Chord
an ensemble machine learning algorithm to identify doublets in single-cell RNA sequencing data
Xiong, Ke-Xu; Zhou, Han-Lin; Lin, Cong; Yin, Jian-Hua; Kristiansen, Karsten; Yang, Huan-Ming; Li, Gui-Bo

Published in:
Communications Biology

DOI:
10.1038/s42003-022-03476-9

Publication date:
2022

Document version
Publisher's PDF, also known as Version of record

Document license:
CC BY

Citation for published version (APA):
Xiong, K-X., Zhou, H-L., Lin, C., Yin, J-H., Kristiansen, K., Yang, H-M., & Li, G-B. (2022). Chord: an ensemble machine learning algorithm to identify doublets in single-cell RNA sequencing data. Communications Biology, 5, [510]. https://doi.org/10.1038/s42003-022-03476-9
Chord: an ensemble machine learning algorithm to identify doublets in single-cell RNA sequencing data

Ke-Xu Xiong1,2,9, Han-Lin Zhou2,3,4,5,6,9,10✉, Cong Lin2,4,5,7, Jian-Hua Yin2,4,5,7, Karsten Kristiansen2,6, Huan-Ming Yang2,8 & Gui-Bo Li2,3,4,5,7,10✉

High-throughput single-cell RNA sequencing (scRNA-seq) is a popular method, but it is accompanied by doublet rate problems that disturb the downstream analysis. Several computational approaches have been developed to detect doublets. However, most of these methods may yield satisfactory performance in some datasets but lack stability in others; thus, it is difficult to regard a single method as the gold standard which can be applied to all types of scenarios. It is a difficult and time-consuming task for researchers to choose the most appropriate software. We here propose Chord which implements a machine learning algorithm that integrates multiple doublet detection methods to address these issues. Chord had higher accuracy and stability than the individual approaches on different datasets containing real and synthetic data. Moreover, Chord was designed with a modular architecture port, which has high flexibility and adaptability to the incorporation of any new tools. Chord is a general solution to the doublet detection problem.
Recently, the development of high-throughput single-cell RNA sequencing (scRNA-seq) has provided convenience for dissecting the cellular heterogeneity of tissues. In contrast to bulk RNA sequencing, profiling transcriptomes at the single-cell resolution has enabled researchers to recognize the molecular characteristics of all cell types at one time and acquire a better insight into physiology, biological development, and disease. Among the current state-of-the-art technologies of high-throughput scRNA-seq, droplet-based technologies are currently commonly employed as an unbiased solution of single-cell transcriptomics. However, these microfluidic methods often encounter the problem of doublets, where one droplet may contain two or more cells with the same barcode during the distribution step of isolating single cells. Then the doublets are counted as a single cell in the data forming technical artefacts. According to the composition of doublets, doublets can be divided into two major classes: homotypic doublets, which originate from the same cell type, and heterotypic doublets, which arise from distinct transcriptional cells generating an artificial hybrid transcriptome. Compared to homotypic doublets, heterotypic doublets are considered to have more impact on downstream analyses, including dimensionality reduction, cell clustering, differential expression, and cell developmental trajectories.

To reduce the number of doublets in experiments, decreasing the concentration of loaded cells is an effective control measure. However, this approach also reduces the number of captured cells and dramatically increases the cost per sample. Several existing experimental techniques can be applied to identify doublets instead of avoiding doublets, such as the cell hashing method using oligo-tagged antibodies as an orthogonal information, MULTI-seq using lipid-tagged indices, and demuxlet using natural genetic variations. However, there are inherent limitations to these experimental techniques. First, since these methods require special experimental operations and additional costs, so they are not helpful for the existing scRNA-seq data. Second, these techniques only experimentally label doublets from different samples but ignore the kind of doublet generated by cells from the same sample or individual. Therefore, several computational approaches have been developed to detect doublets in common scRNA-seq data, including data already generated. A benchmarking study has shown that the performance of these computational methods varies greatly, even the top-performing methods with the noticeable overkill could improve the accuracy of training sets, which is beneficial to model fitting. Therefore, the Chord first roughly estimates the doublets of the input droplet data according to the three built-in methods: DoubletFinder, bcds, and cxds to filter out the likely doublets from the original data before simulating artificial doublets. We called this step “overkill”. Selecting “overkill” could improve the accuracy of training sets, which is beneficial to model fitting. The ensemble algorithm was adopted to integrate these predicted scores, which served as the predictors in the GBM model. Then the doublets output was calculated by the GBM model for the input droplets data.

