Briefings in Bioinformatics, 2026, 27, bbag288
https://doi.org/10.1093/bib/bbag288
Published: 4 June 2026
Problem Solving Protocol

OXFORD

Lense: optimizing data preprocessing in single-cell omics using large language models

Jingyun Liu and Zhicheng Ji*

Department of Biostatistics and Bioinformatics, Duke University School of Medicine, 2424 Erwin Road, Durham, NC, 27705, United States

*Corresponding author. E-mail: zhicheng.ji@duke.edu

Abstract

Data preprocessing is critical for single-cell omics analyses, but default pipelines often underperform on diverse datasets, especially from emerging platforms like spatial transcriptomics. We introduce Lense, a language-model-guided method that automatically selects optimal preprocessing by comparing plots that visualize low-dimensional representations across pipeline variants. Integrated with Seurat, Lense streamlines analysis and improves preprocessing robustness without requiring manual tuning.

Keywords large language model, single-cell genomics, spatial genomics, data preprocessing

Introduction

Single-cell RNA sequencing (scRNA-seq) and single-cell-resolution spatial transcriptomics (ST) platforms have become indispensable for dissecting cellular heterogeneity within complex tissues [1–5]. The reliability of downstream analyses often depends on a carefully engineered preprocessing pipeline that includes data normalization, selection of highly variable genes, dimensionality reduction, and cell clustering. Each stage offers multiple algorithmic alternatives and tunable parameters. For instance, data normalization can involve log transformation, centered-log-ratio (CLR) scaling , or relative count rescaling. Although popular frameworks such as Seurat [6] and Scanpy [7] provide access to these options, analysts frequently adopt default settings, a practice that may lead to suboptimal performance.

The challenge is exacerbated when pipelines optimized for one modality, e.g. scRNA-seq, are applied without modification to data from emerging platforms such as the 10x Xenium ST system. Differences in data distribution between scRNA-seq and single-cell-resolution ST platforms [8] increase the likelihood that parameter settings validated on hundreds of scRNA-seq datasets may perform poorly on ST data. In Fig. 1a, we illustrate this discrepancy by simulating a 10x Xenium dataset from an annotated scRNA-seq sample (Methods), then evaluating the agreement between the resulting cell clusters and the inherited cell type annotations using the adjusted Rand index (ARI). The default scRNA-seq preprocessing parameters yield an ARI far lower than that obtained with an alternative parameter combination, suggesting that pipelines optimized for scRNA-seq do not generalize to ST data and that dataset-specific parameter tuning is essential for accurate cell clustering. However, identifying optimal parameter settings often involves a manual trial-and-error process, in which analysts test multiple combinations and visually compare the results of data processing and cell clustering. This approach is time-consuming and does not scale to large parameter spaces. Furthermore, the absence of a gold standard renders the process highly subjective. Analysts with limited experience may struggle to select appropriate parameters, leading to inconsistent and irreproducible results across research groups. Consequently, a systematic, data-driven strategy is required to pinpoint the optimal preprocessing pipeline rather than relying on intuition and *ad hoc* heuristics.

Large language models (LLMs), such as GPT-4 developed by OpenAI [9], are advanced artificial intelligence systems trained on massive amounts of data and adaptable to a wide range of tasks. Although intended for general purposes, studies have shown that LLMs excel at analyzing genomic and biomedical data [10–15]. In this study, we present a strategy that uses LLMs to automatically select optimal preprocessing pipelines by evaluating plots that visualize low-dimensional representations, such as principal component analysis (PCA) or Uniform Manifold Approximation and Projection (UMAP), produced under different parameter settings. The rationale is that preprocessing quality is often visible in low-dimensional representations, where well-processed data yield large, clearly separated clusters with few small, isolated islands. As illustrated in Fig. 1b, GPT-4o was given two UMAPs generated from distinct pipelines and correctly identified the one with the higher ARI, demonstrating its ability to recognize superior preprocessing configurations. Because generating low-dimensional representations is already a routine step of preprocessing pipelines and remains independent of the sequencing platform, this strategy avoids adding extra analysis steps, making it

Received: October 28, 2025. **Revised:** April 3, 2026. **Accepted:** May 10, 2026

