A Large-Scale Comparative Analysis of Imputation Methods for Single-Cell RNA Sequencing Data

Yuichiro Iwashita[1,2*†], Ahtisham Fazeel Abbasi[1,2†], Koichi Kise[2,3], Andreas Dengel[1,2,4], Muhammad Nabeel Asim[2,3,4]

[1]RPTU University Kaiserslautern-Landau, Department of Computer Science, 67663 Kaiserslautern, Germany.
[2]German Research Center for Artificial Intelligence (DFKI GmbH), 67663 Kaiserslautern, Germany.
[3]Graduate School of Informatics, Osaka Metropolitan University, Osaka 599-8531, Japan.
[4]intelligentX GmbH (intelligentX.com), 67663 Kaiserslautern, Germany.

*Corresponding author(s). E-mail(s): yuichiro.iwashita@cs.rptu.de;
Contributing authors: ahtisham.abbasi@dfki.de; kise@omu.ac.jp;
andreas.dengel@dfki.de; muhammad_nabeel.asim@dfki.de;
†These authors contributed equally to this work.

Abstract

Background: Single-cell RNA sequencing (scRNA-seq) enables gene expression profiling at cellular resolution but is inherently affected by sparsity caused by dropout events, where expressed genes are recorded as zeros due to technical limitations. These artifacts distort gene expression distributions and compromise downstream analyses. Numerous imputation methods have been proposed to recover latent transcriptional signals. These methods range from traditional statistical models to deep learning (DL)-based methods. However, their comparative performance remains unclear, as existing benchmarks evaluate only a limited subset of methods, datasets, and downstream analyses.

Results: We present a comprehensive benchmark of 15 scRNA-seq imputation methods spanning 7 methodological categories, including traditional and DL-based methods. Methods are evaluated across 30 datasets from 10 experimental protocols on 6 downstream analyses. Results show that traditional methods, such as model-based, smoothing-based, and low-rank matrix-based methods,

generally outperform DL-based methods, including diffusion-based, generative adversarial network-based, graph neural network-based, and autoencoder-based methods. In addition, strong performance in numerical gene expression recovery does not necessarily translate into improved biological interpretability in downstream analyses, including cell clustering, differential expression analysis, marker gene analysis, trajectory analysis, and cell type annotation. Furthermore, method performance varies substantially across datasets, protocols, and downstream analyses, with no single method consistently outperforming others. **Conclusions:** Our findings provide practical guidance for selecting imputation methods tailored to specific analytical objectives and underscore the importance of task-specific evaluation when assessing imputation performance in scRNA-seq data analysis.

Keywords: single-cell RNA sequencing, gene expression, imputation, benchmark

1 Background

Single-cell RNA sequencing (scRNA-seq) has become a powerful technology for profiling gene expression at the resolution of individual cells [1–4]. In scRNA-seq, individual cells are isolated from a tissue, and their mRNA content is reverse-transcribed into complementary DNA (cDNA), amplified to increase signal, and sequenced to generate large collections of short DNA reads [5–10]. These reads are subsequently processed through a computational pipeline that includes alignment to a reference genome, quality filtering, and transcript counting [5–8]. The output of this pipeline is a structured gene expression matrix in which rows represent individual cells, columns represent genes or vice versa, and each matrix entry represents the expression of a specific gene in a cell [2].

scRNA-seq plays a significant role in biological research by addressing key questions related to cellular heterogeneity and disease mechanisms [2, 5]. For instance, scRNA-seq enables the discovery of dynamic gene regulatory features [11], the investigation of cellular interactions [12], and the identification of rare cell types [13]. Moreover, scRNA-seq is an essential tool for constructing cell atlases, such as the Human Cell Atlas (HCA) [14] and the Tabula Sapiens [15], and these atlases enable comprehensive mapping of cell types and states across various tissues and organs [13, 16].

A wide range of downstream tasks can be performed on scRNA-seq data to address diverse biological questions [2, 17–20]. These tasks include cell clustering to group cells with similar expression profiles [21, 22], trajectory inference to model dynamic cellular processes [23–27], marker gene analysis to identify genes that define specific cell populations [28], cell type identification to assign biological identities to cells [29], and differential expression (DE) analysis to detect genes that are differentially expressed between different conditions [30].

Despite the widespread use of scRNA-seq, the reliability of results from downstream tasks critically depends on the quality of gene expression data [31, 32]. In practice, achieving high-quality data is challenging due to substantial technical noise

Table 1 Summary of existing benchmarking studies on scRNA-seq data imputation methods

Study	Methods							Datasets	Protocols	Tasks
	Traditional			DL-based						
	Model-based	Smoothing-based	Low-rank Matrix-based	Diffusion-based	GAN-based	GNN-based	AE-based			
Hou et al. [19]	6	3	3	0	0	0	6	16	5	4
Dai et al. [18]	2	3	2	0	1	1	3	8	1	4
Cheng et al. [17]	2	4	1	0	0	0	4	16	3	3
This Study	2	3	**3**	**2**	**2**	**1**	2	**30**	**10**	**6**

inherent to scRNA-seq experiments, arising from the limited amount of mRNA in individual cells [33], inefficiencies in reverse transcription [34], and stochastic variability introduced during mRNA capture and amplification steps [16, 33, 34]. These technical limitations can lead to dropout events, where genes are observed as zero despite being expressed at low levels in the cell [16, 33–36]. However, not all zero counts arise from technical noise; zero counts in scRNA-seq data may also reflect a true biological absence of transcription, often referred to as biological zeros, which are fundamentally distinct from technical dropout events [16, 36, 37]. Since dropout events distort the observed gene expression distribution, they can adversely affect the accuracy and robustness of downstream analyses [16, 36, 38, 39]. In light of these limitations, data imputation strategies have been introduced to address dropout events before performing downstream analyses [2, 17–19, 36].

scRNA-seq imputation methods fall into traditional and deep learning (DL)-based categories [40–54]. Traditional imputation methods typically rely on statistical modeling or similarity-based heuristics [40–47]. Traditional methods can be broadly categorized into 3 methodological classes, namely model-based, smoothing-based, and low-rank matrix-based methods [16, 19, 36], and are briefly described in Section 5.3.1. In contrast, DL-based methods rely on representation learning using deep neural networks (DNNs) [48–54], which is a fundamentally different approach compared to traditional methods. DL-based methods can be broadly categorized into 4 methodological classes, namely diffusion-based, generative adversarial network (GAN)-based, graph neural network (GNN)-based, and autoencoder (AE)-based methods, and are briefly discussed in Section 5.3.2.

Despite the availability of numerous imputation methods, existing benchmarking studies remain limited in their coverage of methods, datasets, experimental protocols, and downstream tasks. Table 1 summarizes 3 existing benchmarking studies on scRNA-seq data imputation, namely Hou et al. [19], Dai et al. [18], and Cheng et al. [17]. Hou et al. [19] evaluated 12 traditional methods and 6 AE-based methods, but do not include any diffusion-based, GAN-based, or GNN-based methods. Dai et al. [18] expanded the scope of DL-based methods by incorporating 1 GAN-based and 1 GNN-based methods in addition to 3 AE-based methods. However, their evaluation is limited to 8 datasets and a single protocol. Cheng et al. [17] assessed 7 traditional methods and 4 AE-based methods, but similarly, they do not evaluate diffusion-based, GAN-based, or GNN-based methods. Furthermore, all 3 studies restricted their

evaluation to at most 3 downstream tasks, which may not adequately capture the multifaceted effects of imputation on biological analyses. These gaps highlight the need for a more comprehensive and robust benchmarking study that spans a wider range of imputation methods, including recent DL architectures, and evaluates their impact across a broader set of downstream tasks and protocols.

In this study, we address these limitations by presenting a comprehensive and systematic benchmark of scRNA-seq data imputation methods. Our evaluation covers 15 representative methods spanning 7 methodological categories, including both traditional methods (model-based, smoothing-based, and low-rank matrix-based) and recent DL-based methods (diffusion-based, GAN-based, GNN-based, and AE-based methods). To ensure a robust and representative assessment, we evaluate these methods across 30 datasets (26 real and 4 simulated) sourced from 10 distinct protocols. Beyond imputing scRNA-seq data using these methods, we further investigate the impact of imputation on a broad range of biologically relevant downstream analyses. Specifically, we assess method performance across 6 key tasks in scRNA-seq data analysis, including numerical gene expression recovery, cell clustering, DE analysis, marker gene analysis, trajectory analysis, and cell type annotation. Together, this study provides a comprehensive benchmarking framework for evaluating scRNA-seq data imputation methods and offers practical insights into their strengths and limitations across diverse analytical settings. Our systematic comparison of methods across heterogeneous datasets, protocols, and downstream tasks provides guidance for selecting appropriate imputation strategies tailored to specific single-cell analysis objectives.

2 Results

2.1 Numerical Gene Expression Recovery

Fig. 1 represents the distribution of log normalized difference (LND) values between imputed and ground truth expression values for 15 imputation methods in terms of 26 real and 4 simulated datasets. The width of each violin plot represents the density of LND values, and the box plot shows the median and interquartile range (IQR) of LND values. LND $= 0$ indicates that all imputed expression values are identical to the ground truth values, LND > 0 indicates over-imputation, where the imputed values are greater than the ground truth values, and LND < 0 indicates under-imputation, i.e., the imputed values are less than the ground truth values.

The comparison of LND distributions across 15 imputation methods in terms of 30 different datasets shows that scTsI, PBLR, and WEDGE achieve the best overall performance, with medians of LND values consistently close to zero. This distribution indicates superior numerical recovery quality and better preservation of the original data structure. Conversely, scIDPMs shows the poorest performance, with a substantial over-imputation across 25 datasets. In contrast, scLRTC exhibits the strongest under-imputation in all datasets. The remaining 10 methods, including PbImpute, scImpute, AcImpute, MAGIC, stDiff, scMultiGAN, scIGANs, scGNN, CPARI, and Bubble, show moderate performance.

Protocol-wise analysis of LND distributions is essential because protocols differ in sparsity, noise, and dropout characteristics, which directly affect imputation behavior.

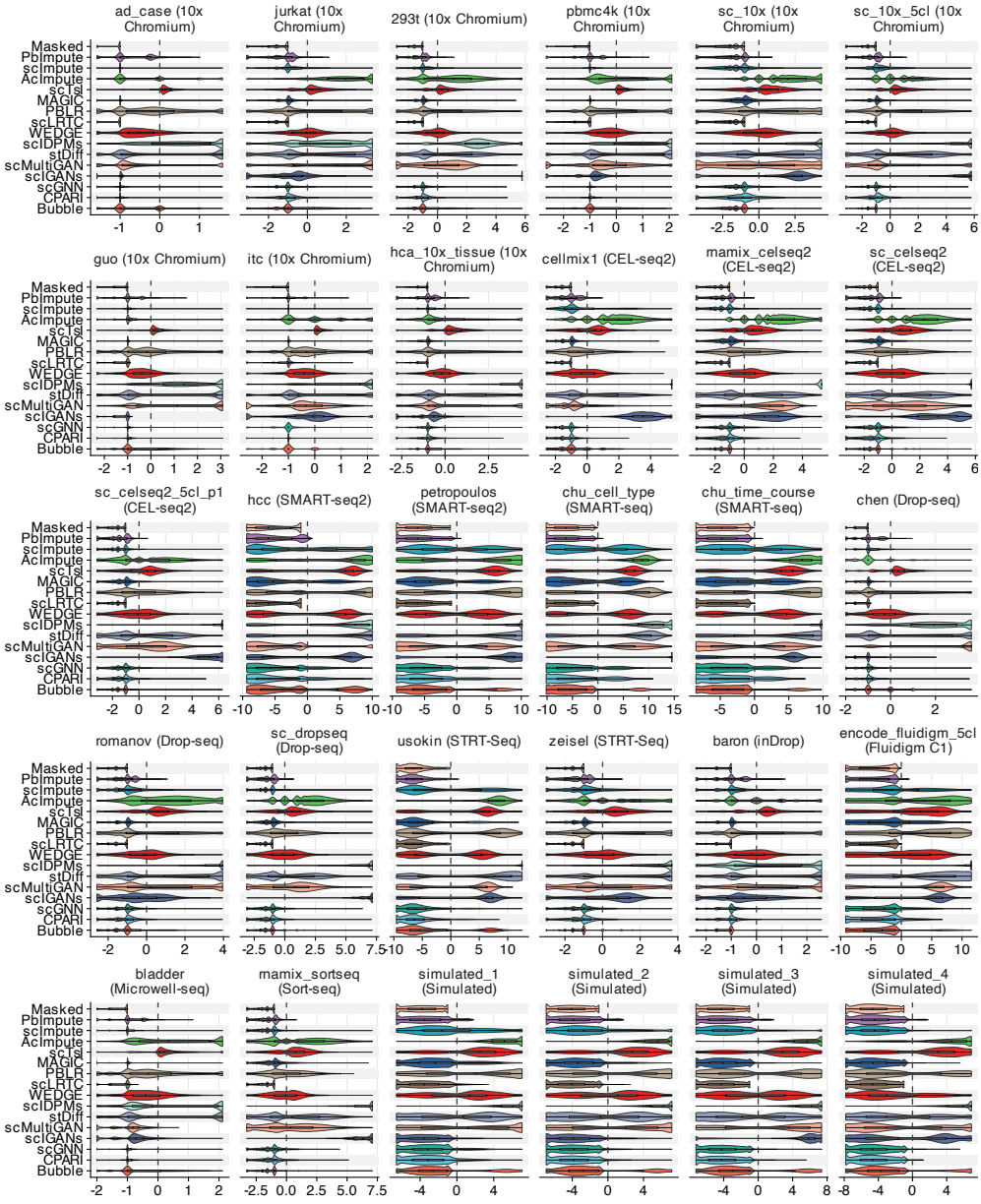

Fig. 1 Distribution of LND between imputed and ground truth expression values for each imputation method. The x-axis represents LND values, and the y-axis represents different imputation methods.

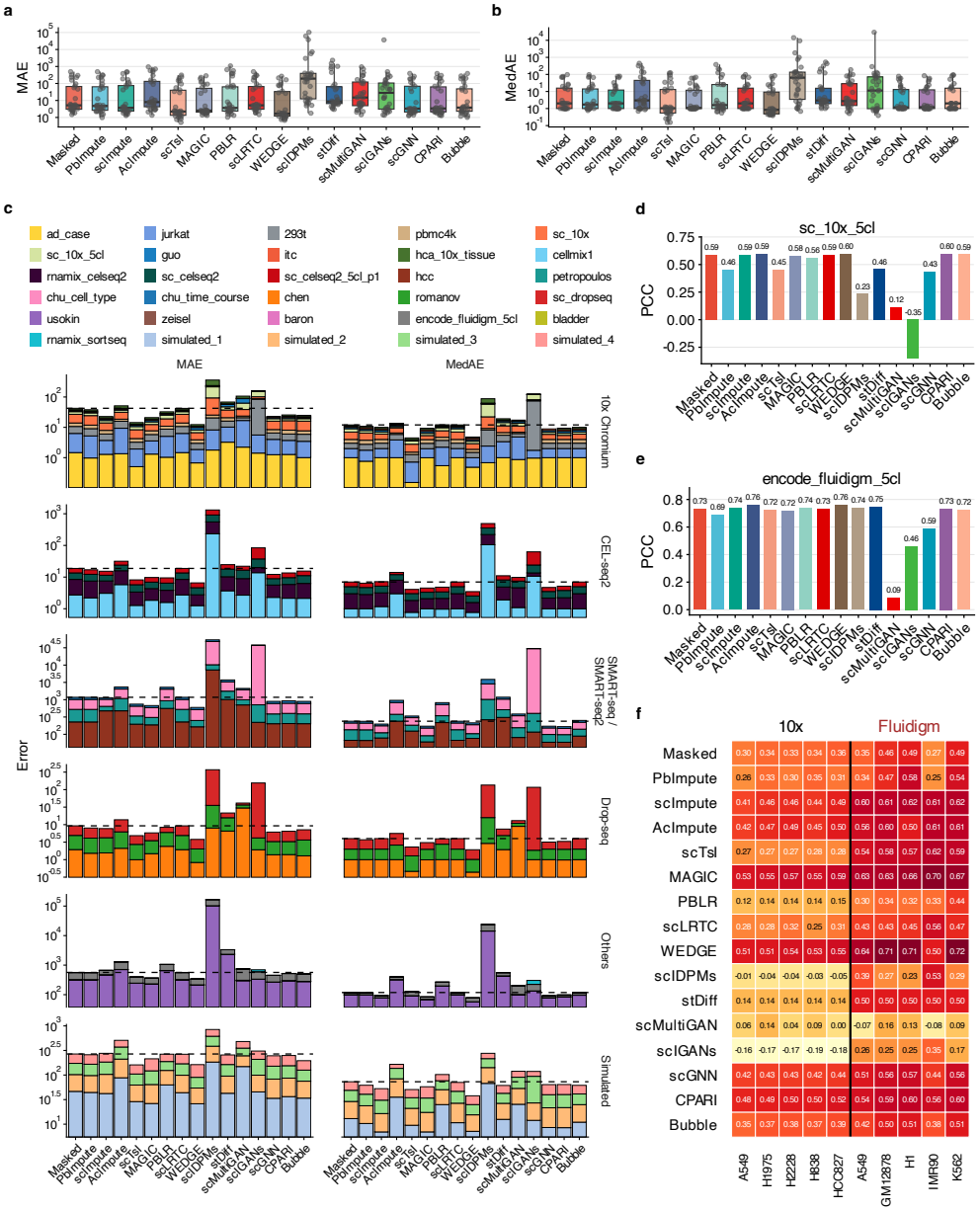

Fig. 2 Numerical gene expression recovery performance. **a**–**b** MAE and MedAE, respectively. The x-axis represents different imputation methods, and the y-axis represents error values in a log scale. **c** Protocol-wise total MAE and MedAE. The x-axis represents different imputation methods, and the y-axis represents total error values in a log scale. **d**–**e** PCC. The x-axis represents different imputation methods, and the y-axis represents PCC values. **f** MCC. The x-axis represents different cell lines, and the y-axis represents different imputation methods.

Across 5 protocols, WEDGE maintains LND values closest to zero with compact distributions, which indicates stable reconstruction and superior preservation of the original data structure. In contrast, scIDPMs frequently exhibits positive LND shifts, which reflect systematic over-imputation. Conversely, scLRTC consistently demonstrates under-imputation in all protocols. Protocols based on full-length sequencing, including SMART-seq, SMART-seq2, and Fluidigm C1, show greater variability overall. All methods display broader or bimodal distributions, and none achieve mode or median LND values close to zero. These patterns suggest increased difficulty for accurate imputation. The remaining methods exhibit intermediate, dataset-dependent behavior with moderate deviations around zero. Collectively, these findings highlight WEDGE as the most protocol-robust approach, while revealing distinct protocol-dependent biases for competing methods.

Figs. 2a and b represent the box plots for mean absolute error (MAE) and median absolute error (MedAE) between imputed and ground truth expression values for 15 imputation methods in terms of 26 real and 4 simulated datasets. MAE shows the overall performance of the methods considering outliers, while MedAE shows the overall performance of the methods without outliers. Lower MAE and MedAE values indicate better recovery performance, as they suggest that the imputed expression values are closer to the ground truth values.

A thorough analysis of MAE and MedAE among the 15 imputation methods reveals that scTsI and WEDGE exhibit the lowest MAE and MedAE. In addition, 12 methods, namely PbImpute, scImpute, AcImpute, MAGIC, PBLR, scLRTC, stDiff, scMultiGAN, scIGANs, scGNN, CPARI, and Bubble, show moderate results, which are similar to the performance using the masked baseline. On the other hand, scIDPMs shows the worst MAE and MedAE.

Fig. 2c shows total MAE and total MedAE across all datasets for each imputation method. WEDGE shows the best MAE across 5 protocols, namely 10x Chromium, CEL-seq2, SMART-seq2, SMART-seq, and Drop-seq. Similarly, WEDGE shows the best MedAE across 3 protocols, namely 10x Chromium, CEL-seq2, and Drop-seq. On the other hand, scIDPMs and scIGANs exhibit the highest MAE and MedAE as their values significantly exceed the masked baseline for all protocols.

Figs. 2d and e show pseudo-bulk correlation coefficient (PCC) between pseudo-bulk and the corresponding bulk RNA-seq data for 15 imputation methods in terms of 2 cell line datasets, namely sc_10x_5cl and encode_fluidigm_5cl. A higher PCC indicates that pseudo-bulk data is highly correlated with the corresponding bulk RNA-seq data. The comparison of PCC across the 15 methods shows that AcImpute, WEDGE, and CPARI achieve the best performance. In addition, 10 methods, namely PbImpute, scImpute, scTsI, MAGIC, PBLR, scLRTC, scIDPMs, stDiff, scGNN, and Bubble, show moderate performance. On the other hand, scMultiGAN and scIGANs show the worst performance.

Fig. 2f shows median correlation coefficient (MCC) between imputed scRNA-seq data and the corresponding bulk RNA-seq data at the cell line level for 15 imputation methods in terms of 2 cell line datasets, namely sc_10x_5cl and encode_fluidigm_5cl. A higher MCC indicates that imputed scRNA-seq data is highly correlated with the corresponding bulk RNA-seq data. The comparison of MCC across the 15 methods

shows that MAGIC and WEDGE achieve the best performance. In addition, 10 methods, namely PbImpute, scImpute, AcImpute, scTsI, PBLR, scLRTC, stDiff, scGNN, CPARI, and Bubble, show moderate performance. On the other hand, scIDPMs, scMultiGAN, and scIGANs show the worst performance.

In summary, for comparison with ground truth data, scTsI, PBLR, and WEDGE show the best overall performance, while scLRTC, scIDPMs, and scIGANs show the worst performance. Furthermore, imputation quality is protocol dependent, as methods show largely consistent behavior on 10x Chromium, CEL-seq2, and Drop-seq datasets, whereas SMART-seq, SMART-seq2, and Fluidigm C1 datasets demonstrate higher instability, characterized by systematic over- or under-imputation across the 15 imputation methods. These methods tend to struggle with SMART-seq, SMART-seq2, and Fluidigm C1 datasets because these datasets are generated only using read-counts, whereas other datasets use unique molecular identifiers (UMIs) [17]. The absence of UMIs can lead to duplicate read counts in the scRNA-seq data, which results in increased technical noise in the data [17]. For comparison with bulk RNA-seq data, WEDGE achieves the best overall performance. On the other hand, scMultiGAN shows poor correlation with bulk RNA-seq data at both the pseudo-bulk and cell line levels, despite its moderate numerical recovery of ground truth data. This suggests that comparable ground truth recovery does not necessarily translate to faithful agreement with bulk RNA-seq data.