To determine whether the ensemble algorithm improves the performance of doublet detection, we first evaluated these methods on ground-truth scRNA-seq datasets that label doublets using the experimental strategies demuxlet and Cell Hashing. The regions of ground-truth doublets in UMAP show enrichment of the Chord’s doublets scores (Fig. 1b, c). The performance results of Chord in each dataset and the average across datasets were evaluated using receiver operating characteristic (ROC) curve analysis and precision-recall (PR) curve analysis (Table 1 and Supplementary Data 1). Chord achieved the highest areas under the ROC curves (AUCs) and the highest area under the PR curve (AUPRC) value on HTO8 dataset (0.815 and 0.599) and DM-A dataset (0.831 and 0.394) respectively (Supplementary Fig. 1b, Supplementary Table 3, Supplementary Table 4, Supplementary Table 5). When the real doublet rate was taken as cutoff, Chord detected 1596 doublets in the HTO8 dataset and 56 doublets in the DM-A dataset, which were higher than any individual built-in method (Supplementary Fig. 2c).

Furthermore, to thoroughly evaluate the performance of Chord, we performed Receiver Operating Characteristic (ROC) curve analysis and Precision-Recall (PR) curve analysis. The ensemble algorithm was adopted to integrate these predicted scores, which served as the predictors in the GBM model. Then the doublets output was calculated by the GBM model for the input droplets data. The Chord workflow is composed of three main steps (Fig. 1a).

(i) Generating training data after coarse removal of doublets using existing methods and generating artificial doublets from the filtered data.
(ii) Generalized Boosted Regression Modeling (GBM) fitting, which integrates and weights the prediction of existing doublet detection tools based on classification performance on the training data.
(iii) Application of the trained GMB model to the original dataset to predict doublets.
Fig. 1 Chord overview and its performance on the DM-A and HTO8 tests. 

(a) Schematic outline of the Chord workflow. First, preliminarily predicted doublets are filtered using bcds, cxds and DoubletFinder, and then the processed dataset is randomly sampled to generate simulation doublets that are added to the training dataset. The second step is to fit the weights of the integrated methods through the GBM algorithm on the training dataset. In the third step, the ensemble model is used to evaluate the original expression matrix and the doublets are identified by the expectation threshold value.

(b) UMAP was embedded for the DM-A and HTO8 tests with experimental doublet labels. The doublets are shown in red, and the singlets are shown in grey. The doublet prediction scores of Chord were visualised on the UMAP plots for the DM-A and HTO8 tests. The DM-A dataset was from human peripheral blood mononuclear cell (PBMC) samples using the experimental demuxlet method to annotate doublets\(^7\). The HTO8 dataset was from the samples of PBMC using eight barcoded antibodies to mark and label doublets\(^9\).

(c) The ROC curves of Chord were drawn for the DM-A and HTO8 tests using the R package PRROC\(^{29}\).

Table 1 Comprehensive performance of each method in real-world scRNA-seq datasets with experimentally annotated doublets.

| Method                  | PAUC800 | PAUC900 | PAUC950 | PAUC975 | AUC   | AUPRC |
|-------------------------|---------|---------|---------|---------|-------|-------|
| bcds                    | 0.598456581 | 0.697609168 | 0.747471204 | 0.772440146 | 0.797428571 | 0.465471429 |
| Chord                   | 0.60219241 | 0.701382997 | 0.751248502 | 0.77621856 | 0.80124286 | 0.464642857 |
| ChordP                  | 0.614164051 | 0.713609623 | 0.763485434 | 0.788453732 | 0.8132 | 0.466514286 |
| cxds                    | 0.576279854 | 0.675031498 | 0.724828357 | 0.749788808 | 0.774785714 | 0.367342857 |
| doubletCells            | 0.396983835 | 0.487445535 | 0.535271112 | 0.559821949 | 0.584685714 | 0.173985714 |
| DoubletDetection        | 0.569370526 | 0.666536657 | 0.715610186 | 0.740426505 | 0.79314286 | 0.500857143 |
| DoubletFinder           | 0.537830717 | 0.636466839 | 0.686221775 | 0.71172069 | 0.736171429 | 0.339428571 |
| Scrublet                | 0.564203075 | 0.663581473 | 0.713449225 | 0.738419459 | 0.763412828 | 0.399771429 |
| Solo                    | 0.604236942 | 0.703224153 | 0.752987364 | 0.777930394 | 0.803142857 | 0.434714286 |

The average performance of various methods in all datasets. The indexes are the pAUC800, pAUC900, pAUC950, pAUC975, AUC and AUPRC.
The performance of doublet detection approaches on ground-truth datasets. In addition to the abovementioned computational doublet detection approaches based on the R environment, some cutting-edge doublet detection software programs based on the Python platform have also been published in recent years. To integrate more doublet identification algorithms to improve the accuracy without losing the usability of Chord and the convenience of the R environment, Chord developed an expandable port allowing integration of more doublet identification algorithms (Supplementary Fig. 1a). This port can take the scoring results of other doublet detection software as input files, integrate these new methods with the individual built-in methods and obtain a training model to further improve the accuracy of doublet identification. Combinations of doublet detection methods were evaluated and the optimal combination (Chord, Scrublet4,6 and DoubletDetection17) was decided based on the mean AUC (Supplementary Fig. 3b, Supplementary Table 7). We used the Chord port to integrate the two Python software (Scrublet and DoubletDetection) (Supplementary Table 1) for an enhanced GBM algorithm model, called Chord Plus version (ChordP) (Supplementary Fig. 3b).

