A unified framework of realistic in silico data generation and statistical model inference for single-cell and spatial omics

Dongyuan Song[1], Qingyang Wang[2], Guanao Yan[2], Tianyang Liu[2], and Jingyi Jessica Li[1,2,3,4,5,6*]

Abstract

In the single-cell and spatial omics field, computational challenges include method benchmarking, data interpretation, and in silico data generation. To address these challenges, we propose an all-in-one statistical simulator, scDesign3, to generate realistic single-cell and spatial omics data, including various cell states, experimental designs, and feature modalities, by learning interpretable parameters from real datasets. Furthermore, using a unified probabilistic model for single-cell and spatial omics data, scDesgin3 can infer biologically meaningful parameters, assess the quality of cell clusters and trajectories, and generate in silico negative and positive controls for benchmarking computational tools.

[1] Bioinformatics Interdepartmental Ph.D. Program, University of California, Los Angeles, CA 90095-7246
[2] Department of Statistics, University of California, Los Angeles, CA 90095-1554
[3] Department of Human Genetics, University of California, Los Angeles, CA 90095-7088
[4] Department of Computational Medicine, University of California, Los Angeles, CA 90095-1766
[5] Department of Biostatistics, University of California, Los Angeles, CA 90095-1772
[6] Radcliffe Institute for Advanced Study, Harvard University, Cambridge, MA 02138
[*] To whom correspondence should be addressed. Email: jli@stat.ucla.edu

Introduction

Single-cell and spatial omics technologies have provided unprecedented multi-modal views of individual cells. As the earliest single-cell technologies, single-cell RNA-seq (scRNA-seq) enabled the measurement of transcriptome-wide gene expression levels and the discovery of novel cell types and continuous cell trajectories [1, 2]. Later, other single-cell omics technologies were developed to measure additional molecular feature modalities, including single-cell chromatin accessibility (e.g., scATAC-seq [3] and sci-ATAC-seq [4]), single-cell DNA methylation [5], and single-cell protein abundance (e.g., single-cell mass cytometry [6]). More recently, single-cell multi-omics technologies were invented to simultaneously measure more than one modality, such as SNARE-seq (gene expression and chromatin accessibility) [7] and CITE-seq (gene expression and surface protein abundance) [8]. In parallel to single-cell omics, spatial transcriptomics technologies were advanced to profile gene expression levels with spatial location information of cell neighborhoods (i.e., multi-cell resolution; e.g., 10x Visium [9] and Slide-seq [10]), individual cells (i.e., single-cell resolution; e.g., Slide-seqV2 [11]), or sub-cellular components (i.e., sub-cellular resolution; e.g., MERFISH [12]).

Thousands of computational methods have been developed to analyze single-cell and spatial omics data for various tasks [13], making method benchmarking a pressing challenge for method developers and users. Fair benchmarking relies on comprehensive evaluation metrics that reflect real data analytical goals; however, meaningful metrics usually require ground truths that are rarely available in real data. (For example, most real datasets contain “cell types” obtained by cell clustering and manual annotation without external validation; using such “cell types” as ground truths would biasedly favor the clustering method used in the original study.) Therefore, fair benchmarking demands in silico data that contain ground truths and mimic real data, calling for realistic simulators.

The demand for realistic simulators motivated two recent benchmark studies, in which 12 and 16 scRNA-seq simulators were evaluated [14, 15]. Due to the complexity of scRNA-seq data, these benchmarked simulators all require training on real scRNA-seq data, and they are more realistic than the de novo simulators that use no real data but generate synthetic data from theoretical models [15]. Although the benchmark studies found that the simulators scDesign2 [16], ZINB-WaVE [17], and muscat [18] can generate realistic scRNA-seq data from discrete cell types [14, 15], few simulators can generate realistic scRNA-seq data from continuous cell trajectories by mimicking real data [15, 19–22]. Moreover, realistic simulators are lacking for single-cell omics other than scRNA-seq, not to mention single-cell multi-omics and spatial transcriptomics. (To our knowledge, simATAC is the only scATAC-seq simulator that learns from real data, but it can only generate discrete cell types [23].) Hence, a large gap exists between the diverse benchmarking needs and the limited functionalities of existing simulators.

To fill in the gap, we introduce scDesign3, a realistic and most versatile simulator to date. As Fig. 1a shows, scDesign3 can generate realistic synthetic data from diverse settings, including cell latent structures (discrete cell types and continuous cell trajectories), feature modalities (e.g., gene expression, chromatin accessibility, methylation, protein abundance, and multi-omics), spatial coordinates, and experimental designs (batches and conditions). Note that the predecessor scDesign2 is a special case of scDesign3 for generating scRNA-seq data from discrete cell types. To our knowledge, scDesign3 offers the first probabilistic model that unifies the generation and inference for single-cell and spatial omics data. Equipped with interpretable parameters and a model likelihood, scDesign3 is beyond a versatile simulator and has unique advantages for generating customized in silico data, which can serve as negative and positive controls for computational analysis, and for assessing the quality of cell clusters and trajectories with statistical rigor (Fig. 2a).

Results

We verified scDesign3 as a realistic and versatile simulator in four exemplar settings where existing simulators have gaps: (1) scRNA-seq data of continuous cell trajectories, (2) spatial transcriptomics data, (3) single-cell epigenomics data, and (4) single-cell multi-omics data (Fig. 1). Under each setting, we show that the synthetic data of scDesign3 resemble the test data (i.e., left-out real data unused for training), confirming that the scDesign3 model fits well but does not overfit the training data.

In the first setting about continuous cell trajectories, scDesign3 mimics three scRNA-seq datasets containing single or bifurcating cell trajectories (datasets EMBRYO, MARROW, and PANCREAS in Table S3). Fig. 1b–c and Figs. S1–S3c–d show that scDesign3 generates realistic synthetic cells that resemble left-out real cells, as evidenced by high values (≥ 1.75) of mLISI (mean Local Inverse Simpson's Index), which indicates the degree of similarity between synthetic and real cells and has a perfect value of 2 [24]. Moreover, scDesign3 preserves five gene- and cell-specific characteristics (i.e., gene expression mean and variance, gene detection frequency, cell library size, and cell detection frequency) and, in particular, gene-gene correlations (Figs. S1–S3a–b). Since no existing simulators can generate cells in continuous trajectories by learning from real data, we benchmarked scDesign3 against ZINB-WaVE, muscat, and SPARSIM—three top-performing simulators for generating discrete cell types in previous benchmark studies [14, 15]—and a deep-learning-based simulator scGAN [25]. The results show that scDesign3 outperforms these four simulators in generating more realistic synthetic cells (by achieving higher mLISI values) and in better preserving the gene- and cell-specific characteristics and gene-gene correlations (Fig. 1b–c and Figs. S1–S3). In addition, scDesign3 can output the pseudotime truths of synthetic cells for benchmarking purposes, a functionality unavailable in existing simulators to our knowledge.

In the second setting about spatial transcriptomics, scDesign3 emulates two spatial transcriptomics datasets generated by the 10x Visium and Slide-seq technologies (datasets VISIUM and SLIDE in Table S3). First, Fig. 1d–e show that scDesign3 recapitulates the expression patterns of spatially variable genes (by achieving high correlations between the corresponding synthetic and real spatial patterns). Second, Figs. S4–S5a–b show that scDesign3 preserves the gene- and cell-specific characteristics mentioned above and gene-gene correlations. Third, Figs. S4–S5c–d use PCA and UMAP embeddings to confirm that the synthetic data of scDesign3 resemble the test data (mLISI values ≥ 1.87). Notably, in these examples, scDesign3 generates spatial transcriptomics data from spatial coordinates without cell type annotations (i.e., scDesign3-spatial; see **Methods** 1.1.2). Figs. S4–S5 show that these synthetic data of scDesign3 are similarly realistic compared to the synthetic data scDesign3 generates under an ideal scenario where annotated cell types are available (i.e., scDesign3-ideal; see **Methods** 1.1.2). These results confirm scDesign3's ability to recapitulate cell heterogeneity without needing cell type annotations. Moreover, by fitting a model for spatial transcriptomics data, scDesign3 can estimate a smooth function for every gene's expected expression levels at spatial coordinates, a functionality unachievable by existing scRNA-seq simulators.

In the third setting about single-cell epigenomics, scDesign3 resembles two single-cell chromatin accessibility datasets profiled by the sci-ATAC-seq and 10x scATAC-seq protocols (datasets SCIATAC and ATAC in Table S3). For both protocols, scDesign3 generates realistic synthetic cells (with each cell represented as a vector of genomic regions' read counts) despite the higher sparsity of single-cell ATAC-seq data compared to scRNA-seq data (Fig. 1g left; Fig. S7). Moreover, coupled with our newly proposed read simulator scReadSim [26], scDesign3 extends the simulation of synthetic cells from the count level to the read level, unblocking its application for benchmarking read-level bioinformatics tools (Fig. 1g right).

In the fourth setting about single-cell multi-omics, scDesign3 mimics a CITE-seq dataset (dataset CITE in Table S3) and simulates a multi-omics dataset from separately measured RNA expression and DNA methylation modalities (dataset SCGEM in Table S3). First, scDesign3 resembles the CITE-seq dataset by simultaneously simulating the expression levels of 1000 highly variable genes and 10 surface proteins. Fig. 1f shows that the RNA and protein expression levels of four exemplary surface proteins are highly consistent between the synthetic data of scDesign3 and the test data. Moreover, scDesign3 recapitulates the correlations between the RNA and protein expression levels of the 10 surface proteins (Fig. S8b). Second, scDesign3 simulates a single-cell multi-omics dataset with joint RNA expression and DNA methylation modalities by learning from (1) two single-omics datasets measuring the two modalities separately (Fig. 1h left) and (2) joint low-dimensional embeddings of the two single-omics datasets. This synthetic multi-omics dataset preserves the cell trajectory in the two single-omics datasets (Fig. 1h right). The functionality to generate multi-omics data from single-omics data allows scDesign3 to benchmark the computational methods that integrate modalities from unmatched cells [27].

Providing the first universal probabilistic model for single-cell and spatial omics data, scDesign3 has broad applications beyond generating realistic synthetic data. We summarize the prominent applications of the scDesign3 model in three aspects: model parameters, model selection, and model alteration (Fig. 2a).

First, the scDesign3 model has an interpretable parametric structure consisting of genes' marginal distributional parameters and pairwise gene correlations, which have direct biological relevance. In addition to being interpretable, the scDesign3 model is flexible to incorporate cell covariates (such as cell type, pseudotime, and spatial coordinates) via the use of generalized additive models (see **Methods** 1.1.2), making the scDesign3 model fit well to various single-cell and spatial omics data—a property confirmed by scDesign3's realistic simulation in the above four settings (Fig. 1). The combined interpretability and flexibility enables scDesign3 to estimate the possibly non-linear relationship between every gene's mean expression and cell covariates, thus allowing statistical inference of gene expression changes between cell types, along cell trajectories (Fig. 2b), and across spatial coordinates (Fig. 2c). Besides inferring every gene's expression characteristics, scDesign3 also estimates pairwise gene correlations conditional on cell covariates, thus providing insights into the possible gene regulatory relationships within each cell type, at a cell differentiation time, or in a spatial region. Specifically, scDesign3 estimates gene correlations by two statistical techniques, Gaussian copula and vine copula, which have complementary advantages (see **Methods** 1.1.3): Gaussian copula is fast to fit but only outputs a gene correlation matrix; vine copula is slow to fit but outputs a hierarchical gene correlation network (a "vine" with the top layer indicating the most highly correlated genes, i.e., "hub genes") and thus more interpretable. As an example application to a dataset containing four human peripheral blood mononuclear cell (PBMC) types (ZHENGMIX4 in Table S3), Fig 2d shows that Gaussian copula reveals similar gene correlation matrices for similar cell types (regulatory T cells vs. naive cytotoxic T cells) and distinct gene correlation matrices for distinct cell types (CD14+ monocytes vs. naive cytotoxic T cells). Moreover, vine copula discovers canonical cell-type marker genes as hub genes: *LYZ* for CD14+ monocytes and *CD79A* for B cells.

Second, scDesign3 outputs the model likelihood, enabling likelihood-based model selection criteria such as Akaike information criterion (AIC) and Bayesian information criterion (BIC). This model selection functionality allows scDesign3 to evaluate the "goodness-of-fit" of a model to data and to compare competing models. A noteworthy application of this functionality is to evaluate how well an inferred latent variable (e.g., cell cluster assignment or cell pseudotime) describes data, thus enabling us to compare cell clustering results and trajectory inference results without needing ground truths. To our knowledge, no existing approaches can evaluate the quality of inferred cell pseudotime without ground truths, so scDesign3 fills this gap. We demonstrate that scDesign3 BIC is a reasonable "unsupervised" criterion for assessing both pseudotime inference and cell clustering quality. For pseudotime inference, scDesign3 BIC is strongly correlated (mean absolute Spearman correlation > 0.95) with the "supervised" R^2, which measures the consistency between the true and inferred (or perturbed) pseudotime values, on multiple synthetic datasets with true pseudotime (Fig. 2e top; Fig. S9a). Further, scDesign3 BIC agrees with UMAP visualization: compared to TSCAN and Monocle3, the pseudotime inferred by Slingshot has the best (smallest) BIC and best agrees with the low-dimensional representation of the cell manifold (Fig. 2e bottom). For cell clustering, we benchmark scDesign3 BIC against the "supervised" adjusted Rand index (ARI), which requires true cell cluster labels, and a newly proposed unsupervised criterion, clustering deviation index (CDI) [28], on eight datasets with known cell types in a published benchmark study [29]. The results show that scDesign3 BIC has good agreement with ARI (mean absolute Spearman correlation > 0.7) and has better or similar performance compared to CDI's performance on six out of the eight datasets (Fig. S9b).

Third, scDesign3 has a model alteration functionality enabled by its transparent probabilistic modeling and interpretable parameters: given the scDesign3 model parameters estimated on real data, users can alter the model parameters to reflect a hypothesis (i.e., a hypothetical truth) and generate the corresponding synthetic data that bear real data characteristics. Hence, users can flexibly generate synthetic data with varying ground truths for comprehensive benchmarking of computational methods. We argue that this functionality is a vital advantage scDesign3 has over deep-learning based simulators [25], which cannot be easily altered to reflect a specific hypothesis. We demonstrate how to use this model alteration functionality in three examples. In the first example, scDesign3 generates synthetic data with different cell-type-specific condition effects (Fig. 2f). In the real data (CONDITION in Table S3), gene *IFI6*'s expression is up-regulated after stimulation in both CD16+ monocytes and B cells (Fig. 2f top-left). With scDesign3's fitted model, users can alter *IFI6*'s mean parameters to make *IFI6*'s expression up-regulated by stimulation in both cell types (Fig. 2f top-right), unchanged by stimulation in both cell types (Fig. 2f bottom-left), or up-regulated by stimulation in CD16+ monocytes only (Fig. 2f bottom-right). In the second example, scDesign3 generates synthetic datasets with or without batch effects (Fig. 2g). Trained on a real dataset (BATCH in Table S3) containing two batches with batch effects (Fig. 2g left), scDesign3's model, if without alteration, can generate synthetic data retaining the batch effects (Fig. 2g middle), or it can have the batch parameter altered to generate synthetic data without batch effects (Fig. 2g right). In the third example, scDesign3 generates synthetic data under two hypotheses: the null hypothesis (H_0) that only one cell type exists and the alternative hypothesis (H_1) that two cell types exist (Fig. 2h). Given a real dataset (ZHENGMIX4 in Table S3) containing two cell types (Fig. 2h left), the scDesign3 model can be fitted in two ways: under H_1, the model is fitted using the cell type information (Fig. 2h middle); under H_0, the model is fitted by assuming all cells are of one type (Fig. 2h right). The two fitted models can generate the corresponding synthetic data under H_1 and H_0. Particularly, the synthetic data under H_0 can serve as the negative control for benchmarking computational pipelines that use cell clustering to identify the possible existence of cell types.

In summary, scDesign3 is the first omnibus model-based simulator for single-cell and spatial omics data to accommodate different cell states (discrete cell types and continuous cell trajectories), diverse omics features (e.g., gene expression, chromatin accessibility, protein abundance, and DNA methylation), and complex experimental designs (e.g., batches and conditions). Besides generating realistic synthetic data, scDesign3 offers a comprehensive interpretation of real data, thanks to its use of transparent modeling and interpretable parameters. Specifically, scDesign3 estimates the relationship between every feature (e.g., gene) and cell covariates, along with pairwise feature correlations. Moreover, scDesign3 allows likelihood-based model selection to assess the quality of inferred cell clusters and trajectories output by computational methods. Uniquely, scDesign3 can generate synthetic data under specific hypotheses (e.g., no differential expression, no batch effects, and no cell types) by altering its model parameters. Overall, scDesign3 is a multi-functional suite for benchmarking computational methods and interpreting single-cell and spatial omics data.