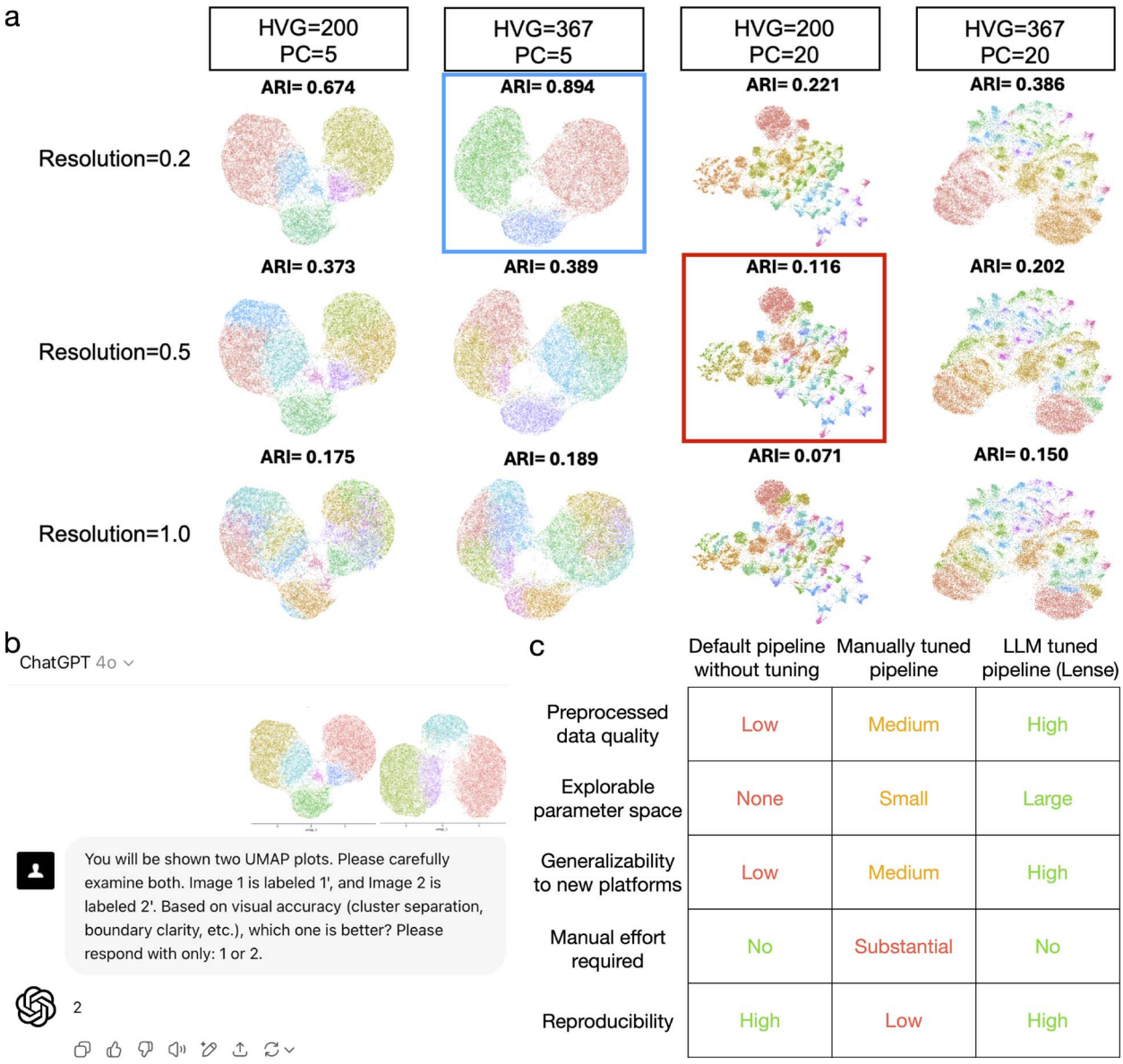

Figure 1 **a**, A simulated human pancreas 10x Xenium dataset illustrating the effects of different log-normalization preprocessing pipelines, with columns showing different numbers of highly variable features and PCs, rows showing different clustering resolutions, ARI values displayed above each subplot, and rectangles highlighting the pipeline closest to Seurat's default settings, second row, third column, and the pipeline with the highest ARI, first row, second column. **b**, A GPT-4o screenshot illustrating the prompt message and GPT-4o output for selecting the better UMAP plot, where the first and second UMAPs have ARI scores of 0.622 and 0.728, respectively. **c**, A table summarizing the advantages and limitations of existing methods and of Lense.

broadly applicable. By incorporating LLMs in place of human analysts, this strategy overcomes several critical limitations of existing methods (Fig. 1c).

Results

To systematically evaluate GPT-4o's ability to identify the better preprocessing pipeline, we collected nine additional scRNA-seq samples generated and analyzed by different research groups and created simulated 10x Xenium datasets. Cell type annotations obtained from the original studies were treated as the gold standard. For each dataset, we applied three different cell filtering criteria with varying levels of stringency, resulting in a total of 30 scenarios. These scenarios represent diverse ranges in the number of cells, number of genes, average total number of reads per cell, and proportion of zeros (Fig. 2a). We designed 72 preprocessing pipelines by combining six data normalization methods, two strategies for selecting highly variable features, two choices for the number of principal components (PCs), and three cell clustering resolutions. Each preprocessing pipeline was applied to every scenario, producing corresponding PCA and UMAP plots. A set of quantitative metrics, including ARI, Silhouette Score, Calinski–Harabasz Index (CHI), Davies–Bouldin Index (DBI), and Mutual Information (MI), was also calculated by comparing the cell clusters generated by each preprocessing pipeline with the ground truth.