To compare the doublet detection performances of Chord, ChordP and the other seven stand-alone software programs, we chose the seven ground-truth scRNA-seq datasets (Supplementary Table 2) to evaluate their overall performance. Chord and ChordP achieved improved accuracy, and what’s more important was that it showed stability across datasets (Fig. 2). Compared with Chord, the AUC of ChordP increased from 0.831 to 0.833 on the DM-A dataset and from 0.815 to 0.835 on the HT08 dataset (Fig. 2a), and ChordP performed better on most datasets. Moreover, partial areas under the ROC curve (pAUC) at 80% (pAUC800), 90% (pAUC900), 95% (pAUC950) and 97.5% (pAUC975) specificity were calculated, the average AUC, pAUC800, pAUC900, pAUC950 and pAUC975 of ChordP across all the datasets were the highest among all methods (Fig. 2d, Table 1), and its average rank value in all datasets reached the highest (2.285, Fig. 2c, Supplementary Data 2). These results showed that ChordP can indeed obtain more accurate results after ensembling 5 methods. In addition, the ranking variance of ChordP was 1.254, which was lower than that of Solo6 (3.047) and bcds (1.773), both of which had the same high accuracy rate (Fig. 2c, Supplementary Data 2). This finding shows that ChordP has better versatility for different datasets than the other methods.

Through the uniform manifold approximation and projection (UMAP) method for visualising the true positive doublets (TP), true negative doublets (TN), false negative doublets (FN) and false positive doublets (FP) (Supplementary Fig. 2), the distributions of the doublets detected by each method were various at cluster level, which intuitively showed the complementarity between the different methods and the necessity of ensembling these methods. Among them, some methods, such as DoubletDetection and DoubletFinder, had a concentrated distribution of FP results in the HT08 dataset. The removal of doublets based on these scoring results may lead to the accidental deletion of such FP cell-enriched clusters, affecting cell type proportion statistics and directly leading to the loss of rare cell subpopulations. In contrast, the FP results of Chord and ChordP were relatively evenly distributed, avoiding becoming independent clusters and affecting subsequent analysis (Supplementary Fig. 2). Above all, the results showed that ChordP, which integrates more algorithms, outperforms Chord and the other methods.

We tested the time consumption of these different software programs under uniform hardware conditions and found that Chord did not significantly increase the time consumption. Cxds was extremely time-efficient, while Solo was the most time-consuming method in a CPU environment (Fig. 2e, Supplementary Data 3).
equivalent to or even outperformed other methods in DEG detection and pseudotime analysis on the synthetic scRNA-seq datasets.

**Applying Chord to real-world scRNA-Seq data.** To investigate the application of Chord in real-world data and whether the effect of downstream analysis has been improved, we tested the Chord method on a real-world scRNA-seq data dataset without doublets labelling information that containing 52,698 cells from lung cancer tumour tissues of 5 patients. Based on the expected doublet rate (0.9% per 1000 cells), we estimated the proportion of doublets in 5 malignant tumour samples (sample 11, 13, 17, 18, 22) with the most expected doublet rate (3.81%, 4.68%, 4.74%,
SciBet25, a cell type annotation tool based on Bayes decision, to
in the UMAP plots (Fig. 4b), clustered at the edge of some
10 had the highest doublets enrichment trend (Fig. 4b, c), in
to the number of predicted doublets in different clusters, cluster
clusters, and some even formed independent clusters. According
to real data. We utilised ROGUE24, an entropy-based metric, to assess
downstream analysis, we evaluate the performances of Chord on
statistics of the proportion of cell types.

By filtering doublets was apparent (Fig. 4e). Next, we used
filter. The accuracy of the annotation results of the B cells
filtered data, and
filtered data to generate a
training set. In this way, the number of undiscovered doublets in
simulated based on the remaining cells are used to generate a
training set may contain undiscovered doublets, which could
cluster 10 had the highest doublets enrichment trend (Fig. 4b, c), in
which markers of T cells and plasma cells are simultaneously
expressed (Fig. 4d). Cluster 10 is shown to be the closest neigh-
bou to both cluster 1 and cluster 8 on the UMAP plot (Fig. 4b).
Cluster 1 is T cell cluster, while cluster 8 is plasma cell cluster
which is a cell subtype of B cells. Obviously, the doublet removal
by Chord can have a great impact on the proportion of cells to
avoid these imbalanced distributions, so that numerous doublets
wont result in becoming noise contamination for the quantitative
of cell types.