Data Availability

The scDesign3 package is available at https://github.com/SONGDONGYUAN1994/scDesign3.

Competing interests

The authors declare no competing interests.

Acknowledgements

The authors appreciate the comments and feedback from the members of the Junction of Statistics and Biology at UCLA (http://jsb.ucla.edu).

Funding

This work was supported by the following grants: National Science Foundation DBI-1846216 and DMS-2113754, NIH/NIGMS R01GM120507 and R35GM140888, Johnson & Johnson WiSTEM2D Award, Sloan Research Fellowship, UCLA David Geffen School of Medicine W. M. Keck Foundation Junior Faculty Award, and the Chan-Zuckerberg Initiative Single-Cell Biology Data Insights Grant (to J.J.L.). J.J.L. was a fellow at the Radcliffe Institute for Advanced Study at Harvard University in 2022-2023 while she was writing this paper.

Figures

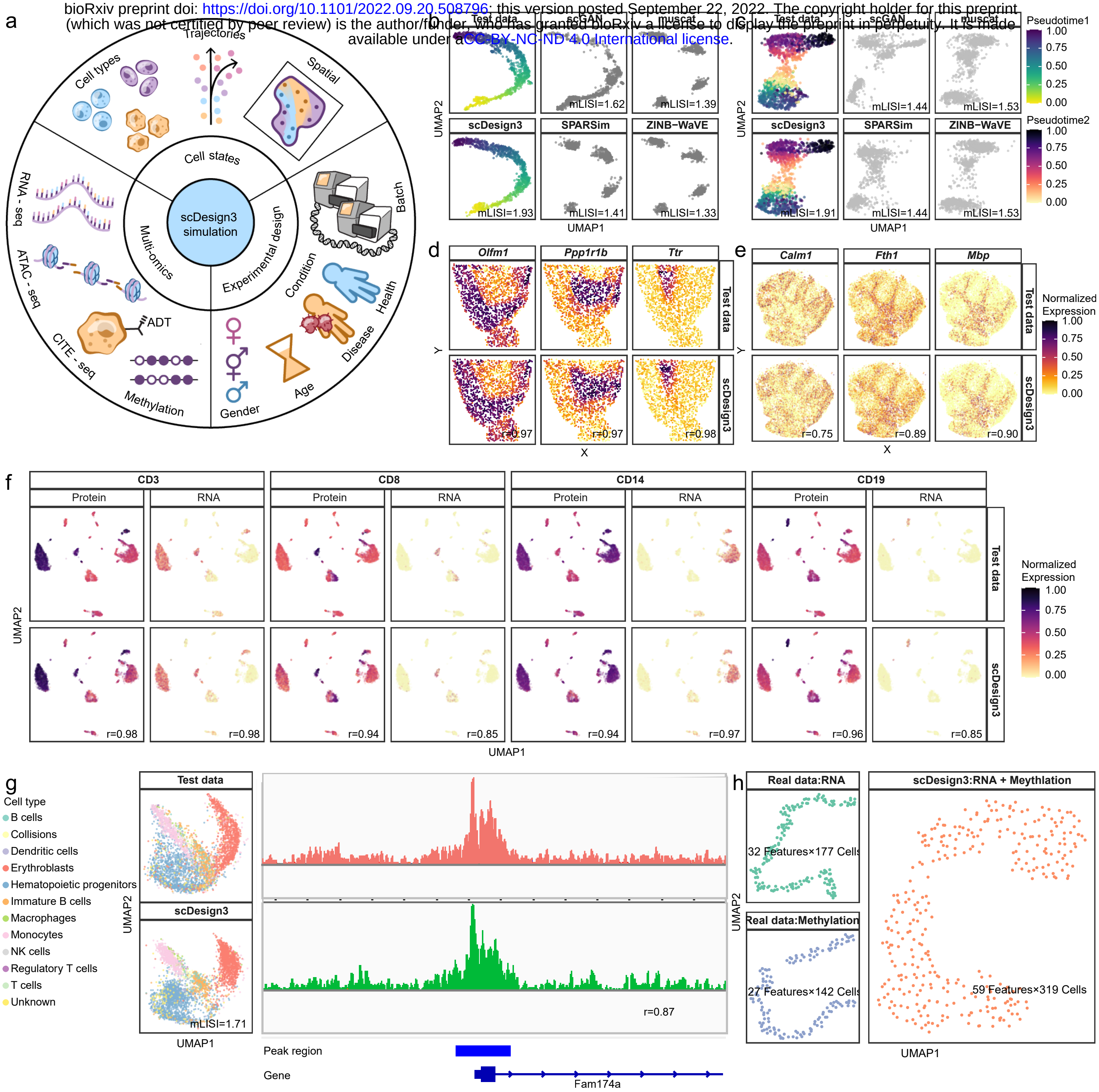


Figure 1: scDesign3 generates realistic synthetic data of diverse single-cell and spatial omics technologies. **a**, An overview of scDesign3's simulation functionalities: cell states (e.g., discrete types, continuous trajectories, and spatial locations); multi-omics modalities (e.g., RNA-seq, ATAC-seq, and CITE-seq); experimental designs (e.g., batches and conditions). **b–c**, scDesign3 outperforms existing simulators scGAN, muscat, SPARSim, and ZINB-WaVE in simulating scRNA-seq datasets with a single trajectory (b) and bifurcating trajectories (c). Larger mLISI values represent better resemblance between synthetic data and test data. **d–e**, scDesign3 simulates realistic gene expression patterns for spatial transcriptomics technologies 10x Visium (d) and Slide-seq (e). Large Pearson correlation coefficients (r) represent similar spatial patterns in synthetic and test data. **f**, scDesign3 simulates realistic CITE-seq data. Four genes' protein and RNA abundances are shown on the cell UMAP embeddings in test data (top) and the synthetic data (bottom). Large r represent similar expression patterns in synthetic and test data. **g**, scDesign3 simulates a realistic sci-ATAC-seq dataset at both the count level (left: UMAP visualizations of real and synthetic cells in terms of peak counts) and the read level (right: pseudobulk read coverages; coupled with scReadSim [26]). **h**, scDesign3 generates a multi-omics (RNA expression + DNA methylation) dataset (right) by learning from real data that only have a single modality (left). The synthetic data preserve the linear cell topology.

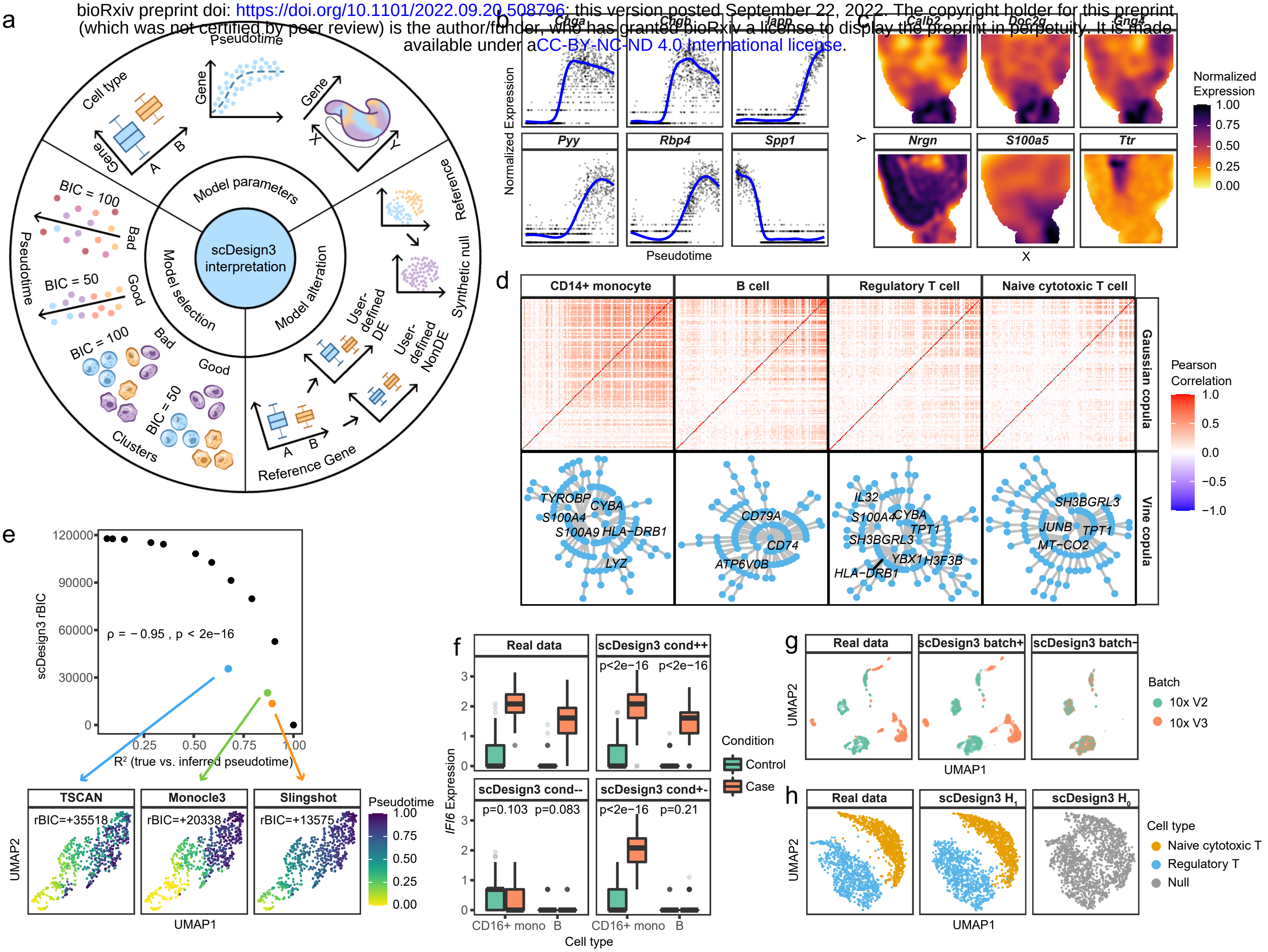


Figure 2: scDesign3 enables comprehensive interpretation of real data. **a**, An overview of scDesign3's interpretation functionalities based on its model parameters, model selection capacity, and model alteration capacity. **b**, scDesign3 estimates six genes' expression trends along cell pseudotime that indicates cell differentiation (scRNA-seq dataset PANCREAS in Table S3). **c**, scDesign3 estimates six genes' expression trends across spatial coordinates (10x Visium spatial dataset VISIUM in Table S3). **d**, scDesign3 estimates cell-type-specific gene correlations in four cell types (scRNA-seq dataset ZHENGMIX4 in Table S3): pairwise gene correlation matrices by Gaussian copula (top); vine representations by vine copula (bottom), with genes in the first layer (roughly the genes strongly correlated) labeled. **e**, scDesign3's model selection functionality allows the evaluation of pseudotime quality using the Bayesian information criterion (BIC). Three pseudotime inference methods—TSCAN, Monocle3, and Slingshot—have BICs evaluated on a synthetic scRNA-seq dataset generated by scDesign3 (based on EMBYRO in Table S3) with true cell pseudotimes. For interpretability, we plot the relative BIC (rBIC) by subtracting the smallest BIC value in **e** so that the rBIC starts from 0. Top: scDesign3 rBIC (calculated without true cell pseudotimes) vs. R^2 between true and inferred pseudotimes (blue: TSCAN; green: Monocle3; orange: Slingshot; black: perturbed true pseudotime as reference, see Methods 1.3.5). The strong negative correlation (Spearman's rank correlation coefficient $\rho = -0.95$) indicates that scDesign3 BIC measures pseudotime quality effectively. Bottom: visualization of the inferred pseudotime by TSCAN, Monocle3, and Slingshot; Slingshot's smallest BIC (best quality) agrees with the visualization. **f**, scDesign3's model alteration functionality allows user to specify the ground truths of cell-type-specific condition effects. In the dataset CONDITION (Table S3), gene *IFI6* is up-regulated in two cell types (CD16+ monocytes and B cells) from control (green) to stimulation (red). With its parametric model, scDesign3 can simulate data where the gene is up-regulated in both cell types (cond$++$), unchanged in both cell types (cond$--$), or only up-regulated in the first cell type CD16+ monocytes (cond$+-$). **g**, scDesign3's model alteration functionality allows it to simulate data with or without batch effects. The real dataset (BATCH in Table S3) contains two batches (10x v2 and v3) (left). scDesign3 can preserve the batch effects in its synthetic data (middle: batch$+$) or generates synthetic data without batch effects (right: batch$-$). **h**, scDesign3's model alteration functionality allows it to synthesize null data that do not have cell clusters. The real dataset (ZHENGMIX4 in Table S3) contains two cell types (left). scDesign3 can resemble the two cell types under the alternative hypothesis (H_1) that two cell types exist (middle). In contrast, under the null hypothesis (H_0) that only one cell type exists, scDesign3 can generate a synthetic null dataset that resembles the real data except the cell type number (right).

References

[1] Fuchou Tang, Catalin Barbacioru, Yangzhou Wang, Ellen Nordman, Clarence Lee, Nanlan Xu, Xiaohui Wang, John Bodeau, Brian B Tuch, Asim Siddiqui, et al. mrna-seq whole-transcriptome analysis of a single cell. *Nature methods*, 6(5):377–382, 2009.

[2] Junyue Cao, Malte Spielmann, Xiaojie Qiu, Xingfan Huang, Daniel M Ibrahim, Andrew J Hill, Fan Zhang, Stefan Mundlos, Lena Christiansen, Frank J Steemers, et al. The single-cell transcriptional landscape of mammalian organogenesis. *Nature*, 566(7745):496–502, 2019.

[3] Jason D Buenrostro, Beijing Wu, Ulrike M Litzenburger, Dave Ruff, Michael L Gonzales, Michael P Snyder, Howard Y Chang, and William J Greenleaf. Single-cell chromatin accessibility reveals principles of regulatory variation. *Nature*, 523(7561):486–490, 2015.

[4] Darren A Cusanovich, Riza Daza, Andrew Adey, Hannah A Pliner, Lena Christiansen, Kevin L Gunderson, Frank J Steemers, Cole Trapnell, and Jay Shendure. Multiplex single-cell profiling of chromatin accessibility by combinatorial cellular indexing. *Science*, 348(6237):910–914, 2015.

[5] Ino D Karemaker and Michiel Vermeulen. Single-cell dna methylation profiling: technologies and biological applications. *Trends in biotechnology*, 36(9):952–965, 2018.

[6] Sean C Bendall, Erin F Simonds, Peng Qiu, El-ad D Amir, Peter O Krutzik, Rachel Finck, Robert V Bruggner, Rachel Melamed, Angelica Trejo, Olga I Ornatsky, et al. Single-cell mass cytometry of differential immune and drug responses across a human hematopoietic continuum. *Science*, 332(6030):687–696, 2011.

[7] Song Chen, Blue B Lake, and Kun Zhang. High-throughput sequencing of the transcriptome and chromatin accessibility in the same cell. *Nature biotechnology*, 37(12):1452–1457, 2019.

[8] Marlon Stoeckius, Christoph Hafemeister, William Stephenson, Brian Houck-Loomis, Pratip K Chattopadhyay, Harold Swerdlow, Rahul Satija, and Peter Smibert. Simultaneous epitope and transcriptome measurement in single cells. *Nature methods*, 14(9):865–868, 2017.

[9] Nikhil Rao, Sheila Clark, and Olivia Habern. Bridging genomics and tissue pathology: 10x genomics explores new frontiers with the visium spatial gene expression solution. *Genetic Engineering & Biotechnology News*, 40(2):50–51, 2020.

[10] Samuel G Rodriques, Robert R Stickels, Aleksandrina Goeva, Carly A Martin, Evan Murray, Charles R Vanderburg, Joshua Welch, Linlin M Chen, Fei Chen, and Evan Z Macosko. Slide-seq: A scalable technology for measuring genome-wide expression at high spatial resolution. *Science*, 363(6434):1463–1467, 2019.

[11] Robert R Stickels, Evan Murray, Pawan Kumar, Jilong Li, Jamie L Marshall, Daniela J Di Bella, Paola Arlotta, Evan Z Macosko, and Fei Chen. Highly sensitive spatial transcriptomics at near-cellular resolution with slide-seqv2. *Nature biotechnology*, 39(3):313–319, 2021.

[12] Jeffrey R Moffitt, Dhananjay Bambah-Mukku, Stephen W Eichhorn, Eric Vaughn, Karthik Shekhar, Julio D Perez, Nimrod D Rubinstein, Junjie Hao, Aviv Regev, Catherine Dulac, et al. Molecular, spatial, and functional single-cell profiling of the hypothalamic preoptic region. *Science*, 362(6416):eaau5324, 2018.