2.2 Cell Clustering

Fig. 3a represents adjusted rand index (ARI) of cell clustering based on the imputed and ground truth data for the 15 imputation methods in terms of 26 real and 4 simulated datasets. Particularly, it demonstrates the consistency between cell clustering using imputed and ground truth data. ARI = 1 shows clusters match perfectly, ARI = 0 shows cell clustering performance is equivalent to randomly assigning clusters, and ARI $<$ 0 shows cell clustering performance is worse than randomly assigning clusters. Out of 15 distinct imputation methods, scLRTC exhibits the highest ARI scores in 13 datasets. In addition, 12 methods, namely PbImpute, scImpute, AcImpute, scTsI, MAGIC, WEDGE, scIDPMs, scMultiGAN, scIGANs, scGNN, CPARI, and Bubble, show moderate ARI scores. On the other hand, PBLR and stDiff show the lowest ARI scores for 8 datasets. Moreover, for 12 datasets, including sc_10x_5cl, hca_10x_tissue, cellmix1, sc_celseq2, hcc, petropoulos, chu_time_course, chen, romanov, usokin, zeisel, and baron, none of the 15 methods exceed the ARI scores of the masked baseline. This indicates that imputation does not necessarily improve cell clustering and can even degrade performance compared to using the masked baseline data.

Fig. 3b represents silhouette coefficient (SC) for the 15 imputation methods in terms of 26 real and 4 simulated datasets. SC = 1 represents cells are perfectly assigned to highly dense and isolated clusters, SC = 0 represents cells are located at the boundaries between 2 clusters, and SC $<$ 0 represents cells are inaccurately assigned to clusters. The comparison of SC scores among the 15 methods shows that MAGIC and WEDGE achieve the best performance for 9 datasets. On the other hand, PBLR

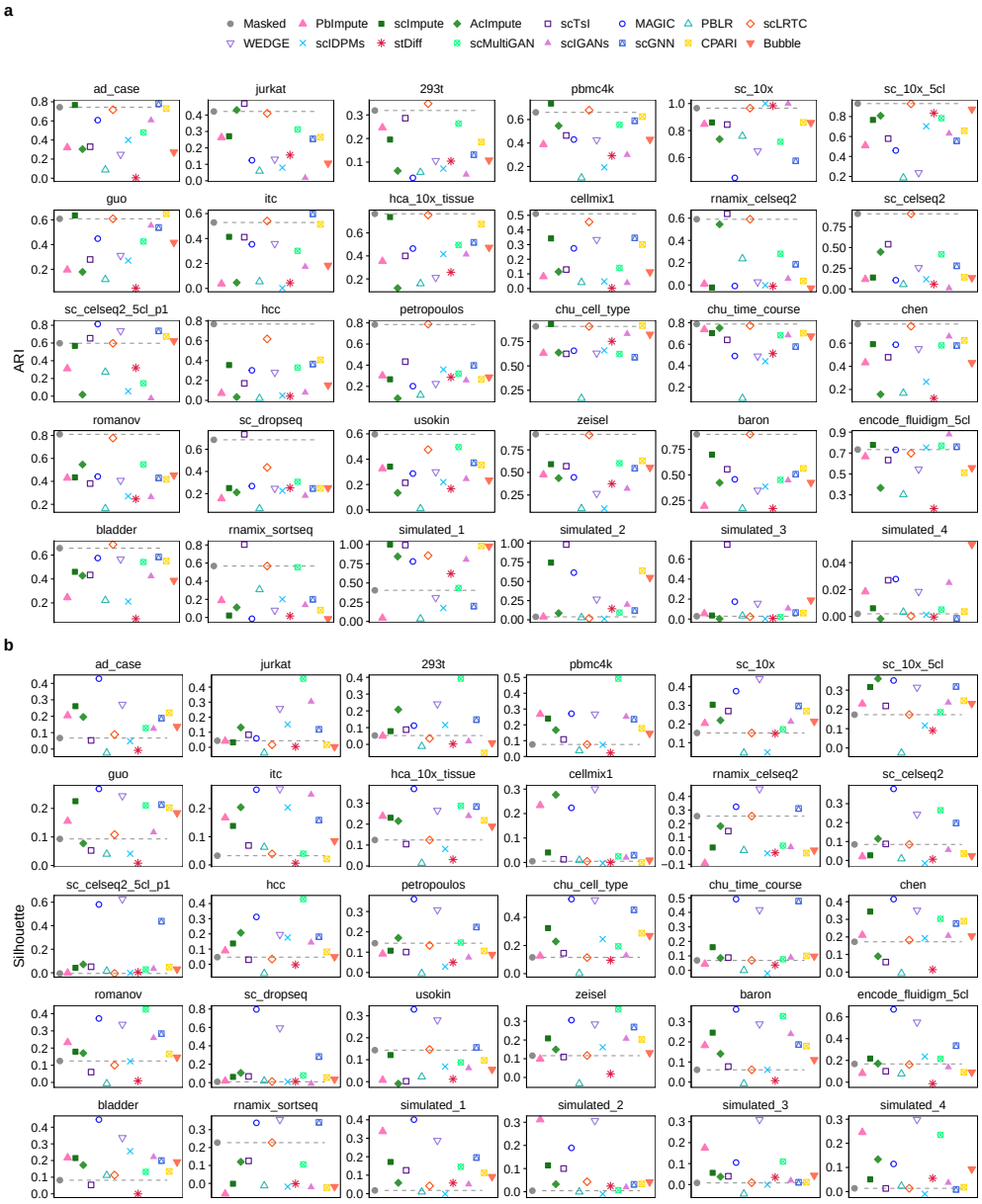

Fig. 3 Cell clustering consistency and coherency performance. **a** ARI. Each plot shows ARI scores of different imputation methods. Each point in a plot represents different methods. The horizontal dashed line represents the masked baseline value. **b** SC. Each plot shows SC scores of different imputation methods. Each point in a plot represents different methods. The horizontal dashed line represents the masked baseline value.

shows the worst SC scores for 13 datasets. The remaining 12 methods, namely PbImpute, scImpute, AcImpute, scTsI, scLRTC, scIDPMs, stDiff, scMultiGAN, scIGANs, scGNN, CPARI, and Bubble, show moderate SC scores.

Figs. 4 and 5 represent uniform manifold approximation and projection (UMAP) visualization of the cell clustering for the 15 imputation methods in terms of 4 real datasets, and 4 simulated datasets, respectively. The datasets with the highest number of cells are selected from 4 protocols, including 10x Chromium, CEL-seq2, SMART-seq2, and Drop-seq, to perform a qualitative evaluation. A qualitative assessment of UMAP plots provides a visual perspective on cluster consistency and coherency that complements the quantitative evaluations. The comparison of cluster structures among the 15 methods shows that MAGIC and WEDGE produce the most visually coherent clusters across 4 real datasets, including sc_celseq2_5cl_p1 with small samples. On the other hand, stDiff and PBLR show the least coherent cluster structures across 4 real datasets, with clusters appearing fragmented or poorly separated compared to the cell clustering based on the ground truth data. In the simulated datasets, 5 methods, namely PbImpute, scImpute, MAGIC, WEDGE, and scMultiGAN, maintain visually distinct clusters across 4 simulated datasets. On the other hand, the remaining 10 methods show limited ability to recover the cluster structure of the ground truth data. This suggests that these 10 methods have lower robustness to dropout-induced sparsity.

Tables 2 and 3 report normalized mutual information (NMI) and purity scores, respectively, comparing cell clustering results from the imputed data with those from ground truth data across the 15 imputation methods for the 26 real and 4 simulated datasets. An NMI score of 1 indicates perfect agreement between the 2 clustering results, whereas 0 indicates independence. Similarly, a purity score of 1 indicates that each predicted cluster contains cells from a single ground truth cluster, while 0 indicates complete mixing of cells from different ground truth clusters. The analyses of NMI and purity scores among the 15 imputation methods show that these scores are largely consistent with the ARI scores, where scLRTC exhibits the best performance in 13 datasets, PBLR and stDiff show the lowest NMI and purity scores in 8 datasets, and the remaining 12 methods, including PbImpute, scImpute, AcImpute, scTsI, MAGIC, WEDGE, scIDPMs, scMultiGAN, scIGANs, scGNN, CPARI, and Bubble, show moderate NMI and purity scores. This consistency across multiple cell clustering metrics reinforces the robustness of the observed performance differences among the 15 methods.

In summary, both quantitative and qualitative evaluations reveal substantial variability in cell clustering performance across the 15 imputation methods in terms of 26 real and 4 simulated datasets. scLRTC shows the best consistency performance, as supported by ARI, NMI, and purity, and MAGIC and WEDGE show the best coherency performance, as supported by SC and UMAP visualization. Conversely, PBLR and stDiff show the worst results in terms of consistency and coherency.

2.3 DE Analysis

Following Hou et al. [19], 3 different DE analyses are conducted, namely DE enrichment analysis, null DE analysis, and effect size analysis, as illustrated in Fig. 6. The

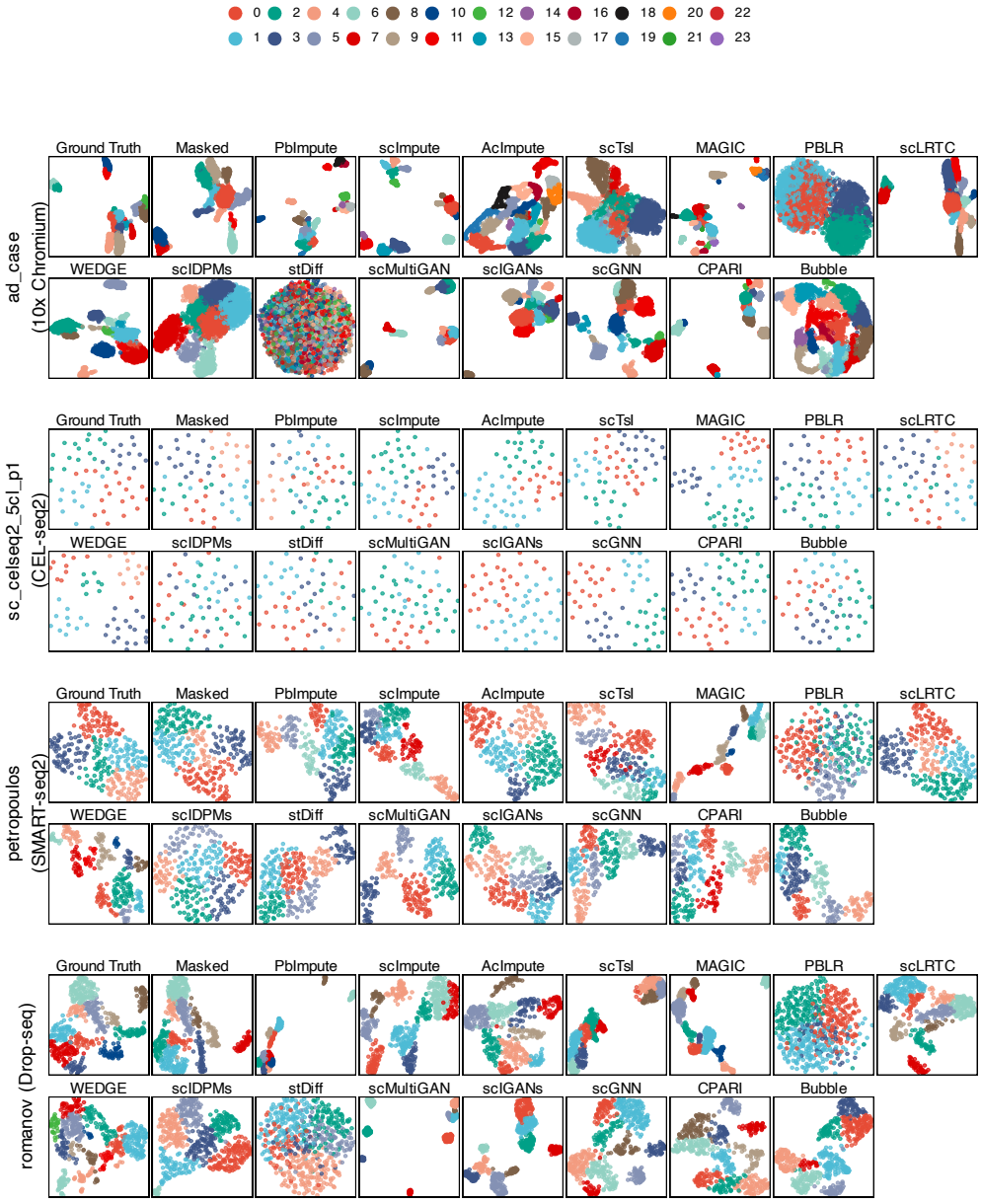

Fig. 4 UMAP visualization of the cell clustering using 4 real datasets with different protocols. Each plot shows the UMAP visualization of the method. Different colors represent different clusters.

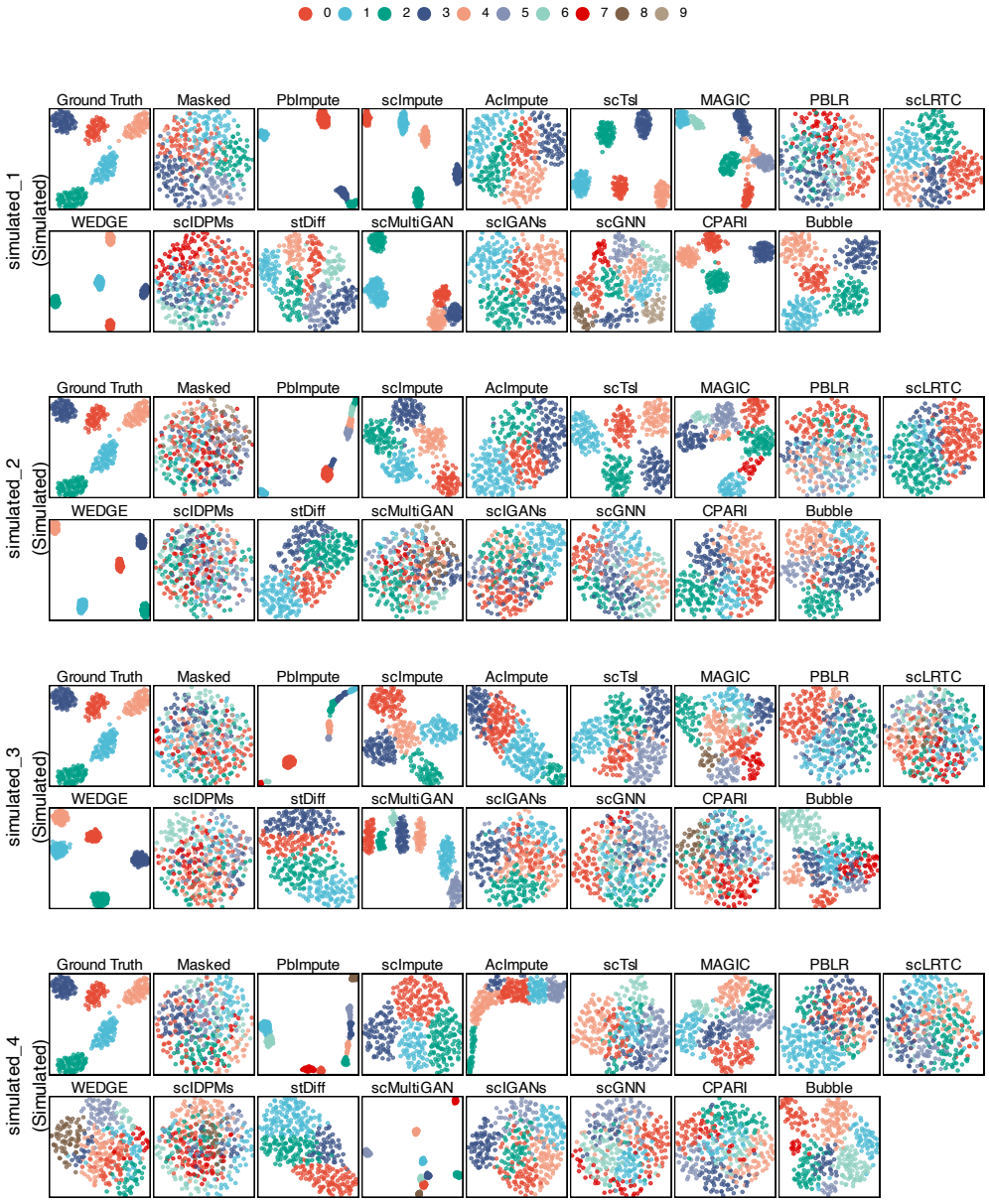

Fig. 5 UMAP visualization of the cell clustering using 4 simulated datasets with different dropout rates. Each plot shows the UMAP visualization of the method. Different colors represent different clusters.

Table 2 NMI of cell clustering based on the imputed and ground truth expression values. The bold values in each row represent the best performance methods.

Dataset	Masked	PbImpute	scImpute	AcImpute	scTsI	MAGIC	PBLR	scLRTC	WEDGE	scIDPMs	stDiff	scMultiGAN	scIGANs	scGNN	CPARI	Bubble
ad_case	0.825	0.547	0.811	0.585	0.491	0.766	0.157	0.812	0.549	0.550	0.028	0.674	0.704	**0.844**	0.830	0.498
jurkat	0.531	0.372	0.356	0.526	**0.572**	0.196	0.093	0.534	0.304	0.148	0.204	0.448	0.041	0.424	0.339	0.196
293t	0.450	0.311	0.274	0.110	0.458	0.089	0.083	**0.466**	0.249	0.146	0.141	0.364	0.139	0.256	0.274	0.141
pbmc4k	0.777	0.551	**0.792**	0.689	0.630	0.628	0.184	0.791	0.630	0.346	0.354	0.711	0.478	0.705	0.758	0.562
sc_10x	0.955	0.882	0.903	0.815	0.880	0.695	0.712	0.955	0.781	**1.000**	0.974	0.826	**1.000**	0.763	0.903	0.902
sc_10x_5cl	0.949	0.761	0.843	0.864	0.751	0.719	0.239	**0.946**	0.509	0.789	0.872	0.884	0.708	0.767	0.815	0.902
guo	0.780	0.447	0.772	0.305	0.530	0.687	0.257	**0.793**	0.545	0.480	0.122	0.620	0.788	0.738	0.786	0.642
itc	0.565	0.096	0.440	0.106	0.513	0.480	0.088	0.567	0.511	0.026	0.084	0.358	0.233	**0.634**	0.573	0.325
hca_10x_tissue	0.850	0.555	0.784	0.215	0.603	0.684	0.277	**0.844**	0.455	0.605	0.394	0.660	0.570	0.707	0.750	0.654
cellmix1	0.546	0.143	0.377	0.124	0.187	0.325	0.075	**0.455**	0.345	0.114	0.048	0.168	0.054	0.351	0.329	0.161
rnamix_celseq2	0.590	0.066	0.004	0.607	**0.619**	0.028	0.258	0.590	0.101	0.024	0.021	0.313	0.138	0.287	0.082	0.019
sc_celseq2	0.869	0.192	0.223	0.550	0.563	0.246	0.126	**0.869**	0.346	0.218	0.177	0.428	0.045	0.378	0.248	0.215
sc_celseq2_5cl_p1	0.696	0.384	0.599	0.091	0.741	**0.816**	0.309	0.696	0.744	0.140	0.351	0.220	0.001	0.744	0.675	0.653
hcc	0.792	0.203	0.509	0.079	0.310	0.536	0.043	**0.724**	0.475	0.095	0.084	0.420	0.152	0.569	0.566	0.301
petropoulos	0.788	0.425	0.382	0.146	0.523	0.381	0.141	**0.782**	0.368	0.443	0.401	0.443	0.406	0.526	0.391	0.419
chu_cell_type	0.932	0.731	0.920	0.748	0.740	0.809	0.295	**0.932**	0.795	0.807	0.809	0.764	0.863	0.745	0.909	0.859
chu_time_course	0.763	0.725	0.671	0.728	0.600	0.604	0.162	**0.747**	0.602	0.537	0.506	0.645	0.648	0.620	0.671	0.628
chen	0.884	0.625	0.785	0.298	0.668	0.777	0.292	**0.875**	0.766	0.511	0.188	0.759	0.824	0.795	0.821	0.676
romanov	0.864	0.603	0.630	0.658	0.579	0.657	0.228	**0.861**	0.635	0.410	0.327	0.683	0.437	0.655	0.637	0.627
sc_dropseq	0.719	0.231	0.388	0.359	**0.773**	0.398	0.168	0.565	0.382	0.314	0.417	0.381	0.308	0.382	0.388	0.383
usokin	0.646	0.457	0.431	0.197	0.298	0.407	0.057	**0.557**	0.415	0.328	0.208	0.553	0.303	0.484	0.442	0.360
zeisel	0.919	0.624	0.712	0.588	0.685	0.674	0.156	**0.908**	0.508	0.212	0.430	0.682	0.474	0.707	0.719	0.704
baron	0.920	0.376	0.753	0.645	0.621	0.672	0.239	**0.920**	0.555	0.597	0.228	0.637	0.608	0.710	0.717	0.617
encode_fluidigm_5cl	0.764	0.708	0.800	0.511	0.660	0.764	0.398	0.723	0.601	0.777	0.312	0.810	**0.874**	0.777	0.606	0.664
bladder	0.802	0.390	0.665	0.641	0.609	0.749	0.402	**0.813**	0.742	0.450	0.123	0.729	0.635	0.766	0.737	0.593
rnamix_sortseq	0.593	0.219	0.082	0.156	**0.788**	0.026	0.275	0.593	0.169	0.273	0.033	0.553	0.181	0.286	0.102	0.026
simulated_1	0.395	0.073	**1.000**	0.814	0.991	0.793	0.075	0.848	0.466	0.209	0.717	0.456	0.774	0.331	0.975	0.967
simulated_2	0.074	0.079	0.707	0.128	**0.975**	0.602	0.052	0.031	0.475	0.045	0.181	0.153	0.217	0.175	0.630	0.539
simulated_3	0.060	0.081	0.049	0.015	**0.714**	0.231	0.072	0.058	0.340	0.024	0.018	0.045	0.128	0.094	0.099	0.227
simulated_4	0.026	0.037	0.015	0.012	0.055	0.048	0.015	0.016	0.059	0.023	0.010	0.030	0.044	0.033	0.024	**0.111**

Table 3 Purity of cell clustering based on the imputed and ground truth expression values. The bold values in each row represent the best performance methods.