To test whether Chord is able to improve the effectiveness of
downstream analysis, we evaluate the performances of Chord on
real data. We utilised ROGUE24, an entropy-based metric, to assess
the purity of cell types in the original and filtered data. The ROGUE
index has been scaled in the range of zero to one where the larger
value means the higher purity. As a result, the ROGUE value of the
filtered data was improved, the increasing trend of ROGUE value
after filtering doublets was apparent (Fig. 4e). Next, we used SciBet25,
a cell type annotation tool based on Bayes decision, to
annotate the cell types from the original data and filtered data, and
calculated the changes in the cell types before and after applying the
dooublet filter. The accuracy of the annotation results of the B cells
(0.726), endothelial cells (0.935), and epithelial cells (0.91) on the
filtered dataset were all greater than those of the B cells (0.705),
endothelial cells (0.908), and epithelial cells (0.904) in the unfiltered
dataset (Supplementary Figure 3d). In addition, more differently
expressed genes can be found after Chord processing, indicating
dooublet removal can improve the effect of DEG analysis (Fig. 4f).
Since a doublet was caused by multiple cells with the same barcode,
doublet cells generally contain a higher unique molecular identifier
(UMI). The UMI of doublets detected by Chord was significantly
higher than singlets in all cell types (Fig. 4g). Since myeloid has the
highest proportion of doublets (8.15%) and have a biological
rationale in tumour microenvironment26. Myeloid cells were
selected to demonstrate that Chord is able to correct the direction of
the cell trajectory. In the pseudotime of myeloid cells in the
dataset, the doublets were unevenly distributed in the dimension-
ality reduction plot and aggregated on the right side. After filtering
the doublets, the direction of the cell trajectory changed, which
might be closer to the real situation (Fig. 4h). We inferred that the
deviation was corrected by removing the doublet data.
Therefore, we believe that Chord’s doublet processing of real
data can improve the purity of cell clusters, allowing researchers
to obtain more accurate cell type identification, accurately
identify DEGs between cell types and obtain better pseudotime
analysis results.

Discussion

A number of tools have been developed to remove doublets from
scRNA data, but most of them cannot perform consistently well
on all datasets (Fig. 2b). For users, it is difficult to evaluate and
choose the most suitable method. To solve this problem, Chord
integrate the results of different methods through the GBM
algorithm. The benefits of each method are retained, and the
disadvantages are minimised. According to our evaluation,
Chord, with a high average ranking and stability, is widely
applicable to various datasets and is able to integrate the doublet
prediction scores from any method. It can accept any updates to
the existing approaches and it will be compatible with any new
approaches in the future (Supplementary Fig. 1a). As novel
methods continue to emerge, the better ones can always be
selectively integrated to improve Chord’s accuracy (Fig. 2b, c). It
will be compatible with some new approaches in the future, and it
can accept any update to the existing approaches (Supplementary
Fig. 1a).

Doublets contained in scRNA data affect not only the quantity
of cell types but also the accuracy of downstream analysis. The
data filtered with Chord were accurately identified in cell type
annotation (Supplementary Fig. 3d), and more potential DEGs
(Fig. 4f) were found with Chord than with other methods. These
results can help researchers obtain more accurate results and
conclusions in subsequent analyses.

We also optimised the construction of simulated training sets.
In this general step, most doublet detection methods add simu-
lated doublets to real data to generate a training set. However,
the training set may contain undiscovered doublets, which could
limit the training of doublet detection models and reduce accu-
rac. Therefore, Chord includes “overkill” step. First, built-in
methods are used to evaluate the data. Then, filtered doublets
identified with various methods are removed, and doublets
simulated based on the remaining cells are used to generate a
training set. In this way, the number of undiscovered doublets in
the training set is greatly reduced, thus improving the accuracy of
training and doublet detection. The use of an accuracy step might
also improve the performance of other methods.

Chord requires the integration of the results of multiple
methods, so it is not optimal in terms of time efficiency. The use
of refactoring the integrated methods may solve this problem.
Also, other pre-processing methods, e.g., ambient mRNA removal
steps for droplets, should be considered to improve the accuracy
of downstream analysis27.