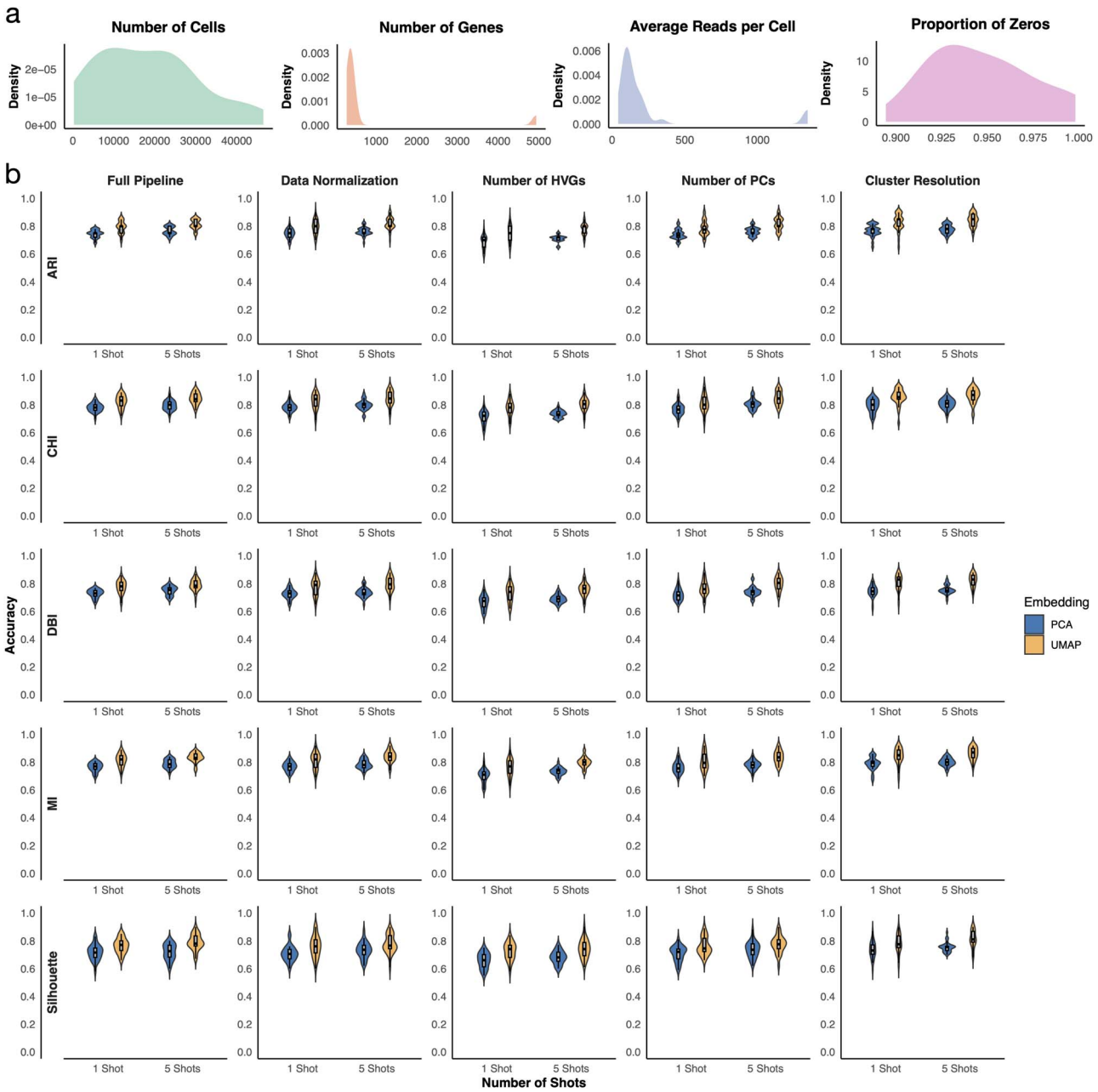

Figure 2 **a**, Distribution of the number of cells, number of genes, average total reads per cell, and proportion of zeros across 30 simulation scenarios, with each data point representing one simulation scenario. **b**, Proportion of cases in which GPT-4o correctly identified the better pipeline according to each metric, with each data point representing one simulation scenario and columns showing randomly chosen pipeline pairs overall or pairs differing in only one preprocessing step, as indicated by the column title.

These metrics are used to determine which preprocessing pipeline yields results that better reflect the underlying biology and therefore represent a superior preprocessing strategy.

For each scenario, we randomly selected 20 pairs of PCA or UMAP plots and prompted GPT-4o to choose the better one. Figure 2b shows that GPT-4o correctly identified the better preprocessing pipeline in most cases. For example, GPT-4o reached an accuracy of 80% using UMAP as input and ARI as the evaluation metric. The accuracy of GPT-4o increases slightly to 82% in a five-shot setting, where GPT-4o was prompted five times and the most frequent answer was used. GPT-4o remains highly accurate across evaluation metrics, and using UMAP as input shows slightly better performance than using PCA as input. We further performed a similar analysis on randomly selected pairs where only one of the four processing steps differed. GPT-4o demonstrated comparable one-shot and five-shot performance. These results suggest that GPT-4o can accurately and robustly identify the better preprocessing pipeline in pairwise comparisons. Note that UMAP and PCA plots are used solely as visual representations for pairwise comparison of preprocessing outcomes, rather than as quantitative validation metrics, and GPT-4o's selections based on these embeddings show strong agreement with multiple quantitative metrics, indicating that they serve as informative proxies for relative comparison of

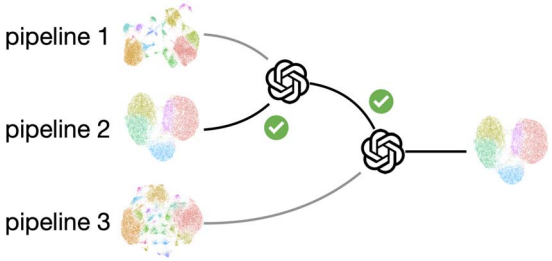

Figure 3 Cartoon illustrating the selection procedure of Lense.

preprocessing quality. Due to the limited improvement and increased computational burden of the five-shot approach, we used the one-shot approach in the subsequent analysis.

Based on these findings, we developed Lense, an LLM-powered preprocessing method for single-cell omics (Fig. 3). Lense fully integrates with the Seurat pipeline and requires only one line of code to perform preprocessing. Lense first runs the Seurat preprocessing pipeline with different combinations of algorithms and parameter values. By default, 72 combinations described above are executed, but users have the option to include fewer or additional runs. A UMAP is generated for each run, and the corresponding preprocessed results are stored. Lense then iteratively prompts GPT-4o to perform pairwise comparisons of UMAPs and selects the best UMAP among the 72 candidates. Finally, Lense retrieves and returns the preprocessing results corresponding to the selected UMAP plot, which can be used for downstream analyses such as cell type annotation and differential analysis. Optionally, Lense can also generate PCA plots and select the optimal preprocessing pipeline based on these plots.

We tested Lense on 30 scenarios and found that the preprocessing pipeline ranked as top by Lense consistently has among the best performance. For example, Lense with UMAP as input selected the preprocessing pipeline with the highest ARI in 73% of cases, and the selected preprocessing pipeline was among the top three in 90% of cases and among the top five in all cases (Fig. 4). The results were similar when PCA was used as input to Lense or when other evaluation metrics were used. The overall performance of Lense is substantially better than that of any single fixed preprocessing pipeline and is very close to the theoretical optimum ("oracle"), where the preprocessing pipeline with the highest ARI or other evaluation metric was used for each dataset (Fig. 5). These results suggest that a single fixed preprocessing pipeline is insufficient for handling datasets with highly diverse data distributions, and even the best-performing single pipeline may fail in certain cases (Supplementary Figures 1 and 2). In comparison, by adaptively selecting dataset-specific pipelines, Lense can effectively improve preprocessing efficacy.

We further evaluated the performance of Lense on two scRNA-seq datasets representing continuous differentiation processes (Fig. 6). The first dataset, derived from human bone marrow, captures the differentiation of CD34+ hematopoietic stem cells. The second dataset, from the mouse dentate gyrus, represents neurogenesis. Using the same evaluation procedure, we compared Lense with single fixed pipelines across multiple evaluation metrics. The results show that Lense achieves the best performance in nearly all scenarios. The pipeline selected by Lense consistently ranks among the top three across both datasets, with either PCA or UMAP as input and across different evaluation metrics (Fig. 6).