[13] Mirjana Efremova and Sarah A Teichmann. Computational methods for single-cell omics across modalities. *Nature methods*, 17(1):14–17, 2020.

[14] Yue Cao, Pengyi Yang, and Jean Yee Hwa Yang. A benchmark study of simulation methods for single-cell rna sequencing data. *Nature communications*, 12(1):1–12, 2021.

[15] Helena L Crowell, Sarah X Morillo Leonardo, Charlotte Soneson, and Mark D Robinson. Built on sand: the shaky foundations of simulating single-cell rna sequencing data. *bioRxiv*, 2021.

[16] Tianyi Sun, Dongyuan Song, Wei Vivian Li, and Jingyi Jessica Li. scdesign2: a transparent simulator that generates high-fidelity single-cell gene expression count data with gene correlations captured. *Genome biology*, 22(1):1–37, 2021.

[17] Davide Risso, Fanny Perraudeau, Svetlana Gribkova, Sandrine Dudoit, and Jean-Philippe Vert. A general and flexible method for signal extraction from single-cell rna-seq data. *Nature communications*, 9(1):1–17, 2018.

[18] Helena L Crowell, Charlotte Soneson, Pierre-Luc Germain, Daniela Calini, Ludovic Collin, Catarina Raposo, Dheeraj Malhotra, and Mark D Robinson. Muscat detects subpopulation-specific state transitions from multi-sample multi-condition single-cell transcriptomics data. *Nature communications*, 11(1):1–12, 2020.

[19] Robrecht Cannoodt, Wouter Saelens, Louise Deconinck, and Yvan Saeys. Spearheading future omics analyses using dyngen, a multi-modal simulator of single cells. *Nature Communications*, 12(1):1–9, 2021.

[20] Payam Dibaeinia and Saurabh Sinha. Sergio: a single-cell expression simulator guided by gene regulatory networks. *Cell systems*, 11(3):252–271, 2020.

[21] Nikolaos Papadopoulos, Parra R Gonzalo, and Johannes Söding. Prosstt: probabilistic simulation of single-cell rna-seq data for complex differentiation processes. *Bioinformatics*, 35(18):3517–3519, 2019.

[22] Jinjin Tian, Jiebiao Wang, and Kathryn Roeder. Esco: single cell expression simulation incorporating gene co-expression. *Bioinformatics*, 37(16):2374–2381, 2021.

[23] Zeinab Navidi, Lin Zhang, and Bo Wang. simatac: a single-cell atac-seq simulation framework. *Genome biology*, 22(1):1–16, 2021.

[24] Ilya Korsunsky, Nghia Millard, Jean Fan, Kamil Slowikowski, Fan Zhang, Kevin Wei, Yuriy Baglaenko, Michael Brenner, Po-ru Loh, and Soumya Raychaudhuri. Fast, sensitive and accurate integration of single-cell data with harmony. *Nature methods*, 16(12):1289–1296, 2019.

[25] Mohamed Marouf, Pierre Machart, Vikas Bansal, Christoph Kilian, Daniel S Magruder, Christian F Krebs, and Stefan Bonn. Realistic in silico generation and augmentation of single-cell rna-seq data using generative adversarial networks. *Nature communications*, 11(1):1–12, 2020.

[26] Guanao Yan and Jingyi Jessica Li. screadsim: a single-cell multi-omics read simulator. *bioRxiv*, 2022.

[27] Ricard Argelaguet, Anna SE Cuomo, Oliver Stegle, and John C Marioni. Computational principles and challenges in single-cell data integration. *Nature biotechnology*, 39(10):1202–1215, 2021.

[28] Jiyuan Fang, Cliburn Chan, Kouros Owzar, Liuyang Wang, Diyuan Qin, Qi-Jing Li, and Jichun Xie. Clustering deviation index (cdi): A robust and accurate unsupervised measure for evaluating scrna-seq data clustering. *bioRxiv*, 2022.

[29] Angelo Duò, Mark D Robinson, and Charlotte Soneson. A systematic performance evaluation of clustering methods for single-cell rna-seq data. *F1000Research*, 7, 2018.

[30] D Mikis Stasinopoulos and Robert A Rigby. Generalized additive models for location scale and shape (gamlss) in r. *Journal of Statistical Software*, 23:1–46, 2008.

[31] Yuqing Zhang, Giovanni Parmigiani, and W Evan Johnson. Combat-seq: batch effect adjustment for rna-seq count data. *NAR genomics and bioinformatics*, 2(3):lqaa078, 2020.

[32] Simon N Wood. *Generalized additive models: an introduction with R*. chapman and hall/CRC, 2006.

[33] EE Kammann and Matthew P Wand. Geoadditive models. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 52(1):1–18, 2003.

[34] Claudia Czado. *Analyzing Dependent Data with Vine Copulas*. Springer, New York, 2019.

[35] Aaron TL Lun, Davis J McCarthy, and John C Marioni. A step-by-step workflow for low-level analysis of single-cell rna-seq data with bioconductor. *F1000Research*, 5, 2016.

[36] Tim Stuart, Avi Srivastava, Shaista Madad, Caleb A Lareau, and Rahul Satija. Single-cell chromatin state analysis with signac. *Nature methods*, 18(11):1333–1341, 2021.

[37] Jiaqiang Zhu, Shiquan Sun, and Xiang Zhou. Spark-x: non-parametric modeling enables scalable and robust detection of spatial expression patterns for large spatial transcriptomic studies. *Genome Biology*, 22(1):184, 2021.

[38] Kelly Street, Davide Risso, Russell B Fletcher, Diya Das, John Ngai, Nir Yosef, Elizabeth Purdom, and Sandrine Dudoit. Slingshot: cell lineage and pseudotime inference for single-cell transcriptomics. *BMC genomics*, 19(1):1–16, 2018.

[39] Tim Bedford and Roger M. Cooke. Vines: A new graphical model for dependent random variables. *Annals of Statistics*, 30(4):1031–1068, 2002.

[40] Ansuman T Satpathy, Jeffrey M Granja, Kathryn E Yost, Yanyan Qi, Francesca Meschi, Geoffrey P McDermott, Brett N Olsen, Maxwell R Mumbach, Sarah E Pierce, M Ryan Corces, et al. Massively parallel single-cell chromatin landscapes of human immune cell development and intratumoral t cell exhaustion. *Nature biotechnology*, 37(8):925–936, 2019.

[41] Jiarui Ding, Xian Adiconis, Sean K Simmons, Monika S Kowalczyk, Cynthia C Hession, Nemanja D Marjanovic, Travis K Hughes, Marc H Wadsworth, Tyler Burks, Lan T Nguyen, et al. Systematic comparison of single-cell and single-nucleus rna-sequencing methods. *Nature biotechnology*, 38(6):737–746, 2020.

[42] Sophie Petropoulos, Daniel Edsgärd, Björn Reinius, Qiaolin Deng, Sarita Pauliina Panula, Simone Codeluppi, Alvaro Plaza Reyes, Sten Linnarsson, Rickard Sandberg, and Fredrik Lanner. Single-cell rna-seq reveals lineage and x chromosome dynamics in human preimplantation embryos. *Cell*, 165(4):1012–1026, 2016.

[43] Hyun Min Kang, Meena Subramaniam, Sasha Targ, Michelle Nguyen, Lenka Maliskova, Elizabeth McCarthy, Eunice Wan, Simon Wong, Lauren Byrnes, Cristina M Lanata, et al. Multiplexed droplet single-cell rna-sequencing using natural genetic variation. *Nature biotechnology*, 36(1):89–94, 2018.

[44] Franziska Paul, Ya'ara Arkin, Amir Giladi, Diego Adhemar Jaitin, Ephraim Kenigsberg, Hadas Keren-Shaul, Deborah Winter, David Lara-Astiaso, Meital Gury, Assaf Weiner, et al. Transcriptional heterogeneity and lineage commitment in myeloid progenitors. *Cell*, 163(7): 1663–1677, 2015.

[45] Aimée Bastidas-Ponce, Sophie Tritschler, Leander Dony, Katharina Scheibner, Marta Tarquis-Medina, Ciro Salinno, Silvia Schirge, Ingo Burtscher, Anika Böttcher, Fabian J Theis, et al. Comprehensive single cell mrna profiling reveals a detailed roadmap for pancreatic endocrinogenesis. *Development*, 146(12):dev173849, 2019.

[46] Lih Feng Cheow, Elise T Courtois, Yuliana Tan, Ramya Viswanathan, Qiaorui Xing, Rui Zhen Tan, Daniel SW Tan, Paul Robson, Yuin-Han Loh, Stephen R Quake, et al. Single-cell

multimodal profiling reveals cellular epigenetic heterogeneity. *Nature methods*, 13(10):833–836, 2016.

[47] Darren A Cusanovich, Andrew J Hill, Delasa Aghamirzaie, Riza M Daza, Hannah A Pliner, Joel B Berletch, Galina N Filippova, Xingfan Huang, Lena Christiansen, William S DeWitt, et al. A single-cell atlas of in vivo mammalian chromatin accessibility. *Cell*, 174(5):1309–1324, 2018.

[48] Satija Lab. *stxBrain.SeuratData: 10X Genomics Visium Mouse Brain Dataset*, 2019. R package version 0.1.1.

[49] Grace XY Zheng, Jessica M Terry, Phillip Belgrader, Paul Ryvkin, Zachary W Bent, Ryan Wilson, Solongo B Ziraldo, Tobias D Wheeler, Geoff P McDermott, Junjie Zhu, et al. Massively parallel digital transcriptional profiling of single cells. *Nature communications*, 8(1):1–12, 2017.

[50] Kai Cao, Yiguang Hong, and Lin Wan. Manifold alignment for heterogeneous single-cell multi-omics data integration using pamona. *Bioinformatics*, 38(1):211–219, 2022.

1 Methods

1.1 The generative model of scDesign3

1.1.1 Mathematical notations of scDesign3's training data

The training data of scDesign3 contain three matrices: a cell-by-feature matrix (e.g., features are genes or chromatin regions), a cell-by-state-covariate matrix (e.g., cell-state covariates include the cell type, pseudotime, or spatial coordinate), and an optional cell-by-design-covariate matrix (e.g., design covariates include the batch or condition).

Mathematically, first, we denote by $\mathbf{Y} = [Y_{ij}] \in \mathbb{R}^{n\times m}$ the cell-by-feature matrix with n cells as rows, m features as columns, and Y_{ij} as the measurement of feature j in cell i. For single-cell sequencing data, $\mathbf{Y}$ is often a count matrix (i.e., $\mathbf{Y} \in \mathbb{N}^{n\times m}$, with Y_{ij} indicating the read or unique molecular identifier (UMI) count of feature j in cell i); then the sequencing depth (i.e., total number of reads) is $N = \sum_{i=1}^{n}\sum_{j=1}^{m} Y_{ij}$.

Second, we denote by $\mathbf{X} = [\mathbf{x}_1, \cdots, \mathbf{x}_n]^\mathsf{T} \in \mathbb{R}^{n\times p}$ the cell-by-state-covariate matrix with n cells as rows and p cell-state covariates as columns. Typical cell-state covariates include the cell type ($p = 1$ categorical variable), the cell pseudotime in p lineage trajectories (p continuous variables), and the 2- or 3-dimensional cell spatial coordinates ($p = 2$ or 3 continuous variables).

Third, we denote by $\mathbf{Z} \in \mathbb{R}^{n\times q}$ the cell-by-design-covariate matrix with n cells as rows and q design covariates as columns. Example design covariates are categorical variables such as the batch and condition. Note that $\mathbf{Z}$ is optional: it is not required if cells are from a single condition and measured in a single batch. To simplify the discussion, in the following text, we write $\mathbf{Z} = [\mathbf{b}, \mathbf{c}]$, where $\mathbf{b} = (b_1, \ldots, b_n)^\mathsf{T}$ has $b_i \in \{1, \cdots, B\}$ representing cell i's batch, and $\mathbf{c} = (c_1, \ldots, c_n)^\mathsf{T}$ has $c_i \in \{1, \cdots, C\}$ representing cell i's condition.

1.1.2 Modeling features' marginal distributions

For each feature $j = 1, \ldots, m$ in every cell $i = 1, \ldots, n$, the measurement Y_{ij}—conditional on cell i's state covariates $\mathbf{x_i}$ and design covariates $\mathbf{z}_i = (b_i, c_i)^\mathsf{T}$—is assumed to follow a distribution $F_j(\cdot \mid \mathbf{x}_i, \mathbf{z}_i\ ;\ \mu_{ij}, \sigma_{ij}, p_{ij})$, which is specified as the generalized additive model for location, scale and shape (GAMLSS) [30] (i.e., the distribution family depends on feature j only, but the parameters depend on both feature j and cell i):

$$\begin{cases} Y_{ij} \mid \mathbf{x}_i, \mathbf{z}_i & \overset{\text{ind}}{\sim} F_j(\cdot \mid \mathbf{x}_i, \mathbf{z}_i\ ;\ \mu_{ij}, \sigma_{ij}, p_{ij}) \\ \theta_j(\mu_{ij}) & = \alpha_{j0} + \alpha_{jb_i} + \alpha_{jc_i} + f_{jc_i}(\mathbf{x}_i) \\ \log(\sigma_{ij}) & = \beta_{j0} + \beta_{jb_i} + \beta_{jc_i} + g_{jc_i}(\mathbf{x}_i) \\ \operatorname{logit}(p_{ij}) & = \gamma_{j0} + \gamma_{jb_i} + \gamma_{jc_i} + h_{jc_i}(\mathbf{x}_i) \end{cases}, \tag{1.1}$$

where $\theta_j(\cdot)$ denotes feature j's specific link function of the mean parameter μ_{ij}, depending on F_j (Table S1); σ_{ij} denotes the scale parameter (e.g., standard deviation or dispersion); p_{ij} denotes the zero-inflation proportion parameter. Note that μ_{ij}, σ_{ij}, and p_{ij} do not always co-exist, depending on the form of F_j (Table S1). To ensure identifiability, for $j = 1, \ldots, m$, we set $\alpha_{jb_i} = \beta_{jb_i} = \gamma_{jb_i} = 0$ when $b_i = 1$ and $\alpha_{jc_i} = \beta_{jc_i} = \gamma_{jc_i} = 0$ when $c_i = 1$.

$\theta_j(\mu_{ij})$ is assumed to have feature j's specific intercept α_{j0}, batch b_i's effect α_{jb_i} (specific to feature j), condition c_i's effect α_{jc_i} (specific to feature j), and cell-state covariates $\mathbf{x}_i$'s effect $f_{jc_i}(\mathbf{x}_i)$ (specific to feature j and condition c_i).

$\log(\sigma_{ij})$ is assumed to have feature j's specific intercept β_{j0}, batch b_i's effect β_{jb_i} (specific to feature j), condition c_i's effect β_{jc_i} (specific to feature j), and cell-state covariates $\mathbf{x}_i$'s effect $g_{jc_i}(\mathbf{x}_i)$ (specific to feature j and condition c_i).

$\text{logit}(p_{ij})$ is assumed to have feature j's specific intercept γ_{j0}, batch b_i's effect γ_{jb_i} (specific to feature j), condition c_i's effect γ_{jc_i} (specific to feature j), and cell-state covariates $\mathbf{x}_i$'s effect $h_{jc_i}(\mathbf{x}_i)$ (specific to feature j and condition c_i).

For $\theta_j(\mu_{ij})$, $\log(\sigma_{ij})$, and $\text{logit}(p_{ij})$, the interaction effects are considered between the condition and cell-state covariates, but not between the batch and cell-state covariates. This modeling choice is made based on empirical observations and the simplicity preference [31].

Note that if only the mean parameter μ_{ij} is assumed to depend on the state covariates $\mathbf{x}_i$, batch b_i, and condition c_i, then the GAMLSS degenerates to a generalized additive model (GAM) [32].

Depending on the modality of feature j (e.g., a gene's UMI count), scDesign3 specifies F_j to be one of the six distributions: Gaussian (Normal), Bernoulli, Poisson, Negative Binomial (NB), Zero-inflated Poisson (ZIP), and Zero-inflated Negative Binomial (ZINB); see Table S1 for the specifications. Different specifications of F_j correspond to different link functions $\theta_j(\cdot)$ and parameters; see Table S1 for the details.

Depending on cell i's cell-state covariates $\mathbf{x}_i$, scDesign3 specifies the functions $f_{jc_i}(\cdot)$, $g_{jc_i}(\cdot)$, and $h_{jc_i}(\cdot)$ in the corresponding forms. See Table S2 for the details. Below are the three typical forms of $f_{jc_i}(\cdot)$.

