NAR Genomics and Bioinformatics, 2024, **6**, lqae134
https://doi.org/10.1093/nargab/lqae134
Advance access publication date: 28 September 2024
Application Notes

scDAPP: a comprehensive single-cell transcriptomics analysis pipeline optimized for cross-group comparison

Alexander Ferrena [1,2], Xiang Yu Zheng [1], Kevyn Jackson[1], Bang Hoang[3], Bernice E. Morrow[1,4] and Deyou Zheng [1,5,6,*]

[1]Department of Genetics, Albert Einstein College of Medicine, Bronx, NY, USA
[2]Institute for Clinical and Translational Research, Albert Einstein College of Medicine, Bronx, NY, USA
[3]Department of Orthopedic Surgery, Montefiore Medical Center, Albert Einstein College of Medicine, Bronx, NY, USA
[4]Departments of Obstetrics and Gynecology, and Pediatrics, Albert Einstein College of Medicine, Bronx, NY, USA
[5]Department of Neurology, Albert Einstein College of Medicine, Bronx, NY, USA
[6]Department of Neuroscience, Albert Einstein College of Medicine, Bronx, NY, USA

[*]To whom correspondence should be addressed. Tel: +1 718 678 1217; Email: deyou.zheng@einsteinmed.edu

Abstract

Single-cell transcriptomics profiling has increasingly been used to evaluate cross-group (or condition) differences in cell population and cell-type gene expression. This often leads to large datasets with complex experimental designs that need advanced comparative analysis. Concurrently, bioinformatics software and analytic approaches also become more diverse and constantly undergo improvement. Thus, there is an increased need for automated and standardized data processing and analysis pipelines, which should be efficient and flexible too. To address these, we develop the **s**ingle-**c**ell **D**ifferential **A**nalysis and **P**rocessing **P**ipeline (scDAPP), a R-based workflow for comparative analysis of single cell (or nucleus) transcriptomic data between two or more groups and at the levels of single cells or 'pseudobulking' samples. The pipeline automates many steps of pre-processing using data-learnt parameters, uses previously benchmarked software, and generates comprehensive intermediate data and final results that are valuable for both beginners and experts of scRNA-seq analysis. Moreover, the analytic reports, augmented by extensive data visualization, increase the transparency of computational analysis and parameter choices, while facilitate users to go seamlessly from raw data to biological interpretation. scDAPP is freely available under the MIT license, with source code, documentation and sample data at the GitHub (https://github.com/bioinfoDZ/scDAPP).

Introduction

Advancements in single-cell transcriptomics technologies and reductions in cost have greatly increased the scale and complexity of experiments using single-cell and single-nucleus RNA-seq (scRNA-seq and snRNA-seq) (1). In 2023, nearly 4000 papers were published using scRNA-seq, illustrating an exponential increase over the past 5 years (PubMed). Not only have these methods been used for routine categorization of cell population in tissues or biological samples, but also increasingly as read-outs of population and gene program changes across experimental conditions, such as genetic knockouts or drug treatments. As the technologies for data acquisition become more sophisticated, the bottleneck of innovation has shifted to efficient and rigorous bioinformatic analysis, to guide investigators to assess data quality rapidly and use benchmarked software for uncovering biological signals efficiently and robustly. Standard, scalable and modular workflows would greatly facilitate this process.

Bioinformatic algorithms and software for single cell data analysis have also evolved rapidly and become challenging for ordinary researchers to follow. Various benchmark comparisons of software, however, have provided excellent recommendations for selecting methods for most data analysis steps, such as statistically rigorous strategies for cross-experimental-condition comparisons, including differential expression analysis and differential cell composition analysis (2–5). Such studies have shown that the most important criterion for robust methodological performance is the capacity to explicitly model biological replicate variability via 'pseudobulking', which is to aggregate information (e.g. counts) for cells of the same type in each replicate. The alternative of treating data of individual cells as independent measurements could easily lead to inflated and inappropriate statistics, due to inherent correlation among cells in the same sample and large cell numbers, resulting in small effect-size discovery that is less biologically relevant and prone to noises. Related to this, advancements in sample multiplexing, such as Cell Hashing, Multi-Seq or 10X Cell Multiplexing, have significantly increased the number of replicates (at reduced cost) and complexity of data analysis (6,7). Additionally, we demonstrated previously that the Reference Principal Component Integration ('RPCI') algorithm, released in the Robust Integration of scRNA-seq data ('RISC') software package, could integrate multiple-sample data with high accuracy, while avoiding over-correction (8).

We present here an R-based pipeline called single-cell Differential Analysis and Processing Pipeline (scDAPP) for

Received: April 12, 2024. **Revised:** July 7, 2024. **Editorial Decision:** September 10, 2024. **Accepted:** September 18, 2024

cross-group comparative analysis of scRNA-seq and snRNA-seq data from 10X Genomics platform. Our design emphasizes scalability, ease of use, user-friendly graphic visualization, transparency, and reproducibility. Compared to similar pipelines, including Cellsnake, Cellenics, scDrake, SingleCAnalyzer, and the Single-Cell Omics workbench (9–13), scDAPP focuses more on comparative analysis across groups or conditions, with implementations specifically for using replicates. Furthermore, it supports complex multi-group comparisons (such as Drug A versus Drug B versus Control) and comprehensive downstream bioinformatics analysis, such as Gene Set Enrichment Analysis (GSEA) and transcription factor target analysis. Additionally, flexible input formats allow for direct data importing from raw CellRanger outputs or pre-processed scRNA-seq data objects. Importantly, using enriched visualization, scDAPP is designed to guide users to examine and understand the selection of parameters in each step of the data analysis, so that they can have a good grasp of the options and make appropriate and rational adjustments. The enriched visualization is especially valuable because it shows the underlying data distributions and moreover how parameter choices affect the analytic results. Overall, scDAPP facilitates and systematizes data processing steps and allows users to quickly delve into biological interpretation, thus advancing the rigor and scalability of single-cell transcriptomic data analysis.

Materials and methods

Overview of scDAPP core functions

The scDAPP wraps previously published software packages into R markdown codes, with critical pipeline-specific implementation (Figure 1A). It starts with several cell filtering functions using parameters learnt from input data's distribution, including doublet removal, followed by individual sample analysis, integration of samples, clustering of the integrated data, and cross-group comparisons of cell cluster abundance and gene expression using either cell-level or pseudobulk-level data (when replicates are used). The core packages include Seurat (v5 and up), RISC/RPCI, Speckle, EdgeR, DESeq2, FGSEA and ClusterProfiler for various analyses (8,14–17), with critical function extensions or modifications specific for scDAPP. Label transfer from existing cell-type annotation is an optional feature to facilitate cell cluster identification.

scDAPP run configuration and input data

Key input options and formats for the pipeline are shown in Figure 1B and C. The input for scDAPP is the raw (non-normalized) Unique Molecular Identifier (UMI) counts matrix for each sample. This can be either the filtered matrices from CellRanger ('filtered_feature_bc_matrix.h5' files) or Seurat objects. The latter allows for flexibility, for example, to use data that have been filtered or processed by other means. In addition, pipeline run configuration files are needed, which specify sample-wise metadata including per-sample information and optional sample nickname codes ('sample_metadata'), and a list of the relevant cross-group comparisons ('comps') (Figure 1B).

