions to conception and design of the study: Wang J, Nelson AC, Winterhoff B
Made substantial contributions to design of the study, performed data acquisition and data analysis: Cepela J, Shetty M

Made substantial contributions to all aspects of the study, including writing the manuscript: Schneider I, Starr TK

Availability of data and materials

All data, including raw UMI counts, R scripts, and R script output files are found in the supplemental files.

Financial support and sponsorship

This work was supported by a grant from the Ovarian Cancer Research Alliance (Liz Tiberius grant to Winterhoff B), a grant from the University of Minnesota Grand Challenges project (to Winterhoff B, Nelson AC and Starr TK), and a grant from the University of Minnesota Masonic Cancer Center (to Starr TK).

Conflicts of interest

All authors declared that there are no conflicts of interest.

Ethical approval and consent to participate

The study was approved by the University of Minnesota's Institutional Review Board (#2018NTLS170). Informed consent was obtained from the patient.

Consent for publication

Not applicable.

REFERENCES

1. Ren F, Shen J, Shi H, Hornicek FJ, Kan Q, Duan Z. Novel mechanisms and approaches to overcome multidrug resistance in the treatment of ovarian cancer. *Biochim Biophys Acta* 2016;1866:266-75.
2. Zheng GX, Terry JM, Belgrader P, et al. Massively parallel digital transcriptional profiling of single cells. *Nat Commun* 2017;8:14049.
3. Brennecke P, Anders S, Kim JK, et al. Accounting for technical noise in single-cell RNA-seq experiments. *Nat Methods* 2013;10:1093-5.
4. Mereu E, Lafzi A, Moutinho C, et al. Benchmarking single-cell RNA-sequencing protocols for cell atlas projects. *Nat Biotechnol* 2020;38:747-55.
5. Sahai E, Astsaturov I, Cukierman E, et al. A framework for advancing our understanding of cancer-associated fibroblasts. *Nat Rev Cancer* 2020;20:174-86.
6. Thorsson V, Gibbs DL, Brown SD, et al; Cancer Genome Atlas Research Network. The immune landscape of cancer. *Immunity* 2019;51:411-2.
7. Stuart T, Butler A, Hoffman P, et al. Comprehensive Integration of Single-Cell Data. *Cell* 2019;177:1888-902.e21.
8. Woo J, Winterhoff BJ, Starr TK, Aliferis C, Wang J. De novo prediction of cell-type complexity in single-cell RNA-seq and tumor microenvironments. *Life Sci Alliance* 2019;2:e201900443.
9. Risso D, Purvis L, Fletcher RB, et al. clusterExperiment and RSEC: a bioconductor package and framework for clustering of single-cell and other large gene expression datasets. *PLoS Comput Biol* 2018;14:e1006378.
10. Kiselev VY, Kirschner K, Schaub MT, et al. SC3: consensus clustering of single-cell RNA-seq data. *Nat Methods* 2017;14:483-6.
11. Lin P, Troup M, Ho JW. CIDR: ultrafast and accurate clustering through imputation for single-cell RNA-seq data. *Genome Biol* 2017;18:59.
12. Aran D, Looney AP, Liu L, et al. Reference-based analysis of lung single-cell sequencing reveals a transitional profibrotic macrophage. *Nat Immunol* 2019;20:163-72.
13. Hie B, Peters J, Nyquist SK, Shalek AK, Berger B, Bryson BD. Computational methods for single-cell RNA sequencing. *Annu Rev Biomed Data Sci* 2020;3:339-64.
14. Waltman L, van Eck NJ. A smart local moving algorithm for large-scale modularity-based community detection. *Eur Phys J B* 2013;86.
15. Lytal N, Ran D, An L. Normalization methods on single-cell RNA-seq data: an empirical survey. *Front Genet* 2020;11:41.
16. Scrucca L, Fop M, Murphy TB, Raftery AE. mclust 5: clustering, classification and density estimation using gaussian finite mixture models. *R J* 2016;8:289-317.