nature communications

Article

https://doi.org/10.1038/s41467-025-67890-3

CellScope: high-performance cell atlas workflow with tree-structured representation

Received: 3 February 2025

Accepted: 11 December 2025

Published online: 30 December 2025

Check for updates

Bingjie Li[1,2,9], Runyu Lin[1,9], Tianhao Ni[1,3,9], Guanao Yan[4,5], Mannix Burns[6], Jingyi Jessica Li[7,8] ✉ & Zhigang Yao[1] ✉

Single-cell sequencing enables comprehensive profiling of individual cells, revealing cellular heterogeneity and function with unprecedented resolution. However, current analysis frameworks lack the ability to simultaneously explore and visualize cellular hierarchies at multiple biological levels. To address these limitations, we present CellScope, a promising framework for constructing high-resolution cell atlases at multiple clustering levels. CellScope employs a two-stage manifold fitting process for gene selection and noise reduction, followed by agglomerative clustering, and integrates UMAP visualization with hierarchical clustering to intuitively represent cellular relationships simultaneously at multiple levels—such as cell lineage, cell type, and cell subtype levels. Compared to established pipelines such as Seurat and Scanpy, CellScope comprehensively improves clustering performance, visualization clarity, computational efficiency, and algorithm interpretability, while reducing dependence on hyperparameters across a multitude of single-cell datasets. Most importantly, it can reveal biological insights that other contemporary methods are unable to detect, thereby deepening our understanding of cellular heterogeneity and function, and potentially informing disease research.

The advent of single-cell sequencing has fundamentally changed our understanding of biology by providing an unprecedented look into the heterogeneity of biological systems at the individual cell level. Over the past decade, the increasing accessibility of single-cell technologies has led to a rise in the generation of large, comprehensive single-cell datasets—collectively known as cell atlases. By providing comprehensive high-resolution maps that identify, characterize, and spatially locate every cell type within an organism or tissue, cell atlases offer invaluable insights into cellular heterogeneity, interactions, and functions[1]. These detailed maps have the potential to revolutionize our understanding of normal development, aging, and disease pathogenesis, paving the way for new diagnostic, prognostic, and therapeutic strategies[2,3]. Consequently, many specialized atlases have emerged, including those focusing on neurodegenerative diseases-mapping cellular changes in Alzheimer's and Parkinson's disease tissues[4–6]. Others have focused on developmental biology, creating time-resolved atlases that track cellular differentiation during organ formation[7–9]. Cancer-specific atlases have also gained prominence, helping to delineate tumor micro-environments and identify new therapeutic targets[10,11]. With this increasing availability of cell atlases,

[1]Department of Statistics and Data Science, National University of Singapore, Singapore, Singapore. [2]Shanghai Institute for Mathematics and Interdisciplinary Sciences, Shanghai, China. [3]School of Mathematical Sciences, Fudan University, Shanghai, China. [4]Department of Statistics, University of California, Los Angeles, Los Angeles, CA, USA. [5]Department of Computational Mathematics, Science and Engineering, Michigan State University, East Lansing, MI, USA. [6]Department of Molecular, Cell and Developmental Biology, University of California, Los Angeles, Los Angeles, CA, USA. [7]Biostatistics Program, Public Health Sciences Division, Fred Hutchinson Cancer Center, Seattle, WA, USA. [8]Department of Biostatistics, University of Washington, Seattle, WA, USA. [9]These authors contributed equally: Bingjie Li, Runyu Lin, Tianhao Ni. ✉e-mail: lijy03@fredhutch.org; zhigang.yao@nus.edu.sg

the ability to extract biologically meaningful information from these datasets is paramount to the progression of our knowledge of system-specific cellular dynamics and disease mechanisms.

Despite remarkable progress in the single-cell field, existing computational methodologies face several limitations that hinder their ability to fully capture the complexity of single-cell data. Commonly used pipelines, such as Seurat[12], Scanpy[13], and SnapATAC[14], rely on conventional unsupervised learning techniques that may not adequately handle the high-dimensionality, sparsity, and noise inherent in single-cell datasets. The highly variable genes (HVG) selection methods employed by both Seurat and Scanpy, as unsupervised approaches based on variance-to-mean ratio calculations, rely purely on statistical dispersion without considering the biological relevance of gene expression patterns across different cell types, and thus cannot consistently identify genes that are truly informative for characterizing cellular heterogeneity[15]. Similarly, clustering algorithms like Louvain[16] and Leiden[17], which optimize community detection through modularity maximization, lack hierarchical structure to capture nested relationships between cell types and cannot effectively use resolution changes to control the merging and further subdivision of cell types[18]. Moreover, popular visualization techniques like t-SNE[19] and UMAP[20] have limitations in representing the global structure and hierarchical organization of cells, often emphasizing local similarities at the expense of preserving the overall topology[21].

Recent advances in single-cell genomics have led to growing recognition of the low-dimensional nature of single-cell data[22,23]. This characteristic can be understood from two perspectives. Firstly, despite the vast number of genes measured in single-cell experiments, only a small subset is typically informative for distinguishing cell types. The majority of genes are housekeeping genes, which maintain basic cellular functions and exhibit relatively constant expression across cell types. Studies such as[15] have demonstrated that focusing on fewer, highly informative genes can lead to improved visualization and analysis outcomes[24]. Secondly, due to the interconnected nature of genes, single-cell data tends to occupy a low-dimensional manifold within the high-dimensional gene expression space[25]. As such, many state-of-the-art single-cell clustering frameworks have begun incorporating this concept of manifolds[26,27]. In particular, manifold fitting[28,29] represents a conceptual framework that aims to preserve data structure while offering high interpretability and theoretical backing, thus emerging as an innovative approach for dimensionality reduction. Building on their previous work[29], Yao et al. developed scAMF[30]—the first framework to implement this manifold fitting concept for single-cell analysis.

Here, we introduce CellScope, a promising method for constructing multi-level, high-resolution cellular atlases. By leveraging manifold fitting and neighborhood graph-based aggregative clustering, CellScope addresses three key challenges in single-cell analysis:

(1) inadequate gene selection and single-level marker gene characterization that fails to capture genes dynamically across cell type subdivisions;
(2) limited clustering resolution and hierarchical structure that cannot support nested organization;
(3) inability to generate tree-structured visualizations that integrate cellular trees with cellular atlases.

To overcome these limitations, CellScope integrates four core innovations: intelligent gene selection that separates signal from noise spaces, precise delineation of similar cell subpopulations through manifold-based denoising, dynamic characterization of cellular landscapes at multiple resolutions, and multi-level functional analysis through dynamic "molecular identity" classification.

We conducted extensive validation across 36 datasets covering various species, organs, and sequencing modalities, demonstrating CellScope's exceptional performance. Notable achievements include the identification of Oligodendrocyte subpopulations and the simultaneous characterization of cell types and health status in COVID-19 patients—discoveries that existing methods like Seurat and Scanpy were unable to detect. Importantly, CellScope achieves these results with exceptional speed, parameter-free operation, and high interpretability, thereby opening new avenues for understanding cellular diversity in complex biological systems.

Results

Overview of cellScope workflow

CellScope uses manifold fitting to model single-cell data, addressing intrinsic complexity and noise to derive crucial biological insights. CellScope assumes that the true biological structure of single-cell data lies on a low-dimensional manifold[31]. This manifold represents the intrinsic, lower-dimensional structure of gene expression that captures the genuine relationships between cells, including cellular states and subtypes. However, the observed single-cell data does not directly reflect this manifold due to two types of noise. The first type of noise refers to the expression of housekeeping genes, which are crucial for basic cellular functions but, due to their ubiquitous expression, do not contribute to distinguishing cell populations. We thus define "*noise space*" to denote the space of housekeeping genes and "*signal space*" to represent the remaining gene expression profiles that reflect cell type differences. The second type of noise lies in this signal space and represents technical noise due to mRNA loss, inefficient molecular capture, and sequencing errors[32]. This stochastic noise may distort the true expression patterns of the marker genes that are key to distinguishing between cell types and states. Together, these two types of noise combine with the underlying biologically meaningful manifold to constitute the observable single-cell gene expression matrix (Fig. 1a).

To recover the essential low-dimensional manifold and improve the quality of downstream analyses, CellScope employs a two-stage manifold fitting process (Fig. 1b). The first stage in our manifold fitting approach aims to mitigate the noise introduced by ubiquitous housekeeping genes, which are irrelevant to cell classification, while preserving critical genes for further analysis (Methods A). We base this process on a widely accepted assumption in manifold learning and translate it to a biological context: low-dimensional representations of individual cells that belong to different cell types lie on distinct submanifolds[33]. These cell type submanifolds are characterized by a high density of cells and are separated from one another by regions of low cell density[34].

Leveraging this principle, CellScope selects multiple sets of distant high-density cells, termed "manifold seeds", along with their neighboring cells, designated as "highly reliable cliques" (Fig. 1c). These cliques originate from multiple separate cell types and help distinguish between noise and signal spaces. Features in the signal space exhibit low variance within the same clique but high variance between different cliques, while features in the noise space lack this property. By exploiting this distinction, CellScope filters out most noise while preserving key genes for determining cell identity.

The second stage ensures proper stratification of different cell types by assigning cells residing in low-density regions, which may represent transitional cell states or have higher levels of technical noise, to the nearest cell type submanifold (Methods B). This denoising stage refines the representation of each cell type, emphasizing genuine biological signals over technical artifacts, and better reflects the underlying cellular heterogeneity.

After manifold fitting, CellScope constructs a cell-to-cell neighborhood similarity graph, where cells with more similar gene expression profiles are assigned higher similarity. Based on this graph, CellScope then performs agglomerative clustering (Methods C).

https://doi.org/10.1038/s41467-025-67890-3

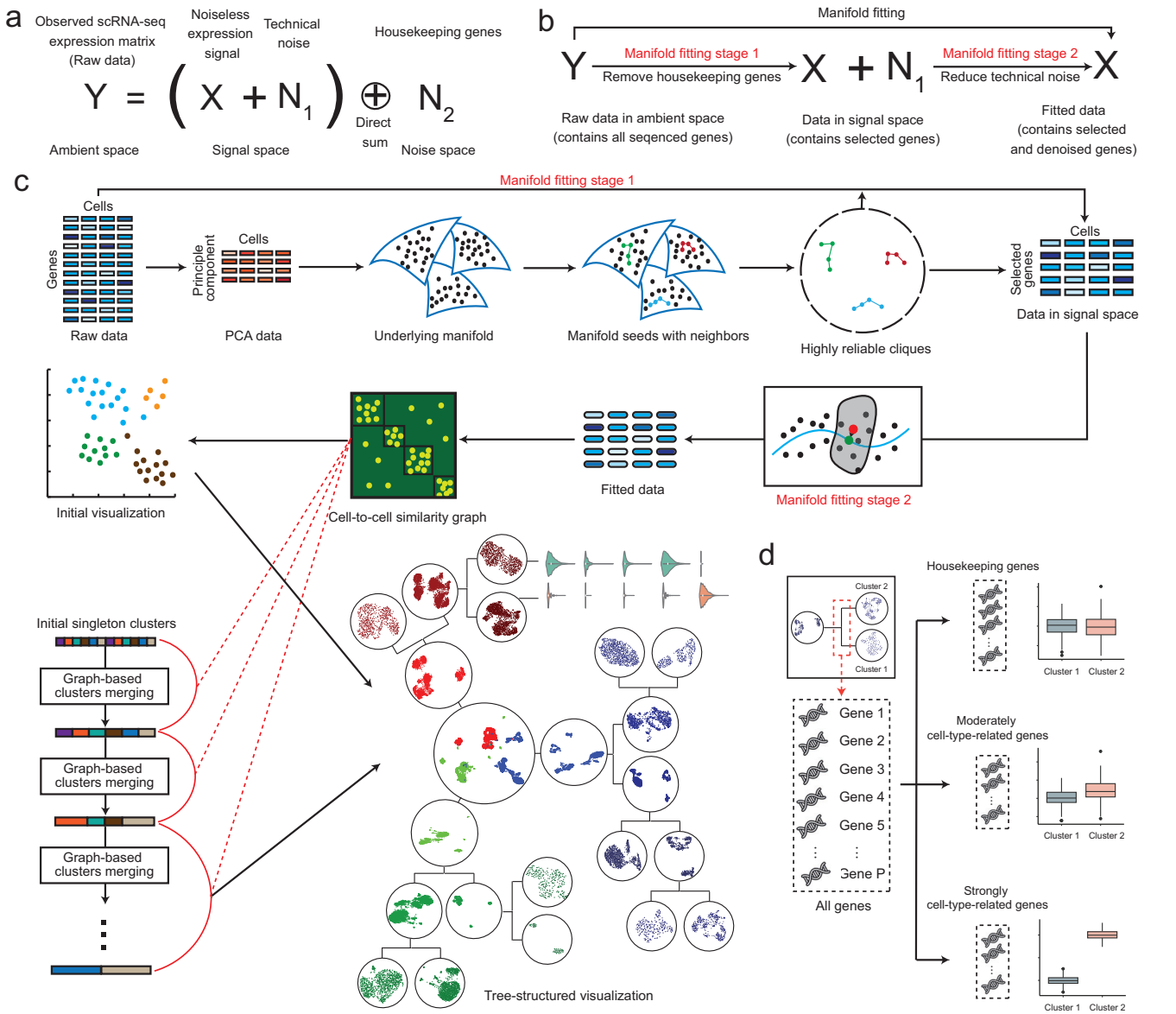


Fig. 1 | CellScope workflow and its underlying manifold modeling strategy. **a** Mathematical modeling of noise in single-cell data. **b** The two stages of manifold fitting and their purpose. **c** Overview of CellScope workflow. CellScope enhances cellular data analysis through a two-stage manifold fitting process. First, it identifies "manifold seeds" and "highly reliable cliques" in the PCA-reduced space to effectively distinguish signal from noise, thereby filtering out housekeeping gene effects. Next, it reduces technical noise by projecting low-density cells onto high-density regions. Subsequently, CellScope constructs a neighborhood similarity graph and performs agglomerative clustering, iteratively merging similar clusters for precise hierarchical classification. Finally, the method generates two key visualizations: a UMAP representation of the manifold-fitted data and a tree-structured visualization combining UMAP with hierarchical clustering. In the "Manifold fitting stage 1" panel, color intensity in the "Raw data", "PCA data", "Data in signal space", and "Fitted data" represents matrix values. Black dots on the manifolds represent individual cells. In the "Manifold fitting stage 2" panel, the blue line represents the fitted manifold, red dots indicate cells to be fitted, green dots show the fitted cell positions, and the gray shaded region denotes the neighborhood of cells used for fitting. In the hierarchical clustering dendrograms and tree-structured visualization, colors distinguish different cell clusters. **d** Each cluster division in the tree-structured visualization produces three unique types of genes: housekeeping genes with minimal variance between classes, moderately cell-type-related genes with partially significant differences, and strongly cell-type-related genes with high variance between classes.