One critical input parameter is 'Pseudobulk_mode' (TRUE/FALSE) that will specify whether to make cross-group comparisons using samples in the same group/condition as replicates (Figure 1A). Setting it to 'TRUE' is highly recommended when biological replicates are available. This has two effects: (i) Differential gene expression analysis will be run in a pseudobulk manner, invoking the EdgeR-Likelihood Ratio Test (or DESeq2) on the aggregated reads from all cells in each cluster in each replicate (2). (ii) Differential cell composition analysis will also be run in a replicate-aware manner via the 'Propeller' test from the 'speckle' package, which was demonstrated to perform more accurately (3). Conversely, setting the 'Pseudobulk_mode' parameter to 'FALSE' will evoke scDAPP to perform differential gene expression analysis by the Wilcoxon or other tests in Seurat at the single cell level (i.e. treating each cell as an independent data point), and differential cell abundance analysis by the two-proportion Z test. This non-replicate option is only recommended for comparative analysis without replicates but can be run on inputs with replicates. It is more prone to false positive, but this is sometimes unavoidable, such as in the context of pilot studies.

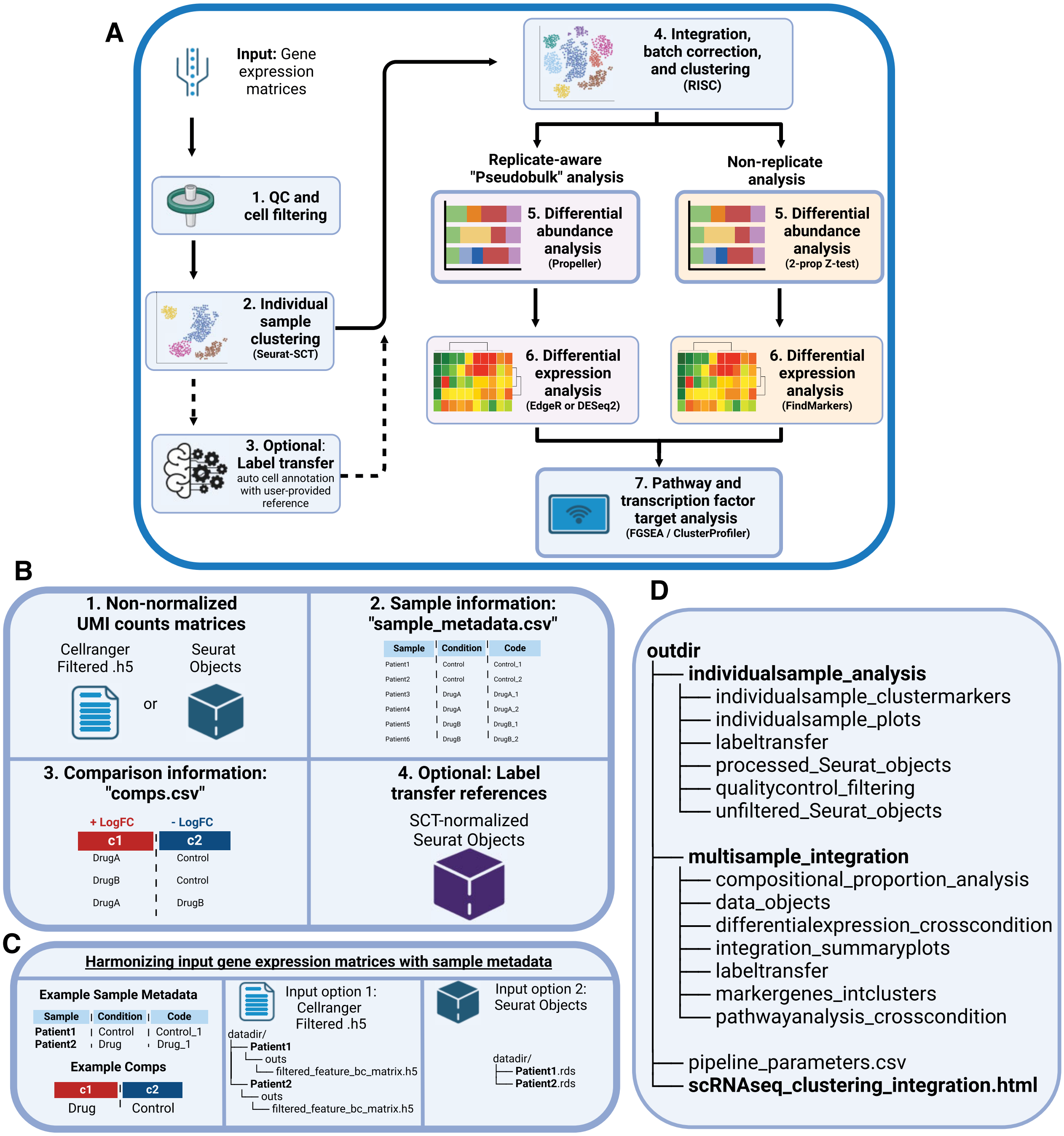
Two other key options are provided (Figure 1A). The first is 'use_labeltransfer' for invoking the use of the Seurat label transfer workflow. If set to 'TRUE', two additional parameters need to be provided: one refers to a Seurat object containing normalized data from the Seurat's SingleCellTransform workflow and a metadata column called 'Celltype' to be used for label transfer, and the other ('m_reference') points to a table file listing marker genes for the reference cell types in the output format of the Seurat 'FindAllMarkers' function. The second is the 'risc_reference' parameter, specifically related to the RISC software. The RPCI algorithm uses one of the input samples to learn the principal component space to project cells in all samples. Users can specify which sample to be the RPCI reference, after they examine the clustering results from all samples. Alternatively, and by default, scDAPP provides an automated RISC reference selection algorithm (described below).

The pipeline input has additional fields related to quality control metric thresholds, tuning of data-driven cell filtering, and hyperparameter selection in clustering analysis, with reasonable defaults, as described below. One more required input is 'species', which is used to search the MSIGDB database for the correct gene symbols during pathway enrichment analysis, via the msigdbr package (18).

