



Accurate and fast cell marker gene identification with COSG

Min Dai , Xiaobing Pei and Xiu-Jie Wang 

Corresponding authors: Xiu-Jie Wang, Institute of Genetics and Developmental Biology, Innovation Academy of Seed Design, Chinese Academy of Sciences, Beijing 100101, China; University of Chinese Academy of Sciences, Beijing 100049, China. Tel: 86-10-64806590; E-mail: xjwang@genetics.ac.cn. Xiaobing Pei, School of Software, Huazhong University of Science and Technology, Wuhan Hubei 430074, China. Tel: 86-027-87792255; E-mail: xiaobingp@hust.edu.cn

Abstract

Accurate cell classification is the groundwork for downstream analysis of single-cell sequencing data, yet how to identify true marker genes for different cell types still remains a big challenge. Here, we report COSine similarity-based marker Gene identification (COSG) as a cosine similarity-based method for more accurate and scalable marker gene identification. COSG is applicable to single-cell RNA sequencing data, single-cell ATAC sequencing data and spatially resolved transcriptome data. COSG is fast and scalable for ultra-large datasets of million-scale cells. Application on both simulated and real experimental datasets showed that the marker genes or genomic regions identified by COSG have greater cell-type specificity, demonstrating the superior performance of COSG in terms of both accuracy and efficiency as compared with other available methods.

Keywords: cell marker gene, cosine similarity, single-cell RNA-seq, single-cell ATAC-seq, spatially resolved transcriptomics

Introduction

With the broad application of various single-cell sequencing technologies, such as single-cell RNA sequencing (scRNA-seq) [1–3] and single-cell assay for transposase-accessible chromatin using sequencing (scATAC-seq) [4–6], as well as the rapid development of spatially resolved transcriptomics (spatial transcriptomics) technology [7–9], how to accurately distinguish cells of interest from others or to characterize novel cell populations is becoming increasingly important [2, 10, 11]. The commonly used methods for cell marker gene identification usually rely on statistical tests to search for genes that are differentially expressed between cells of interest and all other cells in a dataset [12, 13]. However, as statistical tests tend to identify candidates with systematic differences between two groups, when comparing one type of cells (target cells) with multiple other types of cells (nontarget cells), the top-ranked differentially expressed genes selected by statistical methods may not be the true cell markers. For example, a gene could be highly expressed in target cells and a small group of nontarget cells, but almost non-detectable in other cells. Such gene could be selected as a marker gene for the target cells by expression-based statistical methods, although its expression is not restricted to the target cells. Problematically, expression-based statistical methods are the default approaches for marker gene

identification in most single-cell data analysis toolkits, including the commonly used Scanpy [14] and Seurat [15].

Cosine similarity measures the relationship of two n -dimensional vectors using the cosine value of the angle between the vectors in the vector space. Unlike Euclidean distance, which measures the positional difference between two vectors, cosine similarity compares the orientations of two vectors, this means if two genes have identical expression patterns but different scales of expression abundance among a group of cells, they will be considered as equivalent by cosine similarity analysis. Therefore, cosine similarity is expression scale-independent [16] and should be more sensitive to identify genes specifically expressed in target cells.

As the single-cell RNA-seq technology becomes more mature and popular, the number of cells captured by each experiment is rapidly increasing [1], yet the currently available cell marker gene identification methods often suffer from their slow speed when handling data with a large number of cells. In addition, with the development of scATAC-seq [4–6] and spatial transcriptomics technologies [7–9], the needs for a universal method capable of identifying cell marker genes from multiple types of single-cell data modalities are rapidly emerging.

Min Dai is a Ph.D. candidate in Dr. Xiu-Jie Wang's laboratory at the Institute of Genetics and Developmental Biology, Chinese Academy of Sciences and the University of Chinese Academy of Sciences. His major research interests include single-cell genomics, machine learning and epigenetics.

Xiaobing Pei is a full professor at the School of Software, Huazhong University of Science and Technology. His research focuses on machine learning and data mining.

Xiu-Jie Wang is a full professor at the Institute of Genetics and Developmental Biology, Chinese Academy of Sciences and the University of Chinese Academy of Sciences. Her research focuses on bioinformatics, systems biology and epigenetics.

Received: August 13, 2021. **Revised:** December 3, 2021. **Accepted:** December 17, 2021

© The Author(s) 2022. Published by Oxford University Press. All rights reserved. For Permissions, please email: journals.permissions@oup.com

To address the challenges mentioned above, we developed COSine similarity-based marker Gene identification (COSG), a method to identify cell marker genes with better accuracy and faster speed. COSG outperforms existing tools in terms of the expression specificity of identified marker genes and the analysis efficiency for large-scale datasets. In addition to scRNA-seq data, COSG also works well on scATAC-seq and spatial transcriptome data. Therefore, COSG can serve as a general method for cell marker gene identification across different data modalities to facilitate downstream analysis and discoveries.

Results

COSG uses cosine similarity to evaluate the expression specificity of genes

The basic concept of COSG is to compare two genes within a given cell population by evaluating the angles between the vectors representing the expression pattern of each gene in an n -dimensional cell space. Within the cell space, each dimension represents a cell. The representative vector for each gene consists of n -basis (n equals to the number of total detected cells), and the coordinate of each basis represents the gene's expression level in each cell. Therefore, the cosine similarity of two genes equals the cosine value of the angle between the two genes' representative vectors in the cell space. The more similar the expression patterns, the smaller the angle is. If two genes have identical expression patterns, the angle between their representative vectors will be zero, regardless of their expression abundance difference.

The marker gene identification process of COSG starts with multiple groups of cells pre-classified by other single-cell analysis tools. To identify marker genes for each cell group, COSG first creates an artificial gene (λ_k) which only expresses in cells of a given group, e.g. Group k ($G_k, k \in \{1, \dots, K\}$) and does not express in any other cell groups, thus λ_k would be the ideal marker gene for cells belonging to G_k (Figure 1). The representative vector for each expressed gene ($g_i, i \in \{1, \dots, M\}$) will be compared with the representative vector of λ_k , genes whose representative vectors form the smallest angles with the representative vector of λ_k and the largest angles with the representative vectors of other cell groups ($\lambda_t, t \in \{1, \dots, K\}$ and $t \neq k$) will be selected as the marker genes for G_k . Here, we define COSG score as $\text{COSGscore}(g_i, G_k) = \cos(g_i, \lambda_k) * \frac{\cos(g_i, \lambda_k)^2}{\cos(g_i, \lambda_k)^2 + \mu * \sum_{t \in \{1, \dots, K\}, t \neq k} \cos(g_i, \lambda_t)^2}$, where $\cos()$ calculates the cosine similarity between the representative vectors of two genes, and μ ($\mu \geq 0$) is a user-defined hyperparameter as the penalty factor (by default, $\mu = 1$). The output of COSG is a list of candidate marker genes starting with the ones with the highest COSG scores for each cell group. COSG is available both in Python and R, and can be jointly used with Scanpy [14] and Seurat [15].

COSG identifies more indicative marker genes in scRNA-seq data

To test the function of COSG, we first generated 30 simulated scRNA-seq datasets with known marker genes as the ground truth (Methods, Supplementary Table 1), and compared the performance of COSG with other 10 popular methods in the commonly used toolkits Scanpy [14] and Seurat [15] (Supplementary Table 2). We calculated the average overlapping ratios between the top 20 marker genes identified by each method and the 20 true marker genes for each of the 30 simulated datasets. The results showed that COSG outperformed all other tested methods (Figure 2a and Supplementary Figure 1).