(1) When the cell-state covariate is the cell type (out of a total of K_C cell types) and $\mathbf{X} = (x_1, \ldots, x_n)^\mathsf{T}$ with $x_i \in \{1, \ldots, K_C\}$,

$$f_{jc_i}(x_i) = \alpha_{jc_ix_i} \, ,$$

which corresponds to the cell-type x_i's effect on feature j in condition c_i. Note that for identifiability, $\alpha_{jc_ix_i} = 0$ if $c_i = 1$.

(2) When the cell-state covariates are the cell pseudotimes in p lineage trajectories, i.e., $\mathbf{x}_i = (x_{i1}, \ldots, x_{ip})^\mathsf{T}$ with x_{il} indicating cell i's pseudotime in the l-th lineage trajectory,

$$f_{jc_i}(\mathbf{x}_i) = \sum_{l=1}^{p} \sum_{k=1}^{K} b_{jc_ilk}(x_{il})\beta_{jc_ilk} \, ,$$

where $\sum_{k=1}^{K} b_{jc_ilk}(\cdot)\beta_{jc_ilk}$ is a cubic spline function for pseudotime in the l-th lineage. This formulation means that feature j under condition c_i has a specific smooth pattern in lineage l. The exact choice K is not critical as long as K is not too small (see [32]); we set $K = 10$ as default.

(3) When the cell-state covariates are 2-dimensional spatial locations, i.e., $\mathbf{x}_i = (x_{i1}, x_{i2})^{\mathsf{T}}$ with x_{i1} and x_{i2} indicating cell i's spatial coordinates,

$$f_{jc_i}(\mathbf{x}_i) = f_{jc_i}^{\mathrm{GP}}(x_{i1}, x_{i2}, K)\,,$$

a low-rank Gaussian process smoother described in [32, 33], where K is the number of basis functions. This formulation means that feature j under condition c_i has a smooth 2-dimensional function (i.e., a surface). The exact choice K is not critical as long as K is large (see [32]); we set $K = 400$ as default.

The distribution of $(Y_{ij} \mid \mathbf{x}_i, \mathbf{z}_i)$ in (1.1) is fitted by the function `gamlss()` in the R package `gamlss` (version 5.4-3) or the function `gam()` in the R package `mgcv` (version 1.8-40). The fitted distribution is denoted as $\hat{F}_j(\cdot \mid \mathbf{x}_i, \mathbf{z}_i)$, $i = 1, \ldots, n$; $j = 1, \ldots, m$.

1.1.3 Modeling features' joint distribution

For cell $i = 1, \ldots, n$, we denote its measurements of the m features as a random vector $\mathbf{Y}_i = (Y_{i1}, \ldots, Y_{im})^{\mathsf{T}}$, whose joint distribution—conditional on cell i's state covariates $\mathbf{x}_i$ and design covariates $\mathbf{z}_i$—is denoted as $F(\cdot \mid \mathbf{x}_i, \mathbf{z}_i) : \mathbb{R}^m \to [0,1]$. Section 1.1.2 specifies $F_j(\cdot \mid \mathbf{x}_i, \mathbf{z}_i)$, the distribution of $(Y_{ij} \mid \mathbf{x}_i, \mathbf{z}_i)$, $j = 1, \ldots, m$. In scDesign3, the joint cumulative distribution function (CDF) $F(\cdot \mid \mathbf{x}_i, \mathbf{z}_i)$ is modeled from the marginal CDFs $F_1(\cdot \mid \mathbf{x}_i, \mathbf{z}_i), \ldots, F_m(\cdot \mid \mathbf{x}_i, \mathbf{z}_i)$ using the copula $C(\cdot \mid \mathbf{x}_i, \mathbf{z}_i) : [0,1]^m \to [0,1]$:

$$F(\mathbf{y}_i \mid \mathbf{x}_i, \mathbf{z}_i) = C\left(F_1(y_{i1} \mid \mathbf{x}_i, \mathbf{z}_i), \cdots, F_m(y_{im} \mid \mathbf{x}_i, \mathbf{z}_i) \mid \mathbf{x}_i, \mathbf{z}_i\right)\,,$$

where $\mathbf{y}_i = (y_{i1}, \ldots, y_{im})^{\mathsf{T}}$ is a realization of $\mathbf{Y}_i = (Y_{i1}, \ldots, Y_{im})^{\mathsf{T}}$.

The copula $C(\cdot \mid \mathbf{x}_i, \mathbf{z}_i)$ can be (1) the Gaussian copula or (2) the vine copula, specified below.

The Gaussian copula is defined as

$$\begin{aligned} & C\left(F_1(y_{i1} \mid \mathbf{x}_i, \mathbf{z}_i), \cdots, F_m(y_{im} \mid \mathbf{x}_i, \mathbf{z}_i) \mid \mathbf{x}_i, \mathbf{z}_i\right) \\ = {} & \Phi_m\left(\Phi^{-1}(F_1(y_{i1} \mid \mathbf{x}_i, \mathbf{z}_i)), \cdots, \Phi^{-1}(F_m(y_{i1} \mid \mathbf{x}_i, \mathbf{z}_i));\ \mathbf{R}(\mathbf{x}_i, \mathbf{z}_i)\right)\,, \end{aligned}$$

where Φ^{-1} denotes the inverse of the CDF of the standard Gaussian distribution, $\Phi_m(\cdot; \mathbf{R}(\mathbf{x}_i, \mathbf{z}_i))$ denotes the CDF of an m-dimensional Gaussian distribution with a zero mean vector and a covariance matrix equal to the correlation matrix $\mathbf{R}(\mathbf{x}_i, \mathbf{z}_i)$.

An issue with the Gaussian copula is that the likelihood calculation is not straightforward in the high-dimensional case when $\frac{m(m-1)}{2} > n$. Since the sample correlation matrix $\hat{\mathbf{R}}(\mathbf{x}_i, \mathbf{z}_i)$, as an estimator of $\mathbf{R}(\mathbf{x}_i, \mathbf{z}_i)$, is not invertible, the likelihood cannot be computed based on $\hat{\mathbf{R}}(\mathbf{x}_i, \mathbf{z}_i)$. To address this issue, we consider the vine copula.

The vine copula is a way to "decompose" a high-dimensional copula into a sequence of low-dimensional copulas, e.g., bivariate copulas in which every pair of features is modeled as a bivariate Gaussian distribution. In short, the vine copula provides a regular vine (R-vine) structure that uses conditioning to sequentially decompose an m-dimensional copula into a sequence of bivariate copulas; then the m-dimensional copula density function is approximated by the product of the bivariate copula density functions [34]. The vine copula is advantageous to the Gaussian copula because it enables the likelihood calculation in the high-dimensional case. A detailed definition of the vine copula is in **Supplementary Methods** 2.

To estimate $C(\cdot \mid \mathbf{x}_i, \mathbf{z}_i)$ as either the Gaussian or vine copula, we use the plug-in approach that takes the estimated $\hat{F}_1(\cdot \mid \mathbf{x}_i, \mathbf{z}_i), \ldots, \hat{F}_m(\cdot \mid \mathbf{x}_i, \mathbf{z}_i)$ from Section 1.1.2. Specifically, when $\hat{F}_j(\cdot \mid \mathbf{x}_i, \mathbf{z}_i)$ is a continuous distribution, each observed y_{ij} is transformed as $u_{ij} = \hat{F}_j(y_{ij} \mid \mathbf{x}_i, \mathbf{z}_i)$. When $\hat{F}_j(\cdot \mid \mathbf{x}_i, \mathbf{z}_i)$ is a discrete distribution with the support on non-negative integers (e.g., the Poisson distribution), since the Gaussian and vine copulas assume that features follow continuous distributions, we use the distributional transformation as in [16]:

$$u_{ij} = v_{ij}\hat{F}_j(y_{ij} - 1 \mid \mathbf{x}_i, \mathbf{z}_i) + (1 - v_{ij})\hat{F}_j(y_{ij} \mid \mathbf{x}_i, \mathbf{z}_i)\,,\ y_{ij} = 1, 2, \ldots\,,$$

where v_{ij}'s are sampled independently from $\mathrm{Uniform}[0, 1]$, $i = 1, \ldots, n$; $j = 1, \ldots, m$. To unify and simplify our notations, we write $u_{ij} = \tilde{F}_j(y_{ij} \mid \mathbf{x}_i, \mathbf{z}_i)$, where $\tilde{F}_j(\cdot \mid \mathbf{x}_i, \mathbf{z}_i)$ is the CDF of a continuous distribution.

Then $C(\cdot \mid \mathbf{x}_i, \mathbf{z}_i)$ is estimated from $\mathbf{u}_1, \ldots, \mathbf{u}_n$, where $\mathbf{u}_i = (u_{i1}, \ldots, u_{im})^{\mathsf{T}}$. For the Gaussian copula, we use the function `cora()` in the R package `Rfast` (version 2.0.6); specifically, $\hat{\mathbf{R}}(\mathbf{x}_i, \mathbf{z}_i)$ is the sample correlation matrix of $\Phi^{-1}(\mathbf{u}_1), \ldots, \Phi^{-1}(\mathbf{u}_n)$, where $\Phi^{-1}(\mathbf{u}_i) = (\Phi^{-1}(u_{i1}), \ldots, \Phi^{-1}(u_{im}))^{\mathsf{T}}$. For the vine copula, we use the function `vinecop()` in R package `rvinecoplib` (version 0.6.2.1.1).

Then the estimated joint distribution $\hat{F}(\cdot \mid \mathbf{x}_i, \mathbf{z}_i)$ is

$$\hat{F}(\mathbf{y}_i \mid \mathbf{x}_i, \mathbf{z}_i) = \hat{C}\left(\tilde{F}_1(y_{i1} \mid \mathbf{x}_i, \mathbf{z}_i), \cdots, \tilde{F}_m(y_{im} \mid \mathbf{x}_i, \mathbf{z}_i) \,\middle|\, \mathbf{x}_i, \mathbf{z}_i\right). \tag{1.2}$$

1.1.4 Model likelihood, AIC, and BIC

Given (1.2), the estimated probability density function of cell i's m-dimensional feature vector $\mathbf{y}_i$, conditional on the cell-state covariates $\mathbf{x}_i$ and the design covariates $\mathbf{z}_i$, is

$$\hat{f}(\mathbf{y}_i \mid \mathbf{x}_i, \mathbf{z}_i) = \hat{c}\left(\tilde{F}_1(y_{i1} \mid \mathbf{x}_i, \mathbf{z}_i), \cdots, \tilde{F}_m(y_{im} \mid \mathbf{x}_i, \mathbf{z}_i) \,\middle|\, \mathbf{x}_i, \mathbf{z}_i\right) \prod_{j=1}^{m} \tilde{f}_j(y_{ij} \mid \mathbf{x}_i, \mathbf{z}_i)\,,$$

where $\hat{c}(\cdot \mid \mathbf{x}_i, \mathbf{z}_i)$ is the probability density function of $\hat{C}(\cdot \mid \mathbf{x}_i, \mathbf{z}_i)$, and $\tilde{f}_j(\cdot \mid \mathbf{x}_i, \mathbf{z}_i)$ is the probability density function of $\tilde{F}_j(\cdot \mid \mathbf{x}_i, \mathbf{z}_i)$. Hence, the log-likelihood is

$$\begin{aligned}\ell &= \sum_{i=1}^{n} \log \hat{f}(\mathbf{y}_i \mid \mathbf{x}_i, \mathbf{z}_i) \\ &= \sum_{i=1}^{n} \log \hat{c}\left(\tilde{F}_1(y_{i1} \mid \mathbf{x}_i, \mathbf{z}_i), \cdots, \tilde{F}_m(y_{im} \Bigm| \mathbf{x}_i, \mathbf{z}_i) \mid \mathbf{x}_i, \mathbf{z}_i\right) + \sum_{i=1}^{n}\sum_{j=1}^{m} \log \tilde{f}_j(y_{ij} \mid \mathbf{x}_i, \mathbf{z}_i) \\ &= \ell^{\text{Copula}} + \ell^{\text{Marginal}},\end{aligned}$$

so the log-likelihood can be written as the sum of a copula log-likelihood and a marginal log-likelihood.

Given k model parameters and n cells (sample size is the number of cells), the Akaike information criterion (AIC) and Bayesian information criterion (BIC) are

$$\begin{aligned}\text{AIC} &= 2k - 2\ell\,; \\ \text{BIC} &= 2k\log(n) - 2\ell\,.\end{aligned}$$

Because of the likelihood decomposition, the AIC and BIC are also decomposable

$$\begin{aligned}\text{AIC} &= \text{AIC}^{\text{Copula}} + \text{AIC}^{\text{Marginal}}\,, \\ \text{BIC} &= \text{BIC}^{\text{Copula}} + \text{BIC}^{\text{Marginal}}\,,\end{aligned}$$

where $\text{AIC}^{\text{Copula}}$ and $\text{BIC}^{\text{Copula}}$ only include the number of parameters in $\hat{c}(\cdot \mid \mathbf{x}_i, \mathbf{z}_i)$, and $\text{AIC}^{\text{Marginal}}$ and $\text{BIC}^{\text{Marginal}}$ only include the total number of parameters in $\tilde{f}_1(\cdot \mid \mathbf{x}_i, \mathbf{z}_i), \ldots, \tilde{f}_m(\cdot \mid \mathbf{x}_i, \mathbf{z}_i)$.

1.2 Synthetic data generation by scDesign3

To generate a synthetic cell-by-feature matrix $\mathbf{Y}' \in \mathbb{R}^{n' \times m}$, which contains n' synthetic cells and the same m features as in the training data, scDesign3 requires a cell-by-state-covariate matrix $\mathbf{X} \in \mathbb{R}^{n' \times p}$ and a cell-by-design-covariate matrix $\mathbf{Z} \in \mathbb{N}^{n' \times 2}$ specified for the n' synthetic cells. Based on the fitted distributions in Sections 1.1.2 and 1.1.3, we sample n' synthetic cells in the following steps.

First, for each synthetic cell i', given its cell-state covariates $\mathbf{x}_{i'}$ and design covariates $\mathbf{z}_{i'}$, we sample its m-dimensional probability vector from the m-dimensional copula estimated in Section 1.1.3:

$$(U_{i'1}, \ldots, U_{i'm})^{\mathsf{T}} \sim \hat{C}(\cdot \mid \mathbf{x}_{i'}, \mathbf{z}_{i'})\,,\; i' = 1, \ldots, n'\,.$$

Second, based on the m features' fitted marginal distributions in Section 1.1.2, we calculate the conditional distribution of $Y_{i'j}$, the measurement of feature j in synthetic cell i', given the synthetic cell's cell-state covariates $\mathbf{x}_{i'}$ and design covariates $\mathbf{z}_{i'} = (b_{i'}, c_{i'})^{\mathsf{T}}$, where $b_{i'} \in \{1, \ldots, B\}$ and

$c_{i'} \in \{1, \dots, C\}$:

$$Y_{i'j} \mid \mathbf{x}_{i'}, \mathbf{z}_{i'} \sim \hat{F}_j(\cdot \mid \mathbf{x}_{i'}, \mathbf{z}_{i'}) = F_j(\cdot \mid \mathbf{x}_{i'}, \mathbf{z}_{i'} \; ; \; \hat{\mu}_{i'j}, \, \hat{\sigma}_{i'j}, \, \hat{p}_{i'j}) \, ,$$

where

$$\begin{cases} \theta(\hat{\mu}_{i'j}) & = \hat{\alpha}_{j0} + \hat{\alpha}_{jb_{i'}} + \hat{\alpha}_{jc_{i'}} + \hat{f}_{jc_{i'}}(\mathbf{x}_{i'}) \, , \\ \log(\hat{\sigma}_{i'j}) & = \hat{\beta}_{j0} + \hat{\beta}_{jb_{i'}} + \hat{\beta}_{jc_{i'}} + \hat{g}_{jc_{i'}}(\mathbf{x}_{i'}) \, , \\ \operatorname{logit}(\hat{p}_{i'j}) & = \hat{\gamma}_{j0} + \hat{\gamma}_{jb_{i'}} + \hat{\gamma}_{jc_{i'}} + \hat{h}_{jc_{i'}}(\mathbf{x}_{i'}) \, . \end{cases}$$

Note that $\hat{\mu}_{i'j}$, $\hat{\sigma}_{i'j}$, and $\hat{p}_{i'j}$ may not be all required, depending on the form of F_j (Table S1).