Description of step-by-step components

QC and cell filtering

As shown in Figure 1A, the first step in scDAPP is quality control (QC) and filtering of poor-quality cells. On top of pre-set relaxed thresholds, scDAPP tries to learn better cutoffs from the input data directly. Currently scDAPP considers the following information for cell filtering: number of UMIs per cell, number of unique genes (i.e. features) per cell, percent of UMIs from the mitochondrial genes per cell, and the prediction of doublets and multiplets. Additionally, though not commonly implemented in other software, scDAPP evaluates the percent of reads from hemoglobin-related genes, as red blood cell lysis buffer may miss its target population; these cells have the very distinct feature of extremely high hemoglobin gene expression. For each of these QC metrics, users may select initial relaxed thresholds, such as minimum UMIs (500 by default), minimum number of unique genes (200), maximum percent of mitochondrial reads (25%), and maximum percent of hemoglobin read (25%). On top of these user's settings, scDAPP will further optimize the thresholds by


A
Input: Gene expression matrices
1. QC and cell filtering
2. Individual sample clustering (Seurat-SCT)
3. Optional: Label transfer auto cell annotation with user-provided reference
4. Integration, batch correction, and clustering (RISC)
Replicate-aware "Pseudobulk" analysis
Non-replicate analysis
5. Differential abundance analysis (Propeller)
5. Differential abundance analysis (2-prop Z-test)
6. Differential expression analysis (EdgeR or DESeq2)
6. Differential expression analysis (FindMarkers)
7. Pathway and transcription factor target analysis (FGSEA / ClusterProfiler)
B
1. Non-normalized UMI counts matrices
Cellranger Filtered .h5
or
Seurat Objects
2. Sample information: "sample_metadata.csv"
3. Comparison information: "comps.csv"
+ LogFC
- LogFC
4. Optional: Label transfer references
SCT-normalized Seurat Objects
C
Harmonizing input gene expression matrices with sample metadata
Example Sample Metadata
Example Comps
Input option 1: Cellranger Filtered .h5
Input option 2: Seurat Objects
D

Sample	Condition	Code
Patient1	Control	Control_1
Patient2	Control	Control_2
Patient3	DrugA	DrugA_1
Patient4	DrugA	DrugA_2
Patient5	DrugB	DrugB_1
Patient6	DrugB	DrugB_2

c1	c2
DrugA	Control
DrugB	Control
DrugA	DrugB

Sample	Condition	Code
Patient1	Control	Control_1
Patient2	Drug	Drug_1

c1	c2
Drug	Control

```
datadir/
├── Patient1
│   └── outs
│       └── filtered_feature_bc_matrix.h5
└── Patient2
    └── outs
        └── filtered_feature_bc_matrix.h5
```

```
datadir/
├── Patient1.rds
└── Patient2.rds
```

```
outdir
├── individualsample_analysis
│   ├── individualsample_clustermarkers
│   ├── individualsample_plots
│   ├── labeltransfer
│   ├── processed_Seurat_objects
│   ├── qualitycontrol_filtering
│   └── unfiltered_Seurat_objects
│
├── multisample_integration
│   ├── compositional_proportion_analysis
│   ├── data_objects
│   ├── differentialexpression_crosscondition
│   ├── integration_summaryplots
│   ├── labeltransfer
│   ├── markergenes_intclusters
│   └── pathwayanalysis_crosscondition
│
├── pipeline_parameters.csv
└── scRNAseq_clustering_integration.html
```

Figure 1. Overview of steps, inputs and outputs of scDAPP. (**A**) Overview of scDAPP steps and critical options. (**B**) Inputs of scDAPP. (**C**) Example illustrating configuration of inputs, matching gene expression data with metadata. (**D**) Tree structure of scDAPP output files.

analyzing the distribution of the underlying data. The distributions and the corresponding data-driven cutoffs are presented to the users graphically so that they may select thresholds deemed more appropriate. Essentially, the algorithms for deriving these data-driven cutoffs try to emulate the common manual QC, such as visual inspection of violin plots of these variables. First, a 'complexity' filter is applied, removing cells with a lower-than-expected number of unique genes given the number of observed UMIs. This is determined by two regression models, linear regression and LOESS regression with the log(nGenes) as the dependent variable and log(nUMI) as the predictor. Cells are considered low-complexity outliers if their linear regression Cook's distance is greater than 4 / number of cells, and their LOESS regression scaled residual value is less than −5, by default. The residual threshold is passed as a user parameter with higher values increasing the strictness of the filter. Next, scDAPP will learn data-driven cutoffs for the number of UMIs and the percent of mitochondrial reads (%mt). For these, robust statistics methods are applied, where cells with median absolute deviations above +2.5 (by default, tunable) for %mt and below −2.5 (default) for number of UMIs are considered low-quality outliers and removed. Again, users have the option to ignore and overwrite these data-driven cutoffs entirely.

After poor quality cells are removed, scDAPP optionally uses DoubletFinder to predict doublets for removal from further analysis (19). DoubletFinder hyperparameters such as the homotypic doublet rate are automatically estimated for each sample using the number of cells and the empirical multiplet rate provided by 10X Genomics (20).

Individual sample clustering analysis

After cell filtering, individual sample analysis is performed with Seurat using the modified SingleCellTransform (SCT) workflow (14,21). For each sample, scDAPP applies SCT, principal component analysis (PCA), graph construction, Louvain clustering, and Uniform Manifold Approximation and Projection (UMAP) for visualization. The hyperparameters including number of PCs (30 by default) and Louvain resolution (default 0.5) can be specified by the user, and iterated if needed, after examining the result from default settings.

Automated reference-based cell annotation via label transfer

While cluster markers are always computed, cluster annotation by label transfer using Seurat is an option, as described above. Two extensions are made in scDAPP to help users. (i) We apply a hard cutoff of label transfer score of 0.3, below which cells are considered non-classified, and a soft threshold of 0.5, below which cells are considered only putatively classified. (ii) Seurat label transfer gives a score and label for each cell, but we extend this to the cluster level by setting the transferred annotation to the most common predicted cell type for each cluster. To show the quality of label transfer, scDAPP generates cluster-level violin plots for the label transfer scores and provides heatmap visualization of the expression of the reference marker genes. Automated cell calling tools like label transfer are useful for providing a suggested cell annotation, but the results should be carefully examined and thus scDAPP makes all the relevant plots and scores available to the users.

Sample integration and batch correction

Next, scDAPP uses the RISC workflow for integration and batch correction, starting from the raw data matrices before Seurat analysis. RISC is also used for Louvain clustering and UMAP visualization of the integration data. There are scDAPP-specific extensions of the RISC package. By default, RISC uses the intersect of genes from the single cell objects as integration features, but this can leave out genes detected in only one sample. This can be problematic if one cell type (or corresponding marker genes) is absent in one sample or one of the comparison groups, potentially leading to missing marker genes for that cell type after integration. To overcome this, scDAPP reads the raw count data directly to RISC, selects cells filtered in the individual sample analysis, concatenates the cell-filtered matrices, and then performs gene filtering, such that more genes are retained. Several other key parameters are allowed for RISC integration analysis, including the number of PCs, the number of neighbors during graph construction, and clustering resolution. The default for these in scDAPP are reasonable: PCs = 30, resolution = 0.5, neighbors = 10, but they may be modified by users.

Another important extension of the RISC package in scDAPP is automated selection of a reference sample for multiple-sample integration. Currently, this is done manually. RISC users examine a panel of plots generated by the 'InPlot' function of RISC, which describe the number of clusters, variance per PC, and a measure of distributional divergence for each sample (8), and then select a sample for integration reference. We decided to automate this process in scDAPP by calculating a heuristic reference score for each sample. This score is based on the number of clusters, the cluster diversity, and the number of cells in each sample, and generally the sample with the greatest number of clusters, greatest diversity of clusters and highest number of cells gets the highest score and is chosen as the best reference.