Although LLMs can yield variable responses to identical prompts, Lense produces highly reproducible results. Across repeated runs, the Pearson correlation of pipeline rankings exceeds 0.9 in most cases (Supplementary Figure 3). We further assessed Lense using Claude and Gemini as alternative backbone LLMs instead of GPT-4o (Supplementary Figures 4 and 5). In these settings, Lense still ranks the optimal pipeline as the top choice in more than half of the cases and within the top five in nearly all cases. In addition, computational efficiency analysis shows that Lense scales linearly with the number of input cells and completes analysis within one hour for datasets of up to 20 000 cells (Supplementary Figure 6). Together, these results demonstrate that Lense is reproducible, robust, and scalable.

Discussions

In summary, we developed Lense, which selects the optimal preprocessing pipeline by comparing PCA or UMAP plots generated from different pipelines. By formalizing preprocessing as a data-driven selection task enabled by high-level model reasoning, Lense provides a general framework for automating subjective steps in single-cell data analysis, with potential applications across diverse platforms and datasets.

The improved performance of Lense likely reflects not only its systematic exploration of a broad preprocessing space, but also its ability to harness the pattern-recognition capacity of LLMs when applied to visual representations of data structure. Low-dimensional embeddings condense complex, high-dimensional transcriptional relationships into spatial configurations that encode cluster compactness, separation, continuity, and the presence of small, potentially artifactual islands. Human analysts routinely rely on such visual cues to assess preprocessing quality, yet these evaluations are subjective and difficult to standardize across datasets and research groups. In contrast, an LLM trained on large-scale visual and textual corpora may implicitly capture general principles of structural organization, enabling more consistent recognition of embeddings that display clearer inter-cluster separation, reduced fragmentation, and more coherent global geometry. In this way, Lense operationalizes visual intuition as an automated and reproducible decision process, allowing high-level visual reasoning to guide model selection across numerous candidate pipelines. The resulting performance gains suggest that LLM-based visual assessment can detect structural signals aligned with biological validity, underscoring a potential role for multimodal foundation models in the evaluation and optimization of analytical workflows beyond traditional quantitative metrics.

In this study, Lense was primarily evaluated using simulated 10x Xenium data, as the impact of preprocessing on ST data remains less well characterized than in scRNA-seq. Nevertheless, the data distributions in ST can differ substantially from those in scRNA-seq, and may also vary considerably across spatial platforms. Therefore, the extent to which these findings generalize to scRNA-seq or to other ST technologies warrants further investigation. In addition, although the simulation data were generated based on the real dataset, their distribution may still deviate from that of the actual data. This potential discrepancy should be considered when interpreting the results of Lense in real-world applications.

ST datasets often contain a large number of cells, and the original expression matrix is typically too large to be directly provided to LLMs.

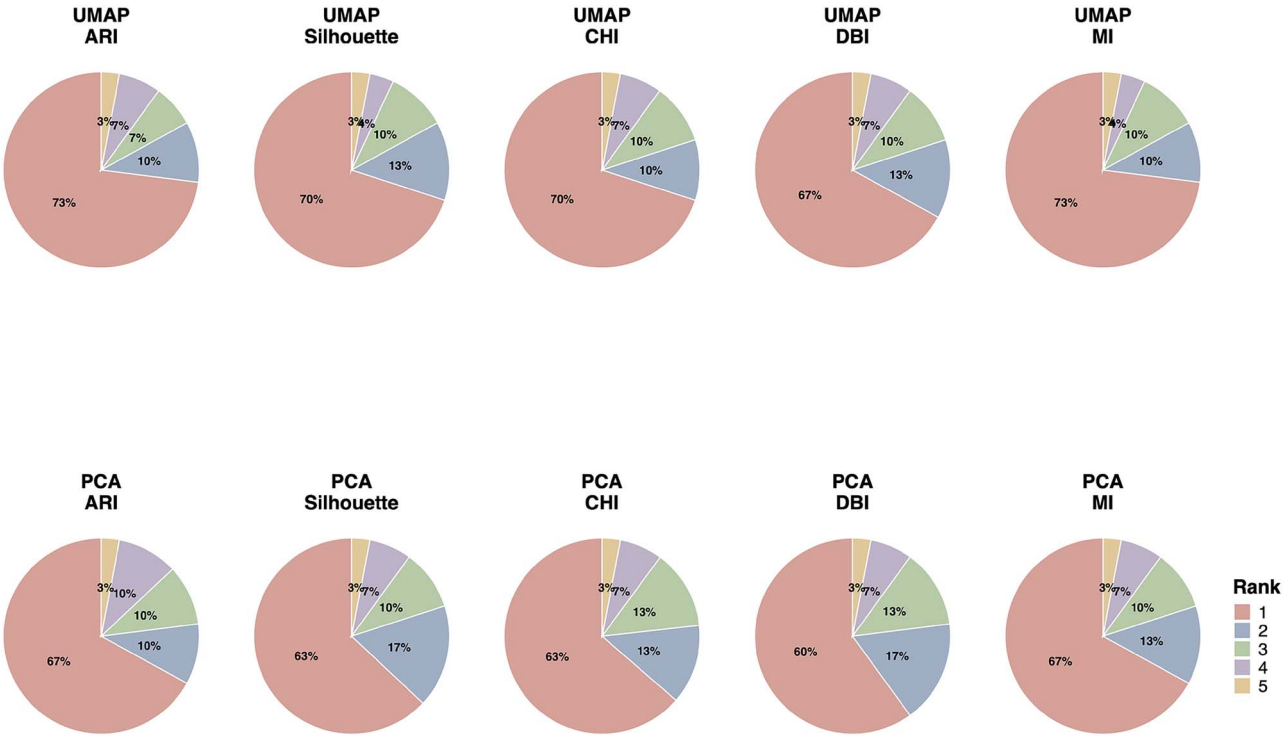

Figure 4 Proportion of simulation scenarios in which the pipeline selected by Lense achieved each rank according to each metric, with rows showing Lense using UMAP or PCA plots as input and columns corresponding to different evaluation metrics.