Then the m-dimensional feature vector of synthetic cell i' is $(Y_{i'1}, \dots, Y_{i'm})^{\mathsf{T}}$, where

$$Y_{i'j} = \hat{F}_j^{-1}(U_{i'j} \mid \mathbf{x}_{i'}, \mathbf{z}_{i'}) \, , \; j = 1, \dots, m \, .$$

Thanks to the parametric form of $\hat{F}_j(\cdot \mid \mathbf{x}_{i'}, \mathbf{z}_{i'})$, users can generate the synthetic data in their demand by modifying the parameters. For instance, if users want the expected sequencing depth of $\mathbf{Y}'$ to change from N (the sequencing depth of $\mathbf{Y}$) to N', they can scale the mean parameter:

$$Y_{i'j} \mid \mathbf{x}_{i'}, \mathbf{z}_{i'} \sim F_j \left(\cdot \;\middle|\; \mathbf{x}_{i'}, \mathbf{z}_{i'} \; ; \; \frac{N'}{N} \hat{\mu}_{i'j}, \, \hat{\sigma}_{i'j}, \, \hat{p}_{i'j} \right) \, .$$

If users want to remove the batch effects, they can set $\hat{\alpha}_{jb_{i'}} = \hat{\beta}_{jb_{i'}} = \hat{\gamma}_{jb_{i'}} = 0, \forall i', j$. If users want to remove the condition effects, they can set $\hat{\alpha}_{jc_{i'}} = \hat{\beta}_{jc_{i'}} = \hat{\gamma}_{jc_{i'}} = 0$, $\hat{f}_{jc_{i'}}(\cdot) = \hat{f}_{j1}(\cdot)$, $\hat{g}_{jc_{i'}}(\cdot) = \hat{g}_{j1}(\cdot)$, and $\hat{h}_{jc_{i'}}(\cdot) = \hat{h}_{j1}(\cdot)$, $\forall i', j$.

1.3 Data analysis

1.3.1 Data preprocessing

Supplementary Table S3 lists the real datasets from 11 published studies. Since scDesign3 can directly model count data, we did not perform data transformation (e.g., log-transformation) on the cell-by-feature count matrices.

For each cell-by-feature count matrix $\mathbf{Y}$, feature screening was used to retain informative features only and save computation time. For every scRNA-seq dataset, we used the R package `scran` (version 1.20.1) [35] to select the top 1000 highly variable genes (HVGs). For the 10x scATAC-seq dataset (ATAC), we used the R package `Signac` (version 1.7.0) [36] to first obtain a cell-by-peak matrix and then select 1133 differentially accessible peaks. For the sci-ATAC-seq data, the preprocessing and feature selection steps are described in [26]. For the 10x Visium dataset (VISIUM), we used the R package `Seurat` (version 4.1.1) to select the top 1000 spatially variable genes (SVGs). For the Slide-seq dataset (SLIDE), we selected the top 1000 genes with the smallest p-values output by `SPARK-X` [37].

For each dataset, the cell-by-state-covariate matrix $\mathbf{X}$ was from the original study (if the cell-state covariates are cell types or spatial locations) or inferred by the R package `Slingshot` (version 2.2.1) [38] (if the cell-state covariates are pseudotime values in trajectory lineages).

For each dataset, the optional cell-by-design-covariate matrix $\mathbf{Z}$ was from the original study if available.

1.3.2 Dimensionality reduction and visualization

To compare scDesign3's synthetic data with real test data, we used the R package `irlba` (version 2.3.5) to calculate the top 50 principal components (PCs) of the test cell-by-feature matrix (after log-transformation); next, we used the R package `umap` (version 0.2.8.0) to project the test cells from the 50-dimensional PC space to the 2-dimensional UMAP space. Then, we applied the same PCA-UMAP projection to scDesign3's synthetic cells using the R function `predict()`. Using the same projection ensures that the test cells and synthetic cells are embedded in the same 2-dimensional space and thus comparable.

Unless otherwise noted, the figures were made by the R package `ggplot2` (version 3.3.6). The coverage plot in Fig. 1g was generated by `IGV` (version 2.12.3).

1.3.3 Evaluation metrics

- **mLISI**: To measure the similarity between test cells and synthetic cells in the 2-dimensional space, we used the mean of local inverse Simpson's index (mLISI) [24] across all cells as the metric. Specifically, if a cell's neighboring cells are from one group (e.g., test cells or synthetic cells), the cell's local inverse Simpson's index (LISI) is 1; otherwise, if a cell's neighboring cells comprise two groups equally, the cell's LISI is 2. The mLISI is the average of all cells'

LISIs. Hence, a mLISI close to 2 means that the test cells and synthetic cells are perfectly mixed.

- **Pearson correlation**: To measure the similarity of between real data and the synthetic data when the cell-state covariates are continuous (e.g., pseudotime, spatial locations, and 2-dimensional UMAP embeddings), we also compared supervised learners trained on the real data and the synthetic data respectively. In detail, for every feature (e.g., gene), we trained a flexible learner, the generalized boosted regression model (GBM), separately on the real data and the synthetic data to predict the feature from cell-state covariates; then, we compared the two GBMs by measuring the Pearson correlation r between their predicted values from the synthetic data's cell-state covariates (note that the cell-state covariates can be replaced by a random sample from the covariate space). An r close to 1 means that the two GBMs are similar, that is, the feature's "relationship" with cell-state covariates is similar in the real data and the synthetic data. If all features have r close to 1, the synthetic data resemble the real data.

- **Summary statistics**: In Supplementary Figures S1–S7, we compared the distributions of six feature-level, cell-level, and feature-pair-level summary statistics between real data and synthetic data. Note that a feature represents a gene in scRNA-seq and spatial transcriptomics data and a peak in scATAC-seq and sci-ATAC-seq data. The six summary statistics are

  1. mean of log expression (feature-level): a feature's mean of log expression values across all cells;
  2. variance of log expression (feature-level): a feature's variance of log expression values across all cells;
  3. feature detection frequency (feature-level): a feature's proportion of non-zero values across all cells;
  4. feature-feature correlation (feature-pair-level): the correlation between two features' log expression values across all cells;
  5. cell library size on the log scale (cell-level): a cell's log-transformed total read or UMI count (i.e., log per-cell sequencing depth);
  6. cell detection frequency (cell-level): a cells' proportion of non-zero values across all features.

  Feature-feature correlations were calculated for the top 100 highly expressed features in each real dataset and the corresponding synthetic datasets. To measure the similarity between the real and synthetic correlation matrices, we calculate the Pearson correlation r across all 100^2 entries.

1.3.4 scDesign3's assessment of clustering quality

To show that scDesign3 can assess clustering quality, we used the 8 datasets from the R package `DuoClustering2018` (version 1.10.0), in which each dataset contains cell type labels ("truth") and various clustering methods' results with varying numbers of clusters. The adjusted Rand index (ARI), a "supervised" measure calculated between each clustering result and cell type labels, was used as the benchmark standard. scDesign3's marginal BIC (Section 1.1.4), an "unsupervised" measure that only uses the clustering result but not the cell type labels, was calculated for each clustering result in each dataset. We used scDesign3's marginal BIC because we observed that it better captures the clustering quality, while scDesign3's BIC is dominated by the copula BIC, which largely reflects the number of parameters instead of the clustering quality.

In Fig. S9b, we benchmarked scDesign3's marginal BIC against the ARI and found them to consistently have negative correlations on the 8 datasets, suggesting that scDesign3's marginal BIC is an effective assessment measure of clustering quality: a lower BIC indicates better clustering.

1.3.5 scDesign3's assessment of pseudotime quality

To show that scDesign3 can assess pseudotime quality, we used 5 synthetic datasets generated by the R package `dyngen` (version 1.0.3) and 3 synthetic datasets generated by scDesign3; each dataset contains cells' true pseudotime values ("truth") ranging from 0 to 1. To generate pseudotime with varying quality, we randomly replaced 0%, 10%, 20%, $\cdots$, 100% of truth pseudotime values with randomly sampled values from the $\text{Uniform}[0,1]$ distribution. The benchmark standard was the "supervised" R^2 between the true pseudotime values and the perturbed pseudotime values. scDesign3's marginal BIC (Section 1.1.4), an "unsupervised" measure that only uses the perturbed pseudotime values but not the true pseudotime values, was calculated for each set of perturbed pseudotime values in each dataset. We used scDesign3's marginal BIC because we observed that it better captures the pseudotime quality, while scDesign3's BIC is dominated by the copula BIC, which largely reflects the number of parameters instead of the pseudotime quality.

In Fig. S9a, we benchmarked scDesign3's marginal BIC against the R^2 and found them to consistently have negative correlations on the 8 datasets, suggesting that scDesign3's marginal BIC is an effective assessment measure of pseudotime quality: a lower BIC indicates better pseudotime quality.

1.3.6 Implementation of other simulators

We compared scDesign3 with existing scRNA-seq simulators including scGAN, muscat, SPARSim, and ZINB-WaVE.

- For scGAN, we used the docker and the tutorial the authors provided on scGAN's GitHub (`https://github.com/imsb-uke/scGAN`; access date: February 7, 2022) to simulate synthetic data.

- For muscat, we first used the R function `prepSim()` to process the training dataset. Then, we ran the R function `simData()` to simulate a synthetic dataset based on the processed training dataset and cell-level information (such as cell types and experimental conditions) of the training dataset. Both functions are from the R package `muscat` (version 1.6.0).

- For SPARSim, we first used the `SPARSim_create_simulation_parameter` function to obtain the parameters for each group of cells in the training dataset, whose cells were grouped by cell types, experimental conditions, or batches. The 3 required input parameters for the `SPARSim_create_simulation_parameter` function (`intensity`, `variability`, and `library_size`) were obtained using the `SPARSim_estimate_intensity`, `SPARSim_estimate_variability`, and `SPARSim_estimate_library_size` functions, respectively, for each cell group. Then, we ran the `SPARSim_simulation` function with the input parameters from the previous step to generate synthetic data. All functions are from the R package `SPARSim` (version 0.9.5).

- For ZINB-WaVE, we used the `zinbFit` function from the R package `zinbwave` (version 1.15.3), with the count matrix and cell-type labels as inputs.

2 Supplementary Methods

2.1 The Vine Copula

An m-dimensional copula C is a multivariate distribution function composed of m $\mathrm{Uniform}[0,1]$ marginal distribution functions. A bivariate copula has $m = 2$. The vine copula provides a regular vine (R-vine) structure that using conditioning and a tree sequence to decompose an m-dimensional copula into bivariate copulas: the m-dimensional copula density function is the product of the bivariate copula density functions [34].

We use a triplet $(\mathcal{F}, \mathcal{R}, \mathcal{C})$ to specify a vine copula, where $\mathcal{F}$ is a vector of marginal distributions, $\mathcal{R}$ is a R-vine tree sequence, and $\mathcal{C}$ is a set of conditional or unconditional bivariate copulas.

$\mathcal{R} = (T_1, \dots, T_{m-1})$ is a R-vine tree sequence if it meets the following conditions [34]:

1. Each tree $T_j = (N_j, E_j)$, where N_j denotes the node set and E_j denotes the edge set, is connected (i.e., there exists an edge path between every two nodes).
2. T_1 is a tree with m nodes corresponding to the m variables.
3. For all $T_j = (N_j, E_j)$, $j \geq 2$, $N_j = E_{j-1}$, i.e., the node set of the T_j is the edge set of T_{j-1}.
4. For all T_j, $j = 2, \dots, m-1$, any two connected nodes $\{n_a, n_b\} \in E_j$ satisfy $|n_a \cap n_b| = 1$. That is, n_a and n_b are two nodes in T_j and two edges in T_{j-1}; n_a and n_b are connected in T_j if and only if they share a node in T_{j-1}.

After obtaining a valid R-vine tree sequence $\mathcal{R}$, we can construct a unique m-dimensional R-vine copula distribution using the triplet $(\mathcal{F}, \mathcal{R}, \mathcal{C})$ that satisfies the following conditions [39]:

1. Given a random vector $\mathbf{X} = (X_1, \dots, X_m)^\mathsf{T}$, $\mathcal{F}$ is a vector of the marginal distribution functions of $\mathbf{X}$. That is, $\mathcal{F} = (F_1, \dots, F_m)$, and F_i is continuous and invertible for $i = 1, \dots, m$.
2. $\mathcal{R}$ is a R-vine tree sequence specified above.
3. $\mathcal{C}$ is a set of bivariate copulas, $\mathcal{C} = \{C_{e_j}\colon e_j \in E_j;\ j = 1, \dots, m-1\}$, where C_{e_j} is a bivariate copula corresponding to the edge e_j, and E_i represents the edge set of tree $T_j \in \mathcal{R}$.

Here is an example R-vine structure to demonstrate the notations above and the decomposition of an m-dimensional joint density function into bivariate copulas. In this example, we have

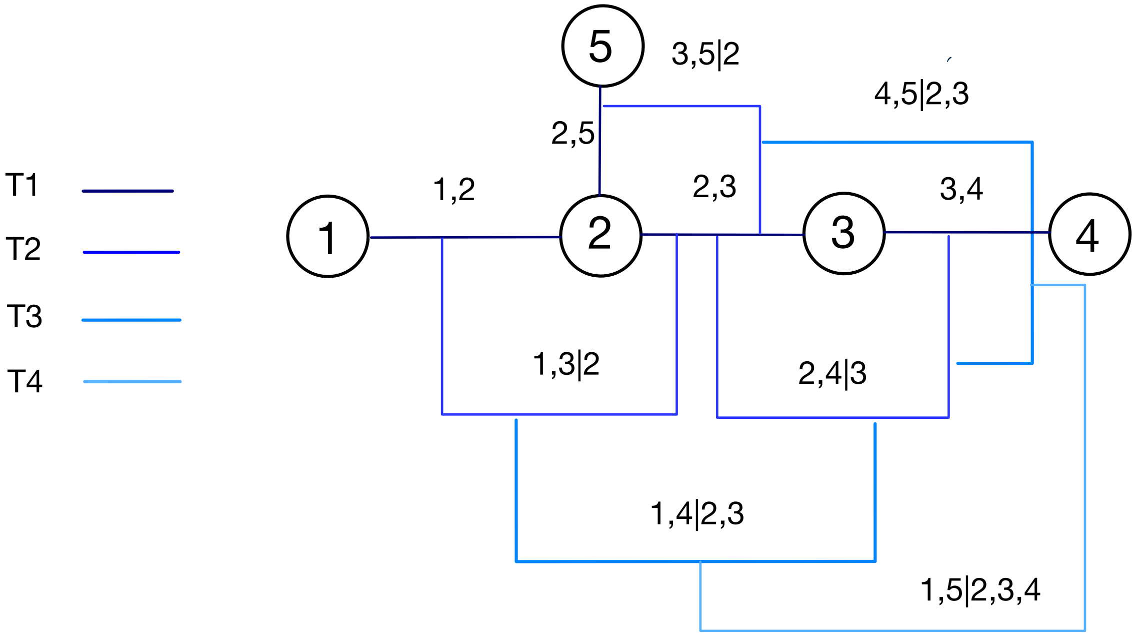
- $\mathbf{X} = (X_1, X_2, X_3, X_4, X_5)^\mathsf{T}$; $m = 5$;
- $\mathcal{R} = (T_1, T_2, T_3, T_4)$;
- $T_1 = (N_1, E_1)$, where $N_1 = \{X_1, X_2, X_3, X_4, X_5\}$ and $E_1 = \{(X_1, X_2), (X_2, X_3), (X_3, X_4), (X_2, X_5)\}$;
- $T_2 = (N_2, E_2)$, where $N_2 = \{(X_1, X_2), (X_2, X_3), (X_3, X_4), (X_2, X_5)\}$ and $E_2 = \{(X_1, X_3|X_2),\ (X_2, X_4|X_3),\ (X_3, X_5|X_2)\}$;

- $T_3 = (N_3, E_3)$, where $N_3 = \{(X_1, X_3|X_2), (X_2, X_4|X_3), (X_3, X_5|X_2)\}$ and $E_3 = \{(X_1, X_4|X_2, X_3), (X_4, X_5|X_2, X_3)\}$;
- $T_4 = (N_4, E_4)$, where $N_4 = \{(X_1, X_4|X_2, X_3), (X_4, X_5|X_2, X_3)\}$ and $E_4 = \{(X_1, X_5|X_2, X_3, X_4)\}$.