Starting with each cell as a cluster, the algorithm iteratively merges the most similar clusters until no two clusters exhibit significant similarity, yielding precise and biologically meaningful classifications.

An innovative aspect of CellScope is its ability to generate an informative tree-structured visualization (Methods D) that integrates UMAP[20] and hierarchical clustering. In addition to an initial UMAP visualization of the manifold-fitted data that provides an intuitive representation of complex cellular relationships, CellScope provides the tree-structured visualization that depicts the hierarchical relationships between cell types, illustrating how different populations emerge, branch, and specialize. By annotating the gene expression differences that drive the emergence of each cell cluster, researchers can gain insight into key regulatory genes and pathways involved in cell fate decisions, development, and functional specialization. Based on the tree-structured visualization, CellScope introduces an innovative multilevel gene identity system, referred to as dynamic "molecular identity". By analyzing the expression differences of genes among different cell clusters within the hierarchical levels of clustering, CellScope classifies genes into distinct identities, including housekeeping genes, moderately cell-type-related genes, and strongly

cell-type-related genes (Fig. 1d and Methods E). By evaluating changes in gene identities across these clustering hierarchies, CellScope transcends the traditional binary classification of genes as either marker or non-marker genes.

CellScope differs fundamentally from existing frameworks through its comprehensive mathematical approach. While Seurat employs variance-mean relationship models for gene selection, linear PCA for dimensionality reduction, and flat Louvain clustering, and Scanpy uses standardized variance-based HVG selection with similar linear PCA and Leiden clustering, CellScope introduces a manifold-based framework with three key innovations. First, it adaptively identifies signal genes through manifold seeds detection rather than parameter-dependent selection. Second, it preserves both local and global cellular relationships via manifold fitting instead of linear subspace assumptions. Third, it constructs multi-level hierarchical structures through agglomerative clustering rather than single-level partitioning. Detailed mathematical comparisons between CellScope, Seurat, and Scanpy are provided in Supplementary Note 1.1 and Supplementary Table 1. While CellScope and scAMF[30] share the conceptual foundation of manifold fitting, their specific mathematical implementations and biological applications are fundamentally distinct. scAMF focuses on general technical noise reduction through a uniform manifold hypothesis, whereas CellScope introduces a biology-driven dual-noise model. Specifically, CellScope explicitly distinguishes housekeeping gene noise from technical noise, implements cell-type-aware submanifold detection using composite metrics, and incorporates ANOVA-based statistical gene selection from biologically reliable cliques. These innovations transform manifold fitting from a general clustering enhancement technique into a specialized biological discovery platform capable of hierarchical validation and dynamic molecular characterization across multiple levels of cellular organization. Detailed comparisons between CellScope and scAMF are provided in Supplementary Note 1.2.

CellScope demonstrates superior performance in cell clustering and gene selection

We evaluated CellScope's performance in cell clustering using 36 distinct scRNA-seq datasets with known cell types (Methods G and Supplementary Tables 2–3). These datasets cover various human and mouse tissues—including brain, pancreas, embryos, and immune cells—and range widely in size (90 to 265,767 cells) and complexity (3 to 20 cell types). Each dataset includes gold-standard cell type labels determined through methods like cell morphology and marker gene expression. We compared CellScope against two widely used single-cell analysis methods (Seurat[35] and Scanpy[13]) and three recent methods (scLEGA[36], scDCCA[37], and CellBRF[38]) with details in Methods F. True cell type labels were used only for post-hoc evaluation. All clustering results shown below are based on analyses performed on the Google Colab platform (44-core CPU, 150 GB RAM) to ensure consistency (Methods H).

CellScope achieves the best cell clustering performance across all datasets regarding accuracy, robustness, and computational efficiency. To quantify clustering performance, we used multiple clustering evaluation metrics: adjusted rand index (ARI)[39], the clustering accuracy (ACC)[40], the normalized mutual information (NMI)[41], and jaccard index (JI)[42], where higher values indicate better clustering performance (Definitions of these metrics are provided in the Supplementary Note 1.3). Among the six methods tested, CellScope, Seurat, and Scanpy successfully completed clustering across all 36 datasets. In contrast, due to computational limitations, scLEGA, scDCCA, and CellBRF could only be executed on datasets containing up to 50,000, 75,000, and 25,000 cells, respectively. As shown in Fig. 2a, CellScope achieved the highest overall average ARI of 0.88 with the lowest standard deviation, ranking first on 32 out of the 36 datasets and second on 3 others, significantly outperforming all other methods (Fig. 2b). Seurat and Scanpy attained lower average ARIs of 0.65 and 0.68, respectively. Among the recent methods, scLEGA, scDCCA, and CellBRF achieved average ARIs of 0.54, 0.74, and 0.49 on datasets where they could be successfully applied. Wilcoxon signed-rank tests further confirmed CellScope's statistically significant superiority, with all p-values against the other five methods being less than 10^{-5}. Similar advantages were observed across other clustering evaluation metrics (see Supplementary Fig. 7 and Tables 4–7). Beyond its accuracy, CellScope also demonstrated competitive computational efficiency across three dataset-size regimes (Fig. 2c, Supplementary Table 8). Among the three methods that could handle all dataset sizes (CellScope, Scanpy, and Seurat), CellScope consistently achieved the fastest or among the fastest runtimes across small (<3000 cells), medium (3000–50,000 cells), and large (>50,000 cells) datasets. The other three methods (scLEGA, scDCCA, and CellBRF) showed computational limitations, with scLEGA and CellBRF failing on large datasets and scDCCA exhibiting prohibitively long runtimes.

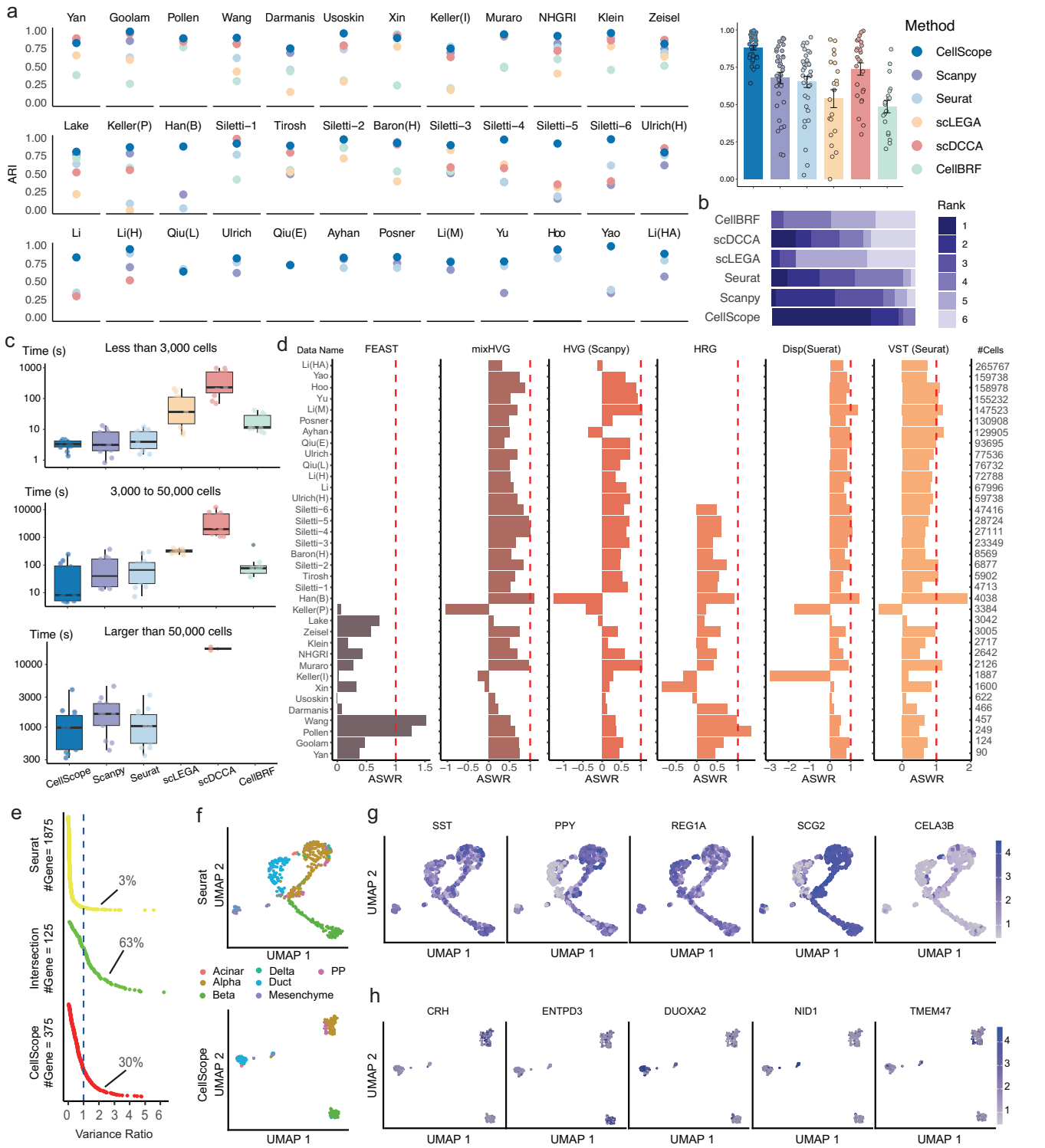
To assess CellScope's stability across different computing environments, we conducted additional tests on two memory-constrained personal computers—a MacBook (8-core CPU, 16GB RAM) and an iMac (10-core CPU, 16GB RAM) (Methods H). Since other methods could not complete all datasets on these platforms, we compared only CellScope and Scanpy. Results show that CellScope typically achieved faster runtimes than Scanpy, demonstrated memory advantages on memory-constrained platforms while performing comparably on high-memory platforms (never exceeding twice Scanpy's memory usage), and exhibited significantly more stable clustering performance than Scanpy (Supplementary Fig. 15 and Tables 23–25).

Gene selection is crucial for clustering results. CellScope exhibits superior performance in this area compared to other methods, including Disp (Seurat)[43], VST (Seurat)[35], HVG (Scanpy)[13], mixHVG[44], FEAST[45], and HRG[46]. We used multiple evaluation metrics to assess the effectiveness of gene selection methods, including Average Silhouette Width (ASW)[47], Variance Ratio, Cell-type Local Inverse Simpson Index[48], KNN Classification Accuracy, and Neighborhood Purity (definitions provided in Supplementary Note 1.5). CellScope consistently achieved higher ASW values on the majority of datasets (Fig. 2d), while other metrics further confirmed its superiority in gene selection effectiveness (Supplementary Tables 9–14). One-sided Wilcoxon signed-rank tests on these gene selection metrics validated CellScope's superiority in gene selection with all p-values < 0.01 (Supplementary Table 15).

To further validate the effectiveness of CellScope's gene selection strategy beyond direct metric comparisons and statistical tests, we combined the gene selection outputs from Seurat and Scanpy with CellScope's graph-based clustering module and compared the results to those of the CellScope pipeline. The complete CellScope workflow achieved the highest average ARI of 0.88, outperforming the hybrid versions using Scanpy (average ARI = 0.75) and Seurat (average ARI = 0.77) gene selection, which themselves performed better than the original Scanpy (average ARI = 0.68) and Seurat (average ARI = 0.65) pipelines (Supplementary Fig. 8a, b). These results confirm that CellScope's gene selection strategy contributes significantly to its superior clustering performance.

Figure 2f compares the visualizations produced by CellScope and Seurat using human pancreatic cells from Wang et al.[49]. Seurat's results incorrectly suggest that the three cell types—Alpha, Duct, and Beta—are interconnected and indistinguishable, whereas CellScope accurately achieves clear separation among the cell types. This difference is largely attributed to CellScope's manifold fitting-based gene selection strategy. As shown in Fig. 2e, only 125 genes overlap between Seurat's 2000 and CellScope's 500 selected genes, with 63% of these shared genes exhibiting significant variation (Variance Ratio >1, defined as the ratio between a gene's inter-cell-type and intra-cell-type expression variance, indicating its power to distinguish cell types). More notably,

https://doi.org/10.1038/s41467-025-67890-3



30% of CellScope-specific genes showed strong differential expression (Variance Ratio >1) between cell classes, whereas only 3% of Seurat-specific genes exhibited such pronounced differences. This indicates that CellScope captures high-quality marker genes and discovers meaningful high-variance genes that Seurat overlooked. For instance, the expression patterns of Seurat-selected genes (e.g., *SST*, *REG1A*) show minimal cell type specificity, resulting in poor cluster separation. In contrast, CellScope uniquely identifies genes (e.g., *CRH*) that are not selected by Seurat and exhibit highly specific expression patterns, facilitating more accurate cell type differentiation (Fig. 2g, h). A comparison between CellScope and Scanpy gene selection can be seen in Supplementary Fig. 8c–f, and results for all other benchmark datasets are available in Supplementary Tables 20–22.

CellScope enhances the ability to distinguish similar cell types, detect rare cell types, and perform multi-level clustering

CellScope demonstrates superior performance in distinguishing similar cell types and detecting rare cell populations. This was exemplified using the human brain cell dataset from NHGRI[50] and the mouse pancreatic cell dataset from Keller (P)[51]. CellScope's visualization results show clearer separation of cell types compared to popular methods such as Scanpy and Seurat (Fig. 3a, b). For instance, in the

https://doi.org/10.1038/s41467-025-67890-3

Fig. 2 | Performance of CellScope on 36 benchmark datasets. a Clustering performance evaluation of CellScope, Scanpy, Seurat, scLEGA, scDCCA, and CellBRF using the Adjusted Rand Index (ARI). The left 36 panels display ARI values for individual datasets, while the top-right panel summarizes the mean ARI with standard errors indicated by vertical bars (CellScope = 0.88 ± 0.014, Scanpy = 0.68 ± 0.038, Seurat = 0.65 ± 0.038, scLEGA = 0.54 ± 0.060, scDCCA = 0.74 ± 0.041, CellBRF = 0.49 ± 0.042). Each point represents the ARI for a specific dataset. **b** The rank distribution based on clustering performance using ARI. **c** Execution time comparison across benchmark datasets of varying sizes. Box-plots display the runtime (in seconds) for six methods across datasets stratified by cell number: less than 3000 cells (small datasets), 3000 to 50,000 cells (medium datasets), and more than 50,000 cells (large datasets). Boxplots display the 25%, 50% (median), and 75% percentiles, where whiskers denote 1.5 times the interquartile range. Each dot represents the runtime on a specific dataset. The sample sizes used for the boxplots are as follows: small datasets (n = [11, 11, 11, 11, 11, 11]), medium datasets (n = [12, 12, 12, 10, 11, 8]), and large datasets (n = [13, 13, 13, 0, 2, 0]). Missing boxes indicate method failure or timeout. **d** Gene selection performance of CellScope compared to six gene selection methods across benchmark datasets. The X-axis depicts the Average Silhouette Width Ratio (ASWR), with CellScope's ASW set as the baseline of 1 (red dashed line). Other methods' ASWs are expressed as multiples of CellScope's. ASWR >1 indicates superior performance to CellScope; ASWR <1 indicates inferior performance. Absent bars indicate method failure or timeout. **e** Gene selection performance comparison of CellScope and Seurat evaluated by Variance Ratio on the Wang dataset. **f** Visualizations produced by CellScope (Bottom) and Seurat (Top) of human pancreatic cells from Wang et al.[49], colored by the true cell types. **g** Visualization of the distribution of marker genes selected by Seurat. Darker colors represent higher gene expression. **h** Visualization of the distribution of marker genes selected by CellScope. Darker colors represent higher gene expression.