To further evaluate the performance of COSG, we chose to compare the results generated by COSG with the ones generated by Logistic regression and Wilcoxon-test (with or without tie correction), as Logistic regression (machine learning based method) and Wilcoxon-test (statistical analysis based method) each represent a main class of scRNA-seq data analysis tools [17–19]. In addition, Wilcoxon-test is currently the most widely used method for single cell data analysis [20] and is the default method used in Seurat. As scRNA-seq data usually contain many zero values (tied values), Wilcoxon-test with tie correction might perform better than traditional Wilcoxon-test, thus we also included Wilcoxon-test with tie correction (denoted as Wilcoxon-test (TIE)) for detailed comparison.

We used two reported scRNA-seq datasets (denoted as scRNA-Hochgerner dataset [21] and scRNA-Stewart dataset [22]) to test these methods (Supplementary Table 3). The scRNA-Hochgerner dataset contains 23 025 cells (classified into 24 cell types by the authors) from the dentate gyrus tissue of perinatal, juvenile and adult mice [21]. The UMAP projection results confirmed the gene expression similarities among cells within each pre-classified cell type (Figure 2b). We then examined the top three marker genes for each cell type identified by COSG and other methods, and found that most marker genes identified by Logistic regression or Wilcoxon-test also had expression in nontarget cell types (Figure 2c). Wilcoxon-test (TIE) works slightly better, but still identified more nonspecific marker genes as compared with COSG (Figure 2c). Enrichment score analysis also showed that marker genes identified by COSG had much higher enrichment scores in the target cell type compared with other cell types (Figure 2d). About 54% marker genes (top three for each cell type) identified by COSG were also reported by Wilcoxon-test (TIE), but only 16% or 8% of them were identified by Logistic regression or Wilcoxon-test, respectively (Supplementary Figure 2a). Similar overlapping proportions were also observed when the top 10 marker genes for each cell type were compared (Supplementary Figure 2b). Expression pattern examination also revealed that the top three marker genes for adult granule cells (GC-adult) identified by other methods also had relatively high expression in at least one type of other cells, such as hippocampus

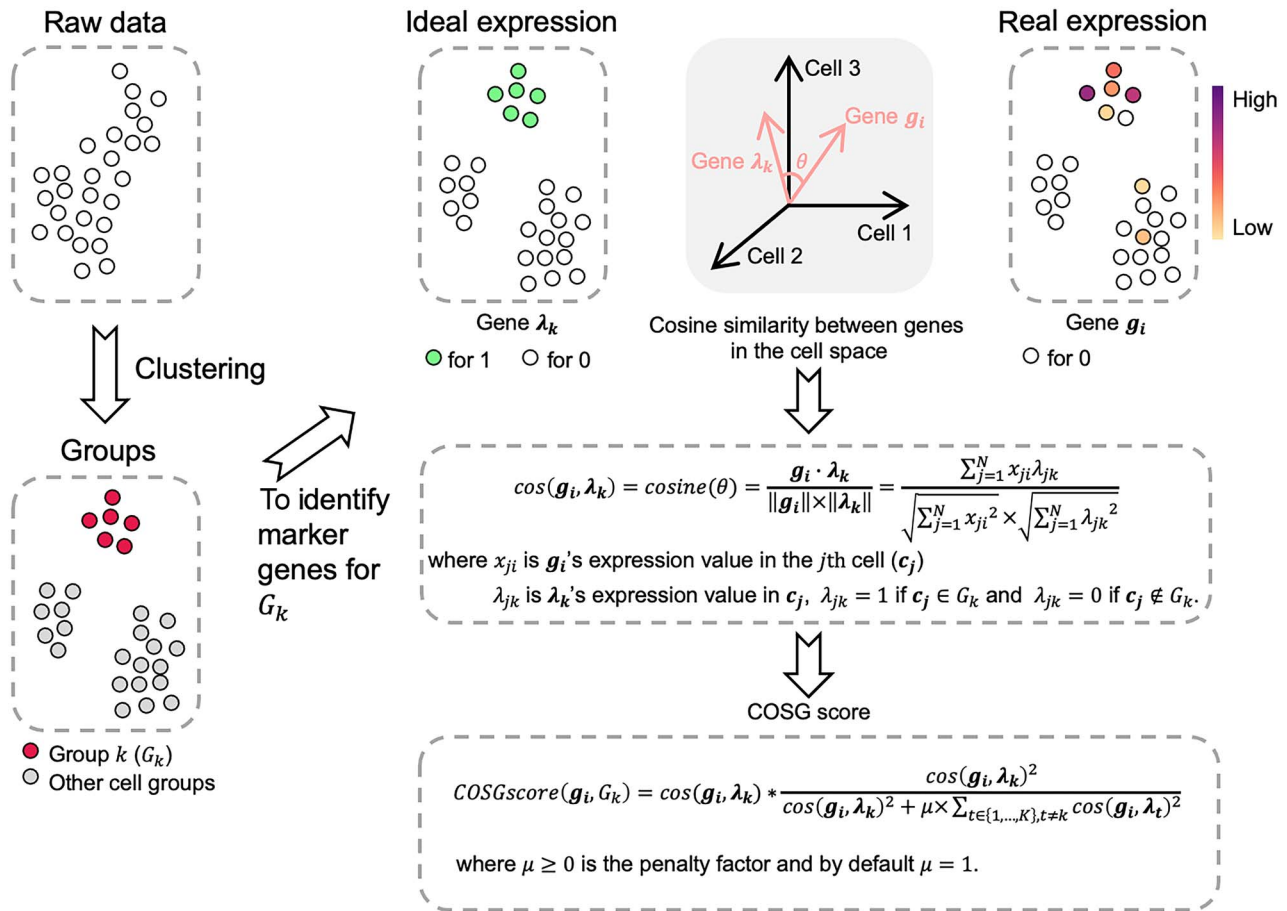


Figure 1. Workflow of COSG. The basic idea of COSG is to identify marker genes within a given group of cells by comparing the cosine values of the angles between the representative vectors of each detected gene and the assumed ideal marker gene. The input data of COSG should be normalized and clustered scRNA-seq data/scATAC-seq data/spatial transcriptome data. For a dataset of N cells (clustered into K groups) with M expressed genes, to identify marker genes for group k (G_k , $k \in \{1, \dots, K\}$), COSG first creates an ideal marker gene λ_k for G_k , which was only detected in cells of G_k with uniform expression value but not in any other group of cells. To examine whether a detected gene, g_i , $i \in \{1, \dots, M\}$, is a good marker gene for G_k , COSG evaluates the expression similarity between gene g_i and gene λ_k among all cells by calculating the cosine values of the angles formed by the representative vectors of g_i and λ_k in the N -dimensional space spanned by all cells, then generates COSG score to reflect the expression specificity of g_i in G_k by comparing the expression values of g_i and λ_k as well as λ_t ($t \in \{1, \dots, K\}$ and $t \neq k$). As λ_t represents the ideal marker genes for cell groups other than G_k , the COSG score reflects the suitability of g_i to serve as a marker gene for G_k . By repeating the above procedures, COSG could identify marker genes for each group of cells.

CA3 pyramidal layer cells (CA3-Pyr) and juvenile GC cells (GC-juv), which were highly similar to GC-adult cells (Supplementary Figure 2c). However, two out of three marker genes identified by COSG had GC-adult cell-specific expression (Supplementary Figure 2c). The scRNA-Stewart dataset contains 40 268 cells (classified into 27 cell types by the authors) from human adult kidney tissue [22], among these, some cell types, especially the immune cells, were less distinguishable from each other by UMAP projection (Supplementary Figure 3a). Again, the marker genes for almost all cell types identified by COSG showed high specificity, yet the other methods failed to reach the same standard (Supplementary Figure 3b and 3c).