Then, the joint density of $\mathbf{X}$ can be written as

$$\begin{aligned}
&f_{12345}(x_1, x_2, x_3, x_4, x_5)\\
=&f_1(x_1) \cdot f_2(x_2) \cdot f_3(x_3) \cdot f_4(x_4) \cdot f_5(x_5)\\
&\cdot c_{1,2}\left(F_1(x_1),\, F_2(x_2)\right) \cdot c_{2,3}\left(F_2(x_2),\, F_3(x_3)\right) \cdot c_{2,5}\left(F_2(x_2),\, F_5(x_5)\right) \cdot c_{3,4}\left(F_3(x_3),\, F_4(x_4)\right)\\
&\cdot c_{1,3|2}\left(F_{1|2}(x_1|x_2),\, F_{3|2}(x_3|x_2)\right) \cdot c_{2,4|3}\left(F_{2|3}(x_2|x_3),\, F_{4|3}(x_4|x_3)\right) \cdot c_{3,5|2}\left(F_{3|2}(x_3|x_2),\, F_{5|2}(x_5|x_2)\right)\\
&\cdot c_{1,4|2,3}\left(F_{1|2,3}(x_1|x_2, x_3),\, F_{4|2,3}(x_4|x_2, x_3)\right) \cdot c_{4,5|2,3}\left(F_{4|2,3}(x_4|x_2, x_3),\, F_{5|2,3}(x_5|x_2, x_3)\right)\\
&\cdot c_{1,5|2,3,4}\left(F_{1|2,3,4}(x_1|x_2, x_3, x_4),\, F_{5|2,3,4}(x_5|x_2, x_3, x_4)\right)\ ,
\end{aligned}$$

where $c_{i,j|D} : [0,1]^2 \to [0,\infty)$ is a bivariate copula density function of $F_{i|D}(X_i)$ and $F_{j|D}(X_j)$ conditional on the variable set $\{X_k : k \in D\}$, and $F_{i|D}$ is the conditional CDF of X_i given $\{X_k : k \in D\}$, $i = 1, \ldots, m$.



Supplementary Figures

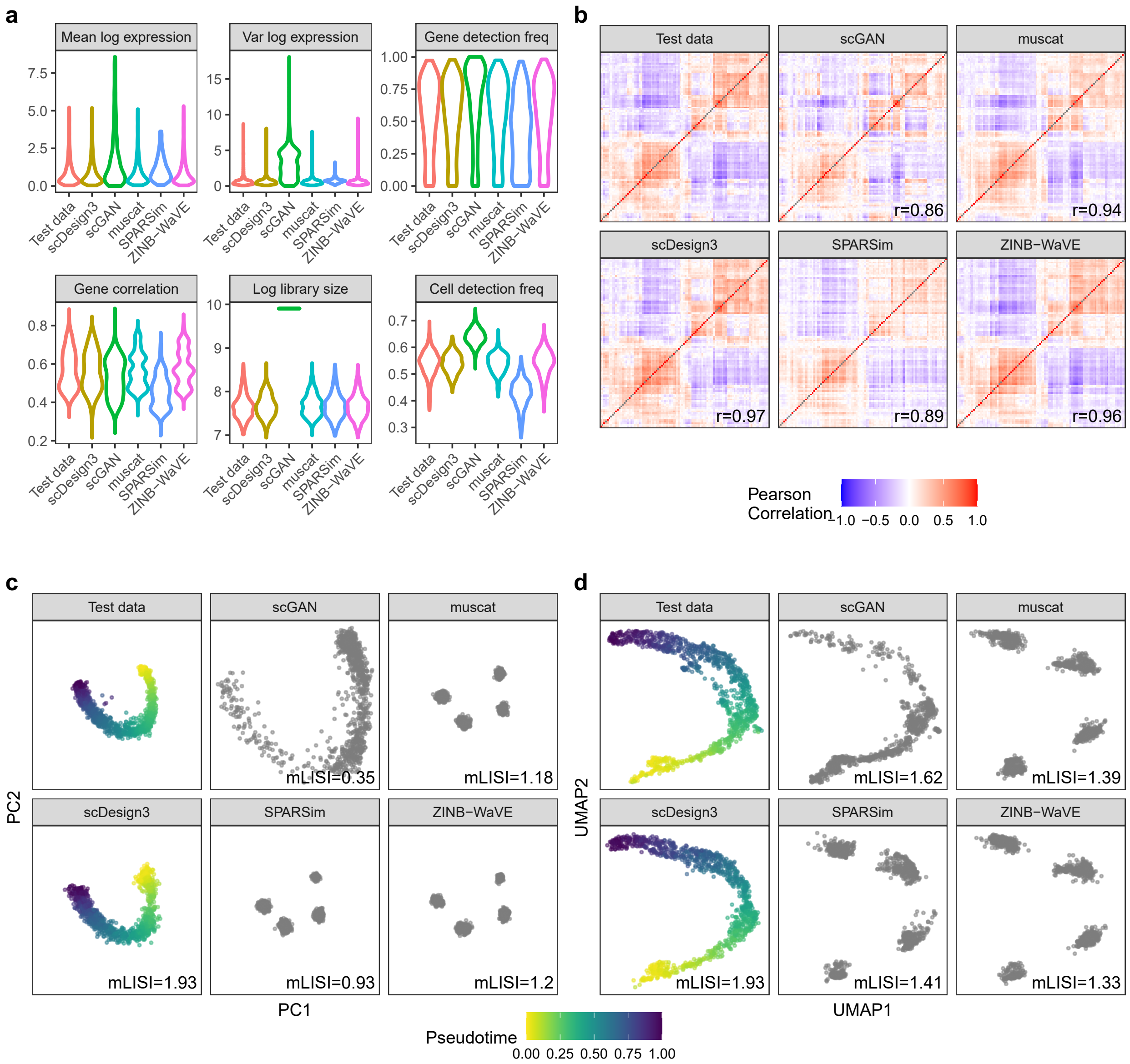


Figure S1: Benchmarking scDesign3 against four existing scRNA-seq simulators (scGAN, muscat, SPARSim, and ZINB-WaVE) for generating scRNA-seq data from a single trajectory (mouse pancreatic endocrinogenesis). **a**, Distributions of six summary statistics in the test data and the synthetic data generated by scDesign3 and the four simulators. **b**, Heatmaps of the gene-gene correlation matrices (showing top 100 highly expressed genes) in the test data and the synthetic data generated by scDesign3 and the four simulators. The Pearson's correlation coefficient r measures the similarity between two correlation matrices, one from the test data and the other from the synthetic data. **c**, PCA visualization (top two PCs) of the test data and the synthetic data generated by scDesign3 and the four simulators. The color labels each cell's pseudotime value; note that only the synthetic data by scDesign3 outputs the pseudotime truths. An mLISI value close to 2 means that the synthetic data resemble the real data well in the low-dimensional space. **d**, UMAP visualization of the real data and the synthetic data generated by scDesign3 and the four simulators.

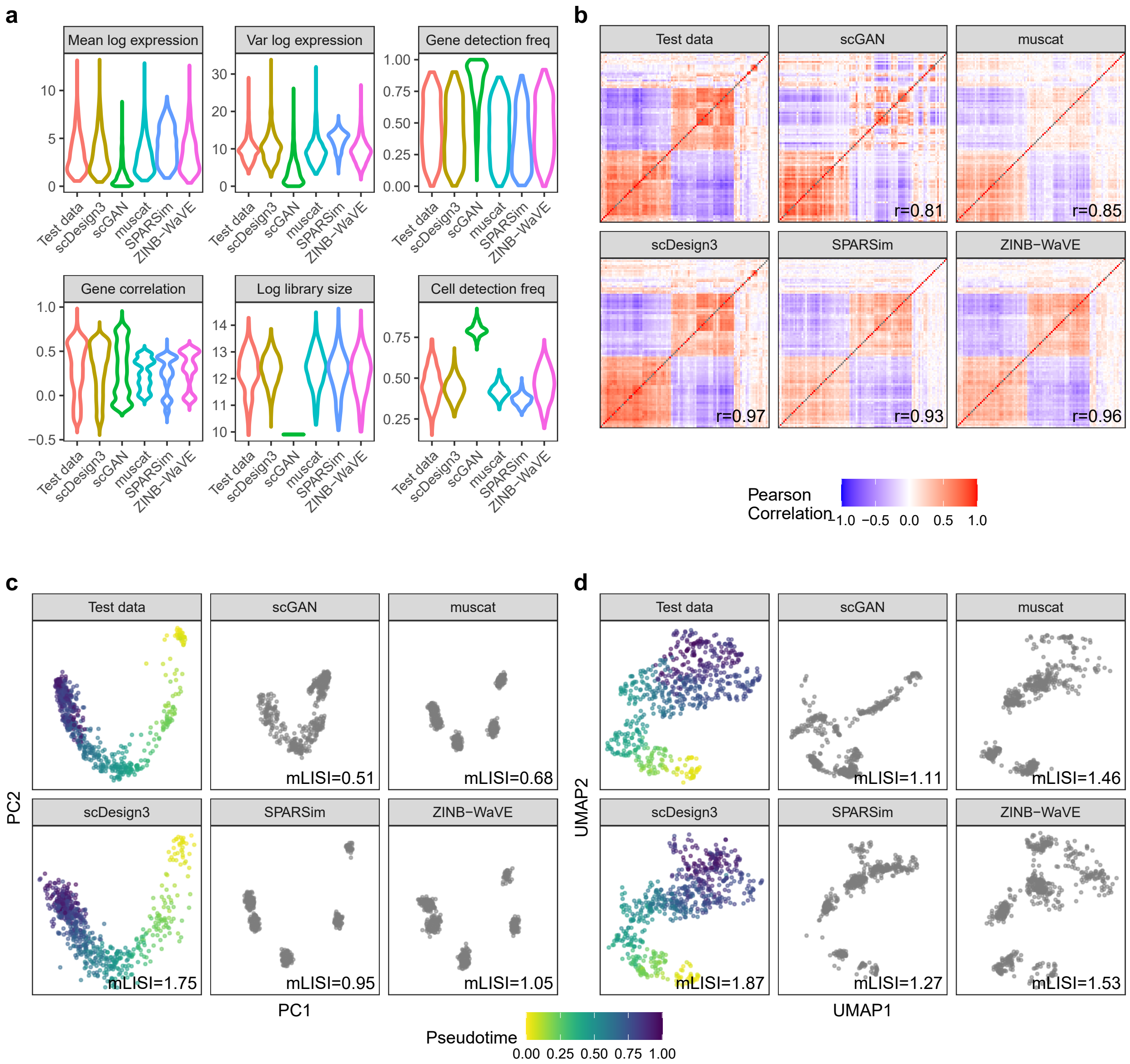


Figure S2: Benchmarking scDesign3 against four existing scRNA-seq simulators (scGAN, muscat, SPARSim, and ZINB-WaVE) for generating scRNA-seq data from a single trajectory (human preimplantation embryos). **a**, Distributions of six summary statistics in the test data and the synthetic data generated by scDesign3 and the four simulators. **b**, Heatmaps of the gene-gene correlation matrices (showing top 100 highly expressed genes) in the test data and the synthetic data generated by scDesign3 and the four simulators. The Pearson's correlation coefficient r measures the similarity between two correlation matrices, one from the test data and the other from the synthetic data. **c**, PCA visualization (top two PCs) of the test data and the synthetic data generated by scDesign3 and the four simulators. The color labels each cell's pseudotime value; note that only the synthetic data by scDesign3 outputs the pseudotime truths. An mLISI value close to 2 means that the synthetic data resemble the real data well in the low-dimensional space. **d**, UMAP visualization of the real data and the synthetic data generated by scDesign3 and the four simulators.

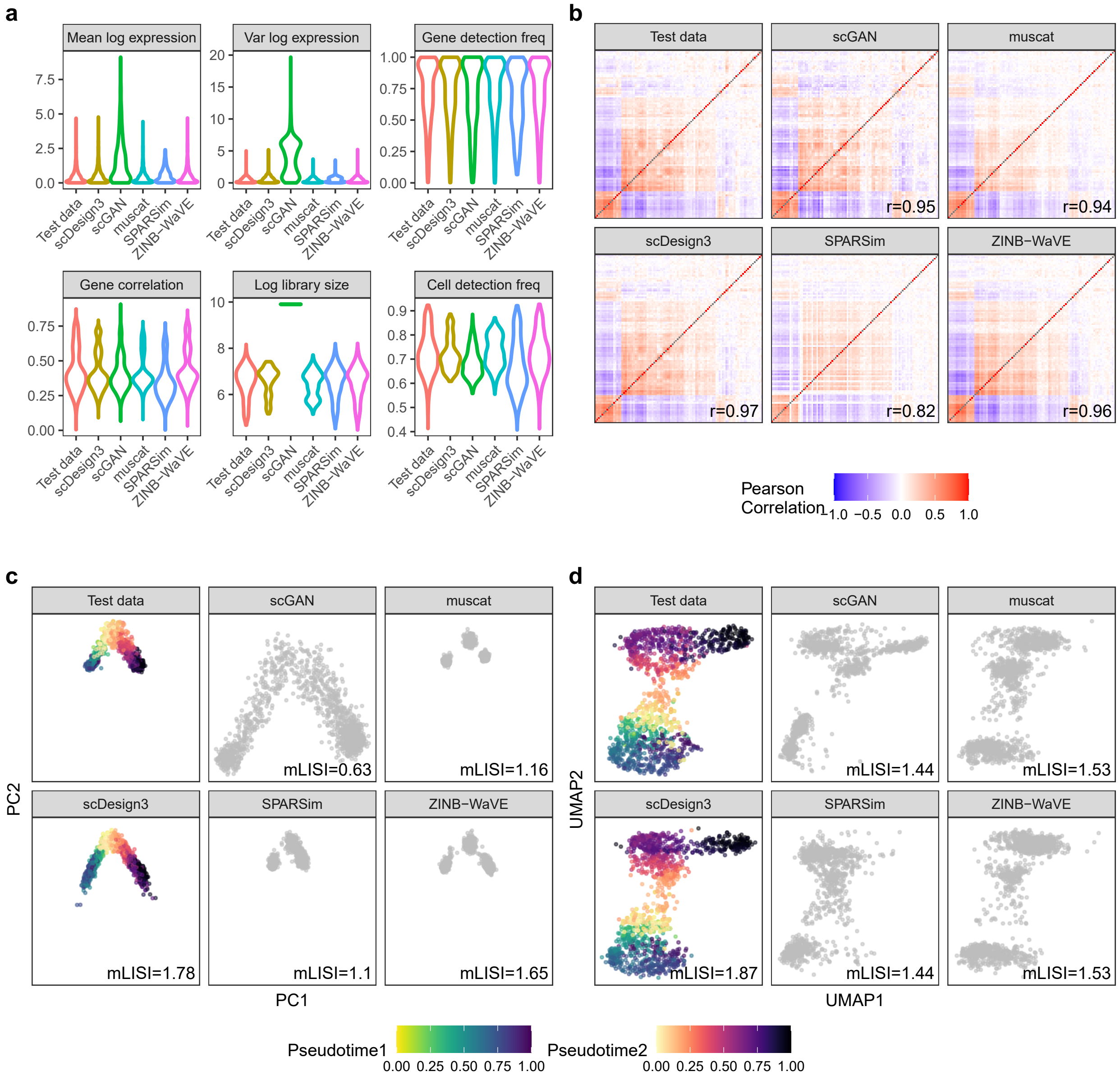


Figure S3: Benchmarking scDesign3 against four existing scRNA-seq simulators (scGAN, muscat, SPARSim, and ZINB-WaVE) for generating scRNA-seq data from bifurcating trajectories (myeloid progenitors in mouse bone marrow). **a**, Distributions of six summary statistics in the test data and the synthetic data generated by scDesign3 and the four simulators. **b**, Heatmaps of the gene-gene correlation matrices (showing top 100 highly expressed genes) in the test data and the synthetic data generated by scDesign3 and the four simulators. The Pearson's correlation coefficient r measures the similarity between two correlation matrices, one from the test data and the other from the synthetic data. **c**, PCA visualization (top two PCs) of the test data and the synthetic data generated by scDesign3 and the four simulators. The color labels each cell's pseudotime value; note that only the synthetic data by scDesign3 outputs the pseudotime truths. An mLISI value close to 2 means that the synthetic data resemble the real data well in the low-dimensional space. **d**, UMAP visualization of the real data and the synthetic data generated by scDesign3 and the four simulators.