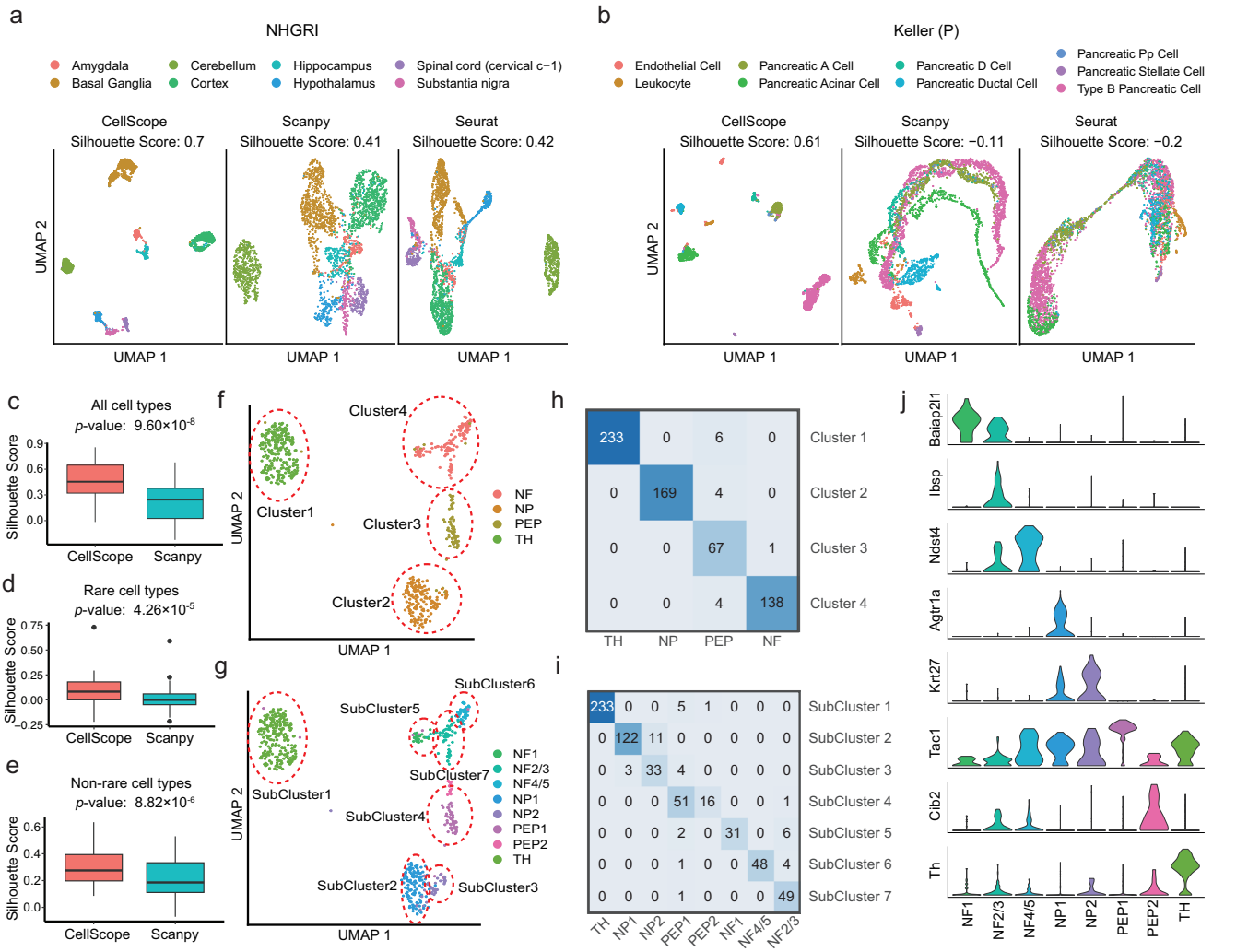


Fig. 3 | The performance of CellScope in distinguishing similar cell types, detecting rare types, and performing multi-level clustering. a UMAP visualizations of human brain cells (NHGRI dataset) generated by CellScope, Seurat, and Scanpy. Colors represent different brain regions. **b** UMAP visualizations of mouse pancreatic cells (Keller (P) dataset) generated by CellScope, Seurat, and Scanpy. Colors represent different cell types. **c** Comparison of visualization quality using Silhouette coefficients for all cell types across 36 benchmark datasets. One-sided Wilcoxon signed-rank test (alternative: greater) comparing CellScope versus Scanpy, $p = 9.60 \times 10^{-8}$. **d** Silhouette coefficients for rare cell types (defined as < 5% of total population) comparing CellScope and Scanpy across 36 datasets (one-sided Wilcoxon signed-rank test, alternative: greater, $p = 4.26 \times 10^{-5}$). **e** Silhouette coefficients for non-rare cell types (≥5% of total population) comparing CellScope and Scanpy across 36 datasets (one-sided Wilcoxon signed-rank test, alternative: greater, $p = 8.82 \times 10^{-6}$). **c**–**e** The sample size for each boxplot is $n = 36$. Boxplots display the 25%, 50% (median), and 75% percentiles, where the whiskers extend to the most extreme data points within 1.5 times the interquartile range. **f** Hierarchical clustering visualization of mouse lumbar sensory neurons (Usoskin dataset) showing major cell types at resolution level 2. **g** Extended hierarchical clustering of the same dataset showing cell subtypes at resolution level 5. **h** Confusion matrix comparing CellScope's level-2 clustering results with annotated major cell types. Perfect classification shown by diagonal values. **i** Confusion matrix comparing CellScope's level-5 clustering results with annotated cell subtypes. **j** Expression heatmap of key marker genes identified by CellScope for different cell types and subtypes in the Usoskin dataset.

NHGRI dataset, Scanpy and Seurat could only distinguish the Cerebellum from other brain regions, while the remaining cell types exhibited mixed clustering without clear boundaries. In contrast, CellScope's results successfully separated Basal Ganglia, Cerebellum, and Cortex from other cell types. Furthermore, Amygdala and Hippocampus clustered together as one cluster, and Hypothalamus, spinal cord, and Substantia nigra formed another, with clear boundaries maintained between each cell type within these clusters. Quantitative analysis revealed that CellScope achieved significantly higher Silhouette Scores for nearly all cell types in the NHGRI dataset compared to Scanpy or Seurat (Supplementary Fig. 9a). The analysis of the Keller (P) dataset further demonstrated CellScope's robust ability to recognize both non-rare and rare cell types. For instance, non-rare cell types such as type B pancreatic cells, comprising approximately 40% of the total population, formed distinct clusters. Meanwhile, rare cell types, including leukocytes, pancreatic PP cells, and pancreatic stellate cells, representing 3.6%, 2.1%, and 1.4% of the population, respectively, were distinctly separated (Supplementary Fig. 9b).

Through systematic analysis of all 36 datasets, CellScope demonstrated significant superiority over Scanpy for visualization Silhouette Scores (Fig. 3c), with the difference statistically confirmed by Wilcoxon signed-rank test ($p = 9.6 \times 10^{-8}$). To further quantify its effectiveness in identifying both non-rare and rare cell types, we defined rare cell types as those comprising less than 5% of the total population and non-rare types as those comprising 5% or more. We then calculated their respective visualization Silhouette Scores (see Supplementary Note 1.4 for detailed calculation). The analysis revealed that CellScope achieved high and stable visualization Silhouette Scores for rare cell types (Fig. 3d, $p = 4.26 \times 10^{-5}$), while demonstrating greater advantages for non-rare cell types (Fig. 3e, $p = 8.82 \times 10^{-6}$). To validate the robustness of these findings, we performed comprehensive sensitivity analyses using multiple thresholds (2%, 10%, 15%). Across all threshold conditions, CellScope maintained consistently superior performance in identifying rare cell populations, while performing as well as or better than Scanpy for non-rare cell identification (Supplementary Fig. 10). These results collectively highlight CellScope's unique capability to resolve low-abundance and high-abundance cell populations with high efficacy.

Additionally, CellScope demonstrated advanced multi-level clustering capabilities by analyzing the mouse lumbar cells from Usoskin dataset[52], which contains four major cell types and a total of eight subtypes. Using our hierarchical clustering, we first mapped major cell types at the second clustering level (Fig. 3f, h), successfully separating tyrosine hydroxylase containing (TH), neurofilament containing (NF), peptidergic nociceptors (PEP), and non-peptidergic nociceptors (NP) cell types. For subtypes without clear boundaries, we extended the clustering process (Fig. 3g, i) to accurately identify NP and NF subtypes. This superior performance can be attributed to CellScope's gene selection strategy, which identifies genes with subtype-specific expression patterns (Fig. 3j). For instance, CellScope selects *Tac1* and *Th* as distinctive markers based on their specific expression in PEP1 and TH cells, respectively. Additionally, CellScope identified *Agtr1a* based on its preferential expression in NP1 neurons. This gene selection approach, combined with multi-level clustering, enables accurate identification of cell types and subtypes while preserving the biological relationships between cell populations. This level of precision could not be achieved by Seurat and Scanpy (Supplementary Fig. 11).

CellScope's tree-structured visualizations refine characterization of brain cell atlases

CellScope's tree-structured visualization effectively displays the hierarchical relationships between cell types and their subtypes. To demonstrate, we implemented CellScope with a dataset from the red nucleus within the human midbrain[53], designated Siletti-1. First, CellScope's tree-structured visualization identifies the majority of previously annotated cell types in the Siletti-1 dataset. Specifically, CellScope categorized cells into nine distinct classes and identified nearly all superclusters previously reported in ref. 53. Notably, CellScope successfully identified Fibroblasts comprising only 26 cells (Fig. 4a).

Second, our analysis revealed that Oligodendrocytes (OLs) were further differentiated into two subtypes, designated as Oligodendrocyte1 (OL1) and Oligodendrocyte2 (OL2), containing 541 and 1592 cells, respectively. To elucidate the hierarchical expression patterns of key marker genes distinguishing these two subtypes, Fig. 4b, c highlights the expression distributions of five differentially expressed genes across three clustering levels: Cluster, SubCluster, and SubSubCluster. Specifically, the high expression of the OL1 marker gene *RBFOX1* indicates that these cells have reached a terminally differentiated[53]. In contrast, the high expression of the OL2 marker gene *OPALIN* reflects active myelination, suggesting that OL2 cells are undergoing active differentiation. Meanwhile, the low expression of *OPALIN* in OL1 further supports the notion that OL1 cells have reached terminal differentiation[54]. Additionally, *CTNND2*[55] plays a critical role in cell adhesion and synapse formation, while Laminin-2, encoded by *LAMA2*[56], regulates the spreading of OLs and myelination in the central nervous system via the integrin signaling pathway. Therefore, the cell population with high expression of these genes (OL2) exhibits enhanced interaction with axons, further indicating active myelination processes[57].

Third, CellScope's multi-layer cell clustering provides an innovative perspective to study the dynamic role of marker genes. In other words, CellScope assigns each gene a new dynamic "molecular identity" depending on whether it is solely unique to one layer of clustering or continues to function as a marker across all layers of clustering. In Siletti-1, CellScope implements a three-level hierarchical clustering system, referred to as Clusters, Subclusters, and SubSubclusters, which progressively identify homogeneous cell groups with increasing resolution (Fig. 4a). By categorizing genes into three dynamic identities—housekeeping genes (HG), moderately cell-type-related genes (MCTRG), and strongly cell-type-related genes (SCTRG)—based on their significance across clustering levels (see Methods E for details), we visualized the relationships between these gene identities using a Sankey diagram (Fig. 4d). Additionally, the overlaps and transitions of these gene identities across the three clustering levels were analyzed (Fig. 4e). As the clustering resolution increases, the number of housekeeping genes gradually increases, while the number of SCTRGs and MCTRGs gradually decreases. This phenomenon reflects the changing roles of genes at different clustering levels. Specifically, as the resolution increases, many genes transition from SCTRGs or MCTRGs to housekeeping genes. During cell differentiation, SCTRGs and MCTRGs are typically involved in establishing cell-specific functions or characteristics, especially during the early and middle stages of development. As cells progress into maturity, more housekeeping genes are activated, indicating that the cells have entered a phase focused on maintaining stable functions, such as protein synthesis and metabolism, which are essential for basic cell maintenance and biological processes.