For each sample i, let n_i denote the number of clusters, and c_i denote a weighted cell number value, defined as:

$$c_i = \frac{number\ of\ cells\ in\ sample\ i}{number\ of\ cells\ in\ the\ sample\ with\ the\ most\ cells}.$$

Next, let v_{ij} denote the variance of cluster j in sample i. We compute a cluster-average variance score s_i for sample i by calculating the mean value of the variances of each cluster:

$$s_i = \frac{1}{n_i}\sum_{j=1}^{n_i} v_{ij}$$

Then, a reference score S_i for sample i is computed as:

$$S_i = n_i \times c_i \times s_i$$

Finally, the sample i^* with the maximum S score is chosen as the reference: $i^* = \arg\max_i S_i$.

This is meant to automate the RISC recommendation that the sample with the most cell types is the preferred integration reference. As with the setting of all other parameters in scDAPP, the pipeline shows all the RISC InPlot graphs so that users can examine the underlying data and manually specify the reference sample via the 'risc_reference' parameter in the run configuration file. Very importantly, scDAPP generates alluvial plots to illustrate cluster relationships between individual samples and integrated data, thus helping users to spot over- or under-integration.

Comparison of cell composition across groups

After RISC integration, scDAPP performs cross-group comparison. First, differential cell composition analysis is applied to study cell population changes between groups by comparing the proportions of cells in each cluster across samples. As mentioned above, this can be done in a replicate-aware manner using the 'Propeller' test from the 'Speckle' package, which is a *t*-test of the proportions for each cluster with samples in each group as replicates. Notably, scDAPP applies the square-root arcsine transformation that is provided as an option by Propeller, as this transformation was shown to perform best in a benchmarking analysis (3). If replicates are not available or not considered, scDAPP will utilize the two proportion Z-test as implemented in the R prop.test function.

Differential expression analysis across groups

As stated above, this can be done via a replicate-aware pseudobulk based method by setting 'Pseudobulk_mode' = 'TRUE'. With this option, by default, the EdgeR likelihood ratio test (EdgeR-LRT) method is used, as this slightly outperformed default EdgeR and other pseudobulk methods like DESeq2 in a recent benchmark (2). Optionally, users may use the 'DE_test' parameter to select either the EdgeR, EdgeR-LRT, DESeq2 or DESeq2-LRT test. The data used for these tests are raw counts (i.e. UMIs) aggregated over cells in each cluster per replicate for individual genes. The statistical outputs are combined with other important single-cell level information including the percent of cells expressing a gene in each condition, allowing downstream prioritization or further filtering of the differentially expressed genes (DEGs). If replicates are not available or not considered, differential expression can be run with 'Pseudobulk_mode' = 'FALSE' and performed using the Wilcoxon Rank-Sum test as implemented in the Seurat 'FindMarkers' function by default, but importantly using the RISC batch-corrected gene expression values. Here too, users may optionally change to other Seurat statistical test (e.g. 'MAST') by passing arguments to the 'DE_test' parameter.

Pathway enrichment analysis

Finally, scDAPP performs multiple types of function enrichment analysis using the differential gene expression results. This includes GSEA as implemented in the 'fgsea' R package (22) and overrepresentation analysis (ORA). Notably, GSEA allows pre-ranking of all genes based on differential expression summary statistics. This obviates the need for arbitrary differential expression thresholds and works well with pseudobulk-based methods. Currently, scDAPP uses the gene sets from the MSIGDB pathway database, including the Hallmarks pathways, Gene Ontology (GO), KEGG, Reactome, and two transcription factor target databases, the Gene Transcription Regulation Database (GTRD) and Xie *et al.* Nature 2005 database (23–27). We draw on these databases using the 'msigdbr' R package, which also allows flexibility across a range of select species with careful multi-source homology support for gene orthologs (18). For the ORA option that is invoked by the 'run_ORA', scDAPP tests for significant overlaps between the DEGs in each cluster and pathway databases via the hypergeometric test as implemented in the ClusterProfiler package (28).

Outputs of the scDAPP pipeline

Once completed, scDAPP will save a variety of critical outputs from each stage of the pipeline, including a detailed HTML file (derived from R Markdown; containing extensive visualizations and tables for QC, intermediate results, and final results), text-format results, intermediate data files, and R objects (Figure 1D). The HTML report file (Supplementary File 1 for an example) summarizes all steps of the pipeline and all processing and results, including quality control information, data behind threshold selections, UMAPs for clustering results, plots for marker genes, and cross-group analysis. Relevant codes are embedded and can be viewed in this report file. We use UMAP for visualization but do not recommend it for direct inference of the degree of similarity between clusters.

Key result data tables are saved as .csv files, including results for marker genes, differential expression, differential abundance, and pathway analysis. High-resolution plots are also stored as .pdf files for each step. All these are included to make it transparent for users to track and understand the nuance and decisions in each step of the scRNA-seq analysis. The data is also valuable for users to adjust the parameters to rerun scDAPP until they are satisfied. In this sense, scDAPP is an excellent education tool for learning scRNA-seq analysis. Additionally, critical data objects are exported and saved, including clustered Seurat objects for each individual sample, a RISC object for the integrated data, and a Seurat object converted from the integrated data in the RISC format, and pseudobulking data. Users with advanced bioinformatics expertise can use them to seamlessly conduct further analyses with their own established workflows. Related to this, scDAPP has some utility R scripts for downstream analysis, such as a wrapper function for the recently described aPEAR algorithm for clustering enriched functions from GSEA (29). Another downstream application includes preparation of interactive web applications. For this purpose, we provide a short vignette linking the output of scDAPP with the ShinyCell package, which is a user-friendly tool to easily export scRNA-seq objects to a web-based Shiny app (30).