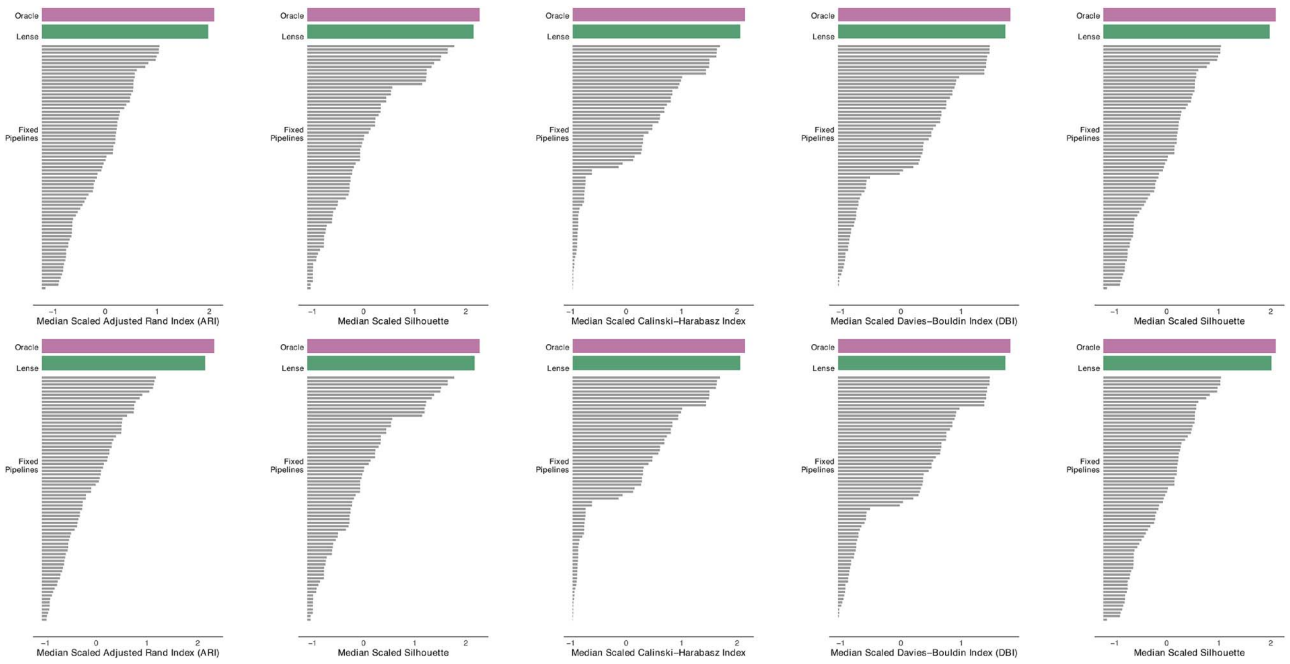

Figure 5 Median scaled evaluation metrics for the theoretical optimum ("oracle"), Lense, and individual fixed preprocessing pipelines across scenarios, with the original metrics scaled to have a mean of 0 and a standard deviation of 1 across methods within each scenario, rows showing Lense using UMAP or PCA plots as input, and columns corresponding to different evaluation metrics.

To efficiently convey the structure of the data, Lense leverages PCA or UMAP visualizations, which capture the low-dimensional embedding and clustering patterns of cells. Consequently, the performance of Lense may depend on the visual quality of the PCA or UMAP representation. For instance, visualization bias can arise when many points are plotted, as points rendered earlier may be obscured by