Overall, we proposed a computational approach for doublet
detection that utilizes an ensemble algorithm model. This is the
first study of its kind to use an ensemble algorithm for doublet
detection. This work could help researchers concisely and effi-
ciently remove doublets from scRNA data.
Methods

Chord overview. Data input. The input format used in Chord was a comma-separated expression matrix, which was a background-filtered, UMI-based count matrix for a single sample. Chord pre-processes the count expression matrix according to the Seurat analysis pipeline. Chord can also directly accept object files generated by the Seurat analysis pipeline. In addition, it is suggested that users estimate a doublet rate (Supplementary Fig. 1) based on the loading conditions so that Chord can simulate a simulated training set that is similar to the real dataset selected. It should be mentioned that the doubletrate, which is the estimated doublet rate parameter, has generally robust, and a certain degree of deviation will not greatly affect the results of Chord (Supplementary Fig. 3e, Supplementary Table 8).

Data pre-treatment. To train the results of bcds, cxds, DoubletFinder, and DoubletCell based on ensemble learning, Chord generates SingleCellExperiment object data conforming in the input format of bcds and cxds through the R package SingleCellExperiment28.
Fig. 3 Evaluation of the doublet detection methods using the realistic synthetic datasets on DEG analysis and pseudotime analysis. a The dataset of labelled DEGs was processed by each doublet detection method, and the top 40% of cells based on the doublet score were excluded. Then, the DEGs were detected using MAST and Wilcoxon rank-sum tests. Taking the DEGs as positive, three accuracy measures (i.e., the TPR, TNR, and accuracy) were calculated. b, c After processing the dataset for the pseudotime analysis using each doublet detection method, the top 20% of cells according to the doublet score were excluded. Monocle (B) Singshot (C) were used to construct the trajectories of these results. d The UMAP was embedded for the two realistic synthetic datasets (DataDEG and DataPSE), in which the doublets are shown in red and the singlets are shown in grey. DataDEG is a simulation dataset containing two synthetic cell types, including 1667 cells, 40% of which are correctly labelled doublets. DataPSE consists of 600 cells, 20% of which are synthetic labelled doublets containing a bifurcating trajectory. e The AUC of each method on DataDEG and DataPSE and their ROC curves.

Preliminary deletion of doublets. We tend to reduce the number of doublets in a sample as much as possible before generating simulated doublets to avoid training datasets that contain real doublets because real doublets are not preliminarily labelled and are thus labelled as singlets in the training set. We applied three doublet detection methods, evaluated the selected datasets, and roughly filtered the doublets.

We used the scds() function, cxds, and bcds to evaluate the data based on the parameters ntop = 500, binThresh = 0, and retRes = T; then, we extracted the doublet scores obtained with the two methods for each cell. Next, we used the no ground-truth process of DoubletFinder for evaluation. The parameters were set as PCs = 110 and p = 0.25, and automatically extracted the pk value corresponding to the highest binomial coefficient, and obtained the doublet score.

Chord introduced an adjustable parameter called overkillrate. According to this parameter, we filtered the doubletrate*overkillrate percentage of cells that were most likely to be doublets according to the evaluation results of each method and obtained the prefiltered data. By default, we set this parameter to 1 to exclude the doublets identified by the three built-in methods at the selected doublet rate.

Generating the simulation dataset. To avoid the generation of doublets synthesised from the same cell type, Chord randomly sampled pairs of cells in the pre-filtered data, generated simulated doublets from the raw UMI count by mixing the gene expression profiles of the selected cell pair and then added simulated doublets to the pre-filtered data:

1. Perform a Seurat standardisation process on the data and call the functions NormalizeData(), FindVariableFeatures(), and ScaleData() with default parameters.
2. Cluster cells, perform dimensionality reduction operations on the data with RunPCA(), taking PC1 to PC30 as inputs, and perform k-means clustering (k = 20). After clustering, the cells were divided into 20 clusters.
3. Randomly sample pairs of cells at a ratio of doubletrate/(1-doubletrate) for each cell type, and weight the cells with introduced biological random number from a N(1, 0.1) distribution which was set to roughly represent experimental randomness.
4. Average the weighted gene expression profiles of the cell pair as simulated doublets.
5. Add the simulated doublets to the pre-filtered data; take this new dataset as the training set.

Model training. For the training set in which all cells were labelled, Chord used the same parameter settings to evaluate the doublet scores through the cxds, bcds, DoubletFinder methods. GBM (from R package gbm) which performed better than AdaBoost, XGBoost, and LightGBM (Supplementary Fig. 3a) was used to combine the prediction scores of the built-in methods to fit a model for robust estimation. In GBM, each individual model consists of classification or regression trees, also called boosted regression trees (BRTs). We defined 1000 trees for fitting, and set parameter shrinkage = 0.01 and cv.folds = 5. The function DBBoostTrain() was defined to implement model training, and it combined the scoring results of these built-in methods into a matrix. Then the matrix was input data into the function gbm() in the R package gbm. The simulated doublets were set as true positives (TPs), and the singlets were set as true negatives (TNs) for model training.