We used three flow patterns as examples (Flow 6 HG-SCTRG-SCTRG, Flow 4 HG-MCTRG-HG, and Flow 19 SCTRG-HG-HG) and identified genes *RBFOX1* in Flow 6, *PPM1H* in Flow 4, and *PRANCR* in Flow 19, which serve as marker genes exclusively at the SubSubCluster, SubCluster, and Cluster levels, respectively (see Fig. 4f, g). Specifically, in Flow 6, *RBFOX1* is widely expressed in the nervous system and regulates various alternative splicing events related to neural development and maturation, including transcription factors, splicing factors, and synaptic proteins[58]. *RBFOX1* shows high expression levels in both Oligodendrocytes (OLs) and Splatter-type cells, while showing minimal expression differences between Oligodendrocyte precursor cells (OPCs) and Oligodendrocytes. This is likely because OPCs are the

https://doi.org/10.1038/s41467-025-67890-3

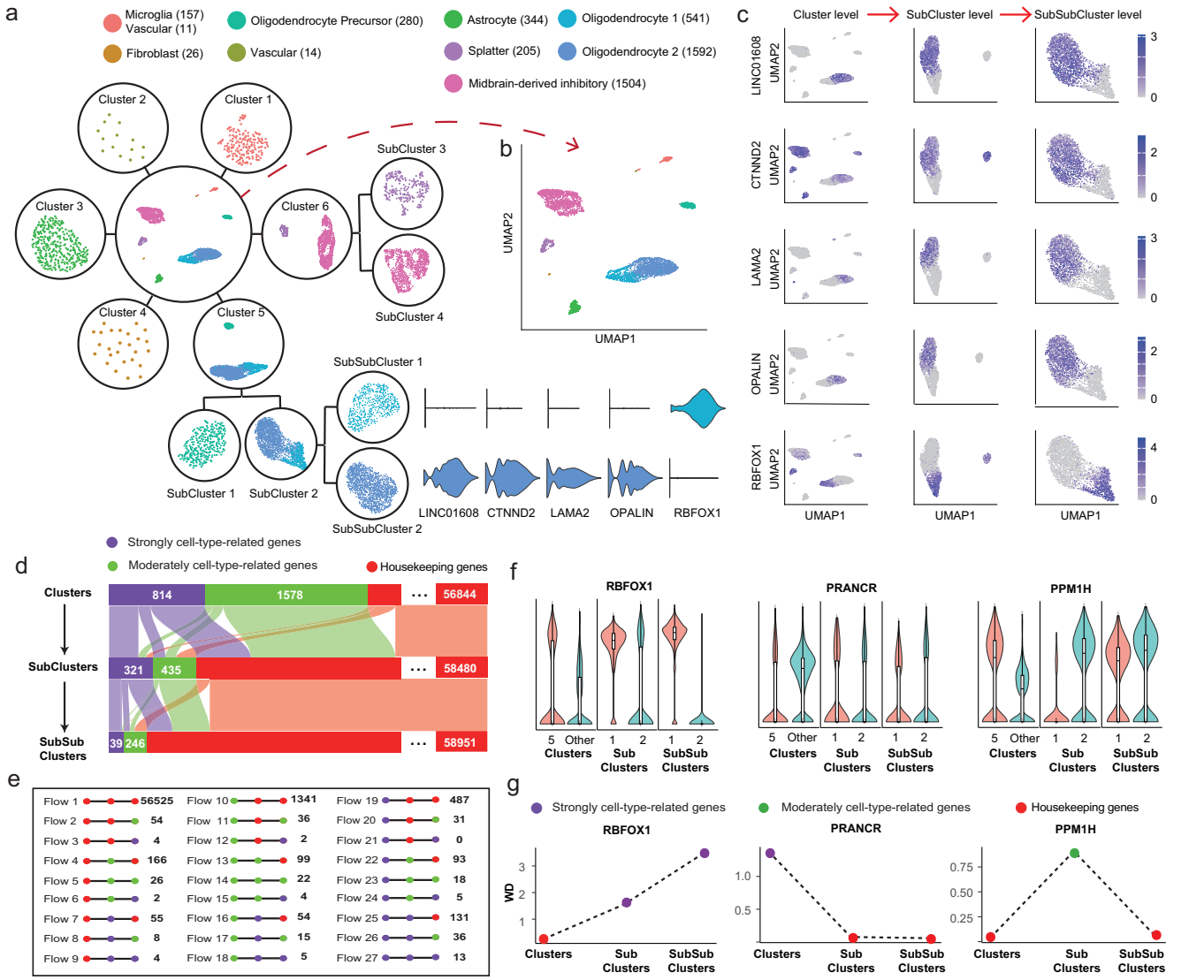


Fig. 4 | Tree-structured analysis of human midbrain red nucleus cells reveals hierarchical gene expression patterns. a CellScope hierarchical clustering tree of human midbrain red nucleus dataset (Siletti-1) showing three resolution levels: Clusters, SubClusters, and SubSubClusters. The number in parentheses indicates the cell count within each cell category. **b** UMAP visualization of the same dataset with cells colored by cluster assignments from **a** by CellScope. **c** Expression heatmaps of five marker genes across three hierarchical levels, demonstrating level-specific expression patterns. **d** Sankey diagram showing transitions of gene classifications (strongly cell-type-related, moderately cell-type-related, and housekeeping genes) across the three clustering levels. Numbers indicate gene counts in each category. **e** Distribution of genes across 27 possible classification transition patterns between clustering levels. **f** Violin plots comparing expression levels of three representative genes across clustering levels: RBFOX1 (transition pattern 6), PRANCR (pattern 19), and PPM1H (pattern 4). Boxplots display the 25%, 50% (median), and 75% percentiles, where the whiskers extend to the most extreme data points within 1.5 times the interquartile range. **g** Wasserstein distances quantifying expression differences between selected cluster pairs for the three representative genes. Lower values indicate more similar expression distributions. Comparisons shown for: Cluster 5 vs. other clusters, and pairwise comparisons within Subclusters and SubSubClusters.

direct precursor cells of OLs, and during this developmental transition, *RBFOX1* primarily supports basic cell development, differentiation, and gene regulation[59], maintaining developmental continuity. As mentioned earlier, there is a significant expression difference between the two OL subtypes, OL1 and OL2, due to their distinct differentiation states.

In Flow 4, gene enrichment analysis using ClusterProfiler[60] reveals that *PPM1H* is enriched in functions related to dephosphorylation and protein regulation. During OL differentiation, *PPM1H* regulates specific protein dephosphorylation events, contributing to myelin protein synthesis. Interestingly, the expression differences of *PPM1H* between different OL subtypes are minimal, likely because mature OLs need to maintain a stable signaling environment for myelin formation, which results in consistent expression of *PPM1H* across different OL subtypes. In Flow 19, *PRANCR*, a long non-coding RNA known to regulate keratinocyte proliferation and cell cycle progression[61], shows low expression in Cluster 5, particularly in OPCs and OLs. This is consistent with its established role in cell proliferation rather than differentiation. Since myelinating cells are specialized and post-mitotic, the low expression of *PRANCR* in these cells aligns with their specialized function in myelination[62].

To further validate CellScope's capability in identifying cellular subtypes within brain cell datasets, we applied it to two distinct datasets: a human thalamic dataset from the same study as Siletti-1[53] and a mouse primary motor cortex dataset[63]. For the human thalamic dataset, CellScope not only successfully resolved nearly all superclusters but also precisely identified two distinct subtypes of Oligodendrocytes. Notably, these subtypes exhibited marker genes similar to

those of Oligodendrocyte populations in the midbrain red nucleus, suggesting conserved maturation pathways across different brain regions. For the mouse primary motor cortex dataset, in addition to successfully identifying nine canonical cell types, our analysis revealed two L5 IT neuronal subtypes (designated L5 IT1 and L5 IT2). Marker gene profiling demonstrated that L5 IT2 is involved in long-range information transmission, while L5 IT1 primarily contributes to local circuit modulation. Detailed analyses are provided in the Supplementary Note 1.8.

CellScope improves analysis of disease-control cell atlases

Disease-control cell atlases are invaluable in modern medical research, offering critical insights into disease mechanisms, potential therapeutic targets, and innovative diagnostic approaches by comparing the cellular composition and functional states of healthy and diseased individuals. In complex diseases like COVID-19, these atlases provide opportunities to help uncover cellular changes during disease progression, track immune responses, and identify cell subtypes associated with disease severity. While existing analysis pipelines[64] have made significant contributions, there remains room for improvement in detecting such signals, particularly in distinguishing cell populations associated with disease states.

To better analyze the disease-control atlas, we developed a CellScope-based analytical pipeline (Supplementary Fig. 12) and applied it to peripheral blood mononuclear cell (PBMC) data from healthy individuals and COVID-19 patients with varying disease severities[64]. Focusing on the monocyte-dendritic cell system, we successfully identified and isolated this system in step 10 of the CellScope analysis (Fig. 5a and Supplementary Fig. 13). CellScope clearly distinguished classical monocytes, non-classical monocytes, and conventional dendritic cells (middle, left, and right clusters), outperforming traditional UMAP (Fig. 5b). It also revealed a continuous differentiation trajectory from classical monocytes to conventional dendritic cells and non-classical monocytes.

Moreover, compared to Seurat, CellScope provides a clearer distinction between the three disease states—severe, moderate, and healthy. Specifically, in step 11, CellScope further refined the separation of COVID-19-associated cells, demonstrating its exceptional capability in identifying disease states. Figure 5c illustrates the expression of eight marker genes in Cluster 1 (mostly COVID-19) and Cluster 2 (mostly healthy). Among them, seven marker genes—*IFIT1, OAS2, OAS3, RNASE2, SIGLEC1, IFI44*, and *IFI27*—exhibit significantly higher expression in Cluster 1, while *HLA-DRB5* shows elevated expression in Cluster 2, providing key molecular markers for distinguishing disease states. The expression levels of these genes in the monocyte-dendritic cell system increase progressively with disease severity (Fig. 5d), highlighting the marked differential expression between COVID-19 and healthy states. Notably, such differences were not observed in other cell types (Fig. 5c). This likely underscores the critical role of the monocyte-dendritic cell system in viral recognition, antiviral responses, and immune regulation. The upregulation of Cluster 1 genes highlights a robust immune defense against SARS-CoV-2, particularly interferon-mediated antiviral responses, which are crucial components of the innate immune system's defense against viral pathogens[65]. This gene expression pattern not only indicates active viral infection but also reveals the complex interactions between the virus and the host immune system.

SARS-CoV-2 appears to impair dendritic cell function by downregulating *HLA-DRB5* expression, leading to a loss of antigen-presentation capacity in infected monocytes and dendritic cells, facilitating viral evasion of T cell-mediated immune responses[66]. Additionally, ClusterProfiler[60] enrichment analysis of the seven genes highly expressed in Cluster 1 (Fig. 5e) revealed their key roles in antiviral immune responses. *IFIT1* and *IFI27* are strongly linked to interferon responses, while *OAS2* and *SIGLEC1* activate interferon signaling, triggering antiviral defenses and inhibiting viral replication. These genes also regulate viral RNA replication, halting virus proliferation. The enrichment analysis further uncovered their roles in multiple stages of the viral life-cycle, from recognition to response and clearance, highlighting their importance in antiviral immunity.

CellScope demonstrates interpretability and robustness

CellScope demonstrates significant advantages in algorithm interpretability, robustness, and user-friendliness. It efficiently identifies biologically meaningful key genes, adapts seamlessly to diverse datasets, and offers intuitive and informative visualization. In contrast, existing tools such as Seurat and Scanpy primarily focus on HVG for gene selection, which have limitations in distinguishing between housekeeping genes and key genes. These tools also often require careful parameter adjustment to achieve optimal results.

First, CellScope enhances algorithm interpretability through its innovative gene selection approach. It selects samples with high density and large distances from each other as "manifold seeds" to effectively distinguish between noise and meaningful signals. This approach is grounded in a fundamental principle of unsupervised learning and cluster analysis—the relationship between local density and distance on manifolds[67]. The intuition behind this strategy is that high-density points have a greater probability of residing at the centers of manifold clusters, where their local neighborhoods typically exhibit higher class purity. Analysis of the Mouse Cerebral Cortex cells from the Zeisel dataset[68] revealed a strong negative correlation between local density and distance from true cluster centers (Fig. 6a), where high-density points consistently exhibited improved neighborhood purity (Fig. 6b), supporting our density-distance based manifold seeds identification method. We also compared the differences between selected and unselected genes within and between clusters (Fig. 6c, d). Relative to the unselected genes, the genes selected by CellScope exhibit significantly larger inter-class differences and smaller intra-class differences ($p = 3.8 \times 10^{-38}$). In contrast, the unselected genes show that intra-class variance is significantly greater than inter-class variance. This enhanced distinction facilitates better cluster separation, demonstrating that our gene selection process effectively captures the most informative genes for distinguishing different cell types.

Second, CellScope demonstrates exceptional robustness and adaptability by minimizing manual parameter tuning and maintaining consistent performance across various datasets. Unlike existing algorithms that often require meticulous manual parameter tuning, CellScope achieves robust performance with minimal user intervention. We performed comprehensive robustness analyses to evaluate key parameters in our gene selection process, including the number of PCA dimensions, the number of selected manifold seeds, and the number of selected genes. First, we assessed how varying the number of PCA dimensions affects clustering performance. The ARI remained stable across different numbers of PCA dimensions (Fig. 6e and Supplementary Table 18), with optimal performance observed at around 100 dimensions. Second, we compared our adaptive method for manifold seeds selection with fixed-number approaches. Our adaptive method consistently outperformed fixed-number methods across different dataset sizes (Fig. 6f and Supplementary Table 17). This adaptability allows CellScope to automatically adjust to the characteristics of each dataset without manual intervention. Third, we investigated the impact of the number of genes selected during gene selection. The clustering performance remained consistent over a wide range of gene counts (Fig. 6g and Supplementary Table 16), with optimal results achieved around 500 genes. However, selecting too few genes (e.g., 50 genes) led to insufficient clustering information due to the omission of key genes that determine cell identity, resulting in poor cluster separation. Conversely, selecting too many genes (e.g., 10,000 genes) introduced redundant information, which masked cell type-specific signals and hindered effective separation of different cell