Dataset	Masked	PbImpute	scImpute	AcImpute	scTsI	MAGIC	PBLR	scLRTC	WEDGE	scIDPMs	stDiff	scMultiGAN	scIGANs	scGNN	CPARI	Bubble
ad_case	0.803	0.611	0.814	0.677	0.515	0.846	0.268	0.803	0.618	0.543	0.204	0.648	0.731	**0.850**	0.830	0.557
jurkat	0.631	0.503	0.545	0.699	**0.730**	0.461	0.326	0.644	0.549	0.381	0.426	0.600	0.279	0.647	0.585	0.426
293t	0.480	0.471	0.496	0.297	0.489	0.315	0.308	**0.523**	0.468	0.337	0.379	0.517	0.299	0.464	0.515	0.353
pbmc4k	0.755	0.637	**0.859**	0.828	0.686	0.727	0.309	0.781	0.735	0.435	0.468	0.729	0.570	0.778	0.775	0.661
sc_10x	0.989	0.994	**1.000**	0.994	0.994	**1.000**	0.912	0.989	**1.000**	**1.000**	0.994	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**
sc_10x_5cl	0.968	0.715	0.980	0.918	0.946	0.980	0.512	0.967	0.796	0.830	0.972	**1.000**	0.925	0.982	0.980	0.980
guo	0.776	0.464	0.776	0.327	0.478	0.744	0.275	0.784	0.573	0.413	0.220	0.506	0.739	0.728	**0.806**	0.574
itc	0.711	0.413	0.659	0.413	0.699	0.756	0.358	0.771	0.764	0.368	0.413	0.607	0.498	**0.823**	0.756	0.612
hca_10x_tissue	0.844	0.618	0.815	0.377	0.671	0.769	0.428	**0.840**	0.554	0.565	0.478	0.676	0.599	0.734	0.778	0.695
cellmix1	0.792	0.491	0.736	0.491	0.566	0.604	0.491	**0.774**	0.698	0.528	0.472	0.566	0.453	0.698	0.679	0.509
rnamix_celseq2	0.725	0.536	0.507	**0.841**	0.754	0.507	0.667	0.725	0.522	0.507	0.507	0.667	0.551	0.696	0.580	0.507
sc_celseq2	0.964	0.564	0.636	0.836	0.836	0.600	0.582	**0.964**	0.691	0.636	0.491	0.745	0.527	0.727	0.636	0.564
sc_celseq2_5cl_p1	0.881	0.695	0.831	0.390	0.847	**0.932**	0.627	0.881	0.898	0.458	0.644	0.508	0.322	0.898	0.864	0.831
hcc	0.902	0.338	0.656	0.300	0.477	0.708	0.232	**0.861**	0.610	0.296	0.262	0.510	0.339	0.737	0.658	0.445
petropoulos	0.898	0.610	0.587	0.407	0.754	0.590	0.416	**0.898**	0.574	0.636	0.603	0.610	0.593	0.715	0.584	0.587
chu_cell_type	0.971	0.883	**0.971**	0.907	0.907	0.966	0.566	**0.971**	0.937	**0.971**	0.937	0.922	0.937	0.932	0.966	0.966
chu_time_course	0.882	0.824	0.824	0.869	0.804	0.824	0.464	**0.876**	0.824	0.804	0.699	0.817	0.817	0.824	0.824	0.810
chen	0.845	0.679	0.761	0.364	0.656	0.777	0.300	**0.829**	0.655	0.421	0.228	0.682	0.793	0.731	0.720	0.636
romanov	0.883	0.672	0.599	0.740	0.576	0.637	0.333	**0.841**	0.627	0.424	0.386	0.698	0.457	0.623	0.620	0.583
sc_dropseq	0.870	0.609	0.652	0.587	**0.957**	0.674	0.609	0.739	0.652	0.652	0.696	0.717	0.674	0.652	0.652	0.652
usokin	0.829	0.653	0.688	0.547	0.606	0.706	0.394	0.753	0.706	0.606	0.512	**0.824**	0.576	0.771	0.718	0.635
zeisel	0.960	0.700	0.765	0.689	0.717	0.805	0.349	**0.955**	0.607	0.388	0.498	0.764	0.566	0.802	0.785	0.777
baron	0.961	0.514	0.829	0.787	0.691	0.740	0.358	**0.961**	0.655	0.688	0.442	0.706	0.673	0.769	0.730	0.631
encode_fluidigm_5cl	0.904	0.863	0.877	0.699	0.808	0.863	0.616	0.890	0.781	0.890	0.562	0.863	**0.959**	0.863	0.671	0.685
bladder	0.705	0.383	0.593	0.639	0.546	0.755	0.409	**0.789**	0.720	0.413	0.252	0.664	0.647	0.729	0.656	0.622
rnamix_sortseq	0.750	0.583	0.483	0.583	**0.917**	0.467	0.667	0.750	0.583	0.633	0.467	0.833	0.567	0.667	0.517	0.450
simulated_1	0.680	0.330	**1.000**	0.935	0.998	0.927	0.360	0.935	0.670	0.540	0.865	0.710	0.917	0.593	0.993	0.990
simulated_2	0.380	0.352	0.885	0.367	**0.993**	0.845	0.325	0.287	0.685	0.300	0.425	0.435	0.497	0.472	0.833	0.792
simulated_3	0.335	0.362	0.323	0.258	**0.890**	0.537	0.330	0.335	0.625	0.270	0.273	0.305	0.388	0.385	0.398	0.520
simulated_4	0.285	0.315	0.275	0.260	0.328	0.318	0.278	0.273	0.352	0.280	0.250	0.305	0.318	0.278	0.278	**0.400**

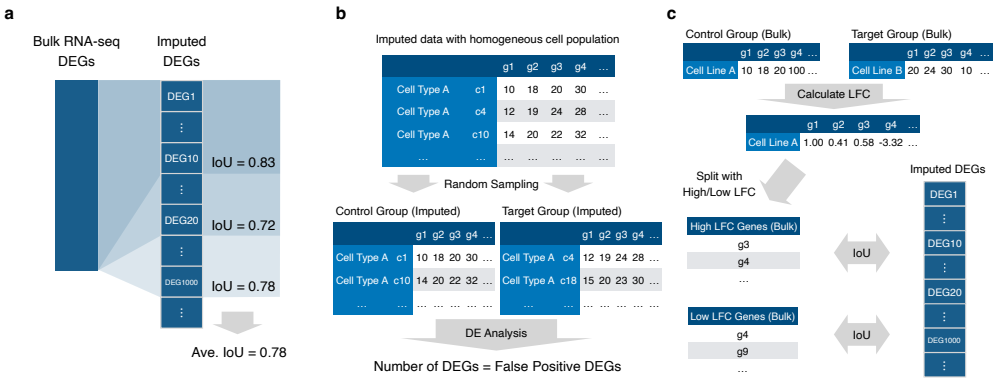

Fig. 6 The overview of 3 complementary analyses of DE analysis. **a** DE enrichment analysis, **b** null DE analysis, and **c** effect size analysis.

DE enrichment analysis evaluates how well differentially expressed genes (DEGs) from imputed scRNA-seq data recover DEGs identified from bulk RNA-seq data, which serve as the ground truth DEGs. The DEGs from imputed scRNA-seq data are ranked by p values or log-fold change (LFC) if there is a tie for p values, and the intersection over union (IoU) between the bulk RNA-seq DEGs and the top $10i$ imputed DEGs is computed for i from 1 to 100. The average IoU across all 100 values of i is used to measure the performance. The null DE analysis assesses the robustness of imputed data to false positive DEGs. Ideally, DEGs should not be identified when control and target groups belong to the same cell population, and any identified DEGs can be treated as false positive DEGs under such conditions. The imputed scRNA-seq data is filtered to a single cell type or cell line to obtain a homogeneous cell population. From this cell population, N_1 and N_2 cells are randomly sampled as control and target groups, respectively, where $N_1 \leq N_2$ and $N_1, N_2 \in \{10, 50, 100\}$. All 6 combinations, namely $(N_1, N_2) = (10, 10), (10, 50), (10, 100), (50, 50), (50, 100), (100, 100)$, are tested to assess robustness across varying sample sizes and group balances, and DE analysis is performed between the control and target groups. The number of false positive DEGs is used to measure the performance, where a lower count indicates greater robustness. The effect size analysis evaluates whether DEGs identified from imputed scRNA-seq data capture genes with both high and low LFC in bulk RNA-seq data. Here, LFC is defined as $\text{LFC} = \log_2(y_{\text{target}}/y_{\text{control}})$, where y_{control} and y_{target} represent the gene expression values of the control and target groups, respectively. Genes in the upper and lower 10 % of the LFC distribution from bulk RNA-seq data are defined as high- and low-LFC genes, respectively. As in the DE enrichment analysis, the IoU is used to measure the overlap between high- or low-LFC genes from bulk RNA-seq data and the DEGs from imputed scRNA-seq data.

Figs. 7a–c show the DE enrichment analysis performance for 15 imputation methods in terms of 3 datasets, namely sc_10x_5cl, encode_fluidigm_5cl, and hca_10x_tissue. These datasets are selected on the basis of the availability of corresponding bulk RNA-seq data. MAST [55] and the Wilcoxon rank-sum test [56–58]

are used to identify DEGs from the imputed data, and limma [59] is used to identify DEGs from the bulk RNA-seq data.

The comparison of the DE enrichment analysis performance across the 15 methods evaluated on the 3 datasets shows that AcImpute and scLRTC achieve the best overall performance. In addition, 10 methods, namely PbImpute, scImpute, scTsI, MAGIC, WEDGE, scIDPMs, scIGANs, scGNN, CPARI, and Bubble, exhibit moderate performance across the 3 datasets. Conversely, scMultiGAN, stDiff, and PBLR show the worst performance on sc_10x_5cl, encode_fluidigm_5cl, and hca_10x_tissue, respectively. However, the masked baseline demonstrates high IoU scores on sc_10x_5cl, and none of the 15 methods significantly outperform it.

Figs. 7d–f show the null DE analysis performance using MAST [55] and the Wilcoxon rank-sum test [56–58] for 15 imputation methods in terms of 3 datasets. The analysis of false positive DEGs in the null DE analysis shows that 5 methods, namely WEDGE, MAGIC, scIGANs, scGNN, and scMultiGAN, produce almost no false positive DEGs across all datasets. In addition, 9 methods, namely PbImpute, scImpute, AcImpute, scTsI, PBLR, scIDPMs, stDiff, CPARI, and Bubble, produce false positive DEGs across 3 datasets. In contrast, scLRTC produces substantial false positive DEGs under MAST in 2 datasets, sc_10x_5cl and hca_10x_tissue.

Figs. 7g and h show the effect size analysis performance of high- and low-LFC genes, respectively, for 15 imputation methods using MAST [55] in sc_10x_5cl. MAST and sc_10x_5cl are selected as representative examples. A thorough evaluation of effect size analysis performance shows that AcImpute, MAGIC, and WEDGE achieve the best performance, with high IoU scores in both high- and low-LFC genes. In addition, 9 methods, namely PbImpute, scImpute, scTsI, scLRTC, scMultiGAN, scIGANs, scGNN, CPARI, and Bubble, exhibit moderate performance. On the other hand, PBLR, scIDPMs, and stDiff show the worst performance, with low IoU scores in high- or low-LFC genes.

In summary, AcImpute achieves the best overall performance, as supported by the highest DE enrichment analysis performance across 3 datasets and the best effect size analysis performance. However, AcImpute produces false positive DEGs under MAST in sc_10x_5cl. MAGIC, WEDGE, scMultiGAN, scIGANs, and scGNN produce nearly 0 false positives across all datasets. Conversely, PBLR shows the worst overall performance, with the worst DE enrichment and effect size analysis performance.

2.4 Marker Gene Analysis

Fig. 8 shows marker gene expression of 4 cell types, namely T cell, B cell, natural killer (NK) cell, and monocyte, in hca_10x_tissue for 15 distinct methods. Generally, *CD3D* and *CD3E* are considered to be marker genes for T cells, *CD79A* and *MS4A1* for B cells, *NKG7* and *GNLY* for NK cells, and *CD14* and *LYZ* for monocytes [17, 60]. The analysis of marker gene expression reveals that scImpute, MAGIC, scIGANs, and CPARI achieve the best performance, as these methods show strong expression levels of marker genes in the corresponding cell types. In addition, 6 methods, namely PbImpute, AcImpute, scTsI, WEDGE, scMultiGAN, and Bubble, show moderate performance, as these methods show relatively strong expression levels of marker genes

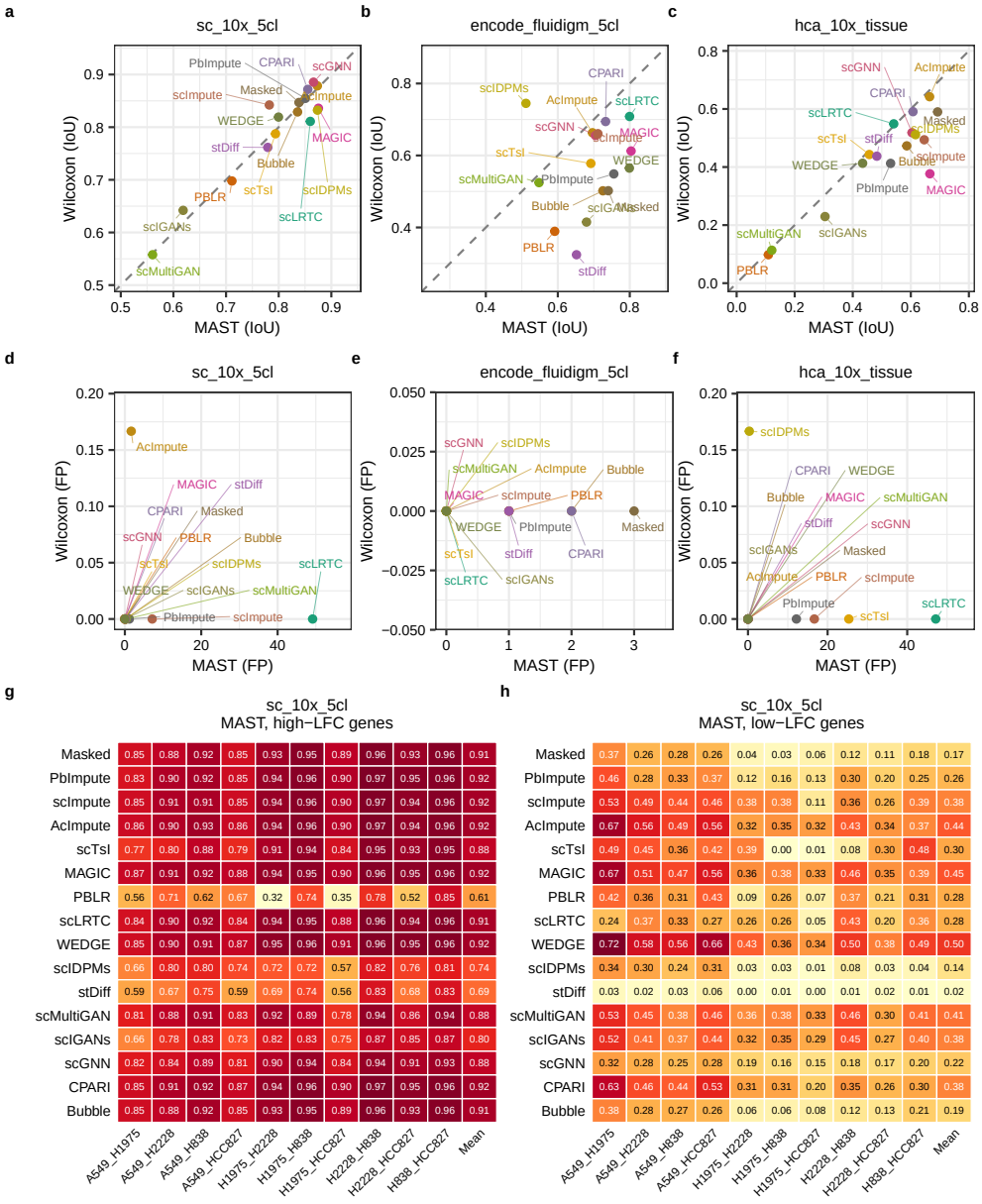

Fig. 7 DE analysis performance. **a**–**c** DE enrichment analysis. IoU between DEGs identified from imputed scRNA-seq data and bulk RNA-seq data using MAST and the Wilcoxon rank-sum test. The dashed line represents equal performance. Each point represents an imputation method. **d**–**f** Null DE analysis. Average number of false positive DEGs across 6 different sample sizes for A549 (**d**), GM12878 (**e**), and monocyte (**f**). **g**–**h** Effect size analysis. IoU between DEGs identified from imputed scRNA-seq data and high- (**g**) or low- (**h**) LFC genes from bulk RNA-seq data.

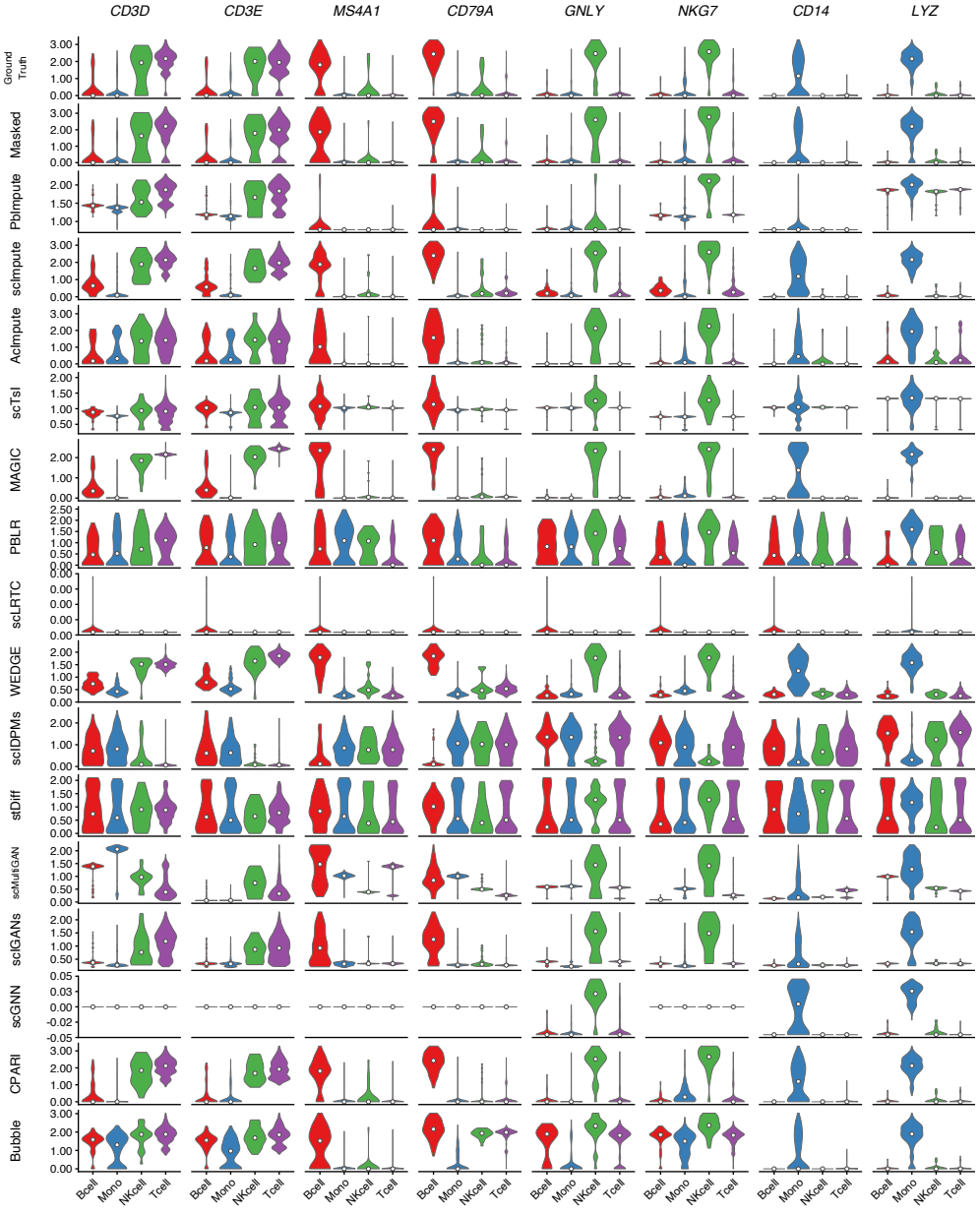

Fig. 8 Comparison of imputation methods for marker gene expression in hca_10x_tissue. Violin plots show the distribution of expression levels for 8 marker genes across 4 cell types, T cell, B cell, NK cell, and monocyte (shown as Mono). The y-axis represents gene expression values.

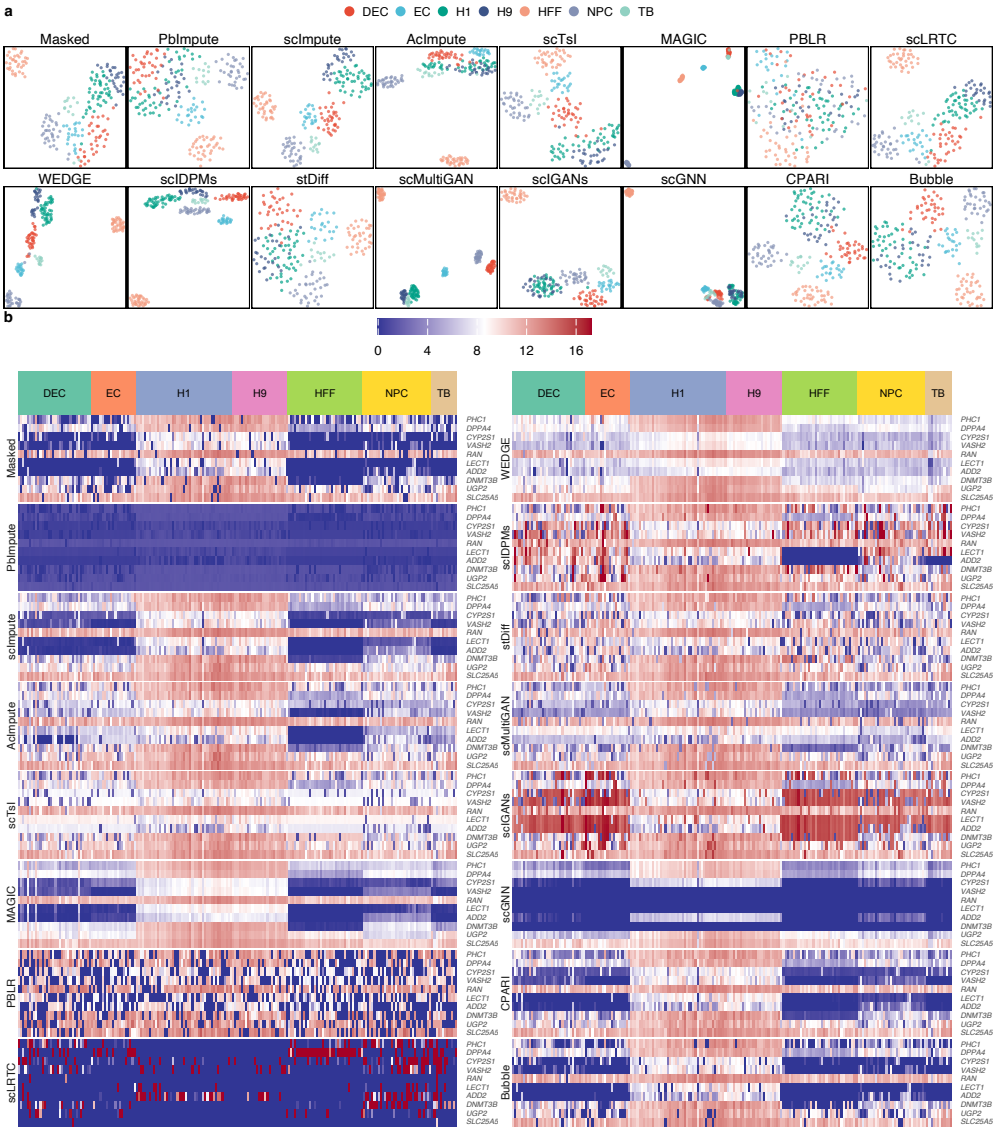

Fig. 9 Marker gene expression performance on chu_cell_type. **a** UMAP visualizations with 7 cell type labels, namely DEC, EC, H1, H9, HFF, NPC, and TB, colored by cell type. Each plot represents an imputation method. **b** Heatmaps of 10 marker gene expression values across the 7 cell types. The x-axis represents individual cells ordered by cell type, and the y-axis represents marker genes.

for B cells and monocytes. Conversely, 5 methods, namely PBLR, scLRTC, scIDPMs, stDiff, and scGNN, show the worst performance, as they produce similar marker gene expression levels across the 4 cell types or show near-zero expression levels.