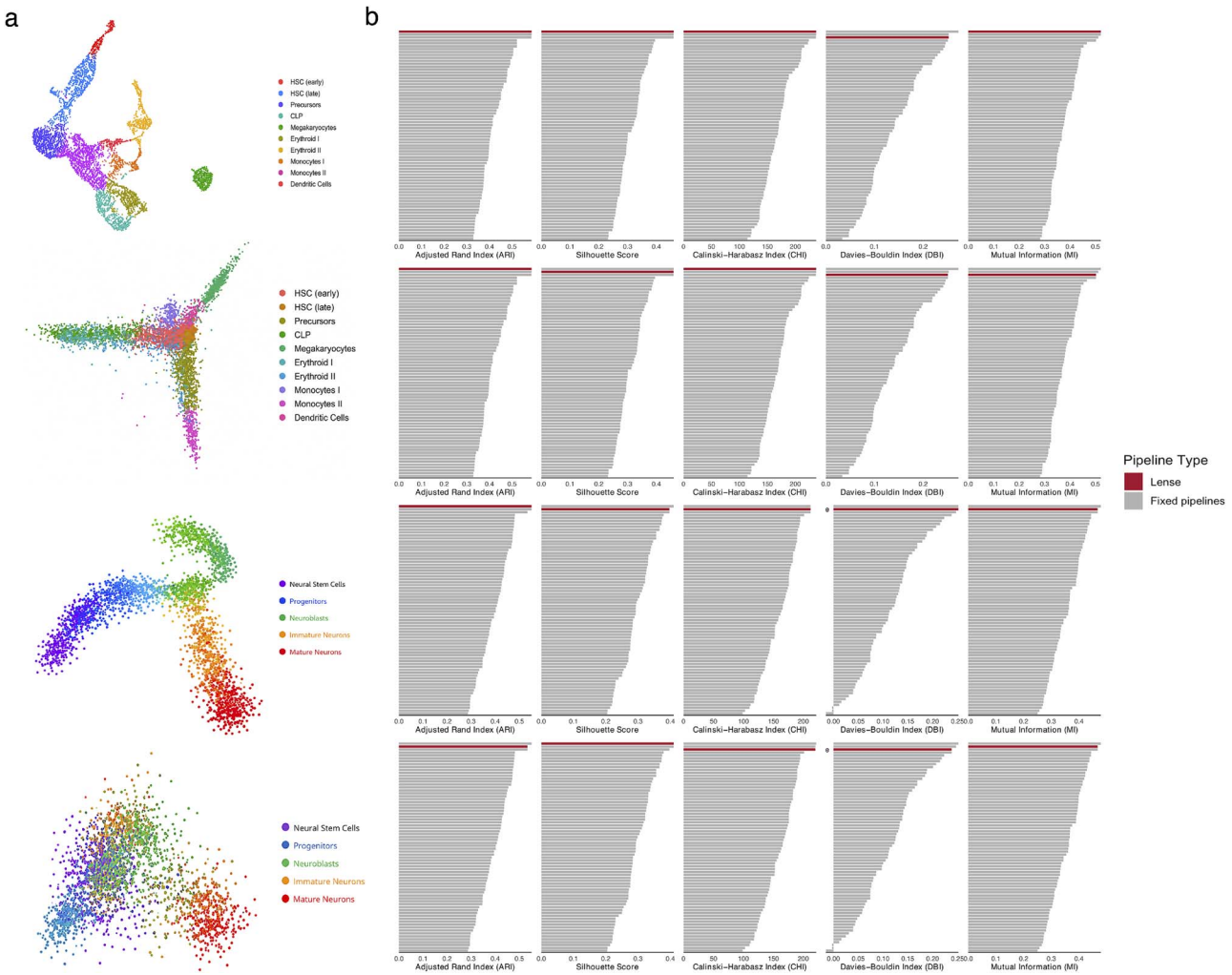

Figure 6 Performance of Lense on two scRNA-seq datasets representing continuous cell differentiation processes, from top to bottom: UMAP of the human CD34^{+} hematopoietic differentiation dataset, PCA of the same dataset, UMAP of the dentate gyrus neurogenesis dataset, and PCA of the dentate gyrus neurogenesis dataset. **a**, PCA or UMAP visualization of each dataset. **b**, Performance comparison between Lense and individual preprocessing pipelines.

those plotted later [16]. In addition, using similar colors for spatially adjacent clusters can make cluster boundaries difficult to distinguish [17]. These limitations may be mitigated by applying specialized visualization strategies [16, 17] to improve clarity and representation quality.

Lense is built upon the application programming interfaces (APIs) of proprietary LLMs, such as GPT-4o, which currently demonstrate strong visual and reasoning performance relative to other available models [12]. However, this reliance on proprietary systems introduces limitations related to reproducibility, transparency, and long-term sustainability. These models are accessed through commercial APIs, while their underlying architectures, training data, and update schedules are not publicly disclosed. Consequently, model behavior may evolve over time without explicit version control, potentially affecting the stability and exact reproducibility of previously reported findings. In addition, the inference costs associated with commercial LLM usage can be substantial, particularly for large-scale analyses. Such financial and infrastructural demands may restrict accessibility for research groups with limited resources.

Methods

Lense

Lense consists of two steps. In the first step, it preprocesses the input single-cell omics data using the Seurat pipeline with various combinations of parameter settings. In the second step, it uses GPT-4o to compare the PCA or UMAP plots generated in the first step and selects the optimal preprocessing pipeline. By default, UMAP plots are used in the second step of the pipeline.

Step 1: preprocessing with different parameters

The input to Lense is a Seurat object that contains a count matrix and has undergone appropriate cell filtering. The data can originate from various single-cell sequencing platforms, such as scRNA-seq or ST technologies with single-cell resolution. Lense then sequentially performs the following preprocessing steps using the Seurat pipeline: data normalization, identification of highly variable features, data scaling, PCA, selection of the number of PCs, cell clustering, and visualization.

By default, Lense evaluates 72 distinct preprocessing pipelines derived from combinations of the parameters described below. Users may optionally select a subset of these pipelines or define additional ones.

Normalization methods: one of six normalization approaches is selected:

- SCTransform (SCT) [18], implemented via the `SCTransform` function in the `sctransform` package
- Log-normalization (Lognorm): total read count normalization followed by log-transformation, implemented via `NormalizeData` in the `Seurat` package
- CLR: implemented via `NormalizeData` in `Seurat`
- Relative counts (RC): total read count normalization, implemented via `NormalizeData` in `Seurat`
- Log transformation: log-transformation, implemented manually
- Raw counts: no normalization applied

Highly variable feature selection: one of two options is selected:

- Using all I features
- Selecting a highly variable features using the `FindVariableFeatures` function in `Seurat` with default settings, where

$$a = 2 \times 10^{\lceil \log_{10}(I/2) \rceil - 1}$$

Number of PCs: the number of PCs is set to either 5 or 20.

Clustering resolution: cell clustering is performed using the `FindClusters` function in `Seurat`, with resolution parameters set to 0.2, 0.5, or 1.

All other steps follow the default parameters of the `Seurat` pipeline. Lense executes each preprocessing pipeline and records both the preprocessing results and the corresponding PCA or UMAP plot.

Step 2: selecting the optimal preprocessing pipeline

Lense then performs pairwise comparisons to select the optimal preprocessing pipeline. Suppose there are N preprocessing pipelines generated in the first step. Lense conducts $N-1$ pairwise comparisons sequentially. In the first comparison, Lense compares the first and second pipelines. In the ith comparison, Lense compares the $(i+1)$th pipeline with the pipeline selected in the $(i-1)$th comparison.

In each pairwise comparison, Lense uses the API provided by OpenAI to query the gpt-4o-2024-08-06 model. Two low-dimensional embeddings, generated using either UMAP or PCA, along with the following text prompt, are used as inputs: "You will be shown two UMAP plots. Please carefully examine both. Image 1 is labeled '1' and Image 2 is labeled '2'. Based on visual accuracy (cluster separation, boundary clarity, etc.), which one is better? Please respond with only: 1 or 2." Due to the randomness and lack of controllability in the output of LLMs, the gpt-4o-2024-08-06 model is prompted repeatedly until its response is either "1" or "2."

Collection of matched spatial and single-cell omics datasets

We collected the gene expression count matrices of ST samples generated using the 10x Genomics Xenium In Situ platform (https://www.10xgenomics.com/datasets/) from 10 tissue types, including human kidney, human liver, human pancreas, human lung, human bone marrow, human brain, human breast, human lung cancer, human heart, and mouse brain.