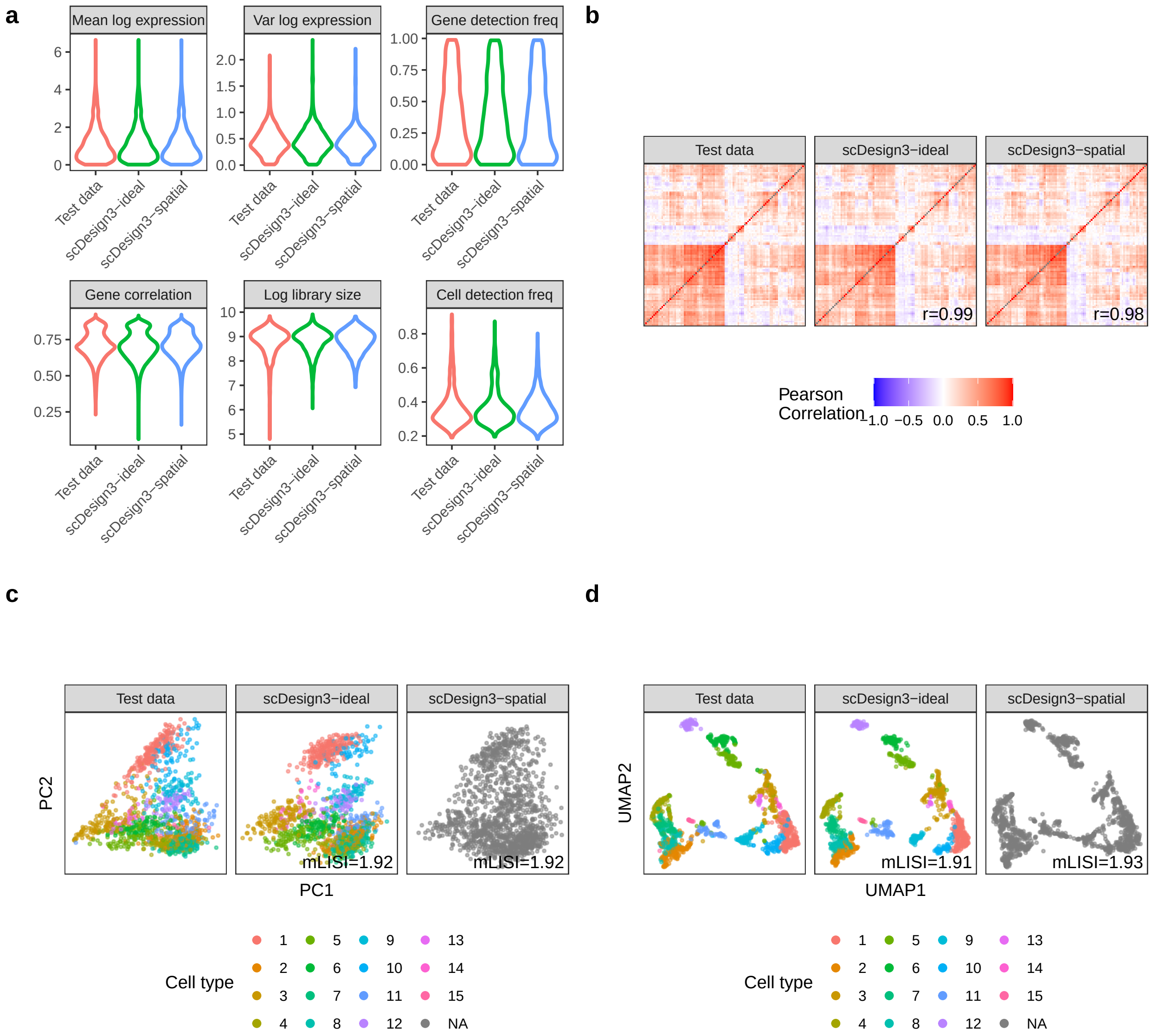


Figure S4: scDesign3 simulates 10x Visium spatial transcriptomics data (sagital mouse brain slices). **a**, Distributions of six summary statistics in the test data and the synthetic data generated by scDesign3 using cell type labels (scDesign3-ideal) and spatial locations (scDesign3-spatial), respectively. **b**, Heatmaps of the gene-gene correlation matrices (showing top 100 highly expressed genes) in the test data and the synthetic data generated by scDesign3-ideal and scDesign3-spatial. The Pearson's correlation coefficient r measures the similarity between two correlation matrices, one from the test data and the other from the synthetic data. **c**, PCA visualization (top two PCs) of the real data and the synthetic data generated by scDesign3-ideal and scDesign3-spatial. The color labels each cell's cell type (cluster). Since the scDesgin3-spatial data only uses spatial locations, it does not rely on cell types. An mLISI value close to 2 means that the synthetic data resemble the real data well in the low-dimensional space. **d**, UMAP visualization of the real data and the synthetic data generated by scDesign3-ideal and scDesign3-spatial. In summary, scDesign3 realistically simulates 10x Visium data based on spatial locations without needing cell type annotations.

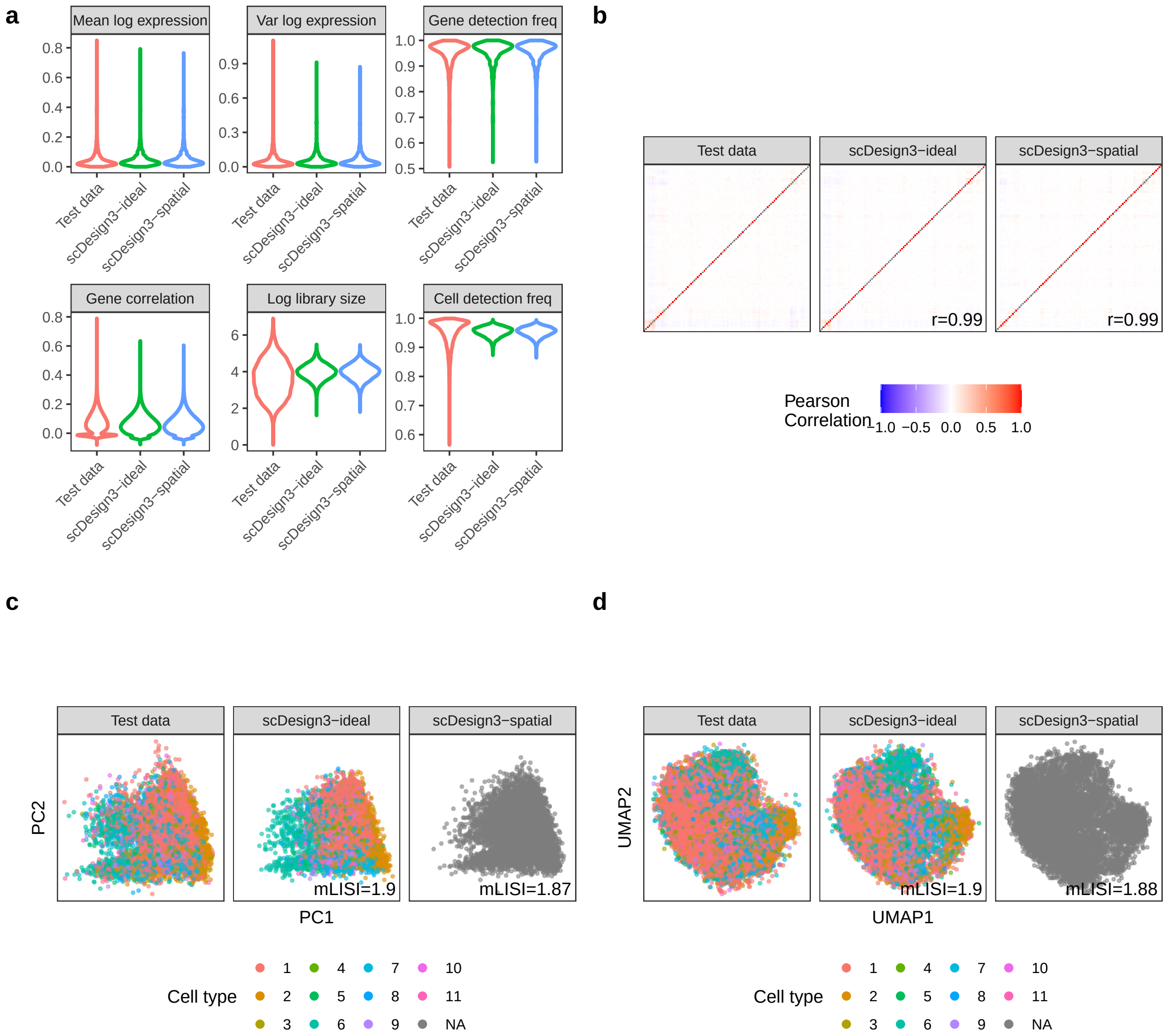


Figure S5: scDesign3 simulates Slide-seq spatial transriptomics data (coronal cerebellum). **a**, Distributions of six summary statistics in the test data and the synthetic data generated by scDesign3 using cell type labels (scDesign3-ideal) and spatial locations (scDesign3-spatial), respectively. **b**, Heatmaps of the gene-gene correlation matrices (showing top 100 highly expressed genes) in the test data and the synthetic data generated by scDesign3-ideal and scDesign3-spatial. The Pearson's correlation coefficient r measures the similarity between two correlation matrices, one from the test data and the other from the synthetic data. **c**, PCA visualization (top two PCs) of the real data and the synthetic data generated by scDesign3-ideal and scDesign3-spatial. The color labels each cell's cell type (cluster). Since scDesgin3-spatial only uses spatial locations, it does not rely on cell types. An mLISI value close to 2 means that the synthetic data resemble the real data well in the low-dimensional space. **d**, UMAP visualization of the real data and the synthetic data generated by scDesign3-ideal and scDesign3-spatial. In summary, scDesign3 realistically simulates Slide-seq data based on spatial locations without needing cell type annotations.

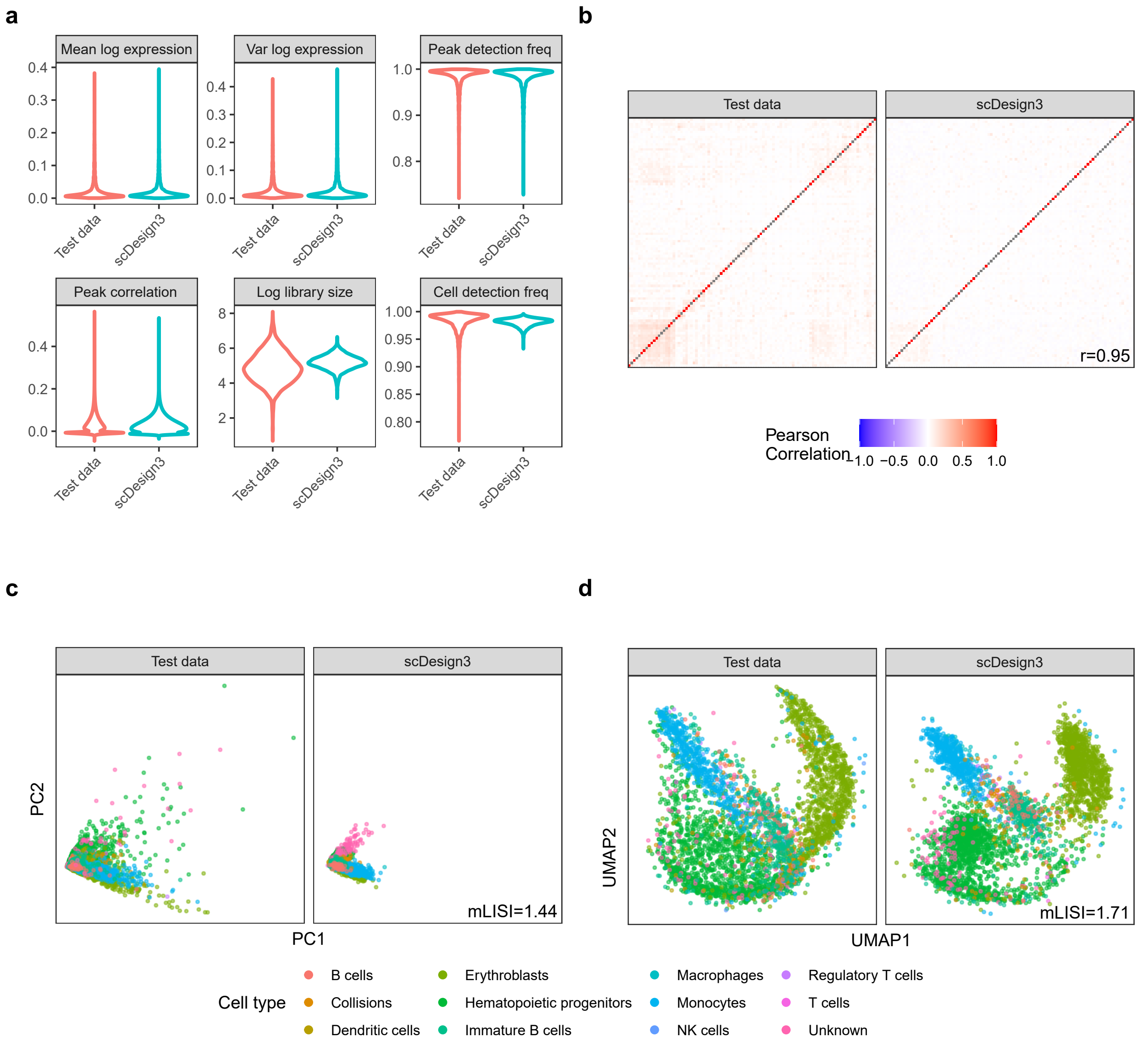


Figure S6: scDesign3 simulates sci-ATAC-seq data (mouse bone marrow). **a**, Distributions of six summary statistics in the test data and the synthetic data generated by scDesign3 using cell type labels. **b**, Heatmaps of the peak-peak correlation matrices in the test data and the synthetic data generated by scDesign3. The Pearson's correlation coefficient r measures the similarity between two correlation matrices, one from the test data and the other from the synthetic data. **c**, PCA visualization (top two PCs) of the test data and the synthetic data generated by scDesign3. The color labels each cell's cell type. An mLISI value close to 2 means that the synthetic data resemble the test data well in the low-dimensional space. **d**, UMAP visualization of the test data and the synthetic data generated by scDesign3.

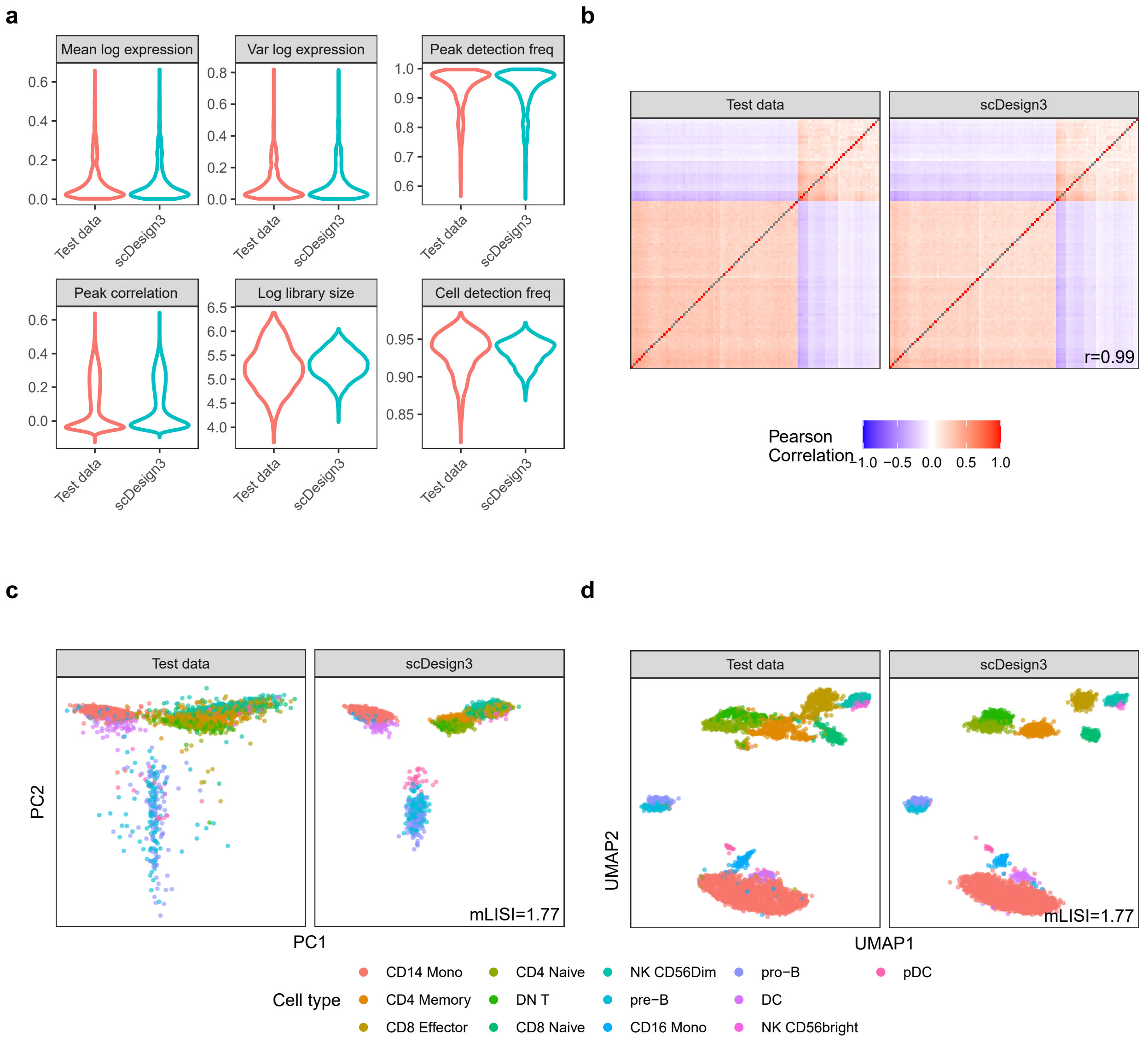


Figure S7: scDesign3 simulates scATAC-seq data (human PBMCs). **a**, Distributions of six summary statistics in the test data and the synthetic data generated by scDesign3 using cell type labels. **b**, Heatmaps of the peak-peak correlation matrices in the test data and the synthetic data generated by scDesign3. The Pearson's correlation coefficient r measures the similarity between two correlation matrices, one from the test data and the other from the synthetic data. **c**, PCA visualization (top two PCs) of the test data and the synthetic data generated by scDesign3. The color labels each cell's cell type. An mLISI value close to 2 means that the synthetic data resemble the test data well in the low-dimensional space. **d**, UMAP visualization of the test data and the synthetic data generated by scDesign3.