COSG outperforms existing methods on large-scale datasets

To evaluate the computational performance and scalability of COSG, we measured the running time of COSG and the other 10 methods mentioned above

(Supplementary Table 2) on 14 scRNA-seq datasets with cell numbers ranging from 1000 to 150 000 (Supplementary Table 4). When processing scRNA-seq data with less than 5000 cells, COSG and five other methods (namely t-test, t-test_overestim_var, Wilcoxon-test, Logistic regression and Wilcoxon-test (TIE)) completed the analysis almost instantly (Figure 3a). Further comparison of these six methods on larger datasets with 10 000 to 150 000 cells demonstrated that COSG ran much faster than other methods, especially when the number of cells reached 150 000 (Figure 3b and Supplementary Table 5). In addition, COSG identified marker genes for over 1 million cells (1331 984 cells) of 37 cell types in less than 2 min (Supplementary Figure 4).

To examine whether the high efficiency of COSG is achieved without sacrificing its accuracy, we further analyzed the expression of the top three marker genes for each cell type identified by Logistic regression, Wilcoxon-test, Wilcoxon-test (TIE) and COSG from the above-mentioned 150 000 cells. Among the 31 cell types

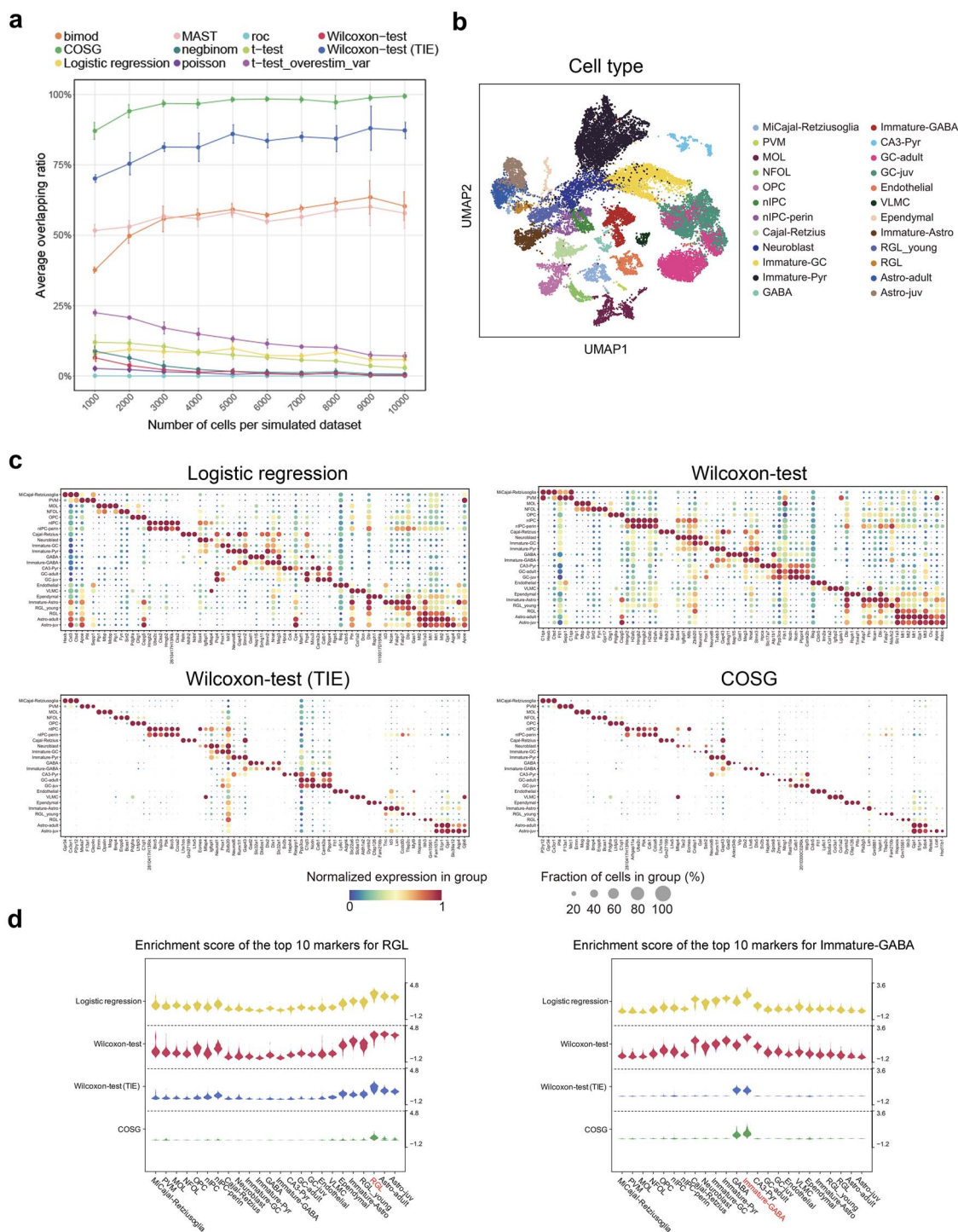


Figure 2. Performance comparison of COSG with other methods on scRNA-seq data. **(A)** Average overlapping ratios of the top 20 marker genes identified by COSG or other 10 popular methods versus the top 20 known marker genes of the 30 simulated datasets. Error bars represent the standard deviation of three datasets. **(B)** UMAP projection of the scRNA-seq data of dentate gyrus cells from perinatal, juvenile and adult mice. **(C)** Expression dot plots of the top three marker genes identified by Logistic regression, Wilcoxon-test, Wilcoxon-test (TIE) and COSG for each cell type. **(D)** Enrichment score comparison of the top 10 marker genes identified by Logistic regression, Wilcoxon-test, Wilcoxon-test (TIE) and COSG for RGL cells and Immature-GABA cells. Violin plots represent the enrichment scores of the top 10 marker genes identified by each method for RGL cells and Immature-GABA cells.

in this dataset, some cell types were difficult to be distinguished from each other by UMAP projections (Figure 3c) or by marker genes identified by Logistic regression or Wilcoxon-test (Figure 3d). Both COSG and Wilcoxon-test (TIE) reported specific marker genes for most cell types, but the processing time used by

COSG was only 1/280 of that used by Wilcoxon-test (TIE), and the marker genes identified by COSG also had higher expression specificity (Figure 3b and 3d, Supplementary Table 5). For example, the top three marker genes for fibroblasts of cardiac tissue identified by COSG all showed fibroblast dominant expression,