Scoring the original data. We defined DBBoostPro(), used the model trained on the training set to predict the doublet scores of the original dataset, and output the doublet score of each cell based on the result of the ensemble model.

Expandable interface. To incorporate version updates of the integrated doublet tools and the release of new doublet tools, Chord included an extendible interface. To incorporate the selected methods, at first the chord() function was used to extract the expression matrix after generating the simulated doublets. Next, we exported prefiltered data, evaluated both the original and prefiltered data using the selected methods, and then imported the scores into Chord. Based on the evaluation scores, Chord used the GBM algorithm to the extra methods together with cxds, bcds and DoubletFinder. At last the ensemble model was used to score the original dataset.

DoubletCells settings. DoubletCells from the R package scrn 1.16.0 with the parameters k = 50 and d = 50 was selected and ran in the R 4.0.2 environment.

Solo settings. Solo was run in the Python 3.7.9 environment, and the parameters were set according to the reference file solo_params_example.json downloaded from https://github.com/calico/solo. The doublet scores of each cell were read through the softmax_scores.npy file.

DoubletDetection settings. Double detection was run in the Python 3.7.9 environment using the operating parameter settings from https://nbviewer.jupyter.org/github/JonathanShor/DoubletDetection/blob/master/tests/notebooks/PMCM_8k_vignette.py. AUC values were obtained with BoostClassifier.fit(voting='average').

Scrublet settings. Scrublet was run in the Python 3.7.9 environment, based on the instructions at https://github.com/AllonKleinLab/scrublet/blob/master/examples/scrublet Basics.ipynb. The doublet scores of each cell were read through Scrublet.scrub_doubles.

Doublet rate gradient. The doublet gradient dataset was composed of random samples from real datasets. The HT08 dataset was randomly sampled from the DM-A dataset with an interval of 0.02, and the ratio of the doublet rate was baried from 0.01 to 0.10. DM-A dataset was randomly sampled from the DM-A dataset at an interval of 0.01, and the ratio of the doublet rates was baried from 0.01 to 10.

Rank variance comparison for different methods. Because of the unstable performance of the methods for different datasets, we calculated the AUC rank variance coefficient to characterise the stability of each method based on different datasets. The methods used in the evaluation, were ranked according to their AUC values, the variance of the AUC ranking of each method was calculated for different datasets as the AUC rank variance coefficient (SDrank). The method with the highest SDrank was unstable for different datasets, it represented the generalisability of methods.

AUROC and AU Precision calculations. Each method was evaluated using the AUC with ground-truth labels for the original datasets or labels for simulated datasets. In addition, the AUROC and partial area under the ROC curve (pAUROC) were calculated. We calculated the AUC and AUROC with the PRROC R package, and plotted the ROC curves for individual method by setting the option 'ROC curve' to 'TRUE'. For the pAUROC at 0.9, a 0.95 specificity value was calculated by using the 'PROC' R package.

Evaluating DEGs with simulated datasets. We used the published simulated single-cell sequence database as a background to analyse the changes in DEGs correctly detected before and after doublet removal by different methods. The simulated single-cell dataset was generated with scDesign and contained 1,667 cells and 18,760 genes. It was divided into two cell types each counted 500 cells, and 667 doublets simulated by those singlets. Among them, high-expression and low-expression DEGs, which were known at the time of data generation, accounted for 6% of all sample (3% upregulated genes and 3% downregulated genes). The dataset without doublets was used as the clean dataset, and the data with doublets were added as the contaminated dataset. After the contaminated dataset was evaluated with a doublet detection method, the dataset of 40% of the cells with the highest score was filtered according to the result. We performed the process described above for each method. Then, we used Seurat’s FindAllMarkers() function with the methods ‘wilcox’ and ‘MAST’ to perform DEG calculations on the dataset. In order to find DEGs, we removed genes with fold changes below 0.25; then, genes with Bonferroni-corrected p values below 0.05 and genes detected in a minimum fraction of 10% cells in either of the two clusters were defined as DEGs. Finally, we calculate the accuracy, TPR, and TNR for all datasets.

Pseudotime analysis of the simulated data. We used the published dataset data of simulated single-cell sequencing to test the effects of different methods on pseudotime analysis. The simulated single-cell dataset, which was generated with Splatter, consisted of 600 cells and 1000 genes. There were two cell tracks containing 250 simulated cells and 100 simulated doublets. The dataset without the doublets was used as a clean dataset, and the data containing the doublets were used as contaminated dataset. After evaluating the contaminated dataset through a doublet detection method, cells predicted to be doublets were deleted. After then,
the R package monocle and Slingshot software were used for a pseudotime analysis of the dataset. We implemented the process described above for each method. In addition, we calculated the trajectories of the clean and contaminated datasets.