Results

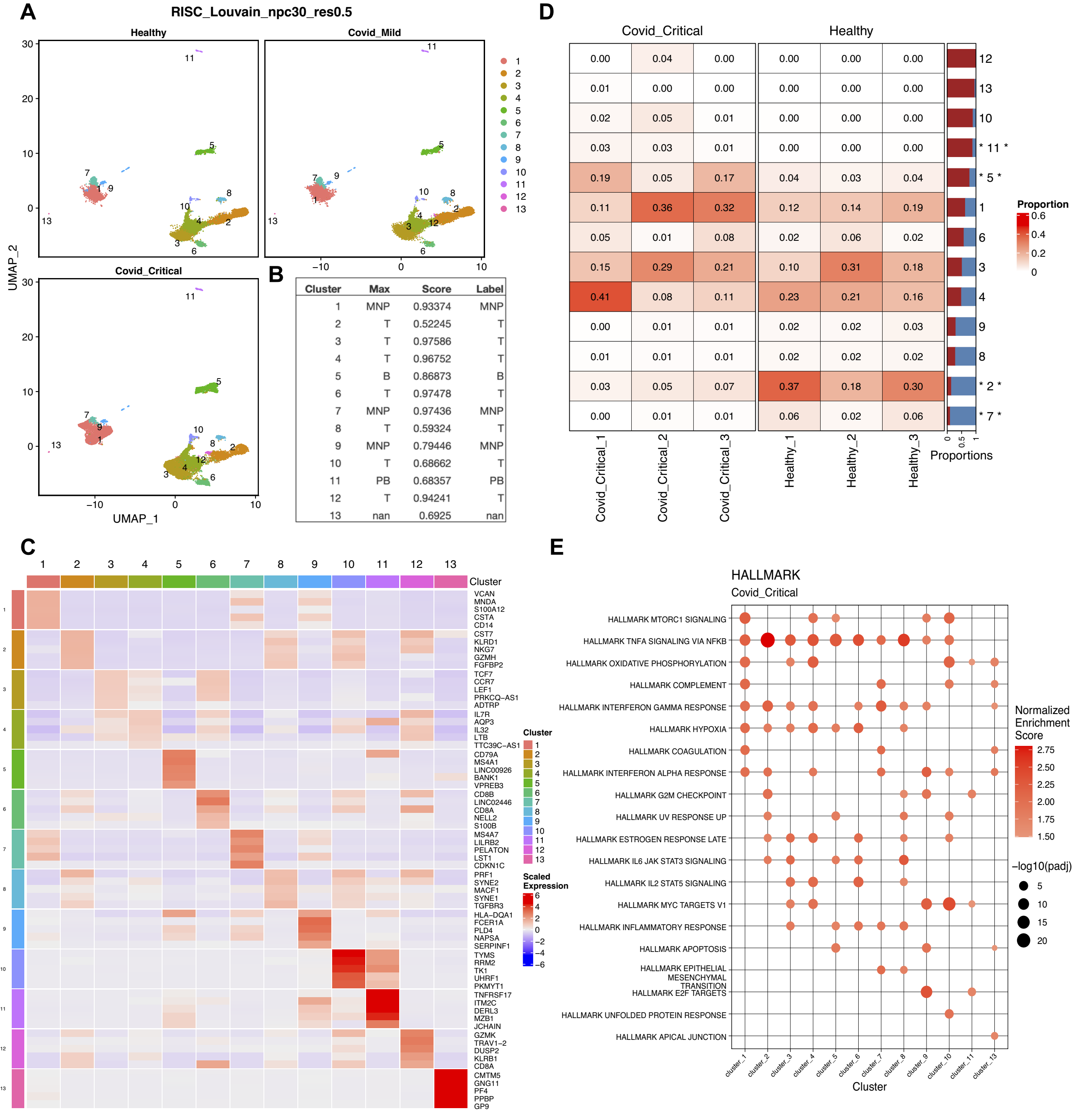
We have applied scDAPP successfully to many scRNA-seq and snRNA-seq data in our own studies. To demonstrate its performance, we included two instances below.

Reanalysis of COVID-19 Blood Atlas dataset with scDAPP revealed high concordance with published findings

We applied scDAPP to an atlas dataset of human blood cells from COVID-19 patients and healthy controls (31). We selected three patient samples each from mild COVID, critical COVID, and controls and ran scDAPP in pseudobulk mode to account for replicates. The input contained nine samples (3 × 3) and a total of 83,356 cells. We ran scDAPP on a SLURM-based high performance cluster (HPC) with an allocation of 100GB memory and 9 CPUs, and completed in ∼14 h (9 × 14 = 126 CPU hours). The main and full HTML report is in a supplemental file (Supplementary File 1), with some key results included in Figure 2 to illustrate scDAPP functions, specifically UMAP of the clustering result, expression heatmap of the top cluster markers, cell population change, and altered pathways from GSEA. The run also used the label transfer option, taking an independent healthy control sample as the reference for cell annotation (provided by the original authors). Overall, our results are in strong concordance with the original publication, for example, increased abundance of plasmablast (PB) and B cells and upregulation of interferon responses in many cell types in critical COVID samples (Figure 2B).

Reanalysis of mouse heart developmental scRNA-seq data with scDAPP recapitulated published results

We next applied scDAPP to a scRNA-seq dataset from mouse neural crest cells (NCCs), collected for studying the congenital heart defects in 22q11.2 deletion syndrome (32). The data were from heart tissues at embryonic day (E)10.5 of control embryos or embryos with *Tbx1* knockout (Wnt1-Cre;Tbx1$^{-/-}$), a gene in the 22q11.2 region and required for cardiac development (33–35). The input contained four E10.5 samples (2 *Tbx1* wild-type and 2 *Tbx1* knockout) and a total of 39,401 cells (Figure 3). We ran scDAPP with standard parameters except for the integrated clustering resolution ('res_int'; set to 1) in order to obtain clusters closely matching the previous publication (32), on a SLURM-based HPC with 50 GB memory allocation and 4 CPUs for ∼12 h. The RPCI reference sample automatically chosen by scDAPP was 'Control2', the same one selected manually in the previous paper (Figure 3). Taken the authors' cell type annotation and markers from an E9.5 sample in the same study for label trans-