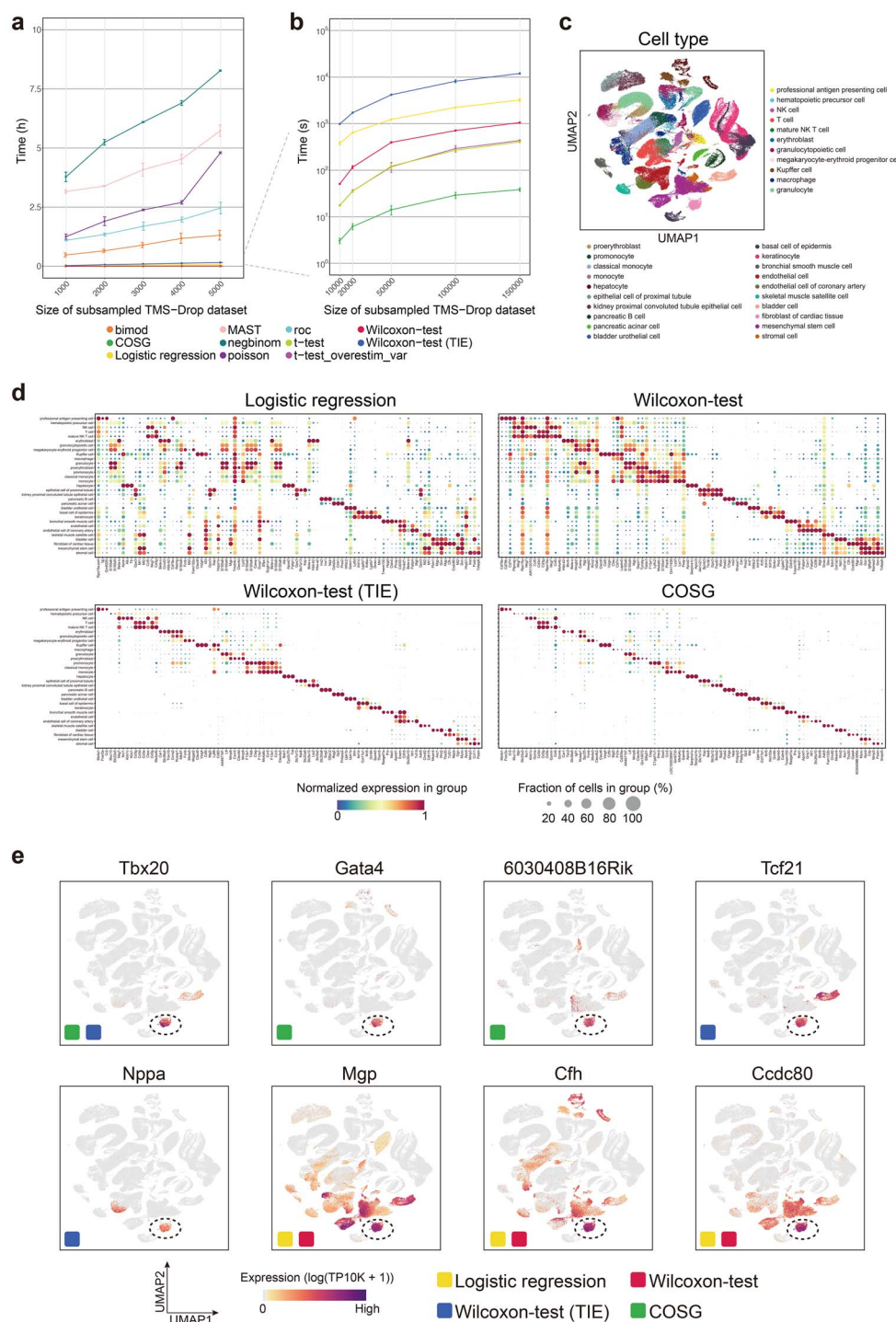


Figure 3. COSG efficiently and accurately identifies more indicative marker genes in large-scale scRNA-seq datasets. **(A)** Running time of COSG and other 10 popular methods on subsampled Drop-seq datasets with cell numbers ranging from 1000 to 5000. Error bars represent the standard deviation of three replicated runs. **(B)** Running time of the six fastest methods, namely COSG, t-test, t-test_overestim_var, Wilcoxon-test, Logistic regression and Wilcoxon-test (TIE) on subsampled Drop-seq datasets with cell numbers ranging from 10 000 to 150 000. Error bars represent the standard deviation of three replicated runs. **(C)** UMAP projection of the scRNA-seq data with 150 000 subsampled cells. **(D)** Expression dot plots of the top three marker genes identified by Logistic regression, Wilcoxon-test, Wilcoxon-test (TIE) and COSG for each group. **(E)** Expression patterns of the top three marker genes for fibroblasts of cardiac tissue identified by Logistic regression, Wilcoxon-test, Wilcoxon-test (TIE) and COSG. Cells classified as fibroblasts of cardiac tissue are indicated by dashed circles.

whereas the top three marker genes identified by other methods also had high expression in one or more types of nontarget cells (Figure 3e). Taken together, these results demonstrated the advantages of COSG in handling large-scale datasets.

COSG correctly identifies cell-type-specific marker regions in scATAC-seq data

We next assessed the performance of COSG on scATAC-seq data, which are much sparser and contain 10–20 times more features than scRNA-seq data [23]. Again, we

compared the results generated by Logistic regression, Wilcoxon-test, Wilcoxon-test (TIE) and COSG using two reported scATAC-seq datasets (Supplementary Table 3). The first dataset (denoted as ATAC-Pijuan-Sala dataset) contains 301 316 detected genomic regions of 19 453 single nuclei from mouse embryos at 8.25 days post-fertilization [4]. The second dataset (denoted as ATAC-Granja dataset [6]) contains 451 999 detected genomic regions of 33 819 bone marrow and peripheral blood mononuclear cells (BMMCs and PBMCs, respectively) from healthy human donors. We first examined the computational efficiency of COSG on scATAC-seq data. The ATAC-Granja dataset was classified into 17 cell types or 23 cell types by the authors via a broad cell type annotation method or a fine cell-type annotation method, respectively [6]. Under both cell type classification conditions, COSG finished the analysis within 2 min, whereas Logistic regression and Wilcoxon-test were about 30 times slower than COSG, and Wilcoxon-test (TIE) was more than 300 times slower than COSG (Figure 4a, Supplementary Table 6).

The UMAP projection result of the ATAC-Pijuan-Sala dataset [4] showed overlaps of some pre-classified cell types, especially those from undifferentiated mesoderm (Figure 4b), such poor discrimination among incompletely differentiated cells has posed great difficulties in marker gene identification. Similar to the results of scRNA-seq data, the top three marker regions identified by COSG were more specific than those identified by other methods (Figure 4c). Majority of marker regions reported by COSG were not identified by Logistic regression or Wilcoxon-test (Figure 4d). Taking forebrain cells as an example, the genomic region 'chr14-48738109-48738610' had specific accessibility in forebrain cells and was identified as one of the top three marker regions only by COSG, yet the marker regions identified by other methods showed high accessibility in non-forebrain cells, namely spinal cord, mid/hindbrain cells or neural crest cells (Figure 4e). Notably, region 'chr2-142589336-142590022', one of the top three marker regions for forebrain cells identified by Wilcoxon-test (TIE), showed much higher accessibility in neural crest cells than in forebrain cells (Figure 4e).

Immune cells, especially subtypes of the same immune cell type (e.g. naive CD4⁺ T cell and memory CD4⁺ T cell), are usually highly similar in terms of molecular features. Analysis results of both the broad cell-type annotation (including 17 cell types) and fine cell-type annotation (including 23 cell types) of the Granja scATAC-seq dataset showed that marker regions identified by COSG had higher cell type specificity than those identified by other methods, especially for different T cell subtypes (Supplementary Figures 5 and 6).