**Time cost.** Based on random sampling from the real DM-2.1 dataset, we constructed test datasets of 1,000 to 12,000 cells with a gradient of 1,000 cells and tested various methods with the same processor. Then, the runtime of each method for the simulated datasets on the same computer server (E5-2678v3 CPU processor and 256 GB memory) was recorded.

**Base analysis process for lung cancer data.** We obtained the Seurat object from SCope (https://gbiomed.kuleuven.be/scRNAseq-NSCLC) including 19 samples from five patients.
After evaluating the predicted double cell rate for each sample based on the number of cells, we selected the 5 tumour samples (sample 11, 13, 17, 18, 22) with the highest predicted double cell rate and applied Chord individually to each sample. Then we performed a standard Seurat analysis:

1. The expression matrix and metadata for samples 11, 13, 17, 18 and 22 (33,694 genes across 24,280 cells) was extracted, and a new Seurat object was created.

2. After the data were normalised, 2,000 variable genes were screened with the function FindVariableFeatures().

3. PCA was used to reduce the dimension of the data to 50 dimensions, and PC1 to PC30 were used for clustering. Through the FindNeighbors function (resolution = 1.5), we divided the cells into 22 clusters, and we computed the UMAP embeddings to display the results.\(^3\)

**Cluster purity.** We calculated the rogue() (R package ROGUE) value for each cell type in each sample. Comparisons between two original datasets and filtered datasets were performed using paired two-tailed t-tests. The parameters of ROGUE were set as "platform = UMI" and "span = 0.6".

**Cell type identification using SciBet.** We used the function SciBet() (R package SciBet\(^29\)) to perform cell type analysis on epithelial, endothelial, myeloid, T, and B cells before and after processing. The reference of human cell types were provided by SciBet (http://sciabet.cancer-pku.cn/major_human_cell_types.csv).

**Calculated number of DEGs.** The number of DEGs were calculated by the Wilcoxon rank-sum test (Seurat) with the following parameters: min.pct = 0.1, and test.use = "wilcoxon", additionally, and the threshold of logFC was varied from 0.25 to 0.75 at an interval of 0.05. Then, we counted the number of DEGs at different log.fc.threshold values.

**Statistics and reproducibility.** The details about the steps and statistics of each analysis were recorded in each part of methods. Statistical analyses were performed using R package ggpubr (https://rpkgs.datanovia.com/ggpubr/) or build-in function of R 4.0.2 (https://www.r-project.org/). A P value less than 0.05 was considered as statistically significant.

**Reporting summary.** Further information on research design is available in the Nature Research Reporting Summary linked to this article.

**Data availability.** The data analysed during this study were collected from public studies or databases. The access to data was shown in Supplementary Table 9.

**Code availability.** The Chord software package, including documentation, tutorials is available at https://github.com/13308204545/Chord.\(^{15}\)

Received: 27 June 2021; Accepted: 11 May 2022; Published online: 30 May 2022

**References**

1. Wu, Y. & Zhang, K. Tools for the analysis of high-dimensional single-cell RNA sequencing data. Nat. Rev. Nephrol. 16, 408–421 (2020).

2. Potter, S. S. Single-cell RNA sequencing for the study of development, physiology and disease. Nat. Rev. Nephrol. 14, 479–492 (2018).

3. Prakadan, M. S., shafta, A. K. & Weitz, D. A. Scaling by shrinking: empowering single-cell 'omics' with microfluidic devices. Nat. Rev. Genet. 18, 345–361 (2017).

4. Wolock, S. L., Lopez, R. & Klein, A. M. Script: computational identification of cell doublets in single-cell transcriptomic data. Cell Syst. 8, 281–291.e289 (2019).

5. McCinnis, C. S., Murrow, L. M. & Gartner, Z. J. DoubletFinder: doublet detection in single-cell RNA sequencing data using artificial nearest neighbors. Cell Syst. 8, 329–337.e324 (2019).

6. Bernstein, N. I. et al. Solo: doublet identification in single-cell RNA-Seq via semi-supervised deep learning. Cell Syst. 11, 95–101.e105 (2020).

7. Xi, N. M. & Li, J. J. Benchmarking computational doublet-detection methods for single-cell RNA sequencing data. Cell Syst. https://doi.org/10.1016/j.cels.2020.11.008 (2020).

8. Zheng, G. X. et al. Massively parallel digital transcriptional profiling of single cells. Nat. Commun. 8, 14049 (2017).