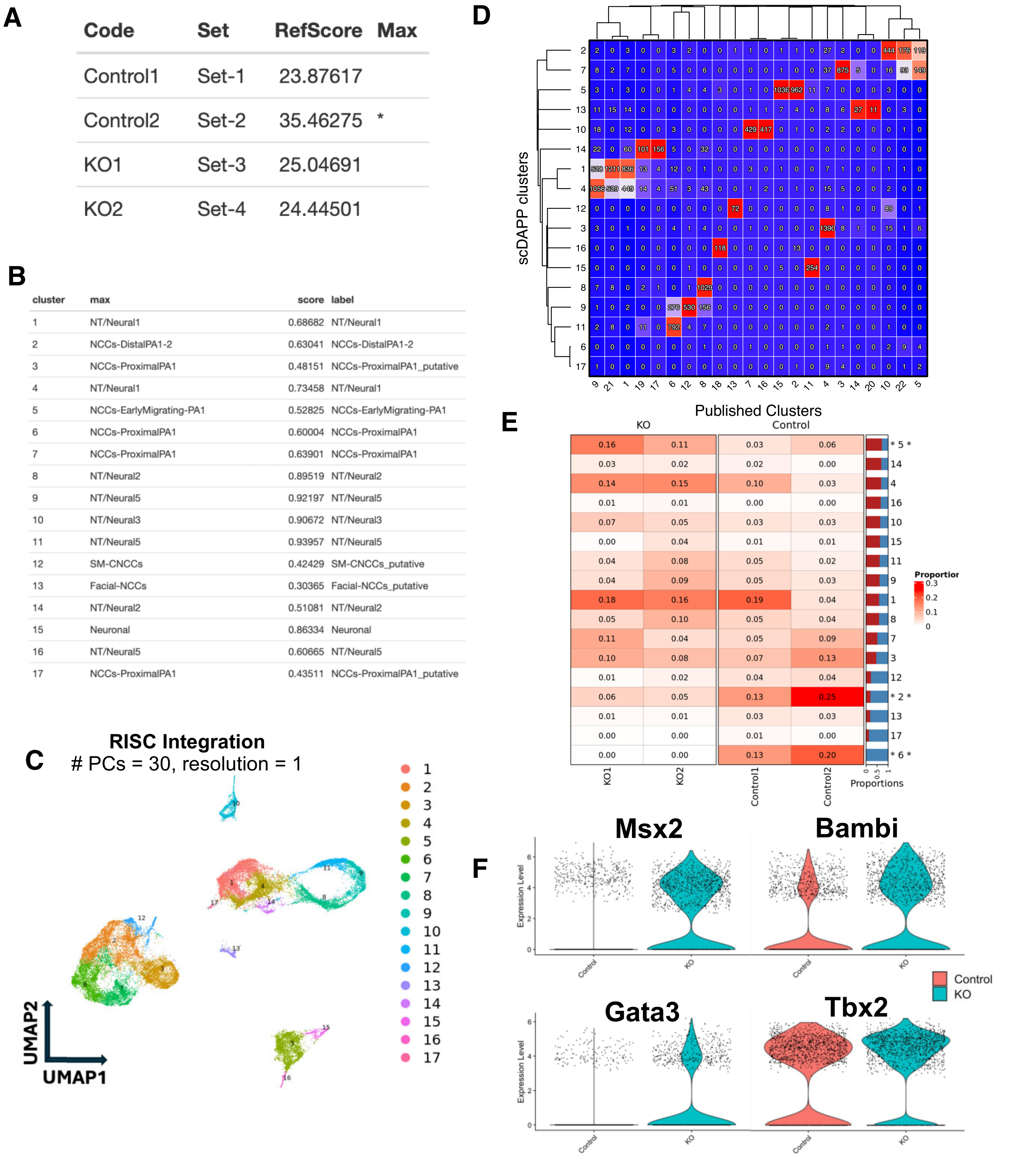


Cluster	Max	Score	Label
1	MNP	0.93374	MNP
2	T	0.52245	T
3	T	0.97586	T
4	T	0.96752	T
5	B	0.86873	B
6	T	0.97478	T
7	MNP	0.97436	MNP
8	T	0.59324	T
9	MNP	0.79446	MNP
10	T	0.68662	T
11	PB	0.68357	PB
12	T	0.94241	T
13	nan	0.6925	nan

	Covid_Critical_1	Covid_Critical_2	Covid_Critical_3	Healthy_1	Healthy_2	Healthy_3
12	0.00	0.04	0.00	0.00	0.00	0.00
13	0.01	0.00	0.00	0.00	0.00	0.00
10	0.02	0.05	0.01	0.00	0.00	0.00
* 11 *	0.03	0.03	0.01	0.00	0.00	0.00
* 5 *	0.19	0.05	0.17	0.04	0.03	0.04
1	0.11	0.36	0.32	0.12	0.14	0.19
6	0.05	0.01	0.08	0.02	0.06	0.02
3	0.15	0.29	0.21	0.10	0.31	0.18
4	0.41	0.08	0.11	0.23	0.21	0.16
9	0.00	0.01	0.01	0.02	0.02	0.03
8	0.01	0.01	0.01	0.02	0.02	0.02
* 2 *	0.03	0.05	0.07	0.37	0.18	0.30
* 7 *	0.00	0.01	0.01	0.06	0.02	0.06

Figure 2. Reanalysis of blood scRNA-seq data from COVID-19 patients and controls with scDAPP. (**A**) Integrated UMAPs colored by clusters and separated by patient groups. (**B**) Table showing the label transfer result using a healthy non-COVID-19 sample from the same study (not included in the clustering and other re-analysis). MNP = 'Mononuclear phagocytes' (monocytes / macrophages), PB = 'plasmablast', 'nan' indicates un-annotated cell types from the original published annotations. (**C**) Heatmap showing the expression of top markers computed for the clusters in (A). (**D**) Heatmap table showing the cell composition change between critical COVID-19 patient and healthy control samples. * $P < 0.05$. (**E**) Differential pathway analysis showing gene sets upregulated in critical COVID-19 patients versus healthy samples. All these figures were taken from the scDAPP output directly (Supplementary File 1).

fer, scDAPP was able to accurately assign cell types (Figure 3B, C). A close examination of the cluster relationship revealed differences, but most cells were clustered similarly by types in scDAPP and previous report (Figure 3D), suggesting that computational choice and software version could make subtle difference. The cell composition change between the controls and *Tbx1* null embryos was also reproduced (difference in the craniofacial and outflow tract NCCs, corresponding to the scDAPP clusters 2 and 6) (Figure 3E). Finally, we compared the DEGs in the cardiac progenitor NCCs and found a large agreement between scDAPP and previous results, e.g. upregulation of *Msx2*, *Bambi*, *Gata3*, and *Tbx2* in the *Tbx1*$^{-/-}$ embryos (Figure 3F), all of which are critical downstream targets of *Tbx1* (32).



A

Code	Set	RefScore	Max
Control1	Set-1	23.87617	
Control2	Set-2	35.46275	*
KO1	Set-3	25.04691	
KO2	Set-4	24.44501	

B

cluster	max	score	label
1	NT/Neural1	0.68682	NT/Neural1
2	NCCs-DistalPA1-2	0.63041	NCCs-DistalPA1-2
3	NCCs-ProximalPA1	0.48151	NCCs-ProximalPA1_putative
4	NT/Neural1	0.73458	NT/Neural1
5	NCCs-EarlyMigrating-PA1	0.52825	NCCs-EarlyMigrating-PA1
6	NCCs-ProximalPA1	0.60004	NCCs-ProximalPA1
7	NCCs-ProximalPA1	0.63901	NCCs-ProximalPA1
8	NT/Neural2	0.89519	NT/Neural2
9	NT/Neural5	0.92197	NT/Neural5
10	NT/Neural3	0.90672	NT/Neural3
11	NT/Neural5	0.93957	NT/Neural5
12	SM-CNCCs	0.42429	SM-CNCCs_putative
13	Facial-NCCs	0.30365	Facial-NCCs_putative
14	NT/Neural2	0.51081	NT/Neural2
15	Neuronal	0.86334	Neuronal
16	NT/Neural5	0.60665	NT/Neural5
17	NCCs-ProximalPA1	0.43511	NCCs-ProximalPA1_putative

D

scDAPP clusters

	9	21	1	19	17	6	12	8	18	13	7	16	15	2	11	4	3	14	20	10	22	5
2	2	0	3	0	0	3	2	0	0	1	1	0	1	1	0	27	2	0	0	444	176	119
7	8	2	7	0	0	5	0	6	0	0	0	0	1	0	0	37	875	5	0	16	93	149
5	3	1	3	0	0	1	4	4	3	0	1	0	1036	962	11	7	0	0	0	3	0	0
13	11	15	14	0	0	0	0	0	0	0	1	1	7	4	0	8	6	27	11	0	3	0
10	18	0	12	0	0	3	0	0	0	0	429	417	0	1	0	2	2	0	0	0	1	0
14	22	0	60	101	156	5	0	32	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	528	1211	836	13	4	12	0	1	0	0	3	0	1	0	0	7	1	0	0	1	1	0
4	1056	529	449	14	4	51	3	43	0	0	1	2	0	1	0	15	5	0	0	2	0	0
12	0	0	0	0	0	0	0	0	0	72	0	0	0	0	0	8	1	0	0	89	0	1
3	3	0	1	0	0	4	1	4	0	0	1	0	0	0	0	1390	8	1	0	15	1	6
16	0	0	0	0	0	0	0	0	118	0	0	0	0	13	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	1	0	0	0	0	0	5	0	254	0	0	0	0	0	0	0
8	7	8	0	2	1	0	1	1029	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	1	0	2	0	0	276	530	156	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	2	8	0	11	0	792	4	7	0	0	0	0	0	0	0	2	1	0	0	1	0	0
6	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	1	1	0	0	2	9	4
17	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	9	4	0	0	0	1	2