COSG holds advantage in analyzing spatial transcriptome data

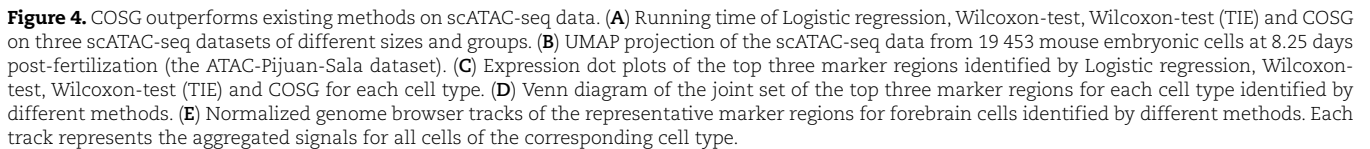
Spatial transcriptome data have emerged as a new data type in recent years, and the analysis of spatial

transcriptome data also relies on marker gene identification to characterize cell types. To test the capability of COSG on analyzing spatial transcriptome data, we first applied it on a dataset (denoted as Spatial-brain_sagitta dataset, Supplementary Table 3) generated by 10x Genomics Visium platform using adult mouse brain. A total of 3355 signal spots were detected in this dataset and clustered into 11 groups according to their gene expression profiles (Figure 5a and 5b). To examine the accuracy of COSG, we compared the top three marker genes for each cell cluster identified by different methods (Figure 5c). It is apparent that most marker genes identified by Logistic regression or Wilcoxon-test did not have cell type specificity. Wilcoxon-test (TIE) worked better, but still picked up more nonspecific cell markers as compared with COSG (Figure 5c). We further examined the spatial expression patterns of the top three marker genes identified by each method for Cluster 0 cells (Figure 5d). The results showed that marker genes identified by COSG had higher and more specific expression among Cluster 0 cells as compared with markers identified by other methods. Similarly, application of the above-mentioned four methods on another 10x Genomics Visium dataset (denoted as Spatial_brain_coronal dataset, Supplementary Table 3) generated using the coronal region of a mouse brain also demonstrated the capability of COSG in identifying more indicative marker genes from noisy data (Supplementary Figure 7).

We next analyzed the performance consistency of COSG across spatial transcriptomics platforms using a dataset of mouse hippocampus generated by the Slide-seqV2 technology [8] (denoted as the Spatial-Slide-seqV2 dataset, Supplementary Table 3). The dataset contains 9319 high-quality beads classified into 13 clusters (Supplementary Figure 8a and 8b). Expression dot plots of the top three markers for each cell cluster demonstrated the superior performance of COSG as compared with other methods (Supplementary Figure 8c). Expression pattern comparison also confirmed that the top three marker genes for Cluster 5 cells identified by COSG had more specific expression, whereas marker genes picked by other methods were broadly expressed (Supplementary Figure 8d).

Discussion

Marker gene identification safeguards the accuracy of cell type discrimination, and is therefore a critical step in single-cell sequencing data or spatial transcriptome data analysis. Here, we present COSG as a more accurate and faster method for marker gene identification from scRNA-seq, scATAC-seq and spatial transcriptome data. COSG should be applied to pre-clustered data to facilitate subsequent cell-type annotations, and the outputs of COSG can also be used to refine cell clustering results. COSG is implemented in both Python and R, and can be



The outstanding performance of COSG is achieved by assuming an ideal marker gene for each cell group and using cosine similarity to compare the expression patterns between the detected genes and the assumed

ideal marker genes. Therefore, unlike those statistics-based marker gene identification methods [13], COSG is more robust to sequencing depth and capture efficiency of cells, as mimicked by the downsampling experiments using both scRNA-seq and scATAC-seq data (Supplementary Figure 9), thus COSG often generates

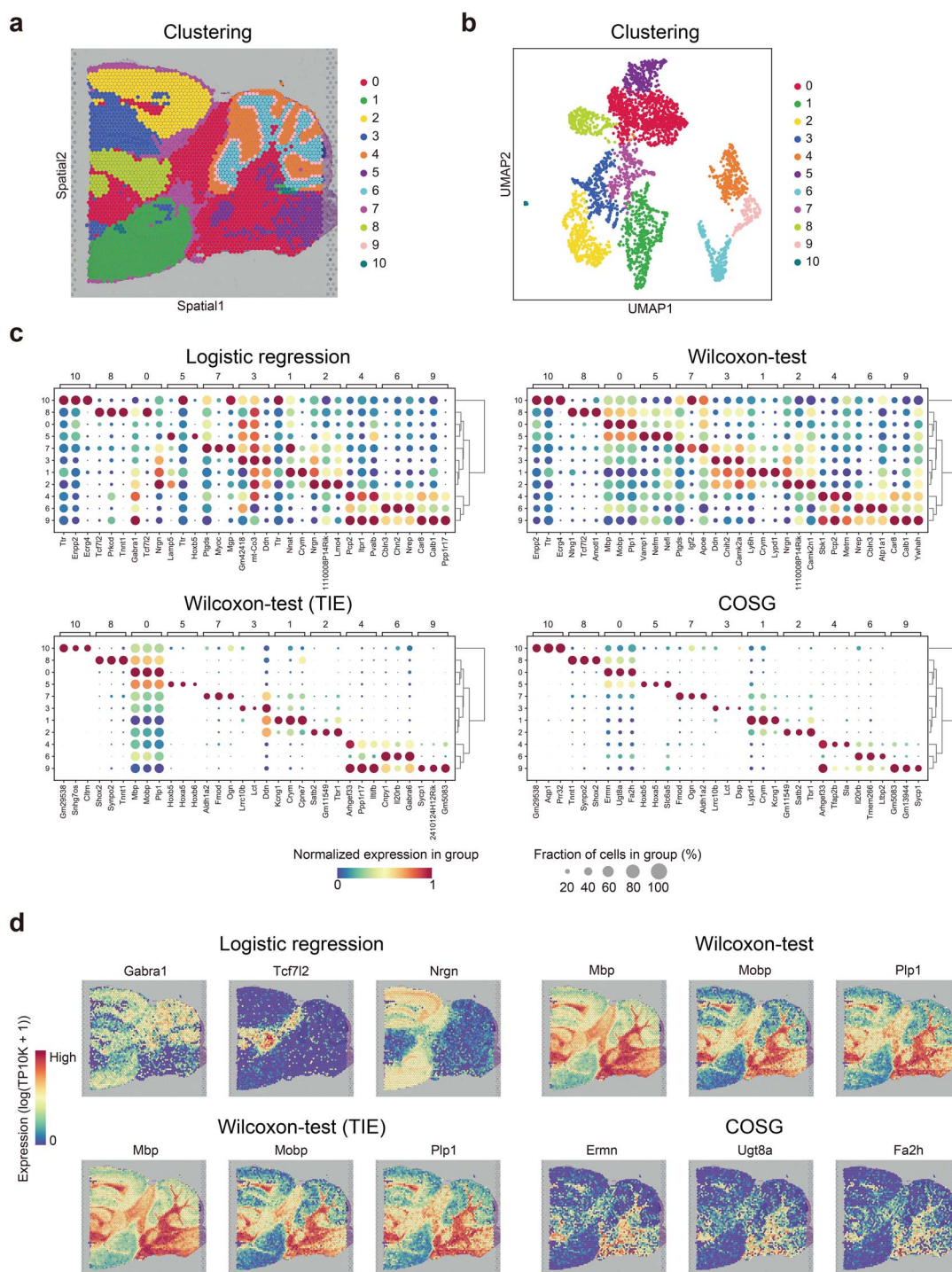


Figure 5. COSG performed well on spatial transcriptome data. (A) Clustering results of the 3355 signal spots detected in adult mouse brain sagittal posterior tissue. (B) UMAP projection of the signal spots shown in (A). (C) Expression dot plots of the top three marker genes identified by Logistic regression, Wilcoxon-test, Wilcoxon-test (TIE) and COSG for each cell cluster. (D) Expression patterns of the top three marker genes identified by different methods for cells in Cluster 0.

more accurate results. The default Wilcoxon-test in Scanpy does not perform tie correction and has been widely used by many published studies [24] and cell atlas projects [25, 26]. Our experiments also showed that due to the high frequency of missing values (zeros, or tied values), doing tie correction is necessary for Wilcoxon-test when it is applied to single-cell sequencing data.