9. Stockiuc, M. et al. Cell Hashing with barcoded antibodies enables multiplexing and doublet detection for single cell genomics. Genome Biol. 19, 224 (2018).

10. McCinnis, C. S. et al. MULTI-seq: sample multiplexing for single-cell RNA sequencing using lipid-tagged indices. Nat. Methods 16, 619–626 (2019).

11. Kang, H. M. et al. Multiplexed droplet single-cell RNA-sequencing using natural genetic variation. Nat. Biotechnol. 36, 89–94 (2018).

12. Ditterich, T. G. in International workshop on multiple classifier systems. 1–15 (Springer).

13. Fang, L. T. et al. An ensemble approach to accurately detect somatic mutations using SomaticSeq. Genome Biol. 16, 197 (2015).

14. Al-Abed, S. J. et al. A boosted ensemble algorithm for determination of plaque stability in high-risk patients on coronary CTA. JACC Cardiovasc Imaging 13, 2162–2173 (2020).

15. Bais, A. S. & Kostka, D. scds: computational annotation of doublets in single-cell transcriptomics. Genome Biol. 19, 94 (2018).

16. Ridgeway, G. Generalized Boosted Models: A guide to the GBM package R package version 2.1.8 (2007).

17. DePasquale, E. A. K. et al. DoubletDecom: deconvoluting doublets from single-cell RNA-seq data. Cell Rep. 29, 1718–1727.e1718 (2019).

18. Fay, M. P. & Proschan, M. A. Wilcoxon-Mann-Whitney or t-test? On assumptions for hypothesis tests and multiple interpretations of decision rules. Stat. Surv. 4, 1–39 (2010).

19. Finak, G. et al. MAST: a flexible statistical framework for assessing transcriptional changes and characterizing heterogeneity in single-cell RNA sequencing data. Genome Biol. 16, 278 (2015).

20. Street, K. et al. Slingshot: cell lineage and pseudotime inference for single-cell transcriptomics. BMC Genomics 19, 477 (2018).

21. Qu, X. et al. Reversed graph embedding resolves complex single-cell trajectories. Nat. Methods 14, 979–982 (2017).

22. Luen, A. T., McCarthy, D. J. & Marioni, J. C. A step-by-step workflow for low-level analysis of single-cell RNA-seq data with Bioconductor. F1000Res 8, 2122 (2016).

23. Lamberterts, D. et al. Phenotype molding of stromal cells in the lung tumor microenvironment. Nat. Med. 24, 1277–1289 (2018).

24. Liu, R. et al. An entropy-based metric for assessing the purity of single cell populations. Nat. Commun. 11, 3155 (2020).

25. Li, C. et al. SciBet as a portable and fast single cell type identifier. Nat. Commun. 11, 1818 (2020).

26. Cheng, S. et al. A pan-cancer single-cell transcriptional atlas of tumor infiltrating myeloid cells. Cell 184, 792–809.e723 (2021).

27. Yang, S. et al. Decontamination of ambient RNA in single-cell RNA-seq with DecontX. Genome Biol. 21, 57 (2020).

28. Amezquita, R. A. et al. Orchestrating single-cell analysis with bioconductor. F1000Res 9, 149 (2020).

29. Grau, J., Grosse, I. & Keilwagen, J. PRROC: computing and visualizing precision-recall and receiver operating characteristic curves in R. Bioinformatics 31, 2595–2597 (2015).
Acknowledgements

We thank Dr. Yong Bai from BGI-Shenzhen for the algorithm suggestions. This research was supported by the Guangdong Enterprise Key Laboratory of Human Disease Genomics (2020B1212070828), and Shenzhen Key Laboratory of Single-Cell Omics (ZDSYS20190902093613831).

Author contributions

H.L.Z. and K.X.X. designed the research, performed the data analyses, and wrote the codes and the manuscript. K.K., J.H.Y., and H.M.Y. made suggestions to optimise the article. C.L. revised the manuscript and provided suggestions to data analysis. G.B.L. provided critical advice and oversight for the research and revised the manuscript. All authors have read and approved the final manuscript.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s42003-022-03476-9.

Correspondence and requests for materials should be addressed to Han-Lin Zhou or Gui-Bo Li.

Peer review information Communications Biology thanks Mehmet Ilyas Cosacak and the other, anonymous, reviewer(s) for their contribution to the peer review of this work. Primary Handling Editors Eirini Marouli and Anam Akhtar. Peer reviewer reports are available.

Reprints and permission information is available at http://www.nature.com/reprints

Publisher’s note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made. The images or other third party material in this article are included in the article’s Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the article’s Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this license, visit http://creativecommons.org/licenses/by/4.0/.

© The Author(s) 2022