Figs. 9a and b show UMAP visualizations with cell type labels, and marker gene expression for the 15 imputation methods in chu_cell_type, respectively. The UMAP

visualizations show that 5 methods, namely scImpute, MAGIC, WEDGE, scIDPMs, and scMultiGAN, clearly separate 3 cell types, including EC, HFF, and NPC. In addition, 5 methods, namely AcImpute, scTsI, scLRTC, scIGANs, and CPARI, show moderate separation of 2 cell types, with HFF and NPC partially separated. Conversely, 5 methods, namely PbImpute, PBLR, stDiff, scGNN, and Bubble, do not show clear separation, as the 7 cell types are mixed together. The analysis of the marker gene expression shows that 3 of the 5 methods that achieve clear separation in the UMAP visualizations, namely scImpute, MAGIC, and WEDGE, exhibit distinct marker gene expression patterns for 2 cell types, namely H1 and H9. In addition, 5 methods, namely AcImpute, scTsI, scMultiGAN, CPARI, and Bubble, show moderate marker gene expression patterns for H1 and H9. In contrast, 7 methods, including PbImpute, PBLR, scLRTC, scIDPMs, stDiff, scIGANs, and scGNN, show the worst performance, with no distinct marker gene expression patterns.

In summary, the marker gene analysis reveals substantial variability in the ability of imputation methods to preserve biologically meaningful gene expression patterns across the 2 datasets. scImpute and MAGIC show the best overall performance. These 2 methods consistently exhibit strong cell-type-specific marker gene expression in hca_10x_tissue and distinct expression patterns in chu_cell_type, which is further supported by clear cell type separation in the UMAP visualizations. scIGANs and CPARI also perform well in hca_10x_tissue, though their performance is less consistent in chu_cell_type. Conversely, PBLR, stDiff, and scGNN show the worst results, as they produce indistinct marker gene expression patterns and poor cell type separation across the 2 datasets.

2.5 Trajectory Analysis

Figs. 10a and b show pseudo-temporal ordering score (POS) and Kendall's rank correlation coefficient (KRCC) for 15 imputation methods in terms of 2 datasets, namely petropoulos and chu_time_course. High POS and KRCC represent proximity to true cell development labels. The comparison of POS and KRCC across 15 methods evaluated on 2 datasets shows that PbImpute, scImpute, scLRTC, CPARI, and Bubble achieve relatively high performance in both datasets. In addition, 7 methods, namely AcImpute, scTsI, MAGIC, PBLR, WEDGE, scMultiGAN, and scGNN, show moderate performance. Conversely, scIDPMs and scIGANs show the worst performance in petropoulos dataset, while stDiff shows the worst performance in chu_time_course dataset. However, the masked baseline shows relatively high performance, and 10 methods, including AcImpute, scTsI, MAGIC, PBLR, WEDGE, scIDPMs, stDiff, scMultiGAN, scIGANs, and scGNN, do not exceed it.

Fig. 10c shows UMAP visualizations of trajectory analysis for 15 imputation methods in terms of 2 datasets. The qualitative analyses through the UMAP visualizations show that the pseudotime changes gradually for PbImpute and scImpute, with cells colored by pseudotime progressing smoothly. On the other hand, the UMAP visualizations of AcImpute, scTsI, PBLR, WEDGE, scIDPMs, stDiff, scMultiGAN, scIGANs, and CPARI show that the pseudotime does not change smoothly, with cells of different time points appearing intermixed along the trajectory. However, since trajectory

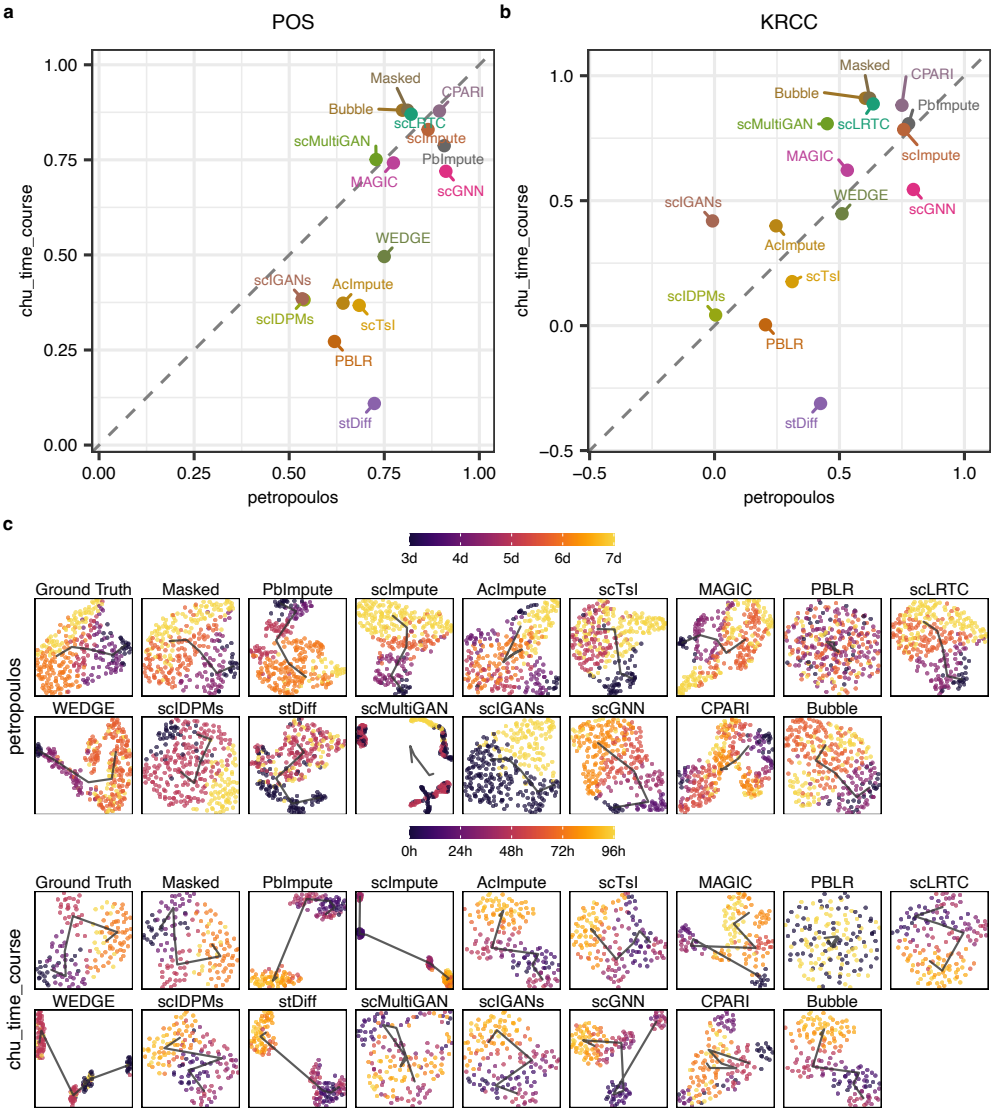

Fig. 10 Trajectory analysis performance. **a**–**b** POS and KRCC between the inferred pseudotime from imputed data and the true cellular development time label, respectively. The x-axis and y-axis represent the performance of petropoulos and chu_time_course, respectively. The dashed line represents equal performance across the 2 datasets. Each point in the plot represents an imputation method. **c** UMAP visualizations with inferred pseudotime trajectories for petropoulos (top) and chu_time_course (bottom). Cells are colored by inferred pseudotime.

analysis is performed in higher dimensions than UMAP, POS and KRCC results may not be fully reflected in the 2-dimensional UMAP projections.

In summary, the trajectory analysis reveals that 5 methods, including PbImpute, scImpute, scLRTC, CPARI, and Bubble, consistently preserve the temporal ordering

of cells across 2 datasets, while scIDPMs, stDiff, and scIGANs perform the worst. However, the fact that 10 methods fail to outperform the masked baseline highlights a critical challenge that imputation can distort the underlying developmental structure of scRNA-seq data. This can potentially lead to less accurate trajectory analysis than simply using the data without imputation.

2.6 Cell Type Annotation

Fig. 11 shows the macro accuracy (ACC), macro precision (PR), macro recall (RC), and macro F1 score (F1) of cell type annotation for the 15 imputation methods in terms of 6 datasets, namely sc_10x_5cl, hca_10x_tissue, chu_cell_type, baron, encode_fluidigm_5cl, and bladder, using 1D convolutional neural network (1D-CNN) and scGPT [61] for annotation. Higher ACC represents better overall accuracy of cell type annotations, higher PR represents better precision in identifying true cell types, higher RC represents better recall of true cell types, and higher F1 represents a better balance between PR and RC.

A thorough analysis of ACC, PR, RC, and F1 for the 15 imputation methods in terms of 6 datasets reveals that MAGIC achieves the best overall performance. In addition, 13 methods, namely PbImpute, scImpute, AcImpute, scTsI, PBLR, scLRTC, WEDGE, scIDPMs, scMultiGAN, scIGANs, scGNN, CPARI, and Bubble, show moderate results. On the other hand, stDiff shows the worst performance. The comparison of the 2 cell type annotation methods, namely 1D-CNN and scGPT, shows that the performance differences between them are substantial in the 2 cell line datasets, namely sc_10x_5cl and encode_fluidigm_5cl, while the differences are relatively small in the 4 tissue datasets, namely hca_10x_tissue, chu_cell_type, baron, and bladder. However, none of the 15 methods significantly outperform the masked baseline in 3 datasets, including sc_10x_5cl, hca_10x_tissue, and chu_cell_type.

In summary, MAGIC shows the best cell type annotation performance, while stDiff shows the worst performance. Furthermore, the choice of cell type annotation methods introduces considerable variability in the cell line datasets, namely sc_10x_5cl and encode_fluidigm_5cl.

3 Discussion

In this study, we conduct a systematic evaluation of 15 imputation methods across 6 downstream tasks, including numerical gene expression recovery, cell clustering, DE analysis, marker gene analysis, trajectory analysis, and cell type annotation. The evaluation is performed using 26 real and 4 simulated datasets generated from 10 different protocols, including 10x Chromium, CEL-seq2, SMART-seq2, SMART-seq, Drop-seq, STRT-Seq, inDrop, Fluidigm C1, Microwell-seq, and Sort-seq. The results reveal substantial variability in the performance of imputation methods across different downstream tasks and datasets, which highlights the importance of carefully selecting imputation methods based on specific downstream analyses and dataset characteristics.

Table 4 summarizes the performance of the 15 imputation methods across 6 downstream tasks. MAGIC shows the best overall performance with the highest

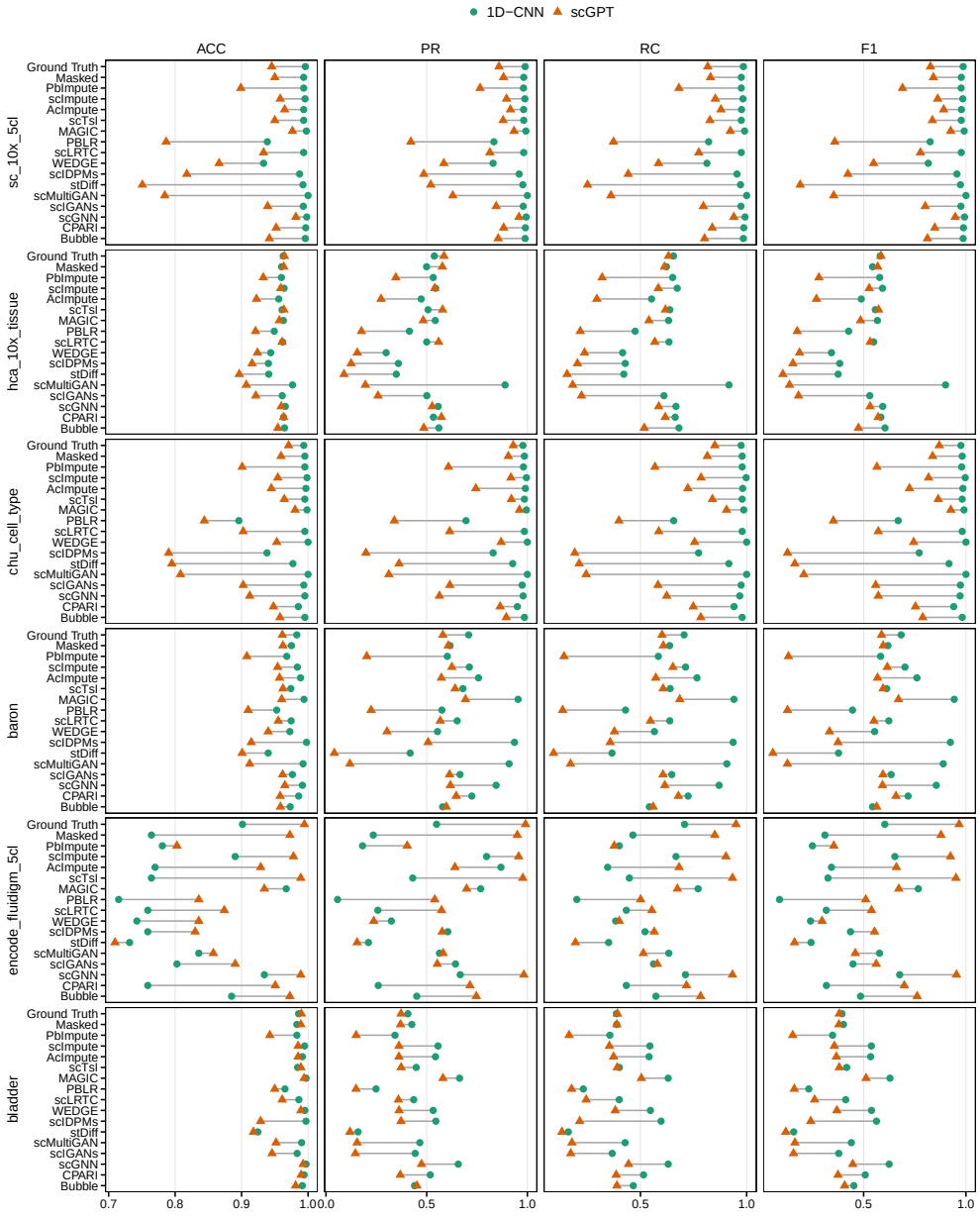

Fig. 11 Cell type annotation performance. Dumbbell plots comparing ACC, PR, RC, and F1 achieved by 1D-CNN (green circles) and scGPT [61] (orange triangles) across 6 datasets, including sc_10x_5cl, hca_10x_tissue, chu_cell_type, baron, encode_fluidigm_5cl, and bladder. The x-axis represents the value of each evaluation measure, and the y-axis represents different imputation methods. Each horizontal line connects the 1D-CNN and scGPT performances for a given method.

Table 4 Summary of the performance of imputation methods. ★, △, and × indicate best, moderate, and worst performance, respectively.

Category	Method	Numerical gene expression recovery	Cell clustering	DE analysis	Marker gene analysis	Trajectory analysis	Cell type annotation
Model-based	PbImpute	△	△	△	△	★	△
	scImpute	△	△	△	★	★	△
Smoothing-based	AcImpute	△	△	★	△	△	△
	scTsI	★	△	△	△	△	△
	MAGIC	△	★	△	★	△	★
Low-rank matrix-based	PBLR	★	×	×	×	△	△
	scLRTC	×	★	△	△	★	△
	WEDGE	★	★	△	△	△	△
Diffusion-based	scIDPMs	×	△	△	△	×	△
	stDiff	△	×	△	×	×	×
GAN-based	scMultiGAN	△	△	△	△	△	△
	scIGANs	×	△	△	△	×	△
GNN-based	scGNN	△	△	△	×	△	△
AE-based	CPARI	△	△	△	△	★	△
	Bubble	△	△	△	△	★	△

performance in 3 tasks, namely cell clustering, marker gene analysis, and cell type annotation, and moderate performance in the remaining 3 tasks. In addition, scImpute and WEDGE show moderately high overall performance with the best performance in 2 tasks, and moderate performance in the remaining 4 tasks. Similarly, PbImpute, AcImpute, scTsI, scMultiGAN, CPARI, and Bubble show moderately low overall performance, with the best performance in 0 or 1 tasks, and moderate performance in the remaining 5 or 6 tasks. Conversely, PBLR, scLRTC, scIDPMs, stDiff, scIGANs, and scGNN show the worst overall performance, with the worst performance in 1 to 3 tasks. We find that traditional methods, such as scImpute, MAGIC, and WEDGE, show the best or moderately high overall performance across the 6 tasks, whereas none of the 7 DL-based methods reach the same level of overall performance. This suggests that traditional methods may be more effective at preserving biologically meaningful information across a wide range of downstream analyses, while the performance of DL-based methods can be more variable and may require careful tuning and validation for specific tasks.

In numerical gene expression recovery, the performance of imputation methods varies significantly across methods. scTsI, PBLR, and WEDGE achieve the best overall LND performance, with medians consistently close to 0. scTsI achieves this through its two-stage strategy, which first imputes dropouts using k-nearest neighbor (k-NN)-based averaging across neighboring cells and genes, and refines the initial estimates through ridge regression. PBLR preserves the data structure through its cell sub-population-based bounded low-rank matrix recovery. The boundaries constrain reconstructed values within biologically plausible ranges. WEDGE shows superior LND performance likely due to its biased low-rank matrix-based approach that assigns

low weights to 0 elements and minimizes approximation error for nonzero elements. This approach effectively separates true biological signal from dropouts without over-imputing zero entries. scTsI and WEDGE also exhibit the lowest MAE and MedAE, which further supports this finding. Protocol-wise analysis reveals that WEDGE maintains LND values closest to zero with compact distributions across all protocols, which indicates that it is the most protocol-robust approach among the 15 methods. For comparison with bulk RNA-seq data, WEDGE also achieves the best overall performance, while MAGIC shows the highest cell line-level correlation through its diffusion-based smoothing, which enforces locally coherent expression profiles aligned with averaged bulk patterns. In contrast, scMultiGAN exhibits the worst correlation with bulk RNA-seq data at both the pseudo-bulk and cell line levels despite its moderate numerical recovery performance from ground truth data. This highlights a trade-off between numerical accuracy and biological fidelity. Its dual-GAN architecture, which minimizes numerical recovery error, may overfit to ground truth distributions at the expense of generalization to bulk RNA-seq data. The over-imputation by scIDPMs may arise from its iterative denoising process that pushes sparse dropout entries toward higher-density non-zero modes across multiple sequential denoising steps. scLRTC consistently under-imputes because its low-rank matrix-based approach compresses the dynamic range of highly variable genes. scIGANs exhibits the highest protocol-wise MAE and MedAE that significantly exceed the masked baseline, and also shows the worst correlation with bulk RNA-seq data. This worst performance likely arises because its adversarially trained generator learns conservative mappings that fail to generalize across protocols and sequencing modalities. Furthermore, methods show largely consistent behavior on 10x Chromium, CEL-seq2, and Drop-seq datasets, while SMART-seq, SMART-seq2, and Fluidigm C1 datasets demonstrate higher instability. This is likely because the latter 3 protocols use only read-counts without UMIs. The absence of UMIs leads to increased technical noise that makes accurate imputation more challenging [17, 62].

In cell clustering, the performance of imputation methods also varies significantly across methods. In terms of consistency, scLRTC achieves the best performance. This is likely because scLRTC reconstructs gene expression values through its low-rank tensor completion that captures global gene expression patterns while preserving the distinct expression signatures that define cell clusters. In terms of coherency, MAGIC and WEDGE achieve the best performance. MAGIC produces tightly grouped clusters, likely because its diffusion-based smoothing over k-NN graphs harmonizes expression profiles within cell neighborhoods, which enhances intra-cluster homogeneity. WEDGE achieves coherent clusters through its biased low-rank matrix decomposition that suppresses noise-driven variability within clusters while preserving inter-cluster separation. The distinction between consistency and coherency suggests that well-separated clusters do not necessarily correspond to accurately recovered expression patterns, as MAGIC and WEDGE achieve the best cluster coherency but not the best cluster consistency. Conversely, PBLR and stDiff show the worst results in both consistency and coherency. For PBLR, this may be due to its bounded low-rank matrix recovery approach that compresses expression variability into a small number of latent factors and can merge expression signatures of distinct but transcriptionally

similar cell populations. For stDiff, its diffusion-based denoising process may over-smooth expression differences between closely related cell populations, which leads to poor cluster separation and inaccurate cluster assignments. Notably, none of the 15 methods exceed the ARI scores of the masked baseline in 12 datasets, which suggests that imputation does not universally improve cell clustering and can even degrade cluster consistency by introducing imputed expression patterns. Furthermore, the UMAP visualization reveals that PbImpute, scImpute, MAGIC, WEDGE, and scMultiGAN maintain visually distinct clusters across simulated datasets with varying dropout rates, while the remaining 10 methods show limited ability to recover cluster structures. This suggests that these 10 methods may be less robust to dropout-induced sparsity, which can lead to poor cluster recovery in datasets with high dropout rates.