For each tissue type, we also collected the gene expression count matrix and the manually annotated cell type labels from a publicly available scRNA-seq dataset generated from the same tissue. Specifically, human liver data [19] were obtained from the Gene Expression Omnibus (GEO) under accession code GSE192742, human pancreas data [20] from GEO under accession code GSE221156, human bone marrow data [21] from GEO under accession code GSE166895, human lung cancer data [22] from ArrayExpress under accession codes E-MTAB-14560 and E-MTAB-14566, adult human heart data [23] from the European Nucleotide Archive (ENA) under accession code ERP123138, human breast data [24] from ArrayExpress under accession code E-MTAB-13664, human lung data [25] from GEO under accession code GSE161382, mouse brain data [26] from CELLxGENE Discover platform [27], human kidney data [28] from GEO under accession code GSE202109, and human brain data [29] from GEO under accession code GSE199762.

For scRNA-seq datasets that only provide Ensembl gene IDs, we used the biomaRt package [30] to convert the gene IDs into gene symbols. Genes without valid gene symbols or with duplicated mappings were removed.

Simulation of 10x Xenium datasets

A simulated 10x Xenium dataset was generated for each pair of real scRNA-seq and real 10x Xenium samples from the same tissue. Only genes shared between the two datasets were retained. A scaling factor was computed as the ratio of the mean library size of the real Xenium data to that of the scRNA-seq data. For each cell in the scRNA-seq dataset, a target library size was calculated by multiplying the cell's total read count by the scaling factor. The reads in that scRNA-seq cell were then randomly subsampled so that the total read count matched the target library size.

Collection of scRNA-seq datasets representing continuous differentiation processes

Two scRNA-seq datasets representing continuous differentiation processes were collected. The first dataset corresponds to human $CD34^+$ hematopoietic differentiation. The raw gene expression count matrix was obtained from the Human Cell Atlas data portal under project ID 091cf39b-01bc-42e5-9437-f419a66c8a45, which is associated with the study by [31]. This dataset comprises scRNA-seq profiles of hematopoietic stem and progenitor cells and captures continuous differentiation trajectories across multiple lineages. Cell type annotations were obtained from the original dataset metadata.

The second dataset corresponds to dentate gyrus neurogenesis and was obtained from the GEO under accession code GSE95753 [32]. This dataset contains scRNA-seq profiles from mouse dentate gyrus across multiple developmental stages, spanning embryonic to postnatal time points, and captures the progression of neurogenesis. Cell type annotations were obtained from the original study.

For both datasets, raw gene expression count matrices were converted into Seurat objects and used as input to the Lense pipeline, ensuring consistent preprocessing across all datasets.

Evaluation of clustering performance

Clustering performance was evaluated by comparing the cell clustering results from a preprocessing pipeline with the ground truth cell

type annotations inherited from the scRNA-seq dataset. Five quantitative metrics were computed: ARI, Silhouette Score, CHI, DBI, and MI. ARI was computed using the `adjustedRandIndex` function from the mclust package in R. The Silhouette Score was calculated using the `silhouette` function from the `cluster` package, based on Euclidean distances in the reduced dimensional space. The CHI and DBI were computed using the `cluster.stats` function from the `fpc` package. MI was calculated using the `mutinformation` function from the `infotheo` package. Since DBI is a metric where lower values indicate better clustering performance, we transformed it as $1 - \mathbf{DBI}$ so that higher values consistently correspond to better performance across all metrics.

Additional LLMs

In addition to the default OpenAI GPT-4o model, we evaluated Lense using Claude 3.5 Sonnet (Anthropic) and `Gemini 1.5 Pro` (Google). All models were accessed via their respective APIs and applied under the same protocol to select optimal preprocessing pipelines based on visual comparisons of low-dimensional embeddings (UMAP or PCA).

Key Points

- Default preprocessing often underperforms on diverse datasets, especially single-cell-resolution spatial transcriptomics, so dataset-specific tuning is needed.
- Lense uses an LLM to compare PCA or UMAP plots from many pipeline variants, then selects the best preprocessing configuration automatically.
- Integrated with Seurat and callable in one line, Lense replaces manual trial-and-error, improving consistency and reproducibility without extra bespoke steps.
- GPT-4o correctly picked the higher-ARI UMAP in about 80% one-shot and 82% five-shot tests across 30 scenarios, showing robust visual-quality judgment.
- On the same scenarios, Lense chose the true best pipeline in 73% of cases, was top three in 90% and top five in 100%, performing far better than any fixed pipeline and close to an oracle.

Author contributions

Jingyun Liu (Formal analysis, Software, Writing—original draft) and Zhicheng Ji (Conceptualization, Writing—original draft).

Supplementary material

Supplementary material is available at *Briefings in Bioinformatics* online.

Competing interests

All authors declare no competing interests.

Funding

The project was supported by the National Institutes of Health under Award Number U54AG075936 and R35GM154865.

Data availability

Lense is freely available on GitHub: https://github.com/jingyun20/Lense.