By default, COSG uses the top three marker genes for each cell type to distinguish cell types; this strategy has been proven to be effective by examples shown in this work. However, it is worth to note that the number of marker genes for a given cell type will be influenced by the presence of other cell types with similar gene expression profiles; under such circumstances more marker

genes are required. Therefore, we recommend users to examine the expression patterns of the ranked marker genes by UMAP or dot plot visualization to determine the right number of marker genes needed for each cell type in a given dataset.

COSG runs remarkably faster than other available methods, and it is capable of identifying marker genes from scRNA-seq data of over 1 million cells in less than 2 min. COSG is a universal method. It has achieved good performance on scATAC-seq and spatial transcriptome data, and also has the potential to be effectively applied to other types of single-cell omics data. The fast speed of COSG would be more beneficial when applied to whole-genome scale single-cell sequencing data, as analysis of these types of data is usually time-consuming.

In short, COSG can serve as a general method for cell marker gene identification across different data modalities to facilitate single-cell data analysis and biomedical discoveries. Because the 10x Visium and Slide-seqV2 technologies have not yet reached single-cell resolution, one spot or bead could contain multiple cells of various types, therefore the marker genes identified by COSG from spatial transcriptome data are not as discriminative as those identified from scRNA-seq data or scATAC-seq data. Enrichment analysis or aggregation of marker gene expression may improve cluster annotations for spatial transcriptome data, which awaits future exploration.

Methods

Overview of COSG algorithm

COSG is designed to identify proper marker genes for pre-classified cell groups. The input data for COSG should first be normalized by other methods. After normalization, COSG generates the gene expression matrix $X \in \mathbb{R}^{N \times M}$, where N is the number of cells and M is the total number of detected genes. The i^{th} gene \mathbf{g}_i 's expression among all cells is the i^{th} column of X :

$$\mathbf{g}_i = \begin{bmatrix} x_{1i} \\ \vdots \\ x_{ni} \end{bmatrix}$$

where x_{ji} is \mathbf{g}_i 's expression value in the j^{th} cell, \mathbf{c}_j , $j \in \{1, \dots, N\}$. Let K represents the number of cell groups predefined by manual annotation or unsupervised cell clustering. In order to identify marker genes for group G_k , $k \in \{1, \dots, K\}$, we first set an ideal marker gene λ_k for G_k :

$$\lambda_k = \begin{bmatrix} \lambda_{1k} \\ \vdots \\ \lambda_{nk} \end{bmatrix}$$

where $\lambda_{jk} = 1$ if $\mathbf{c}_j \in G_k$ and $\lambda_{jk} = 0$ if $\mathbf{c}_j \notin G_k$.

We then calculate the cosine similarity between \mathbf{g}_i and λ_k as $\cos(\mathbf{g}_i, \lambda_k)$:

$$\cos(\mathbf{g}_i, \lambda_k) = \frac{\mathbf{g}_i \cdot \lambda_k}{\|\mathbf{g}_i\| \times \|\lambda_k\|} = \frac{\sum_{j=1}^N x_{ji} \lambda_{jk}}{\sqrt{\sum_{j=1}^N x_{ji}^2} \times \sqrt{\sum_{j=1}^N \lambda_{jk}^2}}$$

To evaluate whether \mathbf{g}_i is a good marker gene for group G_k , we calculate COSG score as:

$$\text{COSGscore}(\mathbf{g}_i, G_k) = \cos(\mathbf{g}_i, \lambda_k) * \frac{\cos(\mathbf{g}_i, \lambda_k)^2}{\cos(\mathbf{g}_i, \lambda_k)^2 + \mu \times \sum_{t \in \{1, \dots, K\}, t \neq k} \cos(\mathbf{g}_i, \lambda_t)^2}$$

where the left term $\cos(\mathbf{g}_i, \lambda_k)$ represents the cosine similarity between any detected gene \mathbf{g}_i and the ideal marker gene λ_k by evaluating their expression pattern among all cells. The right term $\frac{\cos(\mathbf{g}_i, \lambda_k)^2}{\cos(\mathbf{g}_i, \lambda_k)^2 + \mu \times \sum_{t \in \{1, \dots, K\}, t \neq k} \cos(\mathbf{g}_i, \lambda_t)^2}$ is used as the penalty coefficient, and μ ($\mu \geq 0$) is the penalty factor for expression in nontarget cell group G_t , $t \in \{1, \dots, K\}$ and $t \neq k$. The value of μ can be adjusted by users, and a larger μ means a bigger penalization for genes expressed in nontarget cells. When $\mu = 0$, no penalty was given to the expression of gene \mathbf{g}_i in nontarget cell groups, the penalty coefficient equals 1. When $\mu = 1$, equal penalty was given to the expression of gene \mathbf{g}_i in the target cell group and each nontarget cell group. As $\mu \times \sum_{t \in \{1, \dots, K\}, t \neq k} \cos(\mathbf{g}_i, \lambda_t)^2$ represents the total penalty for the expression of gene \mathbf{g}_i in all nontarget cell groups, the more groups gene \mathbf{g}_i expressed, the larger value of $\mu \times \sum_{t \in \{1, \dots, K\}, t \neq k} \cos(\mathbf{g}_i, \lambda_t)^2$, therefore the smaller the penalty coefficient, and the smaller the COSG score. When $\mu > 1$, more penalty was given to the expression of gene \mathbf{g}_i in the nontarget cell groups than in the target cell group. On the other hand, when $0 < \mu < 1$, the expression of gene \mathbf{g}_i in the nontarget cell groups was considered less important than the expression of gene \mathbf{g}_i in the target cell group. In all cases, the larger value of μ or the more groups of nontarget cells with gene \mathbf{g}_i expression will both lead to smaller values of the penalty coefficient and smaller COSG scores. By default, COSG sets $\mu = 1$.

Methods compared with COSG

Besides COSG, 10 commonly used cell marker gene identification methods were evaluated in this study (Supplementary Table 2). Among them, Logistic regression was implemented by `tl.rank_gene_groups` (Scanpy v1.6.1) with 'method' set to 'logreg'. Wilcoxon-test was implemented by `tl.rank_gene_groups` (Scanpy v1.6.1) with 'method' set to 'wilcoxon' and 'tie_correct' set to False. Wilcoxon-test (TIE) was implemented by `tl.rank_gene_groups` (Scanpy v1.6.1) with 'method' set to 'wilcoxon' and 'tie_correct' set to True. For t-test and t-test_overestim_var, we used `tl.rank_gene_groups`

(Scanpy v1.6.1) with 'method' set to 't-test' or 't-test_overestim_var', respectively. For bimod, MAST, negbinom, poisson and roc, we used Seurat v3.2.3's FindAllMarkers function with the parameter 'test.use' set to 'bimod', 'MAST', 'negbinom', 'poisson' or 'roc', respectively. All methods took normalized and log-transformed gene expression data as the input except for the negbinom and poisson methods, which took the raw count data as the input.

Public data resources

scRNA-seq data

For the scRNA-seq dataset of mouse dentate gyrus cells [21], the raw counts of unique molecular identifiers (UMIs) were downloaded from NCBI Gene Expression Omnibus (GEO) database (GSE104323). Data quality control was performed by filtering out genes detected in less than three cells and cells with any of the following features: (1) with fewer than 200 detected genes; (2) with more than 4000 detected genes or more than 15 000 total UMIs; (3) with more than 20% UMIs derived from mitochondrial genomes. For the human kidney scRNA-seq data [22], the preprocessed and normalized UMI counts were downloaded from the COVID-19 Cell Atlas (<https://www.covid19cellatlas.org>) [27].

scATAC-seq data

The raw scATAC-seq data and the processed genome track files of the mouse embryonic scATAC-seq dataset [4] were downloaded from the NCBI GEO repository (GSE133244). The raw scATAC-seq data of the human bone marrow and peripheral blood mononuclear cells (BMMCs and PBMCs, respectively) [6] were downloaded from <https://github.com/GreenleafLab/MPAL-Single-Cell-2019> (File name: scATAC-Healthy-Hematopoiesis-191,120.rds).