https://doi.org/10.1038/s41467-025-67890-3

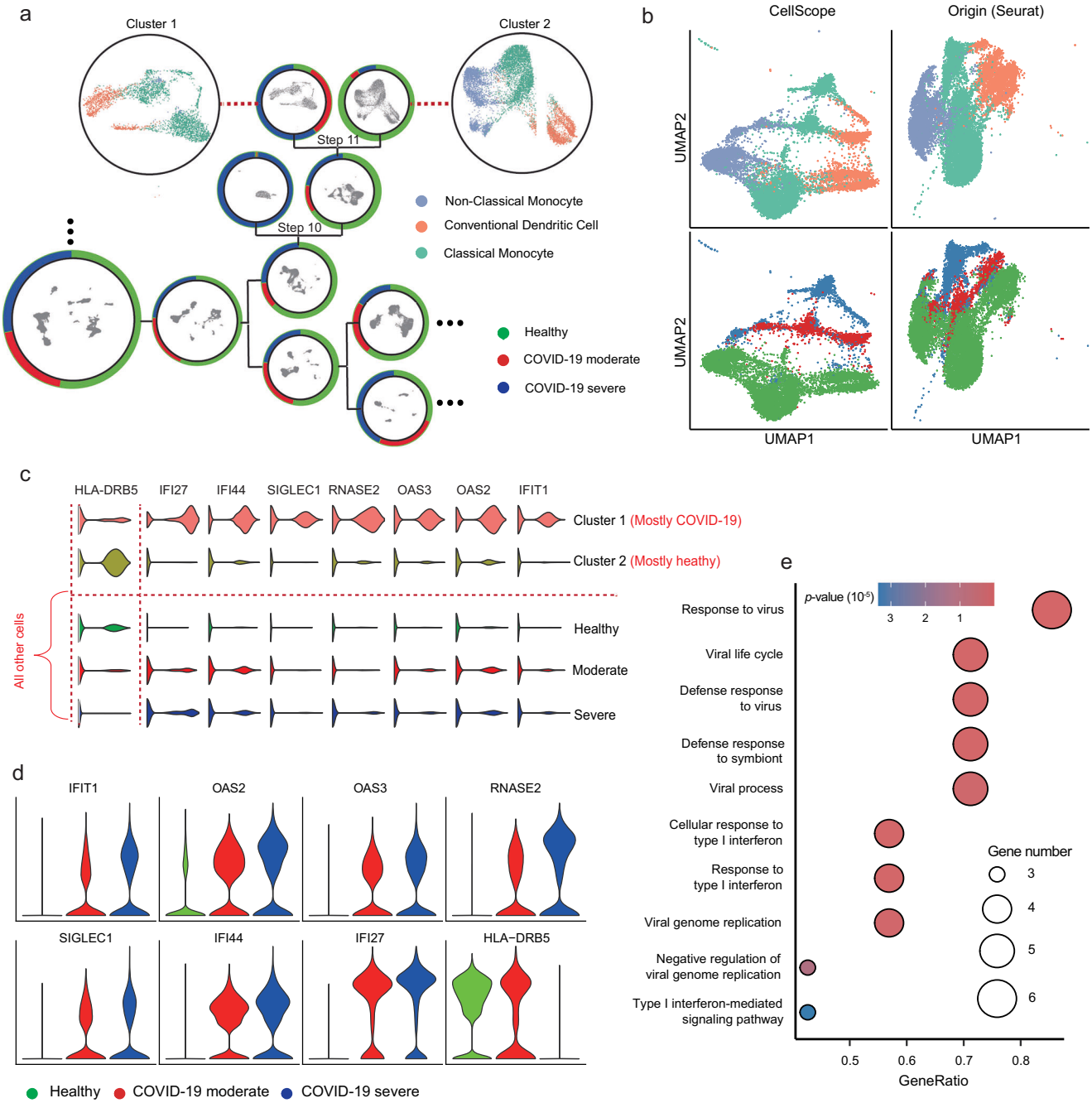


Fig. 5 | Tree-structured visualization, differential gene expression analysis, and GO enrichment for the COVID-19 PBMC dataset[64]. **a** Tree-structured visualization generated by CellScope. The colors inside the circle represent different cell types, while the outer ring colors indicate disease status. **b** UMAP visualization comparison of the monocyte-dendritic cell system provided by CellScope and Seurat, with colors representing cell type and disease severity. **c** Violin plots illustrate the expression levels of 8 marker genes, arranged from top to bottom as Cluster 1, Cluster 2, and cells outside Cluster 1 and Cluster 2, categorized by healthy state, COVID-19 moderate, and COVID-19 severe. **d** Violin plots showing the expression levels of 8 marker genes in the monocyte-dendritic cell system across three disease states. **e** GO enrichment comparison of gene clusters uniquely upregulated in Cluster 1. The *p*-values after Benjamini-Hochberg (BH) correction (Hypergeometric Test) for these functions from top to bottom are: 1.24×10^{-7}, 7.16×10^{-7}, 7.16×10^{-7}, 7.16×10^{-7}, 1.91×10^{-6}, 6.56×10^{-7}, 6.56×10^{-7}, 1.91×10^{-6}, 1.11×10^{-5}, 3.36×10^{-5}.

types. This phenomenon was evident in the analysis of human oral cavity cells from the Tirosh dataset[69] (Fig. 6h). These results emphasize the importance of judicious gene selection in single-cell analysis and highlight CellScope's robustness and adaptability in handling this critical parameter.

To evaluate the importance and necessity of each design component, we conducted a comprehensive evaluation combining targeted component analysis and systematic ablation studies, with all quantitative results provided in Supplementary Table 19. Following the CellScope analysis pipeline, we first validated the necessity of the normalization step through ablation experiments. Normalization helps mitigate technical variability and standardizes data scales across cells, thereby enhancing the reliability of clustering. The results showed that applying normalization improved clustering performance, with an average ARI increase of 0.09. Next, we evaluated the importance of the initial dimensionality reduction via PCA, which serves as a critical step in the first stage of manifold fitting. This step extracts the most informative features from the high-dimensional expression matrix and

https://doi.org/10.1038/s41467-025-67890-3

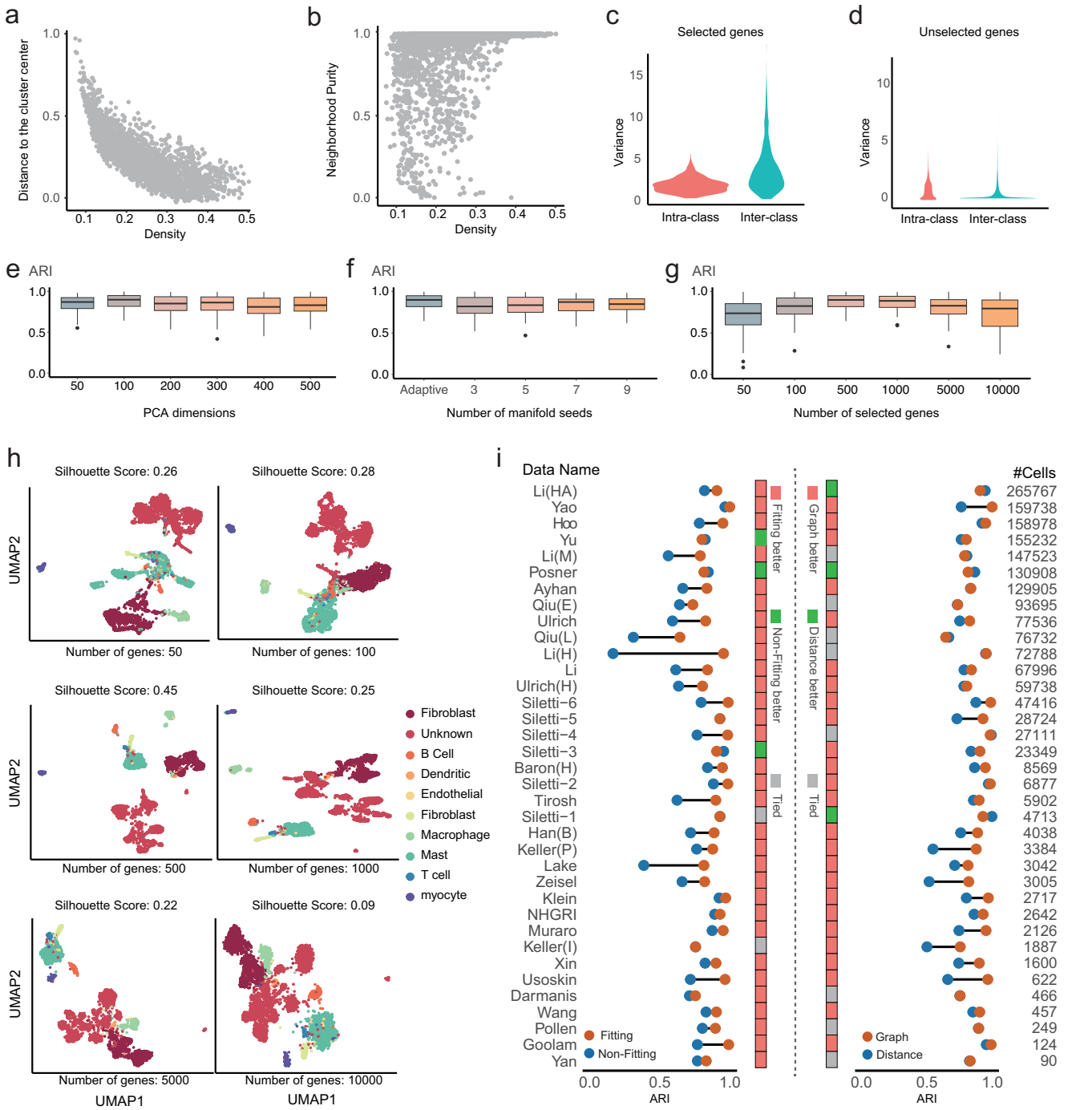


Fig. 6 | CellScope demonstrates interim results and optimal parameter choices with a strong theoretical foundation. **a** Scatter plot of Density vs Distance for Zeisel dataset. The X-axis shows the local density of every cell. The Y-axis shows the distance from the cell to its nearest true class center. **b** Scatter plot of Density vs Purity for Zeisel dataset. The X-axis shows the local density of every cell. The Y-axis shows the 100-nearest neighbor purity of every cell. **c** Violin plot comparing intra-class and inter-class gene variance for selected genes using the CellScope method in the NHGRI dataset. **d** Violin plot comparing intra-class and inter-class gene variance for unselected genes using the CellScope method in the NHGRI dataset. **e** Box plot of ARI values with different dimensions of PCA in the gene selection part of CellScope across all benchmark datasets. **f** Box plot of ARI values with different quantities of manifold seeds in the gene selection part of CellScope across all benchmark datasets. **g** Box plot of ARI values with different numbers of selected genes in the gene selection part of CellScope across all benchmark datasets. **e**–**g** The sample size for each boxplot is $n = 36$. Boxplots display the 25%, 50% (median), and 75% percentiles, where the whiskers extend to the most extreme data points within 1.5 times the interquartile range. **h** Cluster result visualization with Silhouette Scores of Tirosh dataset using different numbers of genes selected in the gene selection part of CellScope.(from left to right: 50, 100, 500, 1000, 5000, 10,000). **i** Visualization of ARI values comparison with methods on whether the manifold is fitted (left) and different hierarchies (right). The Orange above shows that the manifold fitting method and graph-based method have better performance for almost all datasets.

enables the identification of more reliable manifold seeds. The ablation results demonstrated that applying PCA prior to manifold fitting improved clustering accuracy by approximately 0.16 in ARI, compared to skipping this step. We further investigated the impact of the manifold fitting process on CellScope's performance. Figure 6i (left) illustrates that, in 31 out of 36 datasets, the application of manifold fitting yielded superior results compared to those without manifold fitting. We then compared the performance of the graph-based hierarchical clustering method adopted in CellScope with traditional distance-based hierarchical clustering. The ARI values showed that our graph-based clustering achieved equal or better performance than distance-based clustering in 33 out of 36 datasets (Fig. 6i (right)). This can be attributed to the ability of graph-based algorithms to more effectively capture local structures and nonlinear relationships in the data by constructing nodes and edges. This is particularly beneficial in high-dimensional datasets and clusters with complex morphologies, where traditional distance-based algorithms may struggle due to their inability to capture intricate structural properties. To further optimize CellScope, we implemented an adaptive distance metric based on dataset size: Euclidean distance for smaller datasets and Jaccard distance for larger datasets. The results (Supplementary Table 19) confirm the reliability of this adaptive strategy. The effectiveness of this approach stems from the fact that Euclidean distance can introduce dimensional artifacts in large datasets, a phenomenon known as the "curse of dimensionality"[70]. In contrast, the Jaccard distance metric effectively alleviates this issue by focusing on the presence or absence of features rather than their magnitude, making it more suitable for high-dimensional data.

Discussion

To address fundamental challenges in single-cell RNA sequencing analysis, including biased gene selection, oversimplified cellular visualization, and limited capability in discovering and interpreting complex biological phenomena, we present CellScope, a comprehensive computational framework built upon manifold learning principles. CellScope introduces three key innovations: a rapid and accurate gene selection method that minimizes bias while maintaining biological relevance, a tree-structured visualization framework that comprehensively represents cellular hierarchies, and a multi-level characterization system that provides dynamic gene classifications across different resolutions. Based on these innovations, CellScope demonstrates significant advantages: achieving exceptional accuracy, computational efficiency, and interpretability across diverse datasets, while also exhibiting powerful capabilities in discovering cell subpopulations and identifying disease-specific clusters.

A particularly powerful aspect of CellScope is its gene selection methodology, which demonstrates marked improvements over existing approaches. Current single-cell gene selection frameworks exist in two extreme states: overly simplistic or excessively complex. Simplistic algorithms such as HVG, implemented in Seurat or Scanpy, directly select HVG without distinguishing whether the variations arise from biological signals or noise. In contrast, complex algorithms typically employ pre-clustering or consensus approaches, which although performing better than HVG, are computationally intensive and can introduce biases from poor separation in the pre-clustering step. CellScope bridges this gap by introducing a balanced and efficient approach. By constructing reference clusters using only a small subset of trustworthy cells from the centers of high density regions, CellScope is able to identify biologically informative genes with higher reliability and efficiency than other established methods.

Popular visualization techniques like UMAP and t-SNE prioritize global structure preservation at the expense of local relationships, leading to a loss of fine-grained information about cell states and developmental trajectories. CellScope addresses this limitation through its innovative tree-structured visualization framework, which provides a hierarchical representation of cellular relationships across multiple resolutions. Unlike traditional dimensionality reduction methods that compress all information into a single view, our tree structure preserves both broad cellular categories and subtle cell states, enabling researchers to explore cellular hierarchies at different levels of granularity. This multi-resolution visualization approach is particularly powerful when analyzing complex tissues or disease progressions, where cellular states exist along continuous spectrums rather than discrete categories.

An important innovation of CellScope is its introduction of a multi-level identity system for genes, extending beyond traditional binary classifications of marker versus non-marker genes. By characterizing genes through their roles across multiple clustering layers—as housekeeping genes (HG), moderately cell-type-related genes (MCTRG), or strongly cell-type-related genes (SCTRG)—we establish a dynamic "molecular identity" for each gene. This hierarchical gene identity system reveals how genes can play different roles at different levels of cellular organization, providing crucial insights into the context-dependent nature of gene function. This understanding is particularly valuable for disease studies, where genes may acquire new functions in pathological states, and for developmental biology, where genes often switch roles during different stages of cellular differentiation.

Furthermore, CellScope demonstrates several compelling advantages in the analysis of single-cell data through its robust, user-friendly, and interpretable framework. The tool's interpretability is grounded in its theoretically sound approach to manifold fitting[71], allowing it to effectively distinguish signal from noise. In addition, unlike other methods, CellScope does not require extensive parameter tuning, as evidenced by its stability across various parameter settings, including PCA dimensions, gene selection counts, and manifold seeds detection methods. Collectively, CellScope's analytical rigor and practical usability position it as a reliable and accessible tool for single-cell analysis.

Benefiting from these technical advances, CellScope enables significant biological discoveries across diverse applications. For instance, our analysis of the Human Brain Cell Atlas uncovered two previously unrecognized Oligodendrocyte subtypes which exhibit distinct molecular signatures characterized by *RBFOX1* and *OPALIN* expression, respectively, revealing insights into myelination processes. CellScope's capabilities also extend to disease research, as demonstrated in our COVID-19 study. Through analysis of PBMCs from patients with varying disease severities, we successfully distinguished traditional immune cell types while simultaneously identifying disease-specific states. The method revealed eight marker genes showing progressive expression changes with disease severity, specifically within the monocyte-dendritic cell system, providing crucial insights into antiviral immune responses. These discoveries in both steady-state and disease contexts demonstrate CellScope's unique power in revealing both subtle cellular states and disease-associated transitions that are often missed by conventional analysis methods. Overall, CellScope provides a user-friendly, technically sound tool for advancing the field of single-cell omics in an era of increasing availability of large and complex single-cell datasets.