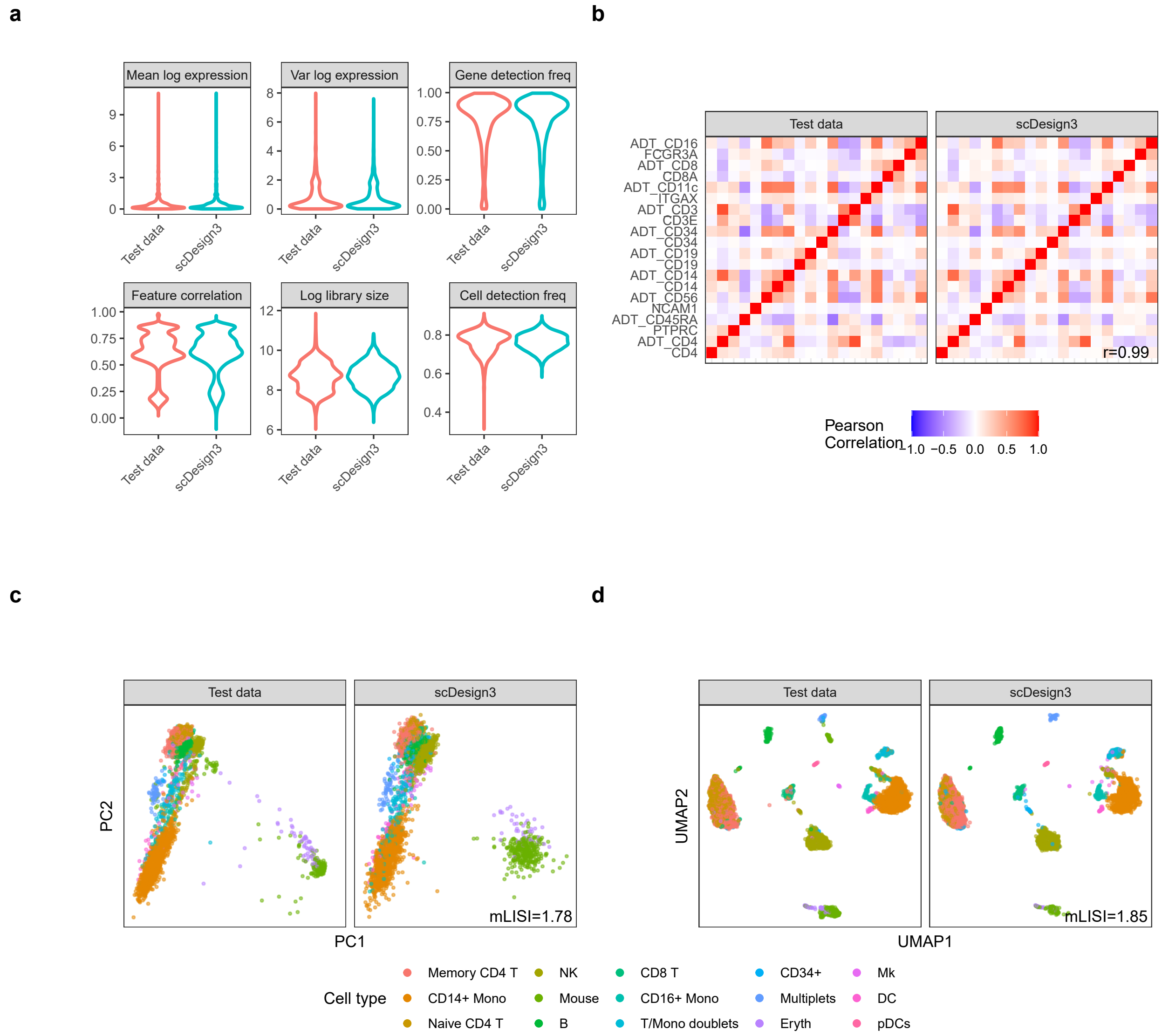


Figure S8: scDesign3 simulates CITE-seq data (human PBMCs). **a**, Distributions of six summary statistics in the test data and the synthetic data generated by scDesign3. The CITE-seq dataset simultaneously measures each cell's gene expression and surface protein abundance by Antibody-Derived Tags (ADTs). **b**, Heatmaps of the gene and protein correlation matrices (10 proteins with names starting with "ADT" and their corresponding genes) from test data and the synthetic data generated by scDesign3. The Pearson's correlation coefficient r measures the similarity between two correlation matrices, one from the test data and the other from the synthetic data. scDesign3 recapitulates the correlations between the RNA and protein expression levels of the 10 surface proteins. **c**, PCA visualization (top two PCs) of the test data and the synthetic data generated by scDesign3. The color labels each cell's cell type. An mLISI value close to 2 means that the synthetic data resemble the real data well in the low-dimensional space. **d**, UMAP visualization of the real data and the synthetic data generated by scDesign3.

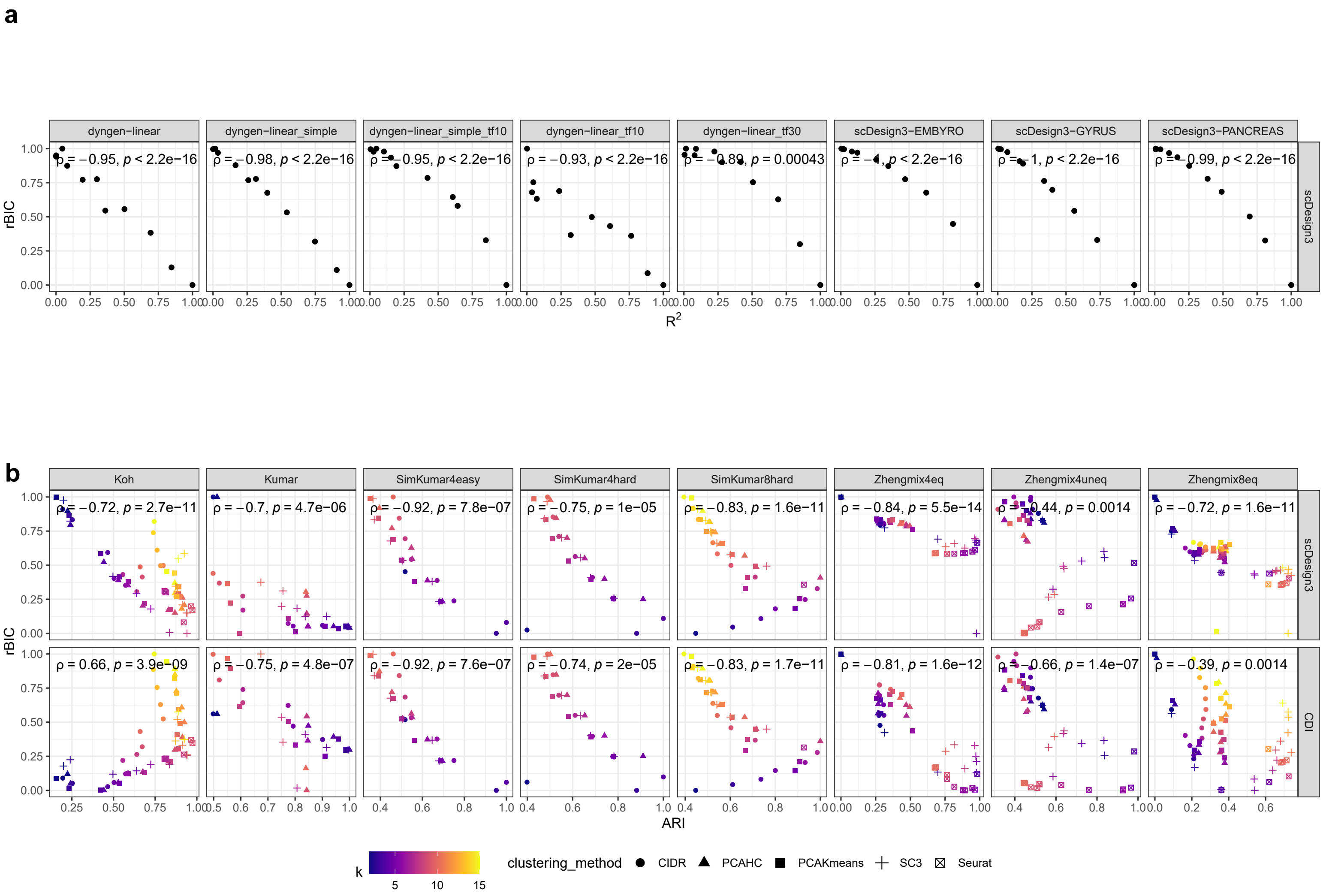


Figure S9: scDesign3 provides an unsupervised quantification of the quality of pseudotime and clusters. For visual clarity, we plot the relative BIC (rBIC) by re-scaling scDesign3's marginal BIC to $[0, 1]$. (a) The scDesign3 rBIC (unsupervised) is negatively correlated with the R^2 (supervised) between the perturbed pseudotime and the true pseudotime in each of the eight datasets. The true pseudotime is the ground truth used for generating the synthetic data. (b) Comparison of scDesign3 rBIC and Clustering Deviation Index (CDI) rBIC. The scDesign3 rBIC (unsupervised) negatively correlates with the ARI (supervised). The scDesign3 rBIC has better or similar performance than CDI's performance on six out of the eight datasets. The color scale shows the number of clusters, and the shapes represent clustering algorithms.

3 Supplementary Tables

Table S1: Choices of feature j's marginal distribution F_j[1]

Distribution	Parameters	Probability density function (PDF) or Probability mass function (PMF)	Link function	Applicable data type
Gaussian	μ: mean σ: standard deviation	$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{x-\mu}{\sigma})^2}; x \in \mathbb{R}$	$\theta_j(\mu) = \mu$	Normalized data (e.g., log-transformed count data)
Bernoulli	μ: mean	$f(x) = \mu^x(1-\mu)^{1-x}; x \in \{0, 1\}$	$\theta_j(\mu) = \log \frac{\mu}{1-\mu}$	Binary data (e.g., DNA methylation)
Poisson	μ: mean	$f(x) = \frac{\mu^x e^{-\mu}}{x!}; x \in \{0, 1, 2, \cdots\}$	$\theta(\mu) = \log \mu$	Count data without over-dispersion
Negative Binomial	μ: mean σ: dispersion	$f(x) = \frac{\Gamma(x+\frac{1}{\sigma})}{\Gamma(\frac{1}{\sigma})\Gamma(x+1)} \left(\frac{1}{1+\sigma\mu}\right)^{\frac{1}{\sigma}} \left(\frac{\sigma\mu}{1+\sigma\mu}\right)^x; x \in \{0, 1, 2, \cdots\}$	$\theta_j(\mu) = \log \mu$	Count data with over-dispersion (e.g., UMI-based scRNA-seq; spatial transcriptomics; protein abundance)
Zero-inflated Poisson	μ: mean p: zero-inflation proportion	$f(x) = \begin{cases} p + (1-p)e^{-\mu};\ x = 0 \\ \dfrac{(1-p)\mu^x e^{-\mu}}{x!};\ x = 1, 2, 3, \cdots \end{cases}$	$\theta_j(\mu) = \log \mu$	Poisson count data with excess zeros (e.g., scATAC-seq)
Zero-inflated Negative Binomial	μ: mean σ: dispersion p: zero-inflation proportion	$f(x) = \begin{cases} p + (1-p)(1+\sigma\mu)^{\frac{-1}{\sigma}};\ x = 0 \\ \dfrac{(1-p)\Gamma(x+\frac{1}{\sigma})}{\Gamma(\frac{1}{\sigma})\Gamma(x+1)} \left(\dfrac{1}{1+\sigma\mu}\right)^{\frac{1}{\sigma}} \left(\dfrac{\sigma\mu}{1+\sigma\mu}\right)^x;\ x = 1, 2, 3, \cdots \end{cases}$	$\theta_j(\mu) = \log \mu$	Negative binomial count data with excess zeros (e.g., full-length non-UMI-based scRNA-seq)

[1] For notation simplicity, we drop the cell index i and feature index j from all parameters' subscripts.

Table S2: Forms of the functions $f_{jc_i}(\cdot)$, $g_{jc_i}(\cdot)$, and $h_{jc_i}(\cdot)$ of cell-state covariates[1]

Covariate type	Covariate form	Function form	Explanation	Geometric meaning
Discrete cell type	$x_i \in \{1, \cdots, K_C\}$	$f_{jc_i}(x_i) = \alpha_{jc_i x_i}$	Cell type x_i has the effect $\alpha_{jc_i x_i}$; for identifiability, $\alpha_{jc_i x_i} = 0$ if $x_i = 1$	One intercept for each cell type
Continuous pseudotime in one lineage	$x_i \in [0, \infty)$	$f_{jc_i}(x_i) = \sum_{k=1}^{K} b_{jc_i k}(x_i)\beta_{jc_i k}$	$b_{jc_i k}(\cdot)$ is a basis function of cubic spline; K is the number of knots	A curve along the pseudotime
Continuous pseudotimes in p lineages	$\mathbf{x}_i = (x_{i1}, \cdots, x_{ip})^\mathsf{T} \in [0, \infty)^p$	$f_{jc_i}(\mathbf{x}_i) = \sum_{l=1}^{p} \sum_{k=1}^{K} b_{jc_i kl}(x_{il})\beta_{jc_i lk}$	$b_{jc_i lk}(\cdot)$ is a basis function of cubic spline; K is the input number of basis functions (default $K = 10$)	A curve along each lineage
Spatial location	$\mathbf{x}_i = (x_{i1}, x_{i2})^\mathsf{T} \in \mathbb{R}^2$	$f_{jc_i}(\mathbf{x}_i) = f_{jc_i}^{\mathrm{GP}}(x_{i1}, x_{i2}, K)$	$f_{jc_i}^{\mathrm{GP}}(\cdot, \cdot, K)$ is a Gaussian process smoother [32] ; K is the input number of basis functions (default $K = 400$)	A smooth surface

[1] For simplicity, we only show the form of $f_{jc_i}(\cdot)$ because $g_{jc_i}(\cdot)$ and $h_{jc_i}(\cdot)$ have the same form.

Table S3: Real datasets used in this study

Dataset name	Protocol	Cell-state covariates	Design covariates	Feature # (m)	Cell/Spot # (n)	Description	Ref
ATAC	10x scATAC-seq	cell type	N/A	1133 peaks	7034	human PBMCs	[40]
BATCH	10x scRNA-seq (V2/V3)	cell type	two batches	1000 genes	6276	human PBMCs	[41]
CITE	CITE-seq	cell type	N/A	1000 genes + 10 proteins	8617	human CBMCs	[8]
EMBYRO	scRNA-seq	cell pseudotime in one trajectory	N/A	1000 genes	1289	human preimplantation embryos	[42]
IFNB	10x scRNA-seq	cell type	case/control	1000 genes	13999	IFNB-stimulated/control PBMCs	[43]
MARROW	MARS-seq	cell pseudotimes in two trajectories	N/A	1000 genes	2660	myeloid progenitors in mouse bone marrow	[44]
PANCREAS	10x scRNA-seq	cell pseudotime in one trajectory	N/A	1000 genes	2087	mouse pancreatic endocrinogenesis	[45]
SCGEM-METH	scGEM	2D UMAP coordinates[1]	N/A	27 methylation loci	142	human foreskin fibroblast reprogramming to iPS	[46]
SCGEM-RNA	scGEM	2D UMAP coordinates[1]	N/A	32 genes	177	same as SCGEM-METH	[46]
SCIATAC	sci-ATAC-seq	cell type	N/A	3836 peaks	4025	mouse bone marrow	[47]
SLIDE	Slide-seq	spatial location	N/A	1000 genes	23372	coronal cerebellum	[37]
VISIUM	10x Visium	spatial location	N/A	1000 genes	2096	a sagital mouse brain slice	[48]
ZHENGMIX4	10x scRNA-seq	cell type	N/A	1556 genes	3555	human PBMCs	[49]

[1] To generate a synthetic dataset with both methylation and gene expression, we used the aligned UMAP space by Pamona [50].