References

1. Tabula Sapiens Consortium, Jones RC, Karkanias J *et al.* The tabula sapiens: a multiple-organ, single-cell transcriptomic atlas of humans. *Science* 2022; **376**:eabl4896.
2. Consortium HuBMAP. The human body at cellular resolution: the NIH human biomolecular atlas program. *Nature* 2019; **574**:187–92.
3. Chen Z, Ji Z, Ngiow SF *et al.* TCF-1-centered transcriptional network drives an effector versus exhausted CD8 t cell-fate decision. *Immunity* 2019; **51**:840–855.e5. https://doi.org/10.1016/j.immuni.2019.09.013
4. Caushi JX, Zhang J, Ji Z *et al.* Transcriptional programs of neoantigen-specific TIL in anti-PD-1-treated lung cancers. *Nature* 2021; **596**:126–32. https://doi.org/10.1038/s41586-021-03752-4
5. Jackson C, Cherry C, Bom S *et al.* Distinct myeloid-derived suppressor cell populations in human glioblastoma. *Science* 2025; **387**:eabm5214. https://doi.org/10.1126/science.abm5214
6. Hao Y, Hao S, Andersen-Nissen E *et al.* Integrated analysis of multimodal single-cell data. *Cell* 2021; **184**:3573–3587.e29. https://doi.org/10.1016/j.cell.2021.04.048
7. Wolf FA, Angerer P, Theis FJ. SCANPY: large-scale single-cell gene expression data analysis. *Genome Biol* 2018; **19**:15.
8. Marco Salas S, Kuemmerle LB, Mattsson-Langseth C *et al.* Optimizing Xenium in situ data utility by quality assessment and best-practice analysis workflows. *Nat Methods* 2025; **22**:813–23. https://doi.org/10.1038/s41592-025-02617-2
9. Achiam J, Adler S, Agarwal S *et al.* GPT-4 technical report. *arXiv preprint arXiv:2303.08774* (2023).
10. Hou W, Ji Z. Assessing GPT-4 for cell type annotation in single-cell RNA-seq analysis. *Nat Methods* 2024; **21**:1462–5.
11. Shang X, Liao X, Ji Z, Hou W. Benchmarking large language models for genomic knowledge with GeneTuring. *Brief Bioinform.* 2025;**26**:bbaf492.
12. Hou W, Liu Q, Ma H *et al.* Assessing large multimodal models for one-shot learning and interpretability in biomedical image classification. *Adv Intell Syst* 2025; **7**:2400947.
13. Chen Y, Zou J. Simple and effective embedding model for single-cell biology built from ChatGPT. *Nat Biomed Eng* 2025; **9**:483–93.
14. Hu M, Alkhairy S, Lee I *et al.* Evaluation of large language models for discovery of gene set function. *Nat Methods* 2025; **22**:82–91. https://doi.org/10.1038/s41592-024-02525-x
15. Ma H, BIOSTAT 824 Student Consortium, Ji Z. Evaluating large language models in biomedical data science challenges through a classroom experiment. *Proc Natl Acad Sci* 2025; **122**:e2521062122.
16. Hou W, Ji Z. Unbiased visualization of single-cell genomic data with SCUBI. *Cell Rep Methods* 2022; **2**:100135. https://doi.org/10.1016/j.crmeth.2021.100135
17. Hou W, Ji Z. Palo: spatially aware color palette optimization for single-cell and spatial data. *Bioinformatics* 2022; **38**:3654–6. https://doi.org/10.1093/bioinformatics/btac368
18. Hafemeister C, Satija R. Normalization and variance stabilization of single-cell RNA-seq data using regularized negative binomial regression. *Genome Biol* 2019; **20**:296. https://doi.org/10.1186/s13059-019-1874-1
19. Guilliams M, Bonnardel J, Haest B *et al.* Spatial proteogenomics reveals distinct and evolutionarily conserved hepatic macrophage niches. *Cell* 2022; **185**:379–396.e38. https://doi.org/10.1016/j.cell.2021.12.018

20. Bandesh K, Motakis E., Nargund S. *et al.* Single-cell decoding of human islet cell type-specific alterations in type 2 diabetes reveals converging genetic-and state-driven β-cell gene expression defects. *bioRxiv* (2025).
21. Jardine L, Webb S, Goh I *et al.* Blood and immune development in human fetal bone marrow and down syndrome. *Nature* 2021; **598**:327–31. https://doi.org/10.1038/s41586-021-03929-x
22. Dong Y, Saglietti C, Bayard Q *et al.* Transcriptome analysis of archived tumors by visium, GeoMx DSP, and chromium reveals patient heterogeneity. *Nat Commun* 2025; **16**:4400. https://doi.org/10.1038/s41467-025-59005-9
23. Litviňuková M, Talavera-López C, Maatz H *et al.* Cells of the adult human heart. *Nature* 2020; **588**:466–72. https://doi.org/10.1038/s41586-020-2797-4
24. Reed AD, Pensa S, Steif A *et al.* A single-cell atlas enables mapping of homeostatic cellular shifts in the adult human breast. *Nat Genet* 2024; **56**:652–62. https://doi.org/10.1038/s41588-024-01688-9
25. Wang A, Chiou J, Poirion OB *et al.* Single-cell multiomic profiling of human lungs reveals cell-type-specific and age-dynamic control of SARS-CoV2 host genes. *Elife* 2020; **9**:e62522. https://doi.org/10.7554/eLife.62522
26. Govek KW, Chen S, Sgourdou P *et al.* Combined single-cell RNA-seq profiling and enhancer editing reveals critical spatiotemporal controls over thalamic nuclei formation in the murine embryo. *bioRxiv* 2022–02 (2022).
27. CELLxGENE Discover. *Developmental Trajectories of Thalamic Nuclei Revealed by Single-Cell Transcriptome Profiling and Shh Perturbation* 2022. https://cellxgene.cziscience.com/collections/d5cad3f0-56b6-4fbe-8f2b-be92a8c7820f. (10 April 2025, date last accessed).
28. McEvoy CM, Murphy JM, Zhang L *et al.* Single-cell profiling of healthy human kidney reveals features of sex-based transcriptional programs and tissue-specific immunity. *Nat Commun* 2022; **13**:7634. https://doi.org/10.1038/s41467-022-35297-z
29. Nascimento MA, Biagiotti S, Herranz-Pérez V *et al.* Protracted neuronal recruitment in the temporal lobes of young children. *Nature* 2024; **626**:1056–65. https://doi.org/10.1038/s41586-023-06981-x
30. Durinck S, Moreau Y, Kasprzyk A *et al.* BioMart and Bioconductor: a powerful link between biological databases and microarray data analysis. *Bioinformatics* 2005; **21**:3439–40. https://doi.org/10.1093/bioinformatics/bti525
31. Setty M, Kiseliovas V, Levine J *et al.* Characterization of cell fate probabilities in single-cell data with palantir. *Nat Biotechnol* 2019; **37**:451–60. https://doi.org/10.1038/s41587-019-0068-4
32. Habib N, Avraham-Davidi I, Basu A *et al.* Massively parallel single-nucleus RNA-seq with DroNc-seq. *Nat Methods* 2017; **14**:955–8. https://doi.org/10.1038/nmeth.4407

Briefings in Bioinformatics, 2026, **27**, bbag288
https://doi.org/10.1093/bib/bbag288
Problem Solving Protocol