While CellScope addresses key limitations of established frameworks like Seurat and Scanpy through its manifold-based gene selection and hierarchical clustering approach, we acknowledge that these established methods possess valuable features, including mature ecosystems, extensive documentation, and broad data type support that are currently beyond CellScope's scope. Seurat's modular R-based architecture and Scanpy's Python integration offer computational accessibility and workflow flexibility that have made them community standards. Recent methods such as scCRT[72] represent another crucial approach to single-cell analysis, which innovatively combines cell-level pairwise modules with cluster-level contrastive learning to preserve

cellular relationships for trajectory inference applications. While both scCRT and CellScope learn highly informative low-dimensional representations and leverage clustering information, they differ in their design focus: scCRT emphasizes continuous processes and trajectory analysis, whereas CellScope is optimized for discrete clustering tasks and multi-resolution cellular characterization. However, CellScope's theoretical foundations—particularly its ability to distinguish biological signal from noise and capture cellular hierarchies—position it as a promising framework for single-cell analysis. As datasets grow larger and more complex, CellScope's parameter-free operation, superior clustering performance, and discovery-oriented design address critical needs that existing methods struggle to meet. Moving forward, we plan to continuously maintain and enhance the CellScope package, including expanding its compatibility with emerging data modalities such as spatial transcriptomics and multimodal omics integration. Additionally, we envision systematic reanalysis of large-scale public databases using CellScope's framework, which could reveal previously undetected cellular subtypes and biological insights across diverse tissues and disease contexts. Through sustained development, community engagement, and comprehensive reanalysis efforts, CellScope aims to advance single-cell analysis by enabling researchers to push the boundaries of cellular biology beyond traditional cell type identification toward comprehensive functional characterization and promising biological discovery.

Methods

CellScope aims to analyze single-cell data through manifold fitting, enabling precise identification of differences between cell types and subtypes. By identifying highly reliable cliques, the method effectively selects type-determined genes with class-specific differences while leveraging manifold fitting to mitigate the impact of technical noise. CellScope then constructs a cell-to-cell similarity graph and performs agglomerative clustering based on this graph to generate a hierarchical structure of cells. A tree-structured visualization intuitively represents the hierarchical relationships and reveals differentiation pathways among cells. Furthermore, CellScope analyzes gene expression changes along differentiation pathways and expression differences within the same hierarchy, providing functional insights into genes.

The CellScope workflow consists of five main steps, as explained in detail below: (A) Manifold fitting stage 1, (B) Manifold fitting stage 2, (C) Graph-based agglomerative clustering, (D) Tree-structured visualization, and (E) Characterization of genes from different categories. The pseudocode and flowchart of the algorithm can be found in Supplementary Note 1.7 and the tutorial website https://cellscope.readthedocs.io/en/latest/, respectively.

We summarize the main notations used in the description of the method as follows: suppose we have a expression counts matrix $\mathcal{X} = \{\mathbf{x}_i\}_{i=1}^N$, where each vector $\mathbf{x}_i$ corresponds to the expression values $[\mathbf{x}_i^{(1)}, \ldots, \mathbf{x}_i^{(D)}]$ of the i-th cell across D genes. We also let $\mathcal{G} = \{\mathbf{g}_i\}_{i=1}^D$, where $\mathcal{G}$ represents the collection of genes related to $\mathcal{X}$. After manifold fitting stage 1, we obtain $\mathcal{Y} = \{\mathbf{y}_i\}_{i=1}^N$, where $\mathcal{Y}$ represents the scRNA-seq data after manifold fitting stage 1. Each vector $\mathbf{y}_i$ corresponds to the expression values $[\mathbf{y}_i^{(1)}, \ldots, \mathbf{y}_i^{(D_1)}]$ of the i-th cell across D_1 selected genes, where $D_1 \ll D$. The set of selected genes is denoted as $\widehat{\mathcal{G}} = \{\widehat{\mathbf{g}}_i\}_{i=1}^{D_1}$, where $\widehat{\mathcal{G}} \subseteq \mathcal{G}$. Subsequently, after manifold fitting stage 2, we obtain $\mathcal{Z} = \{\mathbf{z}_i\}_{i=1}^N$, where $\mathcal{Z}$ represents the fitted scRNA-seq data for subsequent downstream analysis.

Manifold fitting stage 1

Data preprocessing. We applied consistent preprocessing to all single-cell RNA sequencing datasets. First, log normalization (base 2) was applied to the raw data $\mathcal{X}$. Then, we normalized each cell's expression profile, which is a standard procedure prior to downstream analyses. This stage is essential for eliminating differences in total expression levels between cells, which may arise from technical or biological factors. It ensures that highly expressed genes in cells with elevated overall expression do not dominate the dimensionality reduction process, thereby preventing bias in subsequent analyses. Let $\mathcal{X}_P = \{\widetilde{\mathbf{x}}_i\}_{i=1}^N$ denote the preprocessed data, where each vector $\widetilde{\mathbf{x}}_i = [\widetilde{\mathbf{x}}_i^{(1)}, \cdots, \widetilde{\mathbf{x}}_i^{(D)}]$ represents the expression values of the i-th cell after preprocessing. The set of genes after preprocessing is denoted as $\widetilde{\mathcal{G}} = \{\widetilde{\mathbf{g}}_i\}_{i=1}^D$.

Find highly reliable cliques. We begin this process with Principal Component Analysis (PCA)[73], aiming to reduce noise and complexity in the data while retaining the primary sources of variation, thereby clarifying the manifold structure of the data and providing a solid foundation for subsequent manifold exploration. In PCA, we set the target dimensionality to n_1 (defaulting to 100) and apply it to the preprocessed data $\mathcal{X}_P$. This results in a collection of cells represented in a low-dimensional space, denoted as $\mathcal{P} = \{\mathbf{p}_i\}_{i=1}^N$, where $\mathbf{p}_i \in \mathbb{R}^{n_1}$.

We aim to identify highly reliable cliques associated with distinct submanifolds, beginning with the identification of the centers of these submanifolds, referred to as manifold seeds. Inspired by ref. 67, local density reflects the compactness of a cell's surrounding distribution, while relative distance measures the separation of a cell from other cells with higher density. Manifold seeds tend to exhibit significantly higher values in both metrics. Therefore, we evaluate the potential of each cell $\mathbf{p}_i$ to serve as a manifold seed by calculating its local density $\rho(\mathbf{p}_i)$ and relative distance $\delta(\mathbf{p}_i)$, defined as:

$$\rho(\mathbf{p}_i) = \frac{1}{\sum_{j \in \mathcal{N}_i} d(\mathbf{p}_i, \mathbf{p}_j)}, \ \delta(\mathbf{p}_i) = \min_{j \in \mathcal{N}_i, \rho(\mathbf{p}_j) > \rho(\mathbf{p}_i)} d(\mathbf{p}_i, \mathbf{p}_j), \tag{1}$$

where $\mathcal{N}_i$ is the set of k nearest neighbors of cell $\mathbf{p}_i$ (with k = 20 by default), and $d(\cdot, \cdot)$ denotes the distance between two cells, defaulting to Euclidean distance. Next, we compute the composite metric:

$$\gamma(\mathbf{p}_i) = \rho(\mathbf{p}_i)\delta(\mathbf{p}_i), \tag{2}$$

and select the cells with the highest $\gamma(\mathbf{p}_i)$ values as manifold seeds. Specifically, we choose the top m cells with the largest $\gamma(\mathbf{p}_i)$ values as manifold seeds, with m being a predefined hyperparameter which refers to the number of submanifolds hypothesized in advance.

However, since m is related to the number of cell types, an intelligent approach is required to determine the number of manifold seeds m when the number of cell types is unknown. We first address the issue of scale differences between ρ and δ in the calculation of γ. Both ρ and δ are normalized as

$$\rho'(\mathbf{p}_i) = \frac{\rho(\mathbf{p}_i) - \min(\boldsymbol{\rho})}{\max(\boldsymbol{\rho}) - \min(\boldsymbol{\rho})}, \ \delta'(\mathbf{p}_i) = \frac{\delta(\mathbf{p}_i) - \min(\boldsymbol{\delta})}{\max(\boldsymbol{\delta}) - \min(\boldsymbol{\delta})}, \tag{3}$$

where $\boldsymbol{\rho} = \{\rho(\mathbf{p}_i)\}_{i=1}^N$ and $\boldsymbol{\delta} = \{\delta(\mathbf{p}_i)\}_{i=1}^N$. Next, we compute a scaleless index by multiplying the two normalized metrics:

$$\gamma'(\mathbf{p}_i) = \delta'(\mathbf{p}_i) \cdot \rho'(\mathbf{p}_i). \tag{4}$$

We then sort $\boldsymbol{\gamma}' = \{\gamma'(\mathbf{p}_i)\}_{i=1}^N$ in descending order, with γ_j'' representing the j-th value in the sorted list. To further refine the selection of manifold seeds, we introduce the relative rate of change in γ'':

$$R_j = \begin{cases} \frac{\gamma_{j+1}'' - \gamma_{j-1}''}{2} & \text{if } 2 \le j \le N-1, \\ \gamma_2'' - \gamma_1'' & \text{if } j = 1, \\ 0 & \text{if } j = N. \end{cases} \tag{5}$$

Finally, we select the manifold seeds that satisfy the following conditions:

$$\mathcal{C} = \{\mathbf{p}_i | \rho'(\mathbf{p}_i) > \overline{\boldsymbol{\rho}'}, \delta'(\mathbf{p}_i) > \overline{\boldsymbol{\delta}'}, \gamma'(\mathbf{p}_i) > \overline{\boldsymbol{\gamma}'}, R_{I(\mathbf{p}_i)} < \overline{\boldsymbol{R}}\}, \tag{6}$$

where $\bar{\cdot}$ denotes the mean value of the respective set and $\boldsymbol{R} = \{R_j\}_{j=1}^{N}$. The index $I(\mathbf{p}_i)$ refers to the position of $\gamma'(\mathbf{p}_i)$ in $\boldsymbol{\gamma}''$ after sorting $\boldsymbol{\gamma}'$. The combined metric-based strategy ensures that the selected manifold seeds possess both high local density and relative distance. The introduction of the relative rate of change further optimizes this selection process, intelligently selecting all high-confidence manifold seeds. Ultimately, we denote the set of selected seeds as $\mathbf{C} = \{\mathbf{c}_1, \cdots, \mathbf{c}_m\}$, where m represents the number of seeds selected.

Given that the identified manifold seeds are highly likely to reside at the centers of their respective submanifolds, the cells in the immediate neighborhood of each seed are assumed to belong to the same class as the seed. Therefore, for each seed c_i, we select its k_1 nearest neighboring cells (with $k_1 = 5$ by default), denoted as $\mathcal{N}_{\mathbf{c}_i}$. To ensure a high-confidence classification of the sets $\{\mathcal{N}_{\mathbf{c}_1}, \cdots, \mathcal{N}_{\mathbf{c}_m}\}$, we combine the concept of connected components in graphs, and define the following partition:

$$\begin{cases} \mathcal{N}_{\mathbf{c}_i} \text{ and } \mathcal{N}_{\mathbf{c}_j} \text{ belong to different groups}; & \text{if } \mathcal{N}_{\mathbf{c}_i} \cap \mathcal{N}_{\mathbf{c}_j} = \boldsymbol{\phi}, \\ \mathcal{N}_{\mathbf{c}_i} \text{ and } \mathcal{N}_{\mathbf{c}_j} \text{ belong to the same group}; & \text{otherwise}, \end{cases} \tag{7}$$

for $1 \le i \ne j \le m$. The resulting partition of high reliable cliques $\{\mathcal{N}_{\mathbf{c}_1}, \cdots, \mathcal{N}_{\mathbf{c}_m}\}$ is recorded as ℓ_1.

Signal space identification. Due to the significant variance differences of genes in the signal space, both within and between cell clusters, our goal was to identify genes that exhibit notable expression differences, particularly between different cell types, within highly reliable cliques. To achieve this, we leveraged the high-confidence labels ℓ_1 obtained from these reliable cliques and performed a one-way analysis of variance (ANOVA[74]) on each preprocessed gene $\widetilde{\mathbf{g}}_k \in \widetilde{\mathcal{G}}$. For each gene, the corresponding p-value was computed based on its expression across different cell clusters. We then selected the top D_1 genes with the lowest p-values (default: $D_1 = 500$), denoted as $\widehat{\mathcal{G}}$, as the result of gene selection. Furthermore, we retained the genes from the signal space in $\mathcal{X}_P$, denoted as $\mathcal{Y}$.

These selected genes represent those with the most significant expression differences between submanifolds and are considered key to capturing the biological distinctions between different cell types. The set of genes in signal space, $\widehat{\mathcal{G}}$, provides a group of genes that best distinguish the identified cell clusters, facilitating more efficient and biologically meaningful downstream analyses.

Manifold fitting stage 2

To further highlight the true biological signals and better reflect the underlying cellular heterogeneity, we project cells located between submanifolds or near manifold boundaries closer to the centers of their respective submanifolds. This process ensures that the boundaries between submanifolds become clearer after projection. We assume that the density of data points decreases as the distance from the manifold center increases. Therefore, our fitting process focuses on low-density points, as they are more likely to be influenced by noise.

First, we calculate the local density $\rho(\mathbf{y}_i)$ for each cell $\mathbf{y}_i$ to assess its position within the manifold, using the following formula:

$$\rho(\mathbf{y}_i) = \frac{1}{\sum_{\mathbf{y}_j \in \mathcal{N}_k(\mathbf{y}_i)} d(\mathbf{y}_i, \mathbf{y}_j)}. \tag{8}$$

We then select the 5% of cells with the lowest densities to form the set of manifold outliers $\mathcal{O}$.

We assume that the closest high-density point to each outlier belongs to the same class and is closer to the center of its respective manifold. To achieve this, we adopt the projection estimation method from our previous work[30]. For each outlier $\mathbf{y}_i \in \mathcal{O}$, the nearest high-density point $\widehat{\mathbf{y}}_i$ is defined as:

$$\widehat{\mathbf{y}}_i = \arg\min_{\widetilde{\mathbf{y}} \in \mathcal{Y}} d(\widetilde{\mathbf{y}}, \mathbf{y}_i), \text{ with } \rho(\widetilde{\mathbf{y}}) > \rho(\widetilde{\mathbf{y}}), \ \forall \widetilde{\mathbf{y}} \in \mathcal{O}. \tag{9}$$

Subsequently, while preserving the relative position between the outlier $\mathbf{y}_i$ and $\widehat{\mathbf{y}}_i$, we project $\mathbf{y}_i$ closer to the center of the manifold, as given by:

$$\mathbf{z}_i = t\widehat{\mathbf{y}}_i + (1 - t)\mathbf{y}_i, \tag{10}$$

where t is 0.9 by default. For regular points not identified as outliers, we define their projection as $\mathbf{z}_i = \mathbf{y}_i$, if $\mathbf{y}_i \notin \mathcal{O}$. The final dataset after the second stage of manifold fitting is represented as $\mathcal{Z} = \{\mathbf{z}_i\}_{i=1}^{N}$.