Published Clusters

E

KO		Control		
KO1	KO2	Control1	Control2	
0.16	0.11	0.03	0.06	* 5 *
0.03	0.02	0.02	0.00	14
0.14	0.15	0.10	0.03	4
0.01	0.01	0.00	0.00	16
0.07	0.05	0.03	0.03	10
0.00	0.04	0.01	0.01	15
0.04	0.08	0.05	0.02	11
0.04	0.09	0.05	0.03	9
0.18	0.16	0.19	0.04	1
0.05	0.10	0.05	0.04	8
0.11	0.04	0.05	0.09	7
0.10	0.08	0.07	0.13	3
0.01	0.02	0.04	0.04	12
0.06	0.05	0.13	0.25	* 2 *
0.01	0.01	0.03	0.03	13
0.00	0.00	0.01	0.00	17
0.00	0.00	0.13	0.20	* 6 *

Figure 3. Reanalysis of E10.5 Neural Crest Cell scRNA-seq data with scDAPP. (**A**) Table showing the scDAPP-computed RISC reference scores for the four samples, indicating that the sample 'Control2' had the highest score and was selected as the reference for RISC integration, consistent with what was selected manually in the original analysis. (**B**) Table showing the cluster annotation from label transfer using an independent E9.5 NCC dataset, yielding the same cluster annotation as in the original publication. (**C**) UMAP plot of the integrated samples colored by clusters. (**D**) Table showing the relationship between clusters in published versus scDAPP re-analysis. Note that clusters 6 and 17 were not included in the original study, indicating different thresholds for cell filtering. (**E**). Table for cell compositional analysis showing significant decreases in cell number with *Tbx1* KO in the clusters 2 and 6, consistent with the original report. * $P < 0.05$. (**F**) Violin plots showing the differential expression of four key genes highlighted in the paper for cardiac progenitor neural crest cells (cluster 3 in scDAPP). Panels A, B, C and E were taken directly from the scDAPP output.

Discussion

Whether scDAPP is used for a quick end-to-end bioinformatics analysis of the data or recurrent processing of the same data with different options, a main goal of its development is to help users visualize the granularity of sc/nRNA-seq analysis and achieve a quick transition from raw data to biological interpretation, while learning and exploring analytic parameters and options along the way. Based on our experience, we believe that the visualizations in the HTML output are extremely valuable for users, to examine data distribution, cluster marker specificity, cluster relationship, cell population shifts, biological relevant pathways and so on (Supplementary File 1).

Importantly, scDAPP emphasizes on full utilization of replicates for both differential gene expression analysis and differential cell composition analysis. The usage of replicates has repeatedly been demonstrated as a critical factor for specificity in single-cell transcriptomic comparative analysis. To our knowledge, scDAPP is the first such end-to-end pipeline to explicitly feature this replicate-aware approach, but we should note that there are other computational methods that use replicates but not pseudobulking.

As scDAPP is a pipeline, we have not systematically benchmarked all the software by ourselves and rather have taken the recommendations from the community. The modular design, however, provides sufficient flexibility for including additional software in the future. For example, modified Dirichlet models were also shown to perform very well for differential cluster abundance analysis (3), and thus may be included to complement Propeller. Additionally, new methods based on combinatorial indexing are now capable of producing datasets with hundreds of thousands to millions of cells per sample (36). Such technological advancements represent a paradigm shift for the single-cell field in general and may require implementation of specialized tools relying on highly optimized memory storage or combining cells together into metacells, important functions to consider for future scDAPP releases.

Data availability

No new experimental data were collected in this study. Source codes, documentation and sample data of scDAPP are at the GitHub (https://github.com/bioinfoDZ/scDAPP), while a stable release was deposited at the Figshare (https://doi.org/10.6084/m9.figshare.27048388.v1).

Supplementary data

Supplementary Data are available at NARGAB Online.

Acknowledgements

The authors would like to thank the members in the Zheng's lab for testing the software and providing valuable suggestions.

Funding

NIH grants [R01CA255643, R01HL153920, R01HL163667].

Conflict of interest statement

None declared.

References

1. Vandereyken,K., Sifrim,A., Thienpont,B. and Voet,T. (2023) Methods and applications for single-cell and spatial multi-omics. *Nat. Rev. Genet.*, **24**, 494–515.
2. Squair,J.W., Gautier,M., Kathe,C., Anderson,M.A., James,N.D., Hutson,T.H., Hudelle,R., Qaiser,T., Matson,K.J.E., Barraud,Q., *et al.* (2021) Confronting false discoveries in single-cell differential expression. *Nat. Commun.*, **12**, 5692.
3. Simmons,S. (2022) Cell Type Composition Analysis: comparison of statistical methods. bioRxiv doi: https://doi.org/10.1101/2022.02.04.479123, 08 February 2022, preprint: not peer reviewed.
4. Vieth,B., Parekh,S., Ziegenhain,C., Enard,W. and Hellmann,I. (2019) A systematic evaluation of single cell RNA-seq analysis pipelines. *Nat. Commun.*, **10**, 4667.
5. Heumos,L., Schaar,A.C., Lance,C., Litinetskaya,A., Drost,F., Zappia,L., Lücken,M.D., Strobl,D.C., Henao,J., Curion,F., *et al.* (2023) Best practices for single-cell analysis across modalities. *Nat. Rev. Genet.*, **24**, 550–572.
6. Stoeckius,M., Zheng,S., Houck-Loomis,B., Hao,S., Yeung,B.Z., Mauck,W.M., Smibert,P. and Satija,R. (2018) Cell Hashing with barcoded antibodies enables multiplexing and doublet detection for single cell genomics. *Genome Biol.*, **19**, 224.
7. McGinnis,C.S., Patterson,D.M., Winkler,J., Conrad,D.N., Hein,M.Y., Srivastava,V., Hu,J.L., Murrow,L.M., Weissman,J.S., Werb,Z., *et al.* (2019) MULTI-seq: sample multiplexing for single-cell RNA sequencing using lipid-tagged indices. *Nat. Methods*, **16**, 619–626.
8. Liu,Y., Wang,T., Zhou,B. and Zheng,D. (2021) Robust integration of multiple single-cell RNA sequencing datasets using a single reference space. *Nat. Biotechnol.*, **39**, 877–884.
9. Umu,S.U., Rapp Vander-Elst,K., Karlsen,V.T., Chouliara,M., Bækkevold,E.S., Jahnsen,F.L. and Domanska,D. (2022) Cellsnake: a user-friendly tool for single-cell RNA sequencing analysis. *GigaScience*, **12**, giad091.
10. Harvard Medical School DBMI - Cellenics. https://github.com/hms-dbmi-cellenics, (16 March 2024, date last accessed).
11. Tekman,M., Batut,B., Ostrovsky,A., Antoniewski,C., Clements,D., Ramirez,F., Etherington,G.J., Hotz,H.R., Scholtalbers,J., Manning,J.R., *et al.* (2020) A single-cell RNA-sequencing training and analysis suite using the Galaxy framework. *GigaScience*, **9**, giaa102.
12. Kubovčiak,J., Kolář,M. and Novotný,J. (2023) Scdrake: a reproducible and scalable pipeline for scRNA-seq data analysis. *Bioinform. Adv.*, **3**, vbad089.
13. Prieto,C., Barrios,D. and Villaverde,A. (2022) SingleCAnalyzer: interactive Analysis of Single Cell RNA-Seq Data on the Cloud. *Front. Bioinform.*, **2**, 793309.
14. Butler,A., Hoffman,P., Smibert,P., Papalexi,E. and Satija,R. (2018) Integrating single-cell transcriptomic data across different conditions, technologies, and species. *Nat. Biotechnol.*, **36**, 411–420.
15. Phipson,B., Sim,C.B., Porrello,E.R., Hewitt,A.W., Powell,J. and Oshlack,A. (2022) propeller: testing for differences in cell type proportions in single cell data. *Bioinformatics*, **38**, 4720–4726.
16. Robinson,M.D., McCarthy,D.J. and Smyth,G.K. (2010) edgeR: a Bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics*, **26**, 139.
17. Korotkevich,G., Sukhov,V., Budin,N., Shpak,B., Artyomov,M.N. and Sergushichev,A. (2021): Fast gene set enrichment analysis. bioRxiv doi: https://doi.org/10.1101/060012, 01 February 2021, preprint: not peer reviewed.