In DE analysis, AcImpute achieves the best overall performance, with the highest DE enrichment analysis performance across 3 datasets and the best effect size analysis performance. This is likely due to its smoothing-based approach that leverages gene-gene relationships to estimate dropouts and gene expression values, which preserves the relative expression differences between cell populations. AcImpute also achieves high IoU scores in both high- and low-LFC genes, which indicates that it effectively recovers expression differences across a range of effect sizes. However, AcImpute produces false positive DEGs under MAST [55] in sc_10x_5cl, which suggests that its smoothing approach can introduce systematic expression patterns that create false differences between randomly partitioned cell groups. scLRTC also achieves the best DE enrichment analysis performance but produces substantial false positive DEGs under MAST in 2 datasets, sc_10x_5cl and hca_10x_tissue. This indicates that its low-rank tensor completion approach effectively recovers relative expression differences for DEG identification but can simultaneously introduce imputed expression patterns that inflate false positive rates. WEDGE, MAGIC, scIGANs, scGNN, and scMultiGAN produce nearly 0 false positives across all datasets. However, scMultiGAN exhibits the worst DE enrichment performance on sc_10x_5cl, which further reinforces the trade-off between numerical recovery accuracy and biological signal preservation observed in numerical gene expression recovery. The low false positive rates of these methods do not universally translate into accurate DEG identification, which indicates that avoiding false positives is necessary but not sufficient for accurate DE analysis. Conversely, PBLR shows the worst overall performance, with the worst DE enrichment and effect size analysis performance. This indicates that its bounded low-rank matrix recovery fails to preserve the relative expression differences between cell populations, which results in poor DEG identification regardless of effect size. Notably, the masked baseline demonstrates high IoU scores on sc_10x_5cl, and none of the 15 methods significantly outperform it, which suggests that imputation does not always improve DE analysis over data without imputation.

In marker gene analysis, we examine the preservation of biologically meaningful gene expression patterns by comparing marker gene expression levels across different cell types in imputed data. The results reveal that scImpute and MAGIC consistently show the best performance, with strong cell-type-specific marker gene expression patterns across the 2 datasets. scImpute shows the best performance, likely due to its mixture distribution-based approach that selectively imputes values on a per-gene

basis, which allows it to impute only the values that are likely to be technical dropouts while preserving biological zeros. This approach maintains the distinct marker gene expression patterns in their respective cell types while keeping non-expressing cell types at low expression levels. MAGIC achieves strong marker gene expression through its diffusion-based smoothing over a k-NN graph, which propagates expression signals among transcriptionally similar cells. This may amplify marker gene expression within the corresponding cell types without spreading signals across other populations, as the graph structure naturally limits diffusion within cell type boundaries. Conversely, PBLR, stDiff, and scGNN show the worst results across both datasets, as they produce indistinct marker gene expression patterns and poor cell type separation. For PBLR, its bounded low-rank matrix recovery approach compresses gene expression variability into a small number of latent factors, which in turn weakens the distinct expression peaks of marker genes and blurs cell-type-specific signatures. For stDiff, its diffusion-based denoising process can over-smooth localized expression signatures. For scGNN, its graph-based architecture may homogenize expression values across neighboring cells, thereby diminishing cell-type-specific marker gene patterns. Notably, scIGANs and CPARI perform well in hca_10x_tissue but show less consistent results in chu_cell_type, while scIDPMs performs poorly in hca_10x_tissue yet achieves clear cell type separation in the UMAP visualizations for chu_cell_type. This inconsistency between numerical gene expression recovery or cell clustering performance and marker gene analysis further reinforces the finding that overall imputation accuracy does not guarantee the preservation of biologically meaningful expression patterns.

In trajectory analysis, we evaluate the ability of imputation methods to preserve the temporal ordering of cells by comparing inferred pseudotime from imputed data with true cellular development time labels. The results reveal that PbImpute, scImpute, scLRTC, CPARI, and Bubble consistently preserve the temporal ordering of cells across the 2 datasets. This shows that 2 model-based methods, 1 low-rank matrix-based method, and 2 AE-based methods perform well in trajectory analysis. This may be due to the fact that these methods preserve the relative expression differences along developmental trajectories, which is critical for accurate pseudotime inference. The model-based methods, scImpute and PbImpute, selectively target dropout events while preserving the original expression values, which helps maintain the gradual expression changes that define developmental trajectories. scLRTC reconstructs gene expression values through its low-rank tensor completion that captures global gene expression patterns while preserving the distinct expression signatures that define developmental trajectories. The AE-based methods, CPARI and Bubble, learn latent representations using AE architectures that retain the continuous structure of developmental progression without collapsing intermediate states. Conversely, scIDPMs, stDiff, and scIGANs show the worst performance. The diffusion-based methods, namely scIDPMs and stDiff, use an iterative denoising process trained to denoise expression values toward high-density modes, which may collapse the subtle expression gradients between adjacent developmental stages into discrete expression states. This disrupts the continuous pseudotime ordering that trajectory inference relies on,

as cells at intermediate developmental stages are pushed toward the expression profiles of more mature or earlier stages. For scIGANs, its adversarial training aims to match the overall data distribution, but the GAN-based generation process may suffer from mode collapse that concentrates imputed values around high-density expression states rather than preserving the continuous gradients between developmental stages. This can disrupt the temporal ordering of cells by homogenizing expression profiles at intermediate stages. This is consistent with the UMAP visualizations, where cells of different time points appear intermixed along the trajectory for these methods. Furthermore, 10 methods, including AcImpute, scTsI, MAGIC, PBLR, WEDGE, scIDPMs, stDiff, scMultiGAN, scIGANs, and scGNN, fail to outperform the masked baseline, which highlights that imputation can distort the underlying developmental structure of scRNA-seq data and lead to less accurate trajectory inference than simply using the original data. This suggests that trajectory analysis, which depends on the preservation of continuous expression gradients rather than discrete cluster boundaries, is particularly sensitive to imputation artifacts that alter the relative ordering of expression values along developmental axes.

In cell type annotation, we assess the impact of imputation on the accuracy of cell type predictions by comparing annotations derived from imputed data with known cell type labels. The results reveal that MAGIC achieves the best overall performance across the 6 datasets. The strong performance of MAGIC may be attributed to its smoothing-based approach that propagates expression signals across similar cells, which enhances the expression profiles used by annotation classifiers to distinguish cell types without the elimination of cell-type-specific patterns. Conversely, stDiff shows the worst performance. This may be due to its diffusion-based denoising process that over-smooths expression profiles by iteratively refining values toward high-density modes, which can blur the boundaries between transcriptionally similar cell types. The comparison between 1D-CNN and scGPT reveals substantial performance differences in the 2 cell line datasets but relatively small differences in the 4 tissue datasets. This may be because cell line datasets contain more homogeneous populations with subtle transcriptomic differences that are more sensitive to the choice of annotation method, while tissue datasets contain more heterogeneous populations with larger expression differences that are consistently captured by both classifiers. Furthermore, none of the 15 methods significantly outperform the masked baseline in 3 datasets, which is consistent with the observations in cell clustering and DE analysis, and reinforces that imputation does not universally improve downstream task performance, particularly in datasets with sufficient sequencing depth.

4 Conclusions

Overall, the comprehensive evaluation across 6 downstream tasks reveals that although some imputation methods excel in specific tasks, there is no universally superior method across all tasks. This indicates that the choice of imputation method should be carefully tailored to the specific downstream task and dataset characteristics to ensure optimal performance. For instance, scImpute and MAGIC are recommended for tasks that require the preservation of biologically meaningful information, such as

marker gene analysis, trajectory analysis, and cell type annotation, while WEDGE may be more suitable for tasks that prioritize numerical gene expression recovery. The variability in performance across different tasks and datasets also suggests that researchers should consider using multiple imputation methods and comparing their results to ensure the robustness of their findings. In addition, while traditional methods, such as scImpute, MAGIC, and WEDGE, show relatively better performance across a wide range of tasks, the performance of DL-based methods is more variable, with no methods showing the best or moderately high performance, and 4 methods, namely scMultiGAN, scGNN, CPARI, and Bubble, showing moderately low performance across the 6 tasks. This suggests that further development and optimization of DL-based imputation methods are needed to achieve consistent performance across diverse downstream analyses. The trade-off between numerical gene expression recovery performance and biological signal preservation is a critical consideration in the design and selection of imputation methods, as methods that excel in one aspect may perform poorly in the other, which can have significant implications for the interpretation of scRNA-seq data and the biological conclusions drawn from it.

Moreover, this study includes limitations that should be acknowledged. One limitation is the selection of imputation methods, which covers a range of recently developed approaches and 2 well-known traditional methods, i.e., scImpute and MAGIC, but does not encompass all existing methods evaluated in previous benchmarking studies. In addition, while we evaluate performance in terms of 26 real and 4 simulated datasets with 10 different protocols, it may be beneficial to further expand the diversity of datasets, including more complex tissues and disease states. In terms of downstream tasks, we focus on 6 core tasks, including numerical gene expression recovery, cell clustering, DE analysis, marker gene analysis, trajectory analysis, and cell type annotation. Future studies could explore additional open problems, such as RNA velocity analysis and batch effect correction, to further understand the impact of imputation on more complex analyses.

5 Methods

5.1 Summary of the Data Imputation Benchmarking Framework

scRNA-seq data cannot be reliably analyzed directly due to the presence of excessive dropout events, which manifest as false zero expression values and distort observed gene expression distributions [2]. To address this limitation, data imputation methods aim to recover latent gene expression signals and improve the robustness of downstream tasks [17–19, 36, 40–54]. In this study, a comprehensive benchmarking framework is designed to systematically evaluate a diverse set of scRNA-seq imputation methods, spanning 8 traditional methods, including model [40, 41], smoothing [42–44], and low-rank matrix-based methods [45–47], and 7 DL-based methods, including diffusion [48, 49], GAN [51, 54], GNN [53], and AE-based methods [50, 52]. These methods differ substantially in their underlying assumptions, including statistical modeling of dropout mechanisms [40, 41], exploiting cell-cell similarity [42–44], enforcing low-rank structures [45–47], or learning latent representations

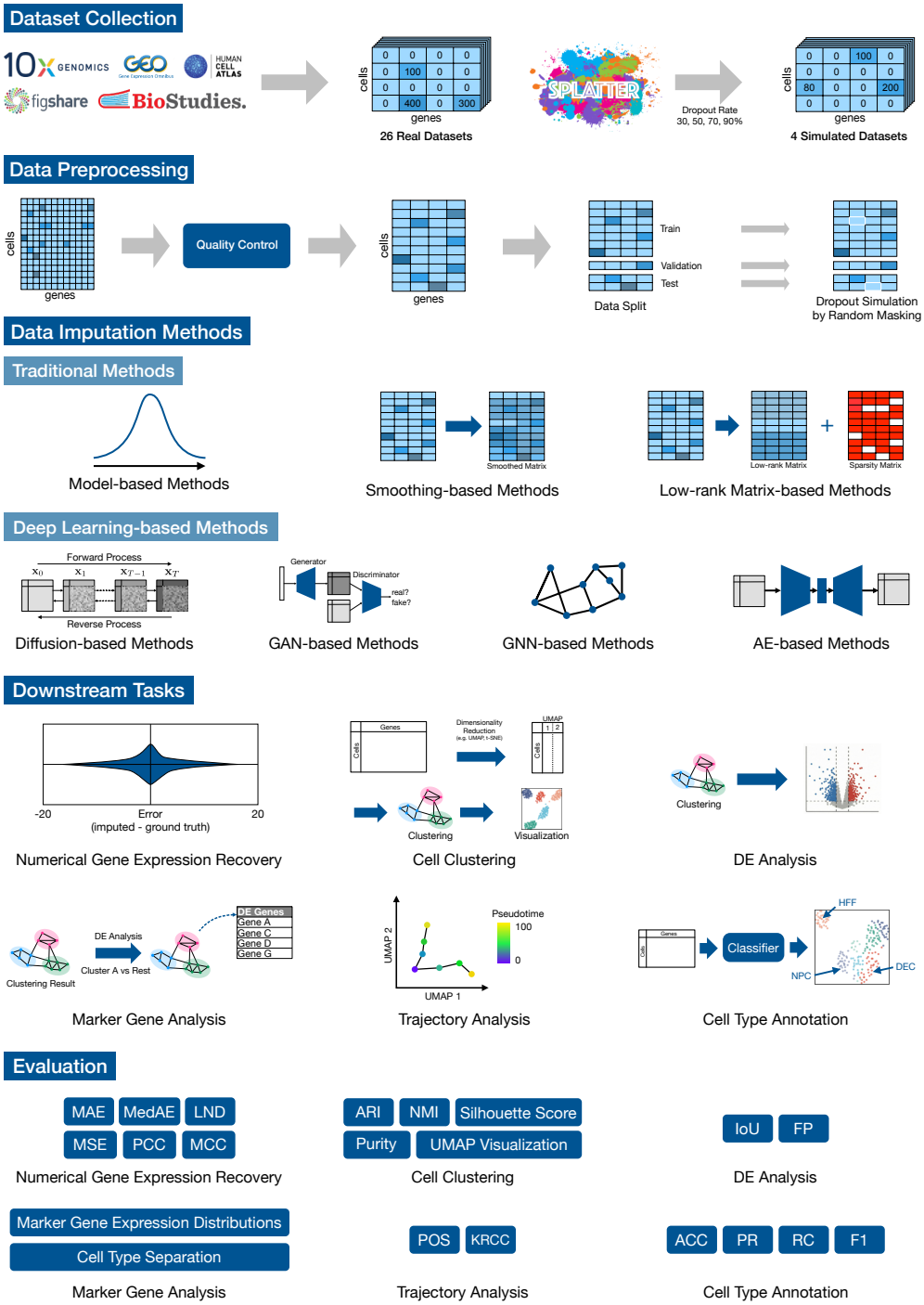

Fig. 12 Data Imputation Benchmarking Framework.

through DNNs [48–54]. Given this diversity, it is essential to assess their effectiveness across heterogeneous datasets and multiple downstream tasks rather than relying on a single evaluation criterion [17–19].

To ensure a fair and unbiased comparison, the benchmarking framework incorporates a carefully curated collection of 26 real and 4 simulated scRNA-seq datasets that vary in size, sparsity level, biological context, and protocol. Each dataset undergoes standardized quality control (QC) and is split into training, validation, and test sets to prevent data leakage during training and evaluation. Dropout events are artificially introduced into each set to establish known ground truth values, enabling objective assessment of numerical gene expression recovery [17–19]. This design allows the framework to isolate the true impact of each imputation method on data quality while avoiding overfitting and biased performance estimates.

Evaluation of data imputation methods is performed from both numerical and functional perspectives using 6 downstream tasks. First, the performance of numerical gene expression recovery is quantified by directly comparing imputed and ground truth values using error-based metrics [17, 18]. Subsequently, biological utility is assessed through a diverse set of downstream tasks, including cell clustering [21, 22, 63], DE analysis [2, 30], marker gene analysis [2, 28], trajectory analysis [2, 23–27], and cell type annotation [2, 29, 63, 64]. These tasks collectively capture core analytical objectives in scRNA-seq studies and provide insight into how data imputation influences biological interpretation [17–19]. Importantly, all downstream tasks are conducted consistently across methods to ensure comparability.

Fig. 12 presents an overview of the proposed benchmarking framework, illustrating the complete pipeline from dataset collection and preprocessing to data imputation, downstream tasks, and performance evaluation. Detailed descriptions of dataset collection and preprocessing are provided in Section 5.2. Data imputation methods, downstream tasks, and evaluation measures are described in Sections 5.3, 5.4 and 5.5, respectively. Together, this framework enables a systematic and reproducible assessment of scRNA-seq data imputation methods, highlighting their strengths, limitations, and suitability for different analytical scenarios.

5.2 Benchmark Datasets

The performance of scRNA-seq imputation methods is heavily influenced by the characteristics of the datasets, such as dataset size, sparsity rate, biological context, and protocol [17–19]. In particular, datasets not only affect the difference between imputed and ground truth expression values but also directly impact downstream tasks [17–19], including cell clustering [21, 22, 63], DE analysis [2, 30], marker gene analysis [2, 28], trajectory analysis[2, 23–27], and cell type annotation [2, 2, 29, 63, 64, 64]. Therefore, the use of representative, well-curated, and high-quality scRNA-seq datasets is fundamental to conducting an unbiased and meaningful benchmark [17, 19].

Table 5: Details of the used scRNA-seq datasets

Dataset	Description	Source	Size (Cells × Genes)	Sparsity Rates (%)	Protocol
ad_case [65]	Human brain cells with Alzheimer's disease	GSE138852	10278 × 13214	94.93	10x Chromium
jurkat	Human Jurkat T-cell leukemia cell line scRNA-seq dataset	10x Genomics[1]	1740 × 13494	81.19	10x Chromium
293t	Human HEK293T embryonic kidney cell line scRNA-seq dataset	10x Genomics[2]	2868 × 16290	81.20	10x Chromium
pbmc4k	Peripheral blood mononuclear cells (PBMCs) from a healthy donor.	10x Genomics[3]	4220 × 16412	92.26	10x Chromium
sc_10x [66, 67]	Single cells from three human lung adenocarcinoma cell lines	GSM3022245	902 × 16468	45.02	10x Chromium
sc_10x_5cl [66, 67]	Single cells from five human lung adenocarcinoma cell lines	GSM3618014	3913 × 11786	63.04	10x Chromium
guo [68]	Mouse early embryonic development scRNA-seq dataset.	GSE150861	18177 × 18538	96.18	10x Chromium
itc [69]	Human innate T cells (ITCs).	GSE124731	2005 × 13260	93.74	10x Chromium
hca_10x_tissue	Bone marrow cells from sample MantonBM6	HCA[4]	6515 × 18203	91.13	10x Chromium
cellmix1 [66, 67]	Pseudo cells from nine cell mixtures	GSE118767	263 × 11798	81.41	CEL-seq2
rnamix_celseq2 [66, 67]	Pseudo cells from RNA mixtures	GSM3305230	340 × 14804	52.07	CEL-seq2
sc_celseq2 [66, 67]	Single cells from three human lung adenocarcinoma cell lines	GSM3336845	273 × 22014	67.82	CEL-seq2
sc_celseq2_5cl_p1 [66, 67]	Single cells from five human lung adenocarcinoma cell lines	GSM3618022	291 × 15564	65.28	CEL-seq2
hcc [70]	T cells from hepatocellular carcinoma (HCC).	GSE98638	5035 × 21576	85.54	SMART-seq2

Continued on next page

[1] https://www.10xgenomics.com/datasets/jurkat-cells-1-standard-1-1-0
[2] https://www.10xgenomics.com/datasets/293-t-cells-1-standard-1-1-0
[3] https://www.10xgenomics.com/datasets/4-k-pbm-cs-from-a-healthy-donor-2-standard-2-1-0
[4] https://explore.data.humancellatlas.org/projects/cc95ff89-2e68-4a08-a234-480eca21ce79

Continued from previous page

Dataset	Description	Source	Size (Cells × Genes)	Sparsity Rates (%)	Protocol
petropoulos [71]	Human preimplantation embryo scRNA-seq dataset across early developmental stages	E-MTAB-3929	1517 × 23583	62.12	SMART-seq2
chu_cell_type [72]	Human embryonic stem cell scRNA-seq dataset with defined cell states.	GSE75748	1018 × 17559	45.24	SMART-seq
chu_time_course [72]	Human embryonic stem cell scRNA-seq dataset following differentiation over time.	GSE75748	758 × 16863	48.45	SMART-seq
chen [73]	Mus musculus scRNA-seq dataset of adult mouse hypothalamus revealing diverse neuronal and non-neuronal cell types	GSE87544	13891 × 17623	92.73	Drop-seq
romanov [74]	Mus musculus brain scRNA-seq dataset profiling hypothalamic neuronal cell types.	GSE74672	3005 × 16979	85.66	Drop-seq
sc_dropseq [66, 67]	Single cells from three human lung adenocarcinoma cell lines	GSM3336849	224 × 15113	62.13	Drop-seq
usokin [75]	Mouse sensory neuron scRNA-seq dataset profiling dorsal root ganglion cell types.	GSE59739	622 × 17777	82.13	STRT-Seq
zeisel [76]	Mouse brain scRNA-seq dataset defining major neuronal and glial cell types.	GSE60361	3005 × 19972	82.15	STRT-Seq
baron [77]	Human pancreas scRNA-seq dataset profiling endocrine and exocrine cell types.	GSM2230757	1918 × 14708	87.03	inDrop
encode_fluidigm_5cl [78]	Single cells from five cell lines	GSE81861	360 × 36092	67.07	Fluidigm C1

Continued on next page

Continued from previous page

Dataset	Description	Source	Size (Cells × Genes)	Sparsity Rates (%)	Protocol
bladder [79]	Mus musculus bladder scRNA-seq dataset from the Mouse Cell Atlas profiling cell types across bladder tissue.	Figshare[5]	1278 × 16387	94.44	Microwell-seq
rnamix_sortseq [66, 67]	Pseudo cells from RNA mixtures	GSM3305231	296 × 15571	60.84	Sort-seq
simulated_1	-	-	2000 × 600	30.79	-
simulated_2	-	-	2000 × 600	50.62	-
simulated_3	-	-	2000 × 600	70.12	-
simulated_4	-	-	2000 × 600	89.61	-

Table 5 presents 30 unique benchmark datasets spanning 10 scRNA-seq data extraction protcols, 12 cell lines, 11 tissues from human and mouse samples, and 6 disease conditions which are collected from 10x Genomics dataset repository [80], Gene Expression Omnibus (GEO) database [81], Figshare [82], HCA [14] and BioStudies [83]. These datasets are selected for representative coverage of the heterogeneity and protocol variations present in real-world scRNA-seq studies.

The curated dataset collection covers a wide range of sizes and sparsity rates to evaluate imputation methods under different number of cells, genes, and dropout conditions. These datasets vary significantly in scale, with number of cells ranging from 224 to 13,891 and number of genes spanning 14,708 to 23,583. This variety helps evaluate if data imputation methods can handle both small samples and large-scale data [17]. Furthermore, the collection includes sparsity rates from 45.24 % to 96.18 %, reflecting realistic dropout levels encountered in scRNA-seq experiments [17]. Conducting benchmarks across these different levels of data density ensures a clear evaluation of how well each method recovers gene expression values in both high and low-quality datasets [17, 19].

Protocols introduce distinct technical noise profiles and dropout patterns that can substantially influence imputation performance [6–8]. This protocol-level heterogeneity is addressed by selecting 26 datasets obtained using 10 different protocols, namely 10x Chromium [8], SMART-seq [84], SMART-seq2 [85], CEL-seq2 [86], Drop-seq [87], inDrop [88], Microwell-seq [79], STRT-seq [89], Sort-seq [90], and Fluidigm C1 [91]. This dataset collection enables a systematic assessment of imputation robustness across platforms with fundamentally different library preparation strategies, read depths, and technical noise profiles [17, 19].