Graph-based agglomerative clustering

In the field of unsupervised learning, distance matrix-based agglomerative clustering methods have been extensively studied and applied[75,76], whereas graph-based clustering aggregation techniques remain relatively under-explored. A similarity graph can capture the local neighborhood relationships between points within a submanifold, reflecting the manifold's local geometric properties. At the same time, through the construction of neighborhood relationships, the graph can represent the connectivity between different submanifolds. Therefore, after obtaining a clear manifold structure, we begin by using the Uniform Manifold Approximation and Projection (UMAP) algorithm[20] to construct the similarity matrix $\mathbf{S} = \{s_{ij}\}_{i,j=1}^{N}$. Specifically, for each cell $\mathbf{x}_i$, we determine its $k = \lceil \log_2(N) \rceil$ nearest neighboring cells. We first set the similarity between the cell and cells outside its k nearest neighbors to 0. Next, using a Gaussian kernel function, we compute the local similarity s_{ij} between two cells $\mathbf{z}_i$ and $\mathbf{z}_j$:

$$s_{ij} = \exp\left(-\frac{\| \mathbf{z}_i - \mathbf{z}_j \|^2}{\sigma_i^2}\right), \tag{11}$$

where $\mathbf{z}_j$ is in the k nearest neighbors of $\mathbf{z}_i$ and $\|\mathbf{z}_i - \mathbf{z}_j\|$ denotes the Euclidean distance, and σ_i is a locally adaptive scale parameter satisfying:

$$\sum_{j=1}^{k} \exp\left(-\frac{\| \mathbf{z}_i - \mathbf{z}_j \|^2}{\sigma_i^2}\right) = \log_2(k). \tag{12}$$

To construct a symmetric similarity graph, we define the edge similarity w_{ij} as:

$$w_{ij} = s_{ij} + s_{ji} - s_{ij} \cdot s_{ji}. \tag{13}$$

The corresponding symmetric similarity matrix is denoted as $\mathbf{W} = \{w_{ij}\}_{i,j=1}^{N}$. Finally, using the average linkage method based on pairwise similarities, we systematically merge data points starting with each cell as a separate cluster, and denote the clustering results after K steps as T_K. T_{K+1} merges the two clusters with the highest average similarity, i.e., clusters A_ℓ and B_ℓ that maximize:

$$[A_\ell, B_\ell] = \text{argmax}_{A,B \in T_K} \frac{1}{|A||B|} \sum_{a \in A} \sum_{b \in B} w_{ab}. \tag{14}$$

For exceptionally large datasets (with over 30,000 data points), we employ the following optimization strategy: First, we randomly select a representative subset of 30,000 points from the complete dataset to form subset S_1, while the remaining cells constitute subset

S_2. After performing hierarchical clustering on S_1 using the average linkage method to obtain clustering result T'_K, we analyze the distribution of clusters within the k nearest neighbors (default $k = 5$) of each cell $\mathbf{z} \in S_2$ in S_1 and assign $\mathbf{z}$ to the cluster with the highest cell count in its k nearest neighbors.

Tree-structured visualization

To comprehensively and systematically explore cell types and their subtypes, we developed a tree-structured visualization method. First, we applied the UMAP algorithm[20] to visualize all cells as the root node. Then, each cluster from T_{N-1} was visualized individually to generate the first layer of child nodes. By comparing the clustering results of T_{N-2} and T_{N-1}, we identified two new subtypes within the cells of the first-layer child nodes and visualized them as the second layer of child nodes. This iterative process was repeated in the same manner, advancing layer by layer.

To clearly illustrate the distribution of each pair of subtypes within their parent nodes, we applied a color-coding scheme to the tree-structured visualization. Specifically, each branch's outermost nodes were colored first, and the colors of the child nodes were then propagated back to their respective parent nodes. This core visualization method effectively reveals the hierarchical relationships between cell types, illustrating how different cell populations emerge, differentiate, and specialize over time. The tree-structured visualization automatically generated by CellScope is shown in Supplementary Note 1.9 and the tutorial at https://cellscope.readthedocs.io/en/latest/.

Characterization of genes from different categories

After generating the Tree-structured visualization, we quantified the gene expression differences between sibling nodes (sharing the same direct parent node) using the Wasserstein distance. This metric measures the minimal transport cost required to transform one probability distribution into another, providing a robust comparison between gene expression profiles from sibling nodes.

For gene $\mathbf{g}_1$, assume its expression in two sibling nodes is represented by $\mathbf{P} = \{p_1, \ldots, p_n\}$ and $\mathbf{Q} = \{q_1, \ldots, q_m\}$, respectively. First, both expression vectors are sorted in ascending order, yielding the ordered vectors $\mathbf{P}' = \{p'_1, \ldots, p'_n\}$ and $\mathbf{Q}' = \{q'_1, \ldots, q'_m\}$. We then compute the cumulative distribution functions (CDFs) for the sorted vectors $\mathbf{P}'$ and $\mathbf{Q}'$, as defined by the following equations

$$F_{\mathbf{P}}(p'_i) = \frac{i}{n},\ i = 1, 2, \ldots, n, \tag{15}$$

$$F_{\mathbf{Q}}(q'_j) = \frac{j}{m},\ j = 1, 2, \ldots, m. \tag{16}$$

We then use linear interpolation to align the two CDFs $F_{\mathbf{P}}$ and $F_{\mathbf{Q}}$ onto the same probability space. For a given probability value $x \in [0, 1]$, if $F_{\mathbf{P}}(p'_i) < x < F_{\mathbf{P}}(p'_{i+1})$, then $F^{-1}_{\mathbf{P}}(x)$ is calculated as

$$F^{-1}_{\mathbf{P}}(x) = p'_i + \frac{x - F_{\mathbf{P}}(p'_i)}{F_{\mathbf{P}}(p'_{i+1}) - F_{\mathbf{P}}(p'_i)} \cdot (p'_{i+1} - p'_i). \tag{17}$$

Similarly, $F^{-1}_{\mathbf{Q}}(x)$ is computed in the same manner. Finally, the Wasserstein distance[77] is defined as the integral of the absolute difference between the two inverse CDFs across the probability space [0, 1]

$$W_1(\mathbf{P}, \mathbf{Q}) = \int_0^1 |F^{-1}_{\mathbf{P}}(x) - F^{-1}_{\mathbf{Q}}(x)|\, dx. \tag{18}$$

Finally, based on the Wasserstein distance calculated from the expression differences of genes between sibling nodes, we classified the genes into three categories

$$\begin{cases} \text{Housekeeping gene} & \text{if } W_1(\mathbf{P}, \mathbf{Q}) < 0.5, \\ \text{Moderately cell} - \text{type} - \text{related gene} & \text{if } 0.5 \le W_1(\mathbf{P}, \mathbf{Q}) < 1, \\ \text{Strongly cell} - \text{type} - \text{related gene} & \text{if } 1 \le W_1(\mathbf{P}, \mathbf{Q}). \end{cases} \tag{19}$$

The threshold selection for Wasserstein distance is based on Supplementary Fig. 14, using SubCluster1 and SubCluster2 in Siletti-1 (Fig. 4) as an example to demonstrate its rationale. When the Wasserstein distance is less than 0.5, the gene expression distributions are similar. For distances between 0.5 and 1, there are notable differences in means, though some overlap remains, indicating moderate differences in gene expression. When the distance exceeds 1, the first quartile for the cluster with higher mean expression surpasses the third quartile of the other, indicating significant differences in gene expression. In addition, we systematically reviewed three traditional differential gene analysis methods in Supplementary Note 1.6.

Benchmark methods

Compared pipelines. We compared CellScope against two widely used single-cell analysis methods (Seurat[35] and Scanpy[13]) and three recent methods (scLEGA[36], scDCCA[37], and CellBRF[38]). Scanpy was implemented from its original source code repository (https://github.com/scverse/Scanpy). HVG were identified based on specified thresholds for mean expression and dispersion, and clustering was performed using the Leiden algorithm across a range of resolutions. The algorithm parameters were set according to the default parameter settings in the tutorial(https://Scanpy-tutorials.readthedocs.io/en/latest/pbmc3k.html).

Seurat was implemented from its source code (https://satijalab.org/seurat) with a scale factor of 10,000. We identified 2000 variable features using the vst selection method as mentioned in Seurat tutorial(https://satijalab.org/seurat/articles/pbmc3k_tutorial). Neighbors were identified using the first 10 principal components, and clustering by Louvain algorithm was performed across a range of resolutions.

scLEGA combines a denoising autoencoder with a graph autoencoder using multi-head attention to fuse expression and topological information (https://github.com/Masonze/scLEGA-main). Following the authors' recommended settings, we used learning rate 0.001, latent dimension 16, 8 attention heads, 2500 HVG, and 200 training epochs.

scDCCA is a deep contrastive clustering method that integrates a denoising autoencoder with dual contrastive learning. We followed the authors' recommended configuration (https://github.com/WJ319/scDCCA): 2000 HVG, $z_{\text{dim}} = 32$, encoder layers [256, 64], and ran 70 pretraining epochs followed by 100 clustering epochs.

CellBRF is a random forest-based method for cell type identification with class balancing strategy. We used the authors' recommended parameters (https://github.com/xuyp-csu/CellBRF): $k = 15$ nearest neighbors, $n_{\text{pcs}} = 50$ principal components, and enabled redundancy removal with correlation threshold 0.8.

In our experiments, we considered a range of clustering resolutions for both Scanpy and Seurat, specifically testing resolutions of 0.1, 0.15, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0, 1.1, 1.2, 1.3, 1.4, 1.5, 1.6, 1.7, 1.8, 1.9, and 2.0. For each method, we computed the Adjusted Rand Index (ARI) between the predicted clusters and the true cell type labels at each resolution. The final results for both methods were selected based on the highest ARI observed across all tested resolutions, ensuring that the best clustering performance was captured for each dataset.

Gene selection methods. We compared several common gene filtering methods and the latest gene selection methods to identify the most effective techniques for analyzing single-cell RNA sequencing

data. Disp[43] and VST[35]are widely used gene filtering methods. Disp, introduced by Seurat, identifies genes with the largest variation after controlling for mean expression variability by z-standardizing dispersion measures within expression bins. VST refines this approach by fitting a loess curve to the log(variance) vs. log(mean) relationship.

In terms of gene selection methods, SAIC[78] uses an iterative k-means clustering method to thoroughly search for the best feature genes. FEAST[45] uses the F statistic to test feature significance and summarize the variance differences between and within groups, similar to the Fisher score. We also selected the ensemble learning-based method, CellBRF[38], which uses a random forest guided by predicted cell labels to identify the most important genes for distinguishing cell types. Finally, we considered the graph-based method, HRG[46], which finds informative genes by optimizing expression patterns in a similarity network between cells, ensuring that these genes exhibit regional expression patterns. Each method provides a unique approach to gene selection.

Benchmark data

We selected 36 benchmark datasets with cell numbers ranging from 90 to 265,767, covering various tissues of humans and mice, including pancreas, brain, intestine, spleen, liver, bone marrow, retina, etc. These datasets also involve a variety of diseases and health conditions, such as human islet cells, mouse cerebral cortex, human cervical cancer, and mouse motor cortex. The number of genes in these datasets ranges from 14,717 to 59,357, and the number of cell types ranges from 3 to 20, covering a wide range of biodiversity to measure the performance of CellScope. The detailed information of all datasets is listed in Supplementary Table 2.

Computational environments

All computational analyses, including the execution of CellScope and comparative algorithms (Scanpy, Seurat, scLEGA, scDCCA, and CellBRF), were performed on the Google Colab platform equipped with 44 CPU cores and 150 GB RAM. To further validate performance metrics, additional benchmarking experiments—focusing on clustering accuracy, runtime efficiency, and memory utilization—were conducted on personal computers. These included an Apple MacBook (M2 chip, 8-core CPU, 16GB RAM) and an Apple iMac (M4 chip, 10-core CPU, 16GB RAM). CellScope was executed in a Python environment, with its Python version and required library dependencies detailed in Supplementary Table 26. All comparative algorithms utilized the latest publicly available versions, with Scanpy version 1.10.3 and Seurat version 5.2.0.

Statistics and reproducibility

In the implementation of CellScope, no statistical method was used to predetermine sample size. No data were excluded from the analyses; the experiments were not randomized; the investigators were not blinded to allocation during experiments and outcome assessment.

Reporting summary

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

Data availability

Publicly available datasets used in this study can be accessed from the National Center for Biotechnology Information Gene Expression Omnibus (GSE36552, GSE83139, GSE67835, GSE59739, GSE81608, GSE85241, GSE65525, GSE60361, GSE132042, GSE108097, GSE103322, GSE84133, GSE178101, GSE228590, GSE160189, and GSE243413), the NCBI Sequence Read Archive (SRP041736), the European Nucleotide Archive (E-MTAB-3321, E-MTAB-13382, E-MTAB-12795, and E-MTAB-10187), the Database of Genotypes and Phenotypes (PHS000833 and PHS000424V9P2), and the BRAIN Initiative Cell Census Network (RRID:SCR_015820) available for download from the Neuroscience Multi-omics Archive (RRID:SCR_016152). Human retina datasets (Li(H) and Li(HA)) are from the Human Retina Cell Atlas (HRCA) project and can be accessed through the HCA Data Portal (https://data.humancellatlas.org/). The specific download links for all datasets can be found in Supplementary Table 3. We deposit the gene expression matrices and their corresponding labels of the benchmark datasets in the Zenodo database, accessible via https://doi.org/10.5281/zenodo.17636503[79]. The source data generated in this study underlying all reported figures are provided in the Supplementary and Source Data files. Source data are provided with this paper.

Code availability

CellScope is implemented in Python and available on GitHub at https://github.com/zhigang-yao/CellScopeand on Zenodo at https://doi.org/10.5281/zenodo.17636503[79]. Detailed tutorials, code instructions and notebooks to reproduce the results of this study are available at https://cellscope.readthedocs.io/en/latest/.