Spatial transcriptome data

The adult mouse brain spatial transcriptome datasets (the sagittal posterior and coronal data) were downloaded from the 10x Genomics Visium spatial transcriptomics platform (https://support.10xgenomics.com/spatial-gene-expression/datasets/1.1.0/V1_Mouse_Brain_Sagittal_Posterior and https://support.10xgenomics.com/spatial-gene-expression/datasets/1.1.0/V1_Adult_Mouse_Brain). Genes detected in less than three spots were filtered out. The raw Slide-seqV2 mouse hippocampus spatial transcriptome data [8] were downloaded from https://singlecell.broadinstitute.org/single_cell/study/SCP815/sensitive-spatial-genome-wide-expression-profiling-at-cellular-resolution (Puck_200115_08). Data quality control was performed by filtering out genes detected in less than three beads and beads met any of the following requirements: (1) with fewer than 200 detected genes or fewer than 1000 total UMIs; (2) with more than 3000 detected genes or more than 5000 total UMIs; (3) with more than 20% UMIs derived from mitochondrial genomes.

Generation of simulated datasets

The simulated datasets used in this study were generated by in-house built R scripts. Genes were simulated to follow negative binomial distribution using the `rnbino()` function in R: $y = \text{rnbino}(n = n, \mu = \mu, \text{size} = 1)$, where n is the number of cells, μ is the mean expression value of each gene and size is defined as the target number of successful trials. The following five types of gene expression patterns were simulated.

Type I expression represents the expression patterns of good marker genes, under which the simulated genes are specifically expressed in and restricted to target cells. For Type I genes, μ was set as 0.2 in target cells and 0.001 in nontarget cells. Type II expression means the simulated genes are widely expressed, but with higher expression levels in target cells than in nontarget cells. Genes with Type II expression pattern were simulated with μ set as 4 for 85% of the target cells, μ set as 2 for 85% of the nontarget cells, and μ set as 2 or 4 for the remaining 15% of the target cells or the nontarget cells, respectively. Type III expression means the simulated genes are not only expressed in the target group ($\mu = 0.4$), but also expressed in limited numbers of nontarget groups (three nontarget groups were created in each simulation, $\mu = 0.2$ for each group). Type IV expression means the simulated genes have detectable but low expression in all cells; these genes were simulated with $\mu = 0.1$. Type V expression means the simulated genes are highly expressed in all cells, and these genes were simulated with $\mu = 2$.

Using the above procedure, we generated 30 simulated datasets (each contains 20 cell groups). The number of cells contained in the simulated datasets ranged from 1000 to 10 000, with the number of cells per dataset increased 1000 per step. At each total cell number, three datasets were generated with different population distributions among cell groups. For all datasets, the minimum number of cells for a cell group was set as 5. A total of 20 genes with Type I expression pattern were generated as the true marker genes for each cell group. In addition, 20 genes with Type II expression and 20 genes with Type III expression were generated for each cell group to serve as the confounding factors for the true marker genes. For all cell groups in each dataset, the numbers of genes with Type IV expression and genes with Type V expression were both set as 500.

Generation of large-scale experimental benchmark datasets

To generate large-scale benchmark datasets, we subsampled the Drop-seq scRNA-seq dataset of *Tabula Muris Senis* [28] (TMS, 245 389 cells of 123 annotated cell types) to generate 14 experimental benchmark datasets (each contained 31 cell types) with sizes ranging from 1000 to 150 000 cells (Supplementary Table 4). The raw UMI count data were downloaded from https://figshare.com/articles/dataset/tms_gene_data_rv1/12827615?file=24351014. Because the purpose of this benchmark test was to

evaluate the scalability and accuracy of COSG on large-scale datasets, cell types with too few or too many cells were likely to be omitted or overrepresented in the subsampled datasets, resulting in cell type imbalance, therefore cell types with too few cells (less than 2000 cells) or too many cells (more than 30 000 cells) were filtered out in this benchmark test to avoid sampling bias. The remaining 156 630 cells from 31 cell types were subsampled using the `pp.subsample` function (Scanpy v1.6.1). Cell replacement was not allowed during the subsampling process.

To generate benchmark datasets from the Mouse Organogenesis Cell Atlas [29], the filtered high-quality scRNA-seq UMI count data (File name: `gene_count_cleaned.RDS`) were downloaded from the Mouse Organogenesis Cell Atlas website (<https://oncoscape.v3.sttrcancer.org/atlas.gs.washington.edu.mouse.mna/downloads>). The downloaded data have 1331 984 cells with annotations. Genes detected in less than three cells were filtered out. Benchmark datasets with 50 000, 100 000, 500 000, 1000 000 and 1331 984 cells, respectively, were generated using the same method mentioned above. Each dataset contains exactly 37 cell types.

Data normalization

Except for the human kidney scRNA-seq dataset, which used the normalized UMI counts provided by the dataset owners, all other datasets were normalized using the below methods. For the scRNA-seq and spatial transcriptome data, normalization was performed by firstly dividing the raw counts of each gene within each cell/spot/bead by the total number of raw counts within that cell/spot/bead, and then multiplying by 10 000, using the `pp.normalize_total` function in Scanpy v1.6.1. The normalized counts were then log-transformed via the `pp.log1p` function in Scanpy v1.6.1. For the scATAC-seq data, the raw scATAC-seq data were normalized by the term frequency-inverse document frequency (TF-IDF) algorithm implemented by the `RunTFIDF` function of Signac v1.1.0 [30].

Dimensionality reduction

We used Principal Component Analysis (PCA) to embed the detected cells/spots/beads of scRNA-seq or spatial transcriptome data into a low-dimensional space. Before PCA, we selected 3000 highly variable genes by the `pp.highly_variable_genes` function (Scanpy v1.6.1) and used them as the input data for PCA. We first used the `StandardScaler` function of scikit-learn v0.24.0 [31] to scale the highly variable genes to unit variance to diminish the effects of expression abundance difference, then applied the `TruncatedSVD` function of scikit-learn v0.24.0 to obtain PCA embedding. The default component number of PCA was set as 50. The PCA embedding results were used for downstream UMAP visualization and Leiden clustering.

Data visualization

The two-dimensional distributions of cells/spots/beads within each dataset were visualized by Uniform Manifold Approximation and Projection (UMAP) [32] plots using the top 50 principal components (PCs) of the scRNA-seq datasets and the top 30 PCs of the spatial transcriptome datasets. UMAP was implemented via the `pp.neighbors` function (parameters: `n_neighbors=15`, `knn=True`, `use_rep='X_pca'` and `method='umap'`) followed by the `tl.umap` function (with default parameters) of Scanpy (v1.6.1). For the human kidney scRNA-seq dataset [22] and the scATAC-seq datasets [4, 6], the two-dimensional coordinates of cells were adopted from the original publications and used for UMAP plot construction.

Cell type annotation

For the scRNA-seq and scATAC-seq datasets, the identities of cells were characterized according to the original publications [4, 6, 21, 22, 28, 29]. For the spatial transcriptome datasets, unsupervised graph-based Leiden clustering algorithm [33] implemented by the `pp.neighbors` (parameters: `n_neighbors=15`, `knn=True`, `use_rep='X_pca'` and `method='umap'`) and `tl.leiden` functions (Scanpy v1.6.1) were used to cluster spots/beads into different groups according to their gene expression similarities. The resolution parameter for `tl.leiden` (Scanpy v1.6.1) was set as 0.3 for the Spatial-brain_sagitta dataset, set as 0.25 for the Spatial_brain_coronal dataset and set as 0.5 for the Slide-seqV2 spatial transcriptome dataset.