18. Dolgalev,I. (2018) msigdbr: mSigDB gene sets for multiple organisms in a tidy data format. R package version 6.2.1. https://CRAN.R-project.org/package=msigdbr (April 2024, date last accessed).
19. McGinnis,C.S., Murrow,L.M. and Gartner,Z.J. (2019) DoubletFinder: doublet detection in single-cell RNA sequencing data using artificial nearest neighbors. *Cell Syst.*, **24**, 329–337.
20. What is the maximum number of cells that can be profiled? –10X Genomics. https://kb.10xgenomics.com/hc/en-us/articles/360001378811-What-is-the-maximum-number-of-cells-that-can-be-profiled, (15 March 2024, date last accessed).
21. Hafemeister,C. and Satija,R. (2019) Normalization and variance stabilization of single-cell RNA-seq data using regularized negative binomial regression. *Genome Biol.*, **20**, 296.
22. Subramanian,A., Tamayo,P., Mootha,V.K., Mukherjee,S., Ebert,B.L., Gillette,M.A., Paulovich,A., Pomeroy,S.L., Golub,T.R., Lander,E.S., *et al.* (2005) Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. *Proc. Natl. Acad. Sci. U.S.A.*, **102**, 15545–15550.
23. Liberzon,A., Birger,C., Thorvaldsdóttir,H., Ghandi,M., Mesirov,J.P. and Tamayo,P. (2015) The molecular signatures database hallmark gene set collection. *Cell Syst.*, **1**, 417–425.
24. Kanehisa,M. and Goto,S.K.E.G.G.: (2000) Kyoto Encyclopedia of Genes and Genomes. *Nucleic Acids Res.*, **28**, 27–30.
25. Ashburner,M., Ball,C.A., Blake,J.A., Botstein,D., Butler,H., Cherry,J.M., Davis,A.P., Dolinski,K., Dwight,S.S., Eppig,J.T., *et al.* (2000) Gene ontology: tool for the unification of biology. *Nat. Genet.*, **25**, 25–29.
26. Kolmykov,S., Yevshin,I., Kulyashov,M., Sharipov,R., Kondrakhin,Y., Makeev,V.J., Kulakovskiy,I.V., Kel,A. and Kolpakov,F. (2021) GTRD: an integrated view of transcription regulation. *Nucleic Acids Res.*, **49**, D104–D111.
27. Xie,X., Lu,J., Kulbokas,E.J., Golub,T.R., Mootha,V., Lindblad-Toh,K., Lander,E.S. and Kellis,M. (2005) Systematic discovery of regulatory motifs in human promoters and 3′ UTRs by comparison of several mammals. *Nature*, **434**, 338–345.
28. Wu,T., Hu,E., Xu,S., Chen,M., Guo,P., Dai,Z., Feng,T., Zhou,L., Tang,W., Zhan,L., *et al.* (2021) clusterProfiler 4.0: a universal enrichment tool for interpreting omics data. *Innovation (Cambridge (Mass))*, **2**, 100141.
29. Kerseviciute,I. and Gordevicius,J. (2023) aPEAR: an R package for autonomous visualisation of pathway enrichment networks. *Bioinformatics*, **39**, btad672.
30. Ouyang,J.F., Kamaraj,U.S., Cao,E.Y. and Rackham,O.J.L. (2021) ShinyCell: simple and sharable visualization of single-cell gene expression data. *Bioinformatics*, **37**, 3374–3376.
31. Ahern,D.J., Ai,Z., Ainsworth,M., Allan,C., Allcock,A., Angus,B., Ansari,M.A., Arancibia-Cárcamo,C.V., Aschenbrenner,D., Attar,M., *et al.* (2022) A blood atlas of COVID-19 defines hallmarks of disease severity and specificity. *Cell*, **185**, 916–938.
32. De Bono,C., Liu,Y., Ferrena,A., Valentine,A., Zheng,D. and Morrow,B.E. (2023) Single-cell transcriptomics uncovers a non-autonomous Tbx1-dependent genetic program controlling cardiac neural crest cell development. *Nat. Commun.*, **14**, 1551.
33. Merscher,S., Funke,B., Epstein,J.A., Heyer,J., Puech,A., Lu,M.M., Xavier,R.J., Demay,M.B., Russell,R.G., Factor,S., *et al.* (2001) TBX1 is responsible for cardiovascular defects in velo-cardio-facial/DiGeorge syndrome. *Cell*, **104**, 619–629.
34. Lindsay,E.A., Vitelli,F., Su,H., Morishima,M., Huynh,T., Pramparo,T., Jurecic,V., Ogunrinu,G., Sutherland,H.F., Scambler,P.J., *et al.* (2001) Tbx1 haploinsufficieny in the DiGeorge syndrome region causes aortic arch defects in mice. *Nature*, **410**, 97–101.
35. Jerome,L.A. and Papaioannou,V.E. (2001) DiGeorge syndrome phenotype in mice mutant for the T-box gene, Tbx1. *Nat. Genet.*, **27**, 286–291.
36. Martin,B.K., Qiu,C., Nichols,E., Phung,M., Green-Gladden,R., Srivatsan,S., Blecher-Gonen,R., Beliveau,B.J., Trapnell,C., Cao,J., *et al.* (2023) Optimized single-nucleus transcriptional profiling by combinatorial indexing. *Nat. Protoc.*, **18**, 188–207.

Received: April 12, 2024. **Revised:** July 7, 2024. **Editorial Decision:** September 10, 2024. **Accepted:** September 18, 2024