In addition to real datasets, 4 simulated datasets with known ground truth to compare the performance of imputation methods under different dropout rates are used in this benchmarking framework [17]. The simulated datasets are generated using the Splatter [92] package, which allows controlled simulation of scRNA-seq data with known ground truth [17, 92]. Each simulated dataset contains 2,000 cells and 10,000

[5] https://figshare.com/s/865e694ad06d5857db4b

genes, and 5 clusters. To compare the performance of imputation methods under different dropout rates, 4 simulated datasets with varying dropout rates of 30.79, 50.62, 70.12, and 89.61 % are generated, which reflect a range of dropout conditions commonly observed in real scRNA-seq experiments [17].

Furthermore, following Hou et al. [19], corresponding bulk RNA-seq datasets are obtained for 3 real scRNA-seq datasets, namely sc_10x_5cl, encode_fluidigm_5cl, and hca_10x_tissue, to evaluate imputation performance at the cell population level. These bulk RNA-seq datasets are downloaded from the processed data repository[6] provided by Hou et al. [19]. For sc_10x_5cl, bulk RNA-seq data with 10 samples of 5 cell lines, namely HCC827, H1975, H2228, H838, and A549, are originally collected from GSE86337 [19, 93]. For encode_fluidigm_5cl, bulk RNA-seq data with 58 samples of 5 cell lines, namely A549, GM12878, H1-hESC, IMR90, and K562, are originally collected from the Encyclopedia of DNA Elements (ENCODE) [19, 94]. For hca_10x_tissue, bulk RNA-seq data with 49 samples of 13 cell types, namely B cell, CD4 T cell, CD8 T cell, CMP, GMP, HSC, MEP, monocyte, MPP, NK cells, CLP, erythroid, and LMPP, are originally collected from GSE74246 [19, 95].

After dataset collection, QC is applied to each real dataset to remove low-quality cells and genes [17]. Following Cheng et al. [17], cells whose number of expressed genes is larger than the 75th percentile or less than the 25th percentile are filtered out [17]. Similarly, genes which are expressed in more than the 75th percentile or fewer than the 25th percentile of the number of cells are filtered out [17]. With this QC procedure, only high-quality cells and genes are retained for subsequent imputation and downstream tasks [17].

The cells are randomly partitioned into training, validation, and test sets. The training set is used for training the data imputation methods, the validation set is used for early stopping of training iteration of DL-based methods, and the test set is used for evaluating the imputation methods including running downstream tasks. This dataset splitting strategy prevents data leakage between training and evaluation phases, ensuring a fair assessment of imputation performance, whereas previous benchmarking studies [17–19] use the same ground truth data for both training and testing, which can lead to overfitting and biased results. The split ratio is 70 % for training, 10 % for validation, and 20 % for testing.

After QC and data splitting, dropout events are artificially introduced into each split set of real scRNA-seq datasets since it is not possible to distinguish between true biological zeros and technical dropout events in real scRNA-seq datasets [16, 36, 37]. To introduce dropout events, following Cheng et al. [17], 10 % of non-zero expression values in each split dataset are selected and masked as zero expression values [17]. The original non-zero expression values before masking are used as ground truth for evaluation.

[6] https://github.com/Winnie09/imputationBenchmark

5.3 Data Imputation Methods

5.3.1 Traditional Methods

Traditional data imputation methods for scRNA-seq can be broadly categorized into 3 methodological classes: model-based methods, smoothing-based methods, and low-rank matrix-based methods. These categories and the specific methods evaluated in this study are described below.

- ***Model-based Methods:*** Model-based methods explicitly model the occurrence of dropout events and the distribution of gene expression values using parametric statistical models to estimate and impute dropout events in scRNA-seq data [40, 41]. 2 different model-based methods are selected in this study, i.e., scImpute [41] and PbImpute [40]. scImpute [41] models the occurrence of dropout events and the distribution of gene expression values using a mixture distribution, in which dropout events are modeled by a Gamma distribution and true expression values are modeled by a normal distribution [41]. An expectation-maximization (EM) algorithm is then used to estimate the dropout probability of each zero expression value [41, 96]. Cells with similar expression patterns are then identified using non-negative least squares regression, and expression values from these similar cells are used to impute values at inferred dropout positions [41]. In contrast, PbImpute [40] addresses the common problem of over-imputation by utilizing a multi-stage method [40]. The process begins with zero-inflated negative binomial (ZINB) modeling to provide robust dropout identification and initial imputation [40]. To enhance data fidelity, the method incorporates a static repair step that corrects over-imputed values by adjusting outlying nonzero values [40]. Moreover, the method identifies residual dropout events using node2vec [97], which capture complex relationships between cells, and impute residual dropout events dynamically [40]. This multi-stage approach allows PbImpute to accurately identify and impute dropout events while minimizing the risk of over-imputation [40].
- ***Smoothing-based Methods:*** Smoothing-based methods leverage the similarity of expression profiles among neighboring cells to reduce noise and preserve biological signals in scRNA-seq data [42–44], a process typically referred to as smoothing [42–44]. 3 representative smoothing-based methods are selected in this study, i.e., MAGIC [44], scTsI [43], and AcImpute [42]. MAGIC is designed based on the concept that gene expression profiles can be shared among similar cells to recover missing values, and utilizes Markov processes to impute dropout events [44]. MAGIC first calculates the cell-cell Euclidean distance matrix and constructs a cell-cell affinity matrix using a Gaussian kernel [44]. The affinity matrix is then normalized using row normalization to obtain a Markov transition matrix, which represents the transition probabilities between cells [44]. The process of calculating the Markov matrix is repeated multiple times to reduce noise and keep the biological signal [44]. Finally, the imputed gene expression matrix is obtained by multiplying the final Markov transition matrix with the observed gene expression matrix [44]. scTsI is a 2-stage smoothing-based imputation method that first imputes the zero expression values using the information of neighboring cells and genes, and then adjusts the imputed values using ridge regression [43, 98]. In the first stage, scTsI imputes

the zero expression values by calculating an average of the expression values from nearest neighbor cells and nearest neighbor genes [43]. In the second stage, scTsI refines the imputed values by fitting a ridge regression model that predicts the expression value of each gene in each cell based on the expression values of other genes in the same cell and the same gene in other cells [43]. While MAGIC diffuses gene expression values equally across all genes, AcImpute applies different diffusion strengths for highly and lowly expressed genes based on the observation that dropout events are more prevalent in lowly expressed genes [42]. AcImpute first normalizes, selects highly variable genes, and reduces dimensionality using principal component analysis (PCA) [42, 99]. Similar to MAGIC, AcImpute then constructs a cell-cell affinity matrix using k-NN-based adaptive kernel, and obtains a Markov transition matrix through row normalization [42]. In addition, AcImpute calculates the power matrix, a locally averaged diffusion operator, by averaging the normalized matrix over its neighboring cells to capture average gene expression patterns across neighboring cells [42]. Finally, AcImpute combines the Markov transition matrix, the power matrix, and the observed gene expression matrix to create a modified Markov transition matrix [42].

- ***Low-rank Matrix-based Methods:*** Low-rank matrix-based methods are built on the idea that gene expression data often have an underlying simple structure [45–47]. Specifically, these methods assume that the true gene expression matrix can be well-approximated by a matrix with low rank, meaning that the expression patterns of thousands of genes across many cells can actually be explained by a small number of shared biological factors, such as cell types or cell states [45–47]. In practice, scRNA-seq data contain a large number of zero values, some of which are caused by dropout events rather than true absence of gene expression [16]. To address this, the observed gene expression matrix can be formulated as $X_{\text{obs}} = X_{\text{true}} + E \in \mathbb{R}^{n \times m}$, where n is the number of cells, m is the number of genes, X_{true} is the true gene expression matrix to be estimated, and E is a sparse noise matrix that captures dropout events [45–47]. The objective of low-rank matrix-based methods is to estimate X_{true} by using the low-rank structure of the gene expression data while accounting for dropout events represented by E [45–47]. 3 representative low-rank matrix-based methods are selected in this study, i.e., PBLR [45], scLRTC [46], and WEDGE [47]. PBLR explicitly incorporates cell heterogeneity into the imputation process [45]. It first identifies cell subpopulations by constructing multiple cell-cell affinity matrices and applying non-negative matrix factorization followed by hierarchical clustering [45]. This step partitions the global expression matrix into more homogeneous submatrices [45]. For each subpopulation-specific submatrix, PBLR performs bounded low-rank matrix recovery, where dropout values are constrained by gene-specific upper bounds estimated from observed expression levels [45]. This bounded formulation prevents unrealistically large imputations and improves recovery accuracy, especially in heterogeneous datasets [45]. scLRTC generalizes matrix-based approaches by modeling scRNA-seq data as a third-order tensor, constructed using cell-cell similarity information [46]. This tensor representation enables simultaneous modeling of gene-gene and cell-cell correlations. scLRTC applies low-rank tensor completion to recover missing values, effectively denoising

the data while preserving higher-order structural relationships [46]. By leveraging tensor decomposition rather than simple matrix factorization, scLRTC can better capture complex dependencies in scRNA-seq data and improve downstream analyses such as clustering and trajectory inference [46]. WEDGE addresses dropout by introducing a biased low-rank matrix decomposition framework [47]. Unlike standard matrix factorization methods that ignore zero entries, WEDGE assigns different weights to zero and non-zero elements in the objective function [47]. Non-zero entries are fitted closely to preserve observed expression, while zero entries are softly penalized using a tunable bias parameter, reducing the risk of over-imputation [47]. The model is optimized via alternating non-negative least squares, ensuring biologically meaningful imputed values [47]. This weighted strategy allows WEDGE to robustly recover expression patterns in highly sparse scRNA-seq datasets [47].

5.3.2 DL-based Methods

DL-based methods can be broadly categorized into 4 methodological classes: diffusion, GAN, GNN, and AE-based methods. These categories and the specific methods evaluated in this study are described below.

- ***Diffusion-based Methods:*** Diffusion-based methods utilize diffusion models [100, 101] to model the underlying data distribution and impute dropout events in scRNA-seq data [48, 49]. Most diffusion models are built on denoising diffusion probabilistic models (DDPMs) [101] composed of 2 Markov processes, the forward process that gradually adds Gaussian noise to the data over multiple time steps, and the reverse process that learns to recover the original data from the noisy input step by step [101]. Moreover, conditional diffusion models can align the output of the reverse denoising process with the given conditions [101]. In this study, 2 representative diffusion-based methods are selected, i.e., scIDPMs [48] and stDiff [49]. scIDPMs identifies potential dropout sites by leveraging intercellular relationships and trains a conditional DDPM conditioned on the observed gene expression values [48]. To train scIDPMs, the method receives the imputation target matrix as input, where it represents the true values and the positions of dropout events, and the observed gene expression matrix as a condition, where it shows gene expression values of the remaining part, and learns the parameters by adding noise to the imputation target matrix and removing the noise from it [48]. During the inference step of scIDPMs, the method receives the imputation target matrix with random noise as input and the observed gene expression matrix as a condition, and outputs an estimated gene expression matrix corresponding to the imputation target matrix [48]. In contrast, stDiff utilizes a conditional DDPM architecture to impute spatial transcriptomics data by learning gene-gene expression relationships from reference scRNA-seq data, rather than modeling cell-cell relationships [49]. To train stDiff, the method first augments the observed gene expression data by adding noise to enhance robustness against batch effects [49]. The augmented gene expression matrices are input to the forward process, and the method adds Gaussian noise step by step [49]. The matrices with Gaussian noise are passed to the reverse process, and the method learns to reconstruct the noised matrices into true expression

values using the Diffusion Transformer (DiT) [49, 102]. During the inference step of stDiff, the method receives a random noise matrix as input, and outputs an estimated gene expression matrix [49]. stDiff is designed to impute spatial transcriptomics data [49], however, here stDiff is adopted for scRNA-seq data imputation to evaluate the performance of multiple diffusion-based methods as diffusion-based imputation methods for scRNA-seq data are still limited.

- ***GAN-based Methods:*** GAN-based methods utilize GANs to learn the underlying distribution of scRNA-seq data and generate imputed values [51, 54]. GANs are composed of 2 neural networks, a generator and a discriminator, that are trained in an adversarial manner [103]. The generator learns to generate realistic data samples from random noise, while the discriminator learns to distinguish between real and generated samples [103]. Through the training process, GANs can learn complex data distributions and generate high-quality samples [103]. Due to their ability to model complex data distributions, GAN-based imputation methods are being proposed [51, 54]. In this study, 2 representative GAN-based methods are included, i.e., scMultiGAN [51] and scIGANs [54]. scIGANs is designed to apply image-generating GANs to scRNA-seq data [54]. scIGANs first converts scRNA-seq data to grayscale square images, which are the format accepted as input by image-generating GANs, by reshaping gene expression vector of a cell into a grayscale square image [54]. The squared images are fed into a GAN and the model learns parameters by generating fake samples and distinguishing between the true samples and the fake samples [54]. During the inference step of scIGANs, the method generates synthetic grayscale square images from the observed scRNA-seq data, selects k-NN cells of the cell to be imputed, and imputes based on the generated image [54]. While scIGANs simply uses a single GAN to generate synthetic cells, scMultiGAN utilizes 3 GANs to learn the complex patterns of scRNA-seq data and generate high-quality imputed values [51]. scMultiGAN performs scRNA-seq imputation using multiple GANs with a two-stage training strategy [51]. In the first stage of scMultiGAN, 2 GANs are trained to learn the distribution of true expression values and dropout events separately [51]. To precisely impute dropout events, in the second stage, it learns the distribution of true expression values precisely by integrating the true expression values generator trained in the first stage, an additional U-Net [104]-based generator, and a discriminator [51]. Finally, the generator from the second stage is used to impute dropout events [51].
- ***GNN-based Methods:*** GNN-based methods leverage GNNs to model the relationships between cells and impute dropout events [53]. By propagating information through the graph structure, GNNs can aggregate neighborhood-level features and effectively capture both local and global cellular relationships [105]. In this study, scGNN [53] is included as a representative GNN-based method. scGNN is a hypothesis-free GNN-based method and it integrates 3 iterative multi-modal AEs, namely feature AE, graph AE, cluster AE, to model heterogeneous gene expression patterns and aggregate cell-cell relationships [53]. The feature AE receives the regularized gene expression matrix calculated through the left-truncated mixture Gaussian model [106] as input and learns low-dimensional cell representations by minimizing the reconstruction loss between the input and output of the AE [53].

Based on the output of the feature AE, scGNN constructs a cell-cell graph using k-NN and feeds it into the graph AE to aggregate neighborhood-level features and learn enhanced cell representations [53]. The cluster AE receives the reconstructed gene expression matrix from the feature AE and an individual encoder is used for each cell cluster to better capture cluster-specific gene expression patterns, which are identified through clustering on the output of the graph AE [53]. The reconstructed gene expression matrices from an individual encoder of the cluster AE are concatenated, and fed into the feature AE and graph AE in the next iteration [53]. This iterative process continues until convergence, and the final reconstructed gene expression matrix from the feature AE is used as the imputed gene expression matrix [53].

- ***AE-based Methods:*** AE-based methods utilize AE architectures to learn low-dimensional representations of scRNA-seq data and reconstruct imputed values [50, 52]. AEs are encoder-decoder architectures that consist of an encoder that maps the input data to a low-dimensional latent representation, and a decoder that reconstructs the original data from the latent representation [107, 108]. By training the AE to minimize the reconstruction loss, which measures the error between the ground truth and reconstructed data, AEs can learn meaningful representations of the input data [107, 108]. AEs are adapted for scRNA-seq data imputation due to their ability to capture complex gene expression patterns and reconstruct dropout events [50, 52]. In this study, 2 representative AE-based methods are included, i.e., Bubble [52] and CPARI [50]. Bubble utilizes an AE to selectively impute dropout events that are identified through statistical analysis of gene expression patterns within cell subpopulations [52]. Bubble consists of 2 main steps, namely identification of dropout events, and imputation [52]. In the first step, Bubble first reduces the dimensionality of the observed gene expression matrix using PCA [99], divides cells into clusters using k-means clustering, and identifies dropout events through predefined statistical rules, which state that if a gene has a high expression rate and low variation in cells within a cluster, then zero expression levels of the gene in the cluster are more likely to be dropout events [52]. In the second step, Bubble trains an AE with the objective of minimizing the total loss function composed of reconstruction loss of the AE, biological loss, which aims to recover non-zero expression values, and alignment loss, which aims to align the aggregated reconstructed gene expression values to the matched bulk RNA-seq data, to impute the identified dropout events [52]. On the other hand, CPARI combines cell partitioning with absolute and relative imputation strategies to effectively distinguish biological zeros from dropout events [50]. In the first step, CPARI selects highly variable genes, and partitions cells into multiple clusters using fuzzy C-means clustering [50, 109]. Absolute imputation is done for each cell cluster by identifying dropout events from the observed gene expression values by statistical rules, and imputing dropout events using an AE [50]. As absolute imputation alone may not fully identify and impute all dropout events, relative imputation is performed by statistical rules based on gene expression patterns within cell subpopulations [50]. Finally, the outputs of absolute and relative imputation are integrated to create the final imputed gene expression matrix [50].

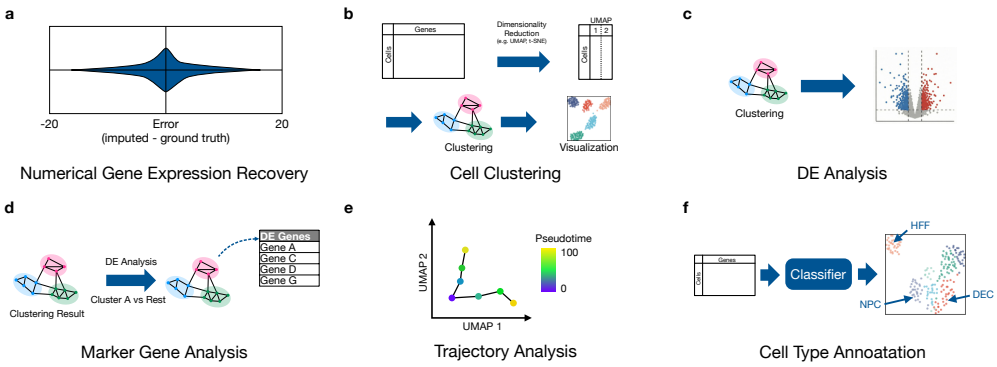

Fig. 13 The overview of downstream tasks used for benchmarking imputation methods. **a** Numerical gene expression recovery, **b** cell clustering, **c** DE analysis, **d** marker gene analysis, **e** trajectory analysis, and **f** cell type annotation.

5.4 Downstream Tasks

The quality of imputed scRNA-seq data directly influences the reliability and performance of computational models in downstream tasks [17–19]. Inaccurate or biased imputation can distort underlying biological signals, leading to misleading conclusions [17–19]. Therefore, a comprehensive evaluation of imputation methods must assess not only numerical recovery of gene expression values but also their impact on biologically meaningful downstream tasks [17–19]. In this section, as illustrated in Fig. 13, we describe 6 distinct downstream tasks used to benchmark imputation methods, namely numerical gene expression recovery, cell clustering, DE analysis, marker gene analysis, trajectory analysis, and cell type annotation.

- ***Numerical Gene Expression Recovery:*** Numerical gene expression recovery can be formulated as a regression task, in which the objective is to predict true gene expression values from sparsely observed scRNA-seq data affected by dropout events [17]. In this setting, the input consists of corrupted expression matrices where zero or near-zero values arise due to technical dropouts, while the target outputs correspond to the original, uncorrupted gene expression values [36]. Ground truth data are obtained from real datasets where artificial dropout is introduced in a controlled manner [17, 19, 36]. Imputation models are trained by learning a mapping from the observed sparse data to the complete expression space, and performance is quantitatively evaluated using regression-based error metrics such as mean squared error (MSE) and MAE computed at the gene, cell, or matrix level [17, 18].
- ***Cell Clustering:*** Cell clustering can be formulated as an unsupervised learning problem, where each cell is treated as an individual sample represented by a high-dimensional gene expression vector [63]. The objective is to group similar cells into clusters based on their expression profiles [63]. Since scRNA-seq data is inherently high-dimensional, with thousands of genes measured per cell, dimensionality reduction techniques, such as PCA [99], t-distributed stochastic neighbor embedding (t-SNE) [110], and UMAP [111], are often applied prior to clustering to reduce noise

and improve computational efficiency [2]. Cell clustering is typically performed as an initial downstream task and serves as a foundation for subsequent downstream tasks, such as marker gene analysis and cell type annotation [2, 63]. Meaningful clusters enable the identification of distinct cell populations and cellular states, whereas inaccurate clustering can lead to misleading biological interpretations [2]. In this benchmark, dimensionality reduction is performed using PCA [99], and subsequently apply the Leiden algorithm [21] for cell clustering. The Leiden algorithm is a graph-based community detection algorithm that operates on a cell-cell similarity graph constructed from the scRNA-seq data and partitions cells into clusters by optimizing modularity, a quality function that measures the density of edges within clusters compared to edges between clusters [21].

- ***DE Analysis:*** DE analysis can be formulated as a feature selection problem, in which the objective is to identify genes that exhibit statistically significant expression differences between conditions, such as disease versus healthy groups or treatment versus control groups [2, 30]. This is typically achieved by testing the null hypothesis that the expression levels of a given gene are identical between the 2 groups [112]. DE analysis enables the identification of genes associated with specific biological processes or disease states and represents a core downstream task for linking gene expression changes to underlying biological phenomena [2]. Accurate identification of DEGs is therefore crucial for understanding molecular mechanisms related to disease progression or treatment response [2]. In this benchmark, 2 common DE analysis methods are used, namely MAST [55], which is a statistical framework using a hurdle model to account for the bimodal distribution of scRNA-seq data [19, 55], and the Wilcoxon rank-sum test [56–58], in order to evaluate the performance and robustness of imputation methods across multiple DE analysis approaches [19].
- ***Marker Gene Analysis:*** Marker gene analysis is an application of DE analysis, and can be formulated as a feature selection problem, in which the objective is to identify genes that best represent each cluster of cells [2, 28]. This task is typically performed in 2 steps. First, DE analysis is conducted to identify genes whose expression levels significantly differ between a given cluster and the remaining clusters [2, 28]. Second, genes are ranked based on LFC or test statistics derived from DE analysis, and the top-ranked genes are selected as marker genes for each cluster [28]. Marker gene analysis plays a crucial role in the interpretation of cell clusters, as marker genes provide insights into the biological functions and identities of distinct cell populations [2, 28]. Accurate identification of marker genes enables reliable interpretation of the biological significance of cell clusters, and a deeper understanding of underlying cellular heterogeneity [2, 28]. In this benchmark, marker gene analysis is performed using the Wilcoxon rank-sum test [56–58]-based DE analysis between each cluster and the remaining clusters.
- ***Trajectory Analysis:*** Trajectory analysis can be formulated as an unsupervised latent-structure inference task, where the objective is to infer continuous cellular progression and lineage relationships from scRNA-seq data [2]. This ordering is commonly represented by pseudotime values assigned to each cell [2]. Pseudotime is a continuous latent variable that captures the relative progression of cells through a

biological process, such as differentiation or development [2, 23, 113]. Pseudotime is typically inferred from cell-cell relationships by measuring distances between cells in the original expression space or in a reduced-dimensional representation [2, 23, 113]. Trajectory analysis is essential for understanding dynamic cellular processes and identifying key regulatory genes involved in these processes [2]. Reliable inference of pseudotime enables the discovery of temporal patterns of gene expression and provides insights into the mechanisms driving cellular transitions [2]. In this benchmark, TSCAN [114] is used to perform trajectory inference. TSCAN first reduces the dimensionality of gene expression data using PCA, then clusters cells in the reduced space, and constructs a minimum spanning tree (MST) connecting cluster centers to represent the trajectory structure [114]. Pseudotime values are subsequently assigned by projecting individual cells onto the nearest edge of the MST [114].

- ***Cell Type Annotation:*** Cell type annotation can be formulated as a supervised multi-class classification problem, in which inputs are gene expression vectors of each cell, and outputs are corresponding cell type labels [2, 63, 64]. The objective of this task is to learn a mapping from gene expression vectors to cell type labels based on known cell type labels [64]. Cell types are predefined at different levels of granularity, such as broad cell types, e.g., T cells, B cells, and monocytes, or fine-grained cell subtypes, e.g., $CD4^+$ T cells, $CD8^+$ T cells, and regulatory T cells [64, 115]. A typical approach to perform cell type annotation is to compare the scRNA-seq data with previously annotated reference datasets using classification models [64]. In this approach, a classifier is trained on a reference dataset to learn the mapping from gene expression vectors to cell type labels, and is subsequently used to predict cell type labels for new scRNA-seq data [64]. This task is essential for the interpretation of scRNA-seq data, as it provides biological context for cells and clusters identified in the data [16, 63, 64]. Robust cell type annotation enables researchers to better understand cellular heterogeneity and the functional roles of different cell types in biological processes [2, 64]. In this benchmark, scGPT [61], a foundation model for scRNA-seq data which supports cell type annotation, and 1D-CNN are used to evaluate the performance of cell type annotation across different imputation methods and scRNA-seq datasets. For scGPT, the pretrained model released by the authors[7], which is trained on 33 million human cells, is used, and for 1D-CNN, the model is trained on the training set of each dataset.

5.5 Evaluation Measures

All 15 imputation methods are evaluated across 6 downstream tasks that capture both numerical accuracy and biological relevance. Since each task serves a different analytical objective, a single metric is insufficient to fully characterize performance. Therefore, task-specific 15 different evaluation measures are utilized to assess reconstruction quality, clustering consistency, statistical agreement, temporal ordering, and classification accuracy. Together, these measures provide a comprehensive and fair comparison of all methods.

[7] https://github.com/bowang-lab/scGPT

- ***Numerical Gene Expression Recovery:*** The evaluation of numerical gene expression recovery can be performed by directly comparing imputed gene expression values with ground truth values or by comparing with corresponding bulk RNA-seq data [17–19]. 4 distinct evaluation measures are used to directly evaluate the numerical gene expression recovery performance of all 15 imputation methods, namely MAE, MedAE, LND, and MSE. Each measure is computed by comparing the imputed gene expression values with the ground truth expression values. MAE is computed as the average of the absolute differences between imputed and ground truth values, which provides a more direct measure of average error [17]. MedAE is computed as the median of the absolute differences between imputed and ground truth values, which provides a similar measure as MAE without outliers [17]. LND is calculated as the log-transformed difference between imputed and ground truth values, which allows assessment of over- or under-imputation [17]. MSE is calculated as the average of the squared differences between imputed and ground truth values, which provides a measure of overall error magnitude [17, 18]. In addition to comparison with ground truth values, comparison with matched bulk RNA-seq data is performed to evaluate the performance of imputation methods in recovering gene expression patterns [19] with 2 distinct evaluation measures, namely PCC and MCC. PCC measures the correlation between pseudo-bulk expression values, calculated by averaging imputed gene expression values across all cells, and bulk RNA-seq expression values [19]. MCC measures the median correlation between imputed gene expression values of individual cells and bulk RNA-seq expression values [19]. Both measures are calculated using Spearman's rank correlation coefficient (SCC) [116].

$$f(x) = \begin{cases} \text{MAE} = \frac{1}{N}\sum_{i=1}^{N} |\hat{y}_i - y_i| \\ \text{MedAE} = \text{median}\{|\hat{y}_i - y_i|\}_{i=1}^{N} \\ \text{LND} = \begin{cases} \log_2(\hat{y}_i - y_i + 1) & \text{if } \hat{y}_i - y_i \geq 0 \\ -\log_2(-\hat{y}_i + y_i + 1) & \text{if } \hat{y}_i - y_i < 0 \end{cases} \\ \text{MSE} = \frac{1}{N}\sum_{i=1}^{N} (\hat{y}_i - y_i)^2 \\ \text{PCC} = \text{SCC}(\hat{y}_{\text{pseudo-bulk}}, y_{\text{bulk}}) \\ \text{MCC} = \text{median}\left\{\text{SCC}(\hat{y}_i, y_{\text{bulk}})\right\}_{i=1}^{N} \end{cases} \tag{1}$$

Here, N is the total number of imputed entries, $\hat{y}_i$ is the imputed expression value of the i-th cell, and y_i is the corresponding ground truth expression value. $\hat{y}_{\text{pseudo-bulk}}$ is the pseudo-bulk expression values calculated by averaging imputed gene expression values across all cells, and y_{bulk} is the matched bulk RNA-seq expression values [19]. SCC is caluculated as $\text{SCC(X, Y)} = 1 - \frac{6\sum_{i=1}^{N} D^2}{N(N^2-1)}$, where N is the total number of genes, and D is the difference between the ranks of the given 2 variables X and Y [19].

- ***Cell Clustering:*** 4 distinct evaluation measures are used to evaluate cell clustering performance, namely ARI [117], NMI, purity, and SC [17–19]. ARI measures the

similarity between 2 clustering results by considering all pairs of cells and counting pairs that are assigned in the same or different clusters in the clustering results of imputed data and ground truth data [17–19]. NMI measures the mutual dependence between the clustering results of imputed data and ground truth data [17]. Purity measures clustering quality by quantifying how homogeneous each predicted cluster is with respect to ground truth labels [17]. SC measures clustering quality by quantifying cohesion within clusters and separation between clusters [17, 18].

$$f(x) = \begin{cases} \text{ARI} = \frac{\sum_{ij}\binom{n_{ij}}{2} - \left[\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}\right] / \binom{n}{2}}{\frac{1}{2}\left[\sum_i \binom{a_i}{2} + \sum_j \binom{b_j}{2}\right] - \left[\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}\right] / \binom{n}{2}} \\ \text{NMI} = \frac{I(X;Y)}{\sqrt{H(X)H(Y)}} \\ \text{Purity} = \frac{1}{n}\sum_{k=1}^{K} \max_j |c_k \cap t_j| \\ \text{SC} = \frac{b(i)-a(i)}{\max\{a(i),b(i)\}} \end{cases} \tag{2}$$

Here, for ARI, n is the total number of cells, i and j are cluster indices in the clustering results of imputed data and ground truth data, respectively, n_{ij} is the number of cells in both cluster i and cluster j, $a_i = \sum_j n_{ij}$, and $b_j = \sum_i n_{ij}$. For NMI, X and Y are the cluster assignments from the clustering results of imputed data and ground truth data, respectively, $I(X;Y)$ is the mutual information between X and Y, and $H(X)$ and $H(Y)$ are the entropies of X and Y, respectively. For purity, n is the total number of cells, K is the number of predicted clusters, c_k is the set of cells in predicted cluster k, and t_j is the set of cells in ground truth cluster j. For SC, $a(i)$ is the average distance between cell i and all other cells in the same cluster, and $b(i)$ is the minimum average distance between cell i and all cells in other clusters.

- ***DE Analysis:*** 2 distinct evaluation measures are used to evaluate DE analysis performance, namely IoU and false positive DEG (FPDEG). IoU measures the overlap between the sets of genes in the 2 different groups, e.g., DEGs identified from imputed scRNA-seq data and from the corresponding bulk RNA-seq data. FPDEG measures the number of genes identified as DEGs that are not true DEGs.

$$f(x) = \begin{cases} \text{IoU} = \frac{|G_\text{A} \cap G_\text{B}|}{|G_\text{A} \cup G_\text{B}|} \\ \text{FPDEG} \end{cases} \tag{3}$$

Here, G_A and G_B are the sets of genes in groups A and B, respectively.

- ***Marker Gene Analysis:*** The evaluation is done qualitatively through visual inspection of marker gene expression distributions and cell type separation. Violin plots are used to compare the distribution of known marker gene expression levels across cell types, which assess whether imputed data retain expected cell-type-specific enrichment patterns. In addition, UMAP visualizations are used to evaluate whether imputed data produce clear separation of distinct cell types in low-dimensional space. Heatmaps of marker gene expression values across cell types further complement this evaluation by illustrating whether imputation methods

recover distinct expression signatures for each cell type. Together, these visualizations assess the degree to which each imputation method preserves the biological signal encoded in established marker genes.

- ***Trajectory Analysis:*** 2 distinct evaluation metrics are used to evaluate trajectory analysis performance, namely POS [18, 114] and KRCC [18, 118]. POS is calculated by summing scores that characterize how well the inferred cell ordering matches the expected ordering based on external information. KRCC is computed to measure the correlation between the inferred pseudotime values and the true cell development labels.

$$f(x) = \begin{cases} \text{POS} = \sum_{i=1}^{n-1} \sum_{j>i}^{n} g(i,j) \\ \text{KRCC} = \frac{4C}{n(n-1)} - 1 \end{cases} \tag{4}$$

  Here, n is the number of cells, and $g(i,j)$ is a score that characterizes how well the order of the i-th and j-th cells in the ordered path matches their expected order based on the external information [18, 114], and C is the number of concordant pairs [18, 118]. See Ji and Ji [114] for a detailed definition of $g(i,j)$.

- ***Cell Type Annotation:*** 4 distinct evaluation measures are used to evaluate cell type annotation performance, namely ACC, PR, RC, and F1. Each measure is computed using a macro-averaging approach to ensure equal weighting for all cells irrespective of their types. ACC is calculated as the average of individual accuracy scores across all cells. For a single cell, accuracy is computed as the ratio of correctly predicted samples to the total samples in that cluster. PR is calculated as the average of the ratio of true positives to total predicted positives for each cell, with single-cell PR computed as true positives divided by predicted positives. RC is the average of the ratio of true positives to total actual positives for each cell, with single-cell RC computed as true positives divided by actual positives. F1 is the harmonic mean of PR and RC, calculated across all cells. For individual cells, the F1 is computed as the harmonic mean of that cell's PR and RC.

$$f(x) = \begin{cases} \text{ACC} = \frac{1}{n} \sum_{i=1}^{n} \frac{\text{TP}_i + \text{TN}_i}{\text{TP}_i + \text{TN}_i + \text{FP}_i + \text{FN}_i} \\ \text{PR} = \frac{1}{n} \sum_{i=1}^{n} \frac{\text{TP}_i}{\text{TP}_i + \text{FP}_i} \\ \text{RC} = \frac{1}{n} \sum_{i=1}^{n} \frac{\text{TP}_i}{\text{TP}_i + \text{FN}_i} \\ \text{F1} = \frac{1}{n} \sum_{i=1}^{n} \frac{2\text{PR}_i\text{RC}_i}{\text{PR}_i + \text{RC}_i} \end{cases} \tag{5}$$

  Here, n is the number of cells, and for each cell i, TP_i, TN_i, FP_i, and FN_i denote the numbers of true positive, true negative, false positive, and false negative cell type annotations compared to ground truth cell type annotations, respectively.

5.6 Experimental Setup

Our benchmarking framework is implemented using Python and R. After collecting datasets, each real dataset is formatted into a `anndata` [119] object, which is a widely used sparse matrix format for scRNA-seq data in Python. Simulated datasets are generated using the `Splatter` [92] package in R. Data imputation methods are implemented based on their original implementations provided by the method authors,

and run with default hyperparameters except for scIDPMs [48] and scMultiGAN [51], whose hyperparameters are adjusted to utilize GPU acceleration. Downstream tasks are performed using the `Scanpy` [58] package in Python, except for DE analysis which is performed using the `MAST` [55] and `limma` [59] package in R, trajectory analysis which is performed using the `TSCAN` [114] pagkage in R, and cell type annotation which is performed using the `scGPT` [61] package in Python and 1D-CNN implemented with `PyTorch` [120] package in Python. The evaluation metrics are calculated on top of the `scikit-learn` [121] package in Python. All visualizations are created using the `ggplot2` [122] package in R.

Supplementary information.

- **Additional file 1**: Supplementary Tables S1–S6. (.xlsx)
  - Table S1: Detailed numerical recovery performance.
  - Table S2: Detailed cell clustering performance.
  - Table S3: Detailed DE enrichment and null DE analysis performance.
  - Table S4: Detailed DE effect size analysis performance.
  - Table S5: Detailed trajectory analysis performance.
  - Table S6: Detailed cell type annotation performance.
- **Additional file 2**: Supplementary Fig. S1. Cell clustering UMAP visualization across all datasets. (.pdf)

Declarations

Ethics approval and consent to participate

Not applicable. This study used only publicly available datasets and did not involve human subjects or animal experiments.

Consent for publication

Not applicable.

Availability of data and materials

All datasets used in this study are publicly available. See Table 5 for more details. The codes are available from the corresponding authors on reasonable request.

Competing interests

The authors declare no competing interests.

Funding

This work was supported in part by JST ASPIRE (Grant No. JPMJAP2403).

Authors' contributions

Y.I.: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Software, Validation, Visualization, and Writing – original draft. A.F.A.: Conceptualization, Data curation, Methodology, Supervision, Validation, Visualization, Writing – original draft, and Writing – review & editing. K.K.: Funding acquisition, Resources, and Writing – review & editing. A.D.: Funding acquisition, Resources, Supervision, and Writing – review & editing. M.N.A.: Conceptualization, Funding acquisition, Supervision, and Writing – review & editing.

Acknowledgements

Not applicable.

References

[1] Tang, F., Barbacioru, C., Wang, Y., Nordman, E., Lee, C., Xu, N., Wang, X., Bodeau, J., Tuch, B.B., Siddiqui, A., Lao, K., Surani, M.A.: mRNA-Seq whole-transcriptome analysis of a single cell. Nature Methods **6**(5), 377–382 (2009) https://doi.org/10.1038/nmeth.1315

[2] Luecken, M.D., Theis, F.J.: Current best practices in single-cell RNA-seq analysis: A tutorial. Molecular Systems Biology **15**(6), 188746 (2019) https://doi.org/10.15252/msb.20188746

[3] Rafi, F.R., Heya, N.R., Hafiz, M.S., Jim, J.R., Kabir, M.M., Mridha, M.F.: A systematic review of single-cell RNA sequencing applications and innovations. Computational Biology and Chemistry **115**, 108362 (2025) https://doi.org/10.1016/j.compbiolchem.2025.108362

[4] Cheng, C., Chen, W., Jin, H., Chen, X.: A Review of Single-Cell RNA-Seq Annotation, Integration, and Cell–Cell Communication. Cells **12**(15), 1970 (2023) https://doi.org/10.3390/cells12151970

[5] Hwang, B., Lee, J.H., Bang, D.: Single-cell RNA sequencing technologies and bioinformatics pipelines. Experimental & Molecular Medicine **50**(8), 1–14 (2018) https://doi.org/10.1038/s12276-018-0071-8

[6] Kolodziejczyk, A.A., Kim, J.K., Svensson, V., Marioni, J.C., Teichmann, S.A.: The Technology and Biology of Single-Cell RNA Sequencing. Molecular Cell **58**(4), 610–620 (2015) https://doi.org/10.1016/j.molcel.2015.04.005

[7] Jovic, D., Liang, X., Zeng, H., Lin, L., Xu, F., Luo, Y.: Single-cell RNA sequencing technologies and applications: A brief overview. Clinical and Translational Medicine **12**(3), 694 (2022) https://doi.org/10.1002/ctm2.694

[8] Zheng, G.X.Y., Terry, J.M., Belgrader, P., Ryvkin, P., Bent, Z.W., Wilson, R., Ziraldo, S.B., Wheeler, T.D., McDermott, G.P., Zhu, J., Gregory, M.T., Shuga,

J., Montesclaros, L., Underwood, J.G., Masquelier, D.A., Nishimura, S.Y., Schnall-Levin, M., Wyatt, P.W., Hindson, C.M., Bharadwaj, R., Wong, A., Ness, K.D., Beppu, L.W., Deeg, H.J., McFarland, C., Loeb, K.R., Valente, W.J., Ericson, N.G., Stevens, E.A., Radich, J.P., Mikkelsen, T.S., Hindson, B.J., Bielas, J.H.: Massively parallel digital transcriptional profiling of single cells. Nature Communications **8**(1), 14049 (2017) https://doi.org/10.1038/ncomms14049

[9] Wen, L., Tang, F.: Single-cell omics sequencing technologies: The long-read generation. Trends in Genetics **42**(1), 46–62 (2026) https://doi.org/10.1016/j.tig.2025.07.012

[10] Hu, T., Chitnis, N., Monos, D., Dinh, A.: Next-generation sequencing technologies: An overview. Human Immunology **82**(11), 801–811 (2021) https://doi.org/10.1016/j.humimm.2021.02.012

[11] Kanton, S., Boyle, M.J., He, Z., Santel, M., Weigert, A., Sanchís-Calleja, F., Guijarro, P., Sidow, L., Fleck, J.S., Han, D., Qian, Z., Heide, M., Huttner, W.B., Khaitovich, P., Pääbo, S., Treutlein, B., Camp, J.G.: Organoid single-cell genomic atlas uncovers human-specific features of brain development. Nature **574**(7778), 418–422 (2019) https://doi.org/10.1038/s41586-019-1654-9

[12] Ramilowski, J.A., Goldberg, T., Harshbarger, J., Kloppmann, E., Lizio, M., Satagopam, V.P., Itoh, M., Kawaji, H., Carninci, P., Rost, B., Forrest, A.R.R.: Correction: Corrigendum: A draft network of ligand-receptor-mediated multicellular signalling in human. Nature Communications **7**(1), 10706 (2016) https://doi.org/10.1038/ncomms10706

[13] Huang, K., Xu, Y., Feng, T., Lan, H., Ling, F., Xiang, H., Liu, Q.: The Advancement and Application of the Single-Cell Transcriptome in Biological and Medical Research. Biology **13**(6), 451 (2024) https://doi.org/10.3390/biology13060451

[14] Regev, A., Teichmann, S.A., Lander, E.S., Amit, I., Benoist, C., Birney, E., Bodenmiller, B., Campbell, P., Carninci, P., Clatworthy, M., Clevers, H., Deplancke, B., Dunham, I., Eberwine, J., Eils, R., Enard, W., Farmer, A., Fugger, L., Göttgens, B., Hacohen, N., Haniffa, M., Hemberg, M., Kim, S., Klenerman, P., Kriegstein, A., Lein, E., Linnarsson, S., Lundberg, E., Lundeberg, J., Majumder, P., Marioni, J.C., Merad, M., Mhlanga, M., Nawijn, M., Netea, M., Nolan, G., Pe'er, D., Phillipakis, A., Ponting, C.P., Quake, S., Reik, W., Rozenblatt-Rosen, O., Sanes, J., Satija, R., Schumacher, T.N., Shalek, A., Shapiro, E., Sharma, P., Shin, J.W., Stegle, O., Stratton, M., Stubbington, M.J.T., Theis, F.J., Uhlen, M., van Oudenaarden, A., Wagner, A., Watt, F., Weissman, J., Wold, B., Xavier, R., Yosef, N., Human Cell Atlas Meeting Participants: The Human Cell Atlas. eLife **6**, 27041 (2017) https://doi.org/10.7554/eLife.27041

[15] The Tabula Sapiens Consortium: The Tabula Sapiens: A multiple-organ, single-cell transcriptomic atlas of humans. Science **376**(6594), 4896 (2022) https://

doi.org/10.1126/science.abl4896

[16] Lähnemann, D., Köster, J., Szczurek, E., McCarthy, D.J., Hicks, S.C., Robinson, M.D., Vallejos, C.A., Campbell, K.R., Beerenwinkel, N., Mahfouz, A., Pinello, L., Skums, P., Stamatakis, A., Attolini, C.S.-O., Aparicio, S., Baaijens, J., Balvert, M., Barbanson, B., Cappuccio, A., Corleone, G., Dutilh, B.E., Florescu, M., Guryev, V., Holmer, R., Jahn, K., Lobo, T.J., Keizer, E.M., Khatri, I., Kielbasa, S.M., Korbel, J.O., Kozlov, A.M., Kuo, T.-H., Lelieveldt, B.P.F., Mandoiu, I.I., Marioni, J.C., Marschall, T., Mölder, F., Niknejad, A., Rączkowska, A., Reinders, M., Ridder, J., Saliba, A.-E., Somarakis, A., Stegle, O., Theis, F.J., Yang, H., Zelikovsky, A., McHardy, A.C., Raphael, B.J., Shah, S.P., Schönhuth, A.: Eleven grand challenges in single-cell data science. Genome Biology **21**(1), 31 (2020) https://doi.org/10.1186/s13059-020-1926-6

[17] Cheng, Y., Ma, X., Yuan, L., Sun, Z., Wang, P.: Evaluating imputation methods for single-cell RNA-seq data. BMC Bioinformatics **24**(1), 302 (2023) https://doi.org/10.1186/s12859-023-05417-7

[18] Dai, C., Jiang, Y., Yin, C., Su, R., Zeng, X., Zou, Q., Nakai, K., Wei, L.: scIMC: A platform for benchmarking comparison and visualization analysis of scRNA-seq data imputation methods. Nucleic Acids Research **50**(9), 4877–4899 (2022) https://doi.org/10.1093/nar/gkac317

[19] Hou, W., Ji, Z., Ji, H., Hicks, S.C.: A systematic evaluation of single-cell RNA-sequencing imputation methods. Genome Biology **21**(1), 218 (2020) https://doi.org/10.1186/s13059-020-02132-x

[20] Saelens, W., Cannoodt, R., Todorov, H., Saeys, Y.: A comparison of single-cell trajectory inference methods. Nature Biotechnology **37**(5), 547–554 (2019) https://doi.org/10.1038/s41587-019-0071-9

[21] Traag, V.A., Waltman, L., van Eck, N.J.: From Louvain to Leiden: Guaranteeing well-connected communities. Scientific Reports **9**(1), 5233 (2019) https://doi.org/10.1038/s41598-019-41695-z

[22] Blondel, V.D., Guillaume, J.-L., Lambiotte, R., Lefebvre, E.: Fast unfolding of communities in large networks. Journal of Statistical Mechanics: Theory and Experiment **2008**(10), 10008 (2008) https://doi.org/10.1088/1742-5468/2008/10/P10008

[23] Haghverdi, L., Büttner, M., Wolf, F.A., Buettner, F., Theis, F.J.: Diffusion pseudotime robustly reconstructs lineage branching. Nature Methods **13**(10), 845–848 (2016) https://doi.org/10.1038/nmeth.3971

[24] Trapnell, C., Cacchiarelli, D., Grimsby, J., Pokharel, P., Li, S., Morse, M., Lennon, N.J., Livak, K.J., Mikkelsen, T.S., Rinn, J.L.: The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering of single cells.

Nature Biotechnology **32**(4), 381–386 (2014) https://doi.org/10.1038/nbt.2859

[25] Bendall, S.C., Davis, K.L., Amir, E.-a.D., Tadmor, M.D., Simonds, E.F., Chen, T.J., Shenfeld, D.K., Nolan, G.P., Pe'er, D.: Single-Cell Trajectory Detection Uncovers Progression and Regulatory Coordination in Human B Cell Development. Cell **157**(3), 714–725 (2014) https://doi.org/10.1016/j.cell.2014.04.005

[26] Street, K., Risso, D., Fletcher, R.B., Das, D., Ngai, J., Yosef, N., Purdom, E., Dudoit, S.: Slingshot: Cell lineage and pseudotime inference for single-cell transcriptomics. BMC Genomics **19**(1), 477 (2018) https://doi.org/10.1186/s12864-018-4772-0

[27] Lotfollahi, M., Wolf, F.A., Theis, F.J.: Generative Modeling and Latent Space Arithmetics Predict Single-Cell Perturbation Response across Cell Types, Studies and Species. bioRxiv (2018). https://doi.org/10.1101/478503

[28] Pullin, J.M., McCarthy, D.J.: A comparison of marker gene selection methods for single-cell RNA sequencing data. Genome Biology **25**(1), 56 (2024) https://doi.org/10.1186/s13059-024-03183-0

[29] Wagner, A., Regev, A., Yosef, N.: Revealing the vectors of cellular identity with single-cell genomics. Nature Biotechnology **34**(11), 1145–1160 (2016) https://doi.org/10.1038/nbt.3711

[30] Scholtens, D., von Heydebreck, A.: Analysis of Differential Gene Expression Studies. In: Gentleman, R., Carey, V.J., Huber, W., Irizarry, R.A., Dudoit, S. (eds.) Bioinformatics and Computational Biology Solutions Using R and Bioconductor, pp. 229–248. Springer, New York, NY (2005). https://doi.org/10.1007/0-387-29362-0_14

[31] Jia, C., Hu, Y., Kelly, D., Kim, J., Li, M., Zhang, N.R.: Accounting for technical noise in differential expression analysis of single-cell RNA sequencing data. Nucleic Acids Research **45**(19), 10978–10988 (2017) https://doi.org/10.1093/nar/gkx754

[32] Andrews, T.S., Hemberg, M.: Identifying cell populations with scRNASeq. Molecular Aspects of Medicine **59**, 114–122 (2018) https://doi.org/10.1016/j.mam.2017.07.002

[33] Marinov, G.K., Williams, B.A., McCue, K., Schroth, G.P., Gertz, J., Myers, R.M., Wold, B.J.: From single-cell to cell-pool transcriptomes: Stochasticity in gene expression and RNA splicing. Genome Research **24**(3), 496–510 (2014) https://doi.org/10.1101/gr.161034.113

[34] Islam, S., Zeisel, A., Joost, S., La Manno, G., Zajac, P., Kasper, M., Lönnerberg,

P., Linnarsson, S.: Quantitative single-cell RNA-seq with unique molecular identifiers. Nature Methods **11**(2), 163–166 (2014) https://doi.org/10.1038/nmeth.2772

[35] Kharchenko, P.V., Silberstein, L., Scadden, D.T.: Bayesian approach to single-cell differential expression analysis. Nature Methods **11**(7), 740–742 (2014) https://doi.org/10.1038/nmeth.2967

[36] Wang, M., Gan, J., Han, C., Guo, Y., Chen, K., Shi, Y.-z., Zhang, B.-g.: Imputation Methods for scRNA Sequencing Data. Applied Sciences **12**(20), 10684 (2022) https://doi.org/10.3390/app122010684

[37] Jiang, R., Sun, T., Song, D., Li, J.J.: Statistics or biology: The zero-inflation controversy about scRNA-seq data. Genome Biology **23**(1), 31 (2022) https://doi.org/10.1186/s13059-022-02601-5

[38] Stegle, O., Teichmann, S.A., Marioni, J.C.: Computational and analytical challenges in single-cell transcriptomics. Nature Reviews Genetics **16**(3), 133–145 (2015) https://doi.org/10.1038/nrg3833

[39] Chen, G., Ning, B., Shi, T.: Single-Cell RNA-Seq Technologies and Related Computational Data Analysis. Frontiers in Genetics **10** (2019) https://doi.org/10.3389/fgene.2019.00317

[40] Zhang, Y., Wang, Y., Liu, X., Feng, X.: PbImpute: Precise Zero Discrimination and Balanced Imputation in Single-Cell RNA Sequencing Data. Journal of Chemical Information and Modeling **65**(5), 2670–2684 (2025) https://doi.org/10.1021/acs.jcim.4c02125

[41] Li, W.V., Li, J.J.: An accurate and robust imputation method scImpute for single-cell RNA-seq data. Nature Communications **9**(1), 997 (2018) https://doi.org/10.1038/s41467-018-03405-7

[42] Zhang, W., Liu, T., Zhang, H., Li, Y.: AcImpute: A constraint-enhancing smooth-based approach for imputing single-cell RNA sequencing data. Bioinformatics **41**(3), 711 (2025) https://doi.org/10.1093/bioinformatics/btae711

[43] Zhang, H., Li, W., Guan, J.: scTsI: An effective two-stage imputation method for single-cell RNA-seq data. Briefings in Bioinformatics **26**(3), 298 (2025) https://doi.org/10.1093/bib/bbaf298

[44] Dijk, D., Sharma, R., Nainys, J., Yim, K., Kathail, P., Carr, A.J., Burdziak, C., Moon, K.R., Chaffer, C.L., Pattabiraman, D., Bierie, B., Mazutis, L., Wolf, G., Krishnaswamy, S., Pe'er, D.: Recovering Gene Interactions from Single-Cell Data Using Data Diffusion. Cell **174**(3), 716–72927 (2018) https://doi.org/10.1016/j.cell.2018.05.061

[45] Zhang, L., Zhang, S.: Imputing single-cell RNA-seq data by considering cell heterogeneity and prior expression of dropouts. Journal of Molecular Cell Biology **13**(1), 29–40 (2021) https://doi.org/10.1093/jmcb/mjaa052

[46] Pan, X., Li, Z., Qin, S., Yu, M., Hu, H.: ScLRTC: Imputation for single-cell RNA-seq data via low-rank tensor completion. BMC Genomics **22**(1), 860 (2021) https://doi.org/10.1186/s12864-021-08101-3

[47] Hu, Y., Li, B., Zhang, W., Liu, N., Cai, P., Chen, F., Qu, K.: WEDGE: Imputation of gene expression values from single-cell RNA-seq datasets using biased matrix decomposition. Briefings in Bioinformatics **22**(5), 085 (2021) https://doi.org/10.1093/bib/bbab085

[48] Zhang, Z., Liu, L.: scIDPMs: Single-Cell RNA-Seq Imputation Using Diffusion Probabilistic Models. IEEE Journal of Biomedical and Health Informatics **29**(4), 3057–3068 (2025) https://doi.org/10.1109/JBHI.2024.3430554

[49] Li, K., Li, J., Tao, Y., Wang, F.: stDiff: A diffusion model for imputing spatial transcriptomics through single-cell transcriptomics. Briefings in Bioinformatics **25**(3), 171 (2024) https://doi.org/10.1093/bib/bbae171

[50] Zhang, Y., Wang, Y., Liu, X., Feng, X.: CPARI: A novel approach combining cell partitioning with absolute and relative imputation to address dropout in single-cell RNA-seq data. Briefings in Bioinformatics **26**(1), 668 (2025) https://doi.org/10.1093/bib/bbae668

[51] Wang, T., Zhao, H., Xu, Y., Wang, Y., Shang, X., Peng, J., Xiao, B.: scMultiGAN: Cell-specific imputation for single-cell transcriptomes with multiple deep generative adversarial networks. Briefings in Bioinformatics **24**(6), 384 (2023) https://doi.org/10.1093/bib/bbad384

[52] Chen, S., Yan, X., Zheng, R., Li, M.: Bubble: A fast single-cell RNA-seq imputation using an autoencoder constrained by bulk RNA-seq data. Briefings in Bioinformatics **24**(1), 580 (2023) https://doi.org/10.1093/bib/bbac580

[53] Wang, J., Ma, A., Chang, Y., Gong, J., Jiang, Y., Qi, R., Wang, C., Fu, H., Ma, Q., Xu, D.: scGNN is a novel graph neural network framework for single-cell RNA-Seq analyses. Nature Communications **12**(1), 1882 (2021) https://doi.org/10.1038/s41467-021-22197-x

[54] Xu, Y., Zhang, Z., You, L., Liu, J., Fan, Z., Zhou, X.: scIGANs: Single-cell RNA-seq imputation using generative adversarial networks. Nucleic Acids Research **48**(15), 85 (2020) https://doi.org/10.1093/nar/gkaa506

[55] Finak, G., McDavid, A., Yajima, M., Deng, J., Gersuk, V., Shalek, A.K., Slichter, C.K., Miller, H.W., McElrath, M.J., Prlic, M., Linsley, P.S., Gottardo, R.: MAST: A flexible statistical framework for assessing transcriptional changes

and characterizing heterogeneity in single-cell RNA sequencing data. Genome Biology **16**(1), 278 (2015) https://doi.org/10.1186/s13059-015-0844-5

[56] Mann, H.B., Whitney, D.R.: On a Test of Whether one of Two Random Variables is Stochastically Larger than the Other. The Annals of Mathematical Statistics **18**(1), 50–60 (1947) 2236101

[57] Wilcoxon, F.: Individual Comparisons by Ranking Methods. Biometrics Bulletin **1**(6), 80–83 (1945) https://doi.org/10.2307/3001968 3001968

[58] Wolf, F.A., Angerer, P., Theis, F.J.: SCANPY: Large-scale single-cell gene expression data analysis. Genome Biology **19**(1), 15 (2018) https://doi.org/10.1186/s13059-017-1382-0

[59] Ritchie, M.E., Phipson, B., Wu, D., Hu, Y., Law, C.W., Shi, W., Smyth, G.K.: limma powers differential expression analyses for RNA-sequencing and microarray studies. Nucleic acids research **43**(7), 47–47 (2015)

[60] Shaffer, A.L., Rosenwald, A., Hurt, E.M., Giltnane, J.M., Lam, L.T., Pickeral, O.K., Staudt, L.M.: Signatures of the Immune Response. Immunity **15**(3), 375–385 (2001) https://doi.org/10.1016/S1074-7613(01)00194-7

[61] Ding, S., Li, J., Luo, R., Cui, H., Wang, B., Chen, R.: scGPT: End-to-end protocol for fine-tuned retinal cell type annotation. Nature Protocols (2025) https://doi.org/10.1038/s41596-025-01220-1

[62] Ziegenhain, C., Vieth, B., Parekh, S., Reinius, B., Guillaumet-Adkins, A., Smets, M., Leonhardt, H., Heyn, H., Hellmann, I., Enard, W.: Comparative Analysis of Single-Cell RNA Sequencing Methods. Molecular Cell **65**(4), 631–6434 (2017) https://doi.org/10.1016/j.molcel.2017.01.023

[63] Kiselev, V.Y., Andrews, T.S., Hemberg, M.: Challenges in unsupervised clustering of single-cell RNA-seq data. Nature Reviews Genetics **20**(5), 273–282 (2019) https://doi.org/10.1038/s41576-018-0088-9

[64] Pasquini, G., Rojo Arias, J.E., Schäfer, P., Busskamp, V.: Automated methods for cell type annotation on scRNA-seq data. Computational and Structural Biotechnology Journal **19**, 961–969 (2021) https://doi.org/10.1016/j.csbj.2021.01.015

[65] Grubman, A., Chew, G., Ouyang, J.F., Sun, G., Choo, X.Y., McLean, C., Simmons, R.K., Buckberry, S., Vargas-Landin, D.B., Poppe, D., Pflueger, J., Lister, R., Rackham, O.J.L., Petretto, E., Polo, J.M.: A single-cell atlas of entorhinal cortex from individuals with Alzheimer's disease reveals cell-type-specific gene expression regulation. Nature Neuroscience **22**(12), 2087–2097 (2019) https://doi.org/10.1038/s41593-019-0539-4

[66] Tian, L., Dong, X., Freytag, S., Lê Cao, K.-A., Su, S., JalalAbadi, A., Amann-Zalcenstein, D., Weber, T.S., Seidi, A., Jabbari, J.S., Naik, S.H., Ritchie, M.E.: Benchmarking single cell RNA-sequencing analysis pipelines using mixture control experiments. Nature Methods **16**(6), 479–487 (2019) https://doi.org/10.1038/s41592-019-0425-8

[67] Tian, L., Su, S., Dong, X., Amann-Zalcenstein, D., Biben, C., Seidi, A., Hilton, D.J., Naik, S.H., Ritchie, M.E.: scPipe: A flexible R/Bioconductor preprocessing pipeline for single-cell RNA-sequencing data. PLOS Computational Biology **14**(8), 1006361 (2018) https://doi.org/10.1371/journal.pcbi.1006361

[68] Guo, C., Li, B., Ma, H., Wang, X., Cai, P., Yu, Q., Zhu, L., Jin, L., Jiang, C., Fang, J., Liu, Q., Zong, D., Zhang, W., Lu, Y., Li, K., Gao, X., Fu, B., Liu, L., Ma, X., Weng, J., Wei, H., Jin, T., Lin, J., Qu, K.: Single-cell analysis of two severe COVID-19 patients reveals a monocyte-associated and tocilizumab-responding cytokine storm. Nature Communications **11**(1), 3924 (2020) https://doi.org/10.1038/s41467-020-17834-w

[69] Gutierrez-Arcelus, M., Teslovich, N., Mola, A.R., Polidoro, R.B., Nathan, A., Kim, H., Hannes, S., Slowikowski, K., Watts, G.F.M., Korsunsky, I., Brenner, M.B., Raychaudhuri, S., Brennan, P.J.: Lymphocyte innateness defined by transcriptional states reflects a balance between proliferation and effector functions. Nature Communications **10**(1), 687 (2019) https://doi.org/10.1038/s41467-019-08604-4

[70] Zheng, C., Zheng, L., Yoo, J.-K., Guo, H., Zhang, Y., Guo, X., Kang, B., Hu, R., Huang, J.Y., Zhang, Q., Liu, Z., Dong, M., Hu, X., Ouyang, W., Peng, J., Zhang, Z.: Landscape of Infiltrating T Cells in Liver Cancer Revealed by Single-Cell Sequencing. Cell **169**(7), 1342–135616 (2017) https://doi.org/10.1016/j.cell.2017.05.035

[71] Petropoulos, S., Edsgärd, D., Reinius, B., Deng, Q., Panula, S.P., Codeluppi, S., Plaza Reyes, A., Linnarsson, S., Sandberg, R., Lanner, F.: Single-Cell RNA-Seq Reveals Lineage and X Chromosome Dynamics in Human Preimplantation Embryos. Cell **165**(4), 1012–1026 (2016) https://doi.org/10.1016/j.cell.2016.03.023

[72] Chu, L.-F., Leng, N., Zhang, J., Hou, Z., Mamott, D., Vereide, D.T., Choi, J., Kendziorski, C., Stewart, R., Thomson, J.A.: Single-cell RNA-seq reveals novel regulators of human embryonic stem cell differentiation to definitive endoderm. Genome Biology **17**(1), 173 (2016) https://doi.org/10.1186/s13059-016-1033-x

[73] Chen, R., Wu, X., Jiang, L., Zhang, Y.: Single-Cell RNA-Seq Reveals Hypothalamic Cell Diversity. Cell Reports **18**(13), 3227–3241 (2017) https://doi.org/10.1016/j.celrep.2017.03.004

[74] Romanov, R.A., Zeisel, A., Bakker, J., Girach, F., Hellysaz, A., Tomer, R.,

Alpár, A., Mulder, J., Clotman, F., Keimpema, E., Hsueh, B., Crow, A.K., Martens, H., Schwindling, C., Calvigioni, D., Bains, J.S., Máté, Z., Szabó, G., Yanagawa, Y., Zhang, M.-D., Rendeiro, A., Farlik, M., Uhlén, M., Wulff, P., Bock, C., Broberger, C., Deisseroth, K., Hökfelt, T., Linnarsson, S., Horvath, T.L., Harkany, T.: Molecular interrogation of hypothalamic organization reveals distinct dopamine neuronal subtypes. Nature Neuroscience **20**(2), 176–188 (2017) https://doi.org/10.1038/nn.4462

[75] Usoskin, D., Furlan, A., Islam, S., Abdo, H., Lönnerberg, P., Lou, D., Hjerling-Leffler, J., Haeggström, J., Kharchenko, O., Kharchenko, P.V., Linnarsson, S., Ernfors, P.: Unbiased classification of sensory neuron types by large-scale single-cell RNA sequencing. Nature Neuroscience **18**(1), 145–153 (2015) https://doi.org/10.1038/nn.3881

[76] Zeisel, A., Muñoz-Manchado, A.B., Codeluppi, S., Lönnerberg, P., La Manno, G., Juréus, A., Marques, S., Munguba, H., He, L., Betsholtz, C., Rolny, C., Castelo-Branco, G., Hjerling-Leffler, J., Linnarsson, S.: Cell types in the mouse cortex and hippocampus revealed by single-cell RNA-seq. Science **347**(6226), 1138–1142 (2015) https://doi.org/10.1126/science.aaa1934

[77] Baron, M., Veres, A., Wolock, S.L., Faust, A.L., Gaujoux, R., Vetere, A., Ryu, J.H., Wagner, B.K., Shen-Orr, S.S., Klein, A.M., Melton, D.A., Yanai, I.: A Single-Cell Transcriptomic Map of the Human and Mouse Pancreas Reveals Inter- and Intra-cell Population Structure. Cell Systems **3**(4), 346–3604 (2016) https://doi.org/10.1016/j.cels.2016.08.011

[78] Li, H., Courtois, E.T., Sengupta, D., Tan, Y., Chen, K.H., Goh, J.J.L., Kong, S.L., Chua, C., Hon, L.K., Tan, W.S., Wong, M., Choi, P.J., Wee, L.J.K., Hillmer, A.M., Tan, I.B., Robson, P., Prabhakar, S.: Reference component analysis of single-cell transcriptomes elucidates cellular heterogeneity in human colorectal tumors. Nature Genetics **49**(5), 708–718 (2017) https://doi.org/10.1038/ng.3818

[79] Han, X., Wang, R., Zhou, Y., Fei, L., Sun, H., Lai, S., Saadatpour, A., Zhou, Z., Chen, H., Ye, F., Huang, D., Xu, Y., Huang, W., Jiang, M., Jiang, X., Mao, J., Chen, Y., Lu, C., Xie, J., Fang, Q., Wang, Y., Yue, R., Li, T., Huang, H., Orkin, S.H., Yuan, G.-C., Chen, M., Guo, G.: Mapping the Mouse Cell Atlas by Microwell-Seq. Cell **172**(5), 1091–110717 (2018) https://doi.org/10.1016/j.cell.2018.02.001

[80] 10x Genomics: 10x Genomics Dataset Repository. https://www.10xgenomics.com/datasets

[81] Edgar, R., Domrachev, M., Lash, A.E.: Gene Expression Omnibus: NCBI gene expression and hybridization array data repository. Nucleic Acids Research **30**(1), 207–210 (2002) https://doi.org/10.1093/nar/30.1.207

[82] Figshare: Figshare. https://figshare.com/

[83] Sarkans, U., Gostev, M., Athar, A., Behrangi, E., Melnichuk, O., Ali, A., Minguet, J., Rada, J.C., Snow, C., Tikhonov, A., Brazma, A., McEntyre, J.: The BioStudies database—one stop shop for all data supporting a life sciences study. Nucleic Acids Research **46**(D1), 1266–1270 (2018) https://doi.org/10.1093/nar/gkx965

[84] Ramsköld, D., Luo, S., Wang, Y.-C., Li, R., Deng, Q., Faridani, O.R., Daniels, G.A., Khrebtukova, I., Loring, J.F., Laurent, L.C., Schroth, G.P., Sandberg, R.: Full-length mRNA-Seq from single-cell levels of RNA and individual circulating tumor cells. Nature Biotechnology **30**(8), 777–782 (2012) https://doi.org/10.1038/nbt.2282

[85] Picelli, S., Faridani, O.R., Björklund, Å.K., Winberg, G., Sagasser, S., Sandberg, R.: Full-length RNA-seq from single cells using Smart-seq2. Nature Protocols **9**(1), 171–181 (2014) https://doi.org/10.1038/nprot.2014.006

[86] Hashimshony, T., Senderovich, N., Avital, G., Klochendler, A., de Leeuw, Y., Anavy, L., Gennert, D., Li, S., Livak, K.J., Rozenblatt-Rosen, O., Dor, Y., Regev, A., Yanai, I.: CEL-Seq2: Sensitive highly-multiplexed single-cell RNA-Seq. Genome Biology **17**(1), 77 (2016) https://doi.org/10.1186/s13059-016-0938-8

[87] Macosko, E.Z., Basu, A., Satija, R., Nemesh, J., Shekhar, K., Goldman, M., Tirosh, I., Bialas, A.R., Kamitaki, N., Martersteck, E.M., Trombetta, J.J., Weitz, D.A., Sanes, J.R., Shalek, A.K., Regev, A., McCarroll, S.A.: Highly Parallel Genome-wide Expression Profiling of Individual Cells Using Nanoliter Droplets. Cell **161**(5), 1202–1214 (2015) https://doi.org/10.1016/j.cell.2015.05.002

[88] Klein, A.M., Mazutis, L., Akartuna, I., Tallapragada, N., Veres, A., Li, V., Peshkin, L., Weitz, D.A., Kirschner, M.W.: Droplet Barcoding for Single-Cell Transcriptomics Applied to Embryonic Stem Cells. Cell **161**(5), 1187–1201 (2015) https://doi.org/10.1016/j.cell.2015.04.044

[89] Islam, S., Kjällquist, U., Moliner, A., Zajac, P., Fan, J.-B., Lönnerberg, P., Linnarsson, S.: Highly multiplexed and strand-specific single-cell RNA 5′ end sequencing. Nature Protocols **7**(5), 813–828 (2012) https://doi.org/10.1038/nprot.2012.022

[90] Muraro, M.J., Dharmadhikari