References

1. Rood, J. E. et al. The human cell atlas from a cell census to a unified foundation model. *Nature* **637**, 1065–1071 (2025).
2. Rozenblatt-Rosen, O., Stubbington, M. J., Regev, A. & Teichmann, S. A. The human cell atlas: from vision to reality. *Nature* **550**, 451–453 (2017).
3. Xu, C. et al. Automatic cell-type harmonization and integration across human cell atlas datasets. *Cell* **186**, 5876–5891 (2023).
4. Zhong, J. et al. Single-cell brain atlas of parkinson's disease mouse model. *J. Genet. Genom.* **48**, 277–288 (2021).
5. Mathys, H. et al. Single-cell atlas reveals correlates of high cognitive function, dementia, and resilience to alzheimer's disease pathology. *Cell* **186**, 4365–4385 (2023).
6. Zhu, B. et al. Single-cell transcriptomic and proteomic analysis of parkinson's disease brains. *Sci. Transl. Med.* **16**, eabo1997 (2024).
7. Haniffa, M. et al. A roadmap for the human developmental cell atlas. *Nature* **597**, 196–205 (2021).
8. Gopee, N. H. et al. A prenatal skin atlas reveals immune regulation of human skin morphogenesis. *Nature* **635**, 679–689 (2024).
9. Zhang, Y. & Liu, F. Multidimensional single-cell analyses in organ development and maintenance. *Trends Cell Biol.* **29**, 477–486 (2019).
10. Nieto, P. et al. A single-cell tumor immune atlas for precision oncology. *Genome Res.* **31**, 1913–1926 (2021).
11. Wu, S. Z. et al. A single-cell and spatially resolved atlas of human breast cancers. *Nat. Genet.* **53**, 1334–1347 (2021).
12. Hao, Y. et al. Dictionary learning for integrative, multimodal and scalable single-cell analysis. *Nat. Biotechnol.* **42**, 293–304 (2024).
13. Wolf, F. A., Angerer, P. & Theis, F. J. Scanpy: large-scale single-cell gene expression data analysis. *Genome Biol.* **19**, 1–5 (2018).
14. Fang, R. et al. Comprehensive analysis of single cell ATAC-seq data with snapatac. *Nat. Commun.* **12**, 1337 (2021).
15. Kiselev, V. Y., Andrews, T. S. & Hemberg, M. Challenges in unsupervised clustering of single-cell RNA-seq data. *Nat. Rev. Genet.* **20**, 273–282 (2019).
16. Blondel, V., Guillaume, J., Lambiotte, R. & Lefebvre, E. Fast unfolding of communities in large networks. *J. Stat. Mech. Theory Exp.* **2008**, P10008 (2008).
17. Traag, V. A., Waltman, L. & Van Eck, N. J. From Louvain to Leiden: guaranteeing well-connected communities. *Sci. Rep.* **9**, 1–12 (2019).
18. Duò, A., Robinson, M. D. & Soneson, C. A systematic performance evaluation of clustering methods for single-cell RNA-seq data. *F1000Research* **7**, 1141 (2018).
19. vd Maaten, L. & Hinton, G. Visualizing data using t-sne. *J. Mach. Learn. Res.* **9**, 2579–2605 (2008).

20. Healy, J. & McInnes, L. Uniform manifold approximation and projection. *Nat. Rev. Methods Primers* **4**, 82 (2024).
21. Kobak, D. & Berens, P. The art of using t-sne for single-cell transcriptomics. *Nat. Commun.* **10**, 5416 (2019).
22. Ding, J., Condon, A. & Shah, S. P. Interpretable dimensionality reduction of single cell transcriptome data with deep generative models. *Nat. Commun.* **9**, 2002 (2018).
23. Sun, X., Liu, Y. & An, L. Ensemble dimensionality reduction and feature gene extraction for single-cell RNA-seq data. *Nat. Commun.* **11**, 5853 (2020).
24. Amodio, M. et al. Exploring single-cell data with deep multitasking neural networks. *Nat. Methods* **16**, 1139–1145 (2019).
25. Verma, A. & Engelhardt, B. E. A robust nonlinear low-dimensional manifold for single cell RNA-seq data. *BMC Bioinform.* **21**, 1–15 (2020).
26. Liang, S. et al. Single-cell manifold-preserving feature selection for detecting rare cell populations. *Nat. Comput. Sci.* **1**, 374–384 (2021).
27. Xu, Y. et al. Structure-preserving visualization for single-cell RNA-seq profiles using deep manifold transformation with batch-correction. *Commun. Biol.* **6**, 369 (2023).
28. Fefferman, C., Ivanov, S., Lassas, M. & Narayanan, H. Fitting a manifold of large reach to noisy data. *J. Topol. Anal.* **17**, 315–396 (2025).
29. Yao, Z., Su, J., Li, B. & Yau, S.-T. Manifold fitting. Preprint at https://doi.org/10.48550/arXiv.2304.07680 (2023).
30. Yao, Z., Li, B., Lu, Y. & Yau, S.-T. Single-cell analysis via manifold fitting: A framework for rna clustering and beyond. *Proc. Natl. Acad. Sci. USA* **121**, e2400002121 (2024).
31. Moon, K. R. et al. Visualizing structure and transitions in high-dimensional biological data. *Nat. Biotechnol.* **37**, 1482–1492 (2019).
32. Kim, J. K., Kolodziejczyk, A. A., Ilicic, T., Teichmann, S. A. & Marioni, J. C. Characterizing noise structure in single-cell RNA-seq distinguishes genuine from technical stochastic allelic expression. *Nat. Commun.* **6**, 8687 (2015).
33. Belkin, M., Niyogi, P. & Sindhwani, V. Manifold regularization: a geometric framework for learning from labeled and unlabeled examples. *J. Mach. Learn. Res.* **7**, 2399–2434 (2006).
34. Chapelle, O., Schölkopf, B. & Zien, A. *Semi-supervised learning* (MIT Press, 2006).
35. Stuart, T. et al. Comprehensive integration of single-cell data. *cell* **177**, 1888–1902 (2019).
36. Liu, Z., Liang, Y., Wang, G. & Zhang, T. Sclega: an attention-based deep clustering method with a tendency for low expression of genes on single-cell RNA-seq data. *Brief. Bioinform.* **25**, bbae371 (2024).
37. Wang, J., Xia, J., Wang, H., Su, Y. & Zheng, C.-H. scdcca: deep contrastive clustering for single-cell RNA-seq data based on auto-encoder network. *Brief. Bioinform.* **24**, bbac625 (2023).
38. Xu, Y. et al. Cellbrf: a feature selection method for single-cell clustering using cell balance and random forest. *Bioinformatics* **39**, i368–i376 (2023).
39. Hubert, L. & Arabie, P. Comparing partitions. *J. classif.* **2**, 193–218 (1985).
40. Van Dongen, S. Performance criteria for graph clustering and Markov cluster experiments. *Rep. Inform. Syst.* **12**, 1–36 (2000).
41. Strehl, A. & Ghosh, J. Cluster ensembles—a knowledge reuse framework for combining multiple partitions. *J. Mach. Learn. Res.* **3**, 583–617 (2002).
42. Murphy, A. H. The finley affair: a signal event in the history of forecast verification. *Weather Forecast.* **11**, 3–20 (1996).
43. Satija, R., Farrell, J. A., Gennert, D., Schier, A. F. & Regev, A. Spatial reconstruction of single-cell gene expression data. *Nat. Biotechnol.* **33**, 495–502 (2015).
44. Zhao, R., Lu, J., Zhou, W., Zhao, N. & Ji, H. A systematic evaluation of highly variable gene selection methods for single-cell RNA-sequencing. *Genom. Biol.* **26**, 424 (2024).
45. Su, K., Yu, T. & Wu, H. Accurate feature selection improves single-cell RNA-seq cell clustering. *Brief. Bioinform.* **22**, bbab034 (2021).
46. Wu, Y. et al. Highly regional genes: graph-based gene selection for single-cell RNA-seq data. *J. Genet. Genom.* **49**, 891–899 (2022).
47. Rousseeuw, P. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *J. Comput. Appl. Math.* **20**, 53–65 (1987).
48. Korsunsky, I. et al. Fast, sensitive and accurate integration of single-cell data with harmony. *Nat. Methods* **16**, 1289–1296 (2019).
49. Wang, Y. J. et al. Single-cell transcriptomics of the human endocrine pancreas. *Diabetes* **65**, 3028–3038 (2016).
50. Consortium, G. The gtex consortium atlas of genetic regulatory effects across human tissues. *Science* **369**, 1318–1330 (2020).
51. Almanzar, N., Antony, J., Baghel, A. S., Bakerman, I. & Zou, J. A single-cell transcriptomic atlas characterizes ageing tissues in the mouse. *Nature* **583**, 590–595 (2020).
52. Usoskin, D. et al. Unbiased classification of sensory neuron types by large-scale single-cell RNA sequencing. *Nat. Neurosci.* **18**, 145–153 (2015).
53. Siletti, K. et al. Transcriptomic diversity of cell types across the adult human brain. *Science* **382**, eadd7046 (2023).
54. Kippert, A., Trajkovic, K., Fitzner, D., Opitz, L. & Simons, M. Identification of tmem10/opalin as a novel marker for oligodendrocytes using gene expression profiling. *BMC Neurosci.* **9**, 1–12 (2008).
55. Yuan, L., Seong, E., Beuscher, J. L. & Arikkath, J. δ-catenin regulates spine architecture via cadherin and PDZ-dependent interactions. *J. Biol. Chem.* **290**, 10947–10957 (2015).
56. Chun, S. J., Rasband, M. N., Sidman, R. L., Habib, A. A. & Vartanian, T. Integrin-linked kinase is required for laminin-2–induced oligodendrocyte cell spreading and CNS myelination. *J. Cell Biol.* **163**, 397–408 (2003).
57. Hoshina, N. et al. Protocadherin 17 regulates presynaptic assembly in topographic corticobasal ganglia circuits. *Neuron* **78**, 839–854 (2013).
58. Fogel, B. L. et al. Rbfox1 regulates both splicing and transcriptional networks in human neuronal development. *Hum. Mol. Genet.* **21**, 4171–4186 (2012).
59. Hughes, E. G. & Stockton, M. E. Premyelinating oligodendrocytes: mechanisms underlying cell survival and integration. *Front. Cell Dev. Biol.* **9**, 714169 (2021).
60. Yu, G., Wang, L.-G., Han, Y. & He, Q.-Y. clusterprofiler: an R package for comparing biological themes among gene clusters. *Omics: J. Integr. Biol.* **16**, 284–287 (2012).
61. Cai, P. et al. A genome-wide long noncoding RNA CRISPR screen identifies prancr as a novel regulator of epidermal homeostasis. *Genome Res.* **30**, 22–34 (2020).
62. Raff, M., Apperly, J., Kondo, T., Tokumoto, Y. & Tang, D. Timing cell-cycle exit and differentiation in oligodendrocyte development. in *The Cell Cycle and Development: Novartis Foundation Symposium*. **237**, 100–113 (Wiley Online Library, 2001).
63. Yao, Z. et al. A transcriptomic and epigenomic cell atlas of the mouse primary motor cortex. *Nature* **598**, 103–110 (2021).
64. Arunachalam, P. S. et al. Systems biological assessment of immunity to mild versus severe COVID-19 infection in humans. *Science* **369**, 1210–1220 (2020).
65. Lee, J. S. et al. Immunophenotyping of COVID-19 and influenza highlights the role of type i interferons in development of severe COVID-19. *Sci. Immunol.* **5**, eabd1554 (2020).
66. Shi, W. et al. High-dimensional single-cell analysis reveals the immune characteristics of COVID-19. *Am. J. Physiol.-Lung Cell. Mol. Physiol.* **320**, L84–L98 (2021).
67. Rodriguez, A. & Laio, A. Clustering by fast search and find of density peaks. *Science* **344**, 1492–1496 (2014).

68. Zeisel, A. et al. Cell types in the mouse cortex and hippocampus revealed by single-cell RNA-seq. *Science* **347**, 1138–1142 (2015).
69. Puram, S. V. et al. Single-cell transcriptomic analysis of primary and metastatic tumor ecosystems in head and neck cancer. *Cell* **171**, 1611–1624 (2017).
70. François, D., Wertz, V. & Verleysen, M. The concentration of fractional distances. *IEEE Trans. Knowl. Data Eng.* **19**, 873–886 (2007).
71. Kour, G. & Saabne, R. Fast classification of handwritten on-line arabic characters. In *Soft Computing and Pattern Recognition (SoCPaR), 2014 6th International Conference of*, 312–318 (IEEE, 2014).
72. Shi, Y., Wan, J., Zhang, X., Liang, T. & Yin, Y. sccrt: a contrastive-based dimensionality reduction model for scRNA-seq trajectory inference. *Brief. Bioinform.* **25**, bbae204 (2024).
73. Abdi, H. & Williams, L. J. Principal component analysis. *Wiley Interdiscip. Rev. Comput. Stat.* **2**, 433–459 (2010).
74. Fisher, R. A. Statistical methods for research workers. in *Breakthroughs in Statistics: Methodology and Distribution* 66–70 (Springer, 1970).
75. Murtagh, F. & Contreras, P. Algorithms for hierarchical clustering: an overview. *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.* **2**, 86–97 (2012).
76. Szekely, G. J. et al. Hierarchical clustering via joint between-within distances: extending ward's minimum variance method. *J. Classif.* **22**, 151–184 (2005).
77. Peyré, G. et al. Computational optimal transport: with applications to data science. *Found. Trends® Mach. Learn.* **11**, 355–607 (2019).
78. Yang, L., Liu, J., Lu, Q., Riggs, A. D. & Wu, X. Saic: an iterative clustering approach for analysis of single cell RNA-seq data. *BMC Genom.* **18**, 9–17 (2017).
79. Li, B. et al. Cellscope: high-performance cell atlas workflow with tree-structured representation. https://doi.org/10.5281/zenodo.17636503 (2025).

Acknowledgements

Z.Y. acknowledges support from the Singapore Ministry of Education Tier 2 grant (A-0008520-00-00, A-8001562-00-00) and the Tier 1 grant (A-8000987-00-00 and A-8002931-00-00) at the National University of Singapore.

Author contributions

B.L. contributed to literature collection, project planning and designing, framework development, code implementation, figure creation, and paper writing. R.L. conducted literature collection, participated in framework development and experimental code debugging, and contributed to figure creation, and paper writing. T.N. implemented the Python version of the original code, expanded the algorithm's functionality, and contributed to experimental code debugging, figure creation, and paper writing. G.Y. reviewed the article, provided suggestions for experimental design and article writing. M.B. reviewed the article, helped revise the article. Z.Y. and J.J.L. developed the idea, supervised the project, and wrote the manuscript. All co-authors read and approved the paper.

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/s41467-025-67890-3.

Correspondence and requests for materials should be addressed to Jingyi Jessica Li or Zhigang Yao.

Peer review information *Nature Communications* thanks Yin Wang and Yuyu Yin for their contribution to the peer review of this work. A peer review file is available.