Enrichment score calculation

The enrichment score of a given gene set was calculated by the `tl.score_genes()` function in Scanpy v1.6.1.

Running time evaluation

The running time for each tested marker gene identification method was measured by the time module in Python. All methods were run on a 2.00GHz Intel Xeon E7-4830v4 central processing unit (CPU) with 512GB of RAM. Except for the MAST method, which by default uses multiple CPU cores, other methods were restricted to use one CPU core.

Code availability

COSG is available at <https://github.com/genecell/COSG> (Python) and <https://github.com/genecell/COSGR> (R).

Author contributions

X.W. and X.P. supervised the study. M.D. developed the algorithm, built the computational tools, and performed the analysis. X.W. and M.D. wrote the manuscript. All authors read and approved the final manuscript.

Key Points

- COSG utilizes cosine similarity to identify cell marker genes or genomic regions with better accuracy and greater cell-type specificity.
- COSG is a general method for cell marker gene identification across different data modalities, including scRNA-seq, scATAC-seq and spatially resolved transcriptome data.
- COSG is ultrafast for large-scale datasets.

Supplementary Data

Supplementary data are available online at <https://academic.oup.com/bib>.

Funding

This work was supported by the National Key Research and Development Program of China (2019YFA0802203), Natural Science Foundation of China (81790622 and 91940304), CAS Strategic Priority Research Program (XDA16020801) and Beijing Natural Science Foundation of China (Z200020) to X.-J. W.

References

1. Svensson V, Vento-Tormo R, Teichmann SA. Exponential scaling of single-cell RNA-seq in the past decade. *Nat Protoc* 2018;**13**(4): 599–604.
2. Wagner A, Regev A, Yosef N. Revealing the vectors of cellular identity with single-cell genomics. *Nat Biotechnol* 2016;**34**(11): 1145–60.
3. Ding J, Adiconis X, Simmons SK, et al. Systematic comparison of single-cell and single-nucleus RNA-sequencing methods. *Nat Biotechnol* 2020;**38**(6):737–46.
4. Pijuan-Sala B, Wilson NK, Xia J, et al. Single-cell chromatin accessibility maps reveal regulatory programs driving early mouse organogenesis. *Nat Cell Biol* 2020;**22**(4):487–97.
5. Corces MR, Buenrostro JD, Wu B, et al. Lineage-specific and single-cell chromatin accessibility charts human hematopoiesis and leukemia evolution. *Nat Genet* 2016;**48**(10):1193–203.
6. Granja JM, Klemm S, McGinnis LM, et al. Single-cell multiomic analysis identifies regulatory programs in mixed-phenotype acute leukemia. *Nat Biotechnol* 2019;**37**(12):1458–65.
7. Marx V. Method of the year: spatially resolved transcriptomics. *Nat Methods* 2021;**18**(1):9–14.
8. Stickels RR, Murray E, Kumar P, et al. Highly sensitive spatial transcriptomics at near-cellular resolution with slide-seqV2. *Nat Biotechnol* 2021;**39**(3):313–9.
9. Moffitt JR, Bambach-Mukku D, Eichhorn SW, et al. Molecular, spatial, and functional single-cell profiling of the hypothalamic preoptic region. *Science* 2018;**362**(6416):eaau5324.
10. Lähnemann D, Köster J, Szczurek E, et al. Eleven grand challenges in single-cell data science. *Genome Biol* 2020;**21**(1):31.
11. Stegle O, Teichmann SA, Marioni JC. Computational and analytical challenges in single-cell transcriptomics. *Nat Rev Genet* 2015;**16**(3):133–45.
12. Soneson C, Robinson MD. Bias, robustness and scalability in single-cell differential expression analysis. *Nat Methods* 2018;**15**(4):255–61.
13. Finak G, McDavid A, Yajima M, et al. MAST: a flexible statistical framework for assessing transcriptional changes and characterizing heterogeneity in single-cell RNA sequencing data. *Genome Biol* 2015;**16**(1):1–13.
14. Wolf FA, Angerer P, Theis FJ. SCANPY: large-scale single-cell gene expression data analysis. *Genome Biol* 2018;**19**(1):15.
15. Butler A, Hoffman P, Smibert P, et al. Integrating single-cell transcriptomic data across different conditions, technologies, and species. *Nat Biotechnol* 2018;**36**(5):411–20.
16. Haghverdi L, Lun ATL, Morgan MD, et al. Batch effects in single-cell RNA-sequencing data are corrected by matching mutual nearest neighbors. *Nat Biotechnol* 2018;**36**(5):421–7.
17. Ntranos V, Yi L, Melsted P, et al. A discriminative learning approach to differential expression analysis for single-cell RNA-seq. *Nat Methods* 2019;**16**(2):163–6.
18. Pratt JW. Remarks on zeros and ties in the Wilcoxon signed rank procedures. *J Am Stat Assoc* 1959;**54**(287):655–67.
19. Wilcoxon F. Individual comparisons by ranking methods. *Biometrics* 1945;**1**(6):80–3.
20. Squair JW, Gautier M, Kathe C, et al. Confronting false discoveries in single-cell differential expression. *Nat Commun* 2021;**12**(1):5692.
21. Hochgerner H, Zeisel A, Lönnerberg P, et al. Conserved properties of dentate gyrus neurogenesis across postnatal development revealed by single-cell RNA sequencing. *Nat Neurosci* 2018;**21**(2): 290–9.
22. Stewart BJ, Ferdinand JR, Young MD, et al. Spatiotemporal immune zonation of the human kidney. *Science* 2019;**365**(6460): 1461–6.
23. Wang C, Sun D, Huang X, et al. Integrative analyses of single-cell transcriptome and regulome using MAESTRO. *Genome Biol* 2020;**21**(1):198.
24. Han X, Zhou Z, Fei L, et al. Construction of a human cell landscape at single-cell level. *Nature* 2020;**581**(7808):303–9.
25. Reynolds G, Vegh P, Fletcher J, et al. Poised cell circuits in human skin are activated in disease. *bioRxiv preprint*, 2020. 2020.11.05.369363.
26. Litviňuková M, Talavera-López C, Maatz H, et al. Cells of the adult human heart. *Nature* 2020;**588**(7838):466–72.
27. Sungnak W, Huang N, Bécavin C, et al. SARS-CoV-2 entry factors are highly expressed in nasal epithelial cells together with innate immune genes. *Nat Med* 2020;**26**(5):681–7.
28. Almanzar N, Antony J, Baghel AS, et al. A single-cell transcriptomic atlas characterizes ageing tissues in the mouse. *Nature* 2020;**583**(7817):590–5.
29. Cao J, Spielmann M, Qiu X, et al. The single-cell transcriptional landscape of mammalian organogenesis. *Nature* 2019;**566**(7745): 496–502.
30. Stuart T, Srivastava A, Lareau C, et al. Multimodal single-cell chromatin analysis with Signac. *Nat Methods* 2021;**18**(11):1333–41.
31. Pedregosa F, Varoquaux G, Gramfort A, et al. Scikit-learn: machine learning in python. *J Mach Learn Res* 2011;**12**(85): 2825–30.
32. McInnes L, Healy J, Saul N, et al. UMAP: uniform manifold approximation and projection. *J Open Source Softw* 2018;**3**(29): 861.
33. Traag VA, Waltman L, van Eck NJ. From Louvain to Leiden: guaranteeing well-connected communities. *Sci Rep* 2019;**9**(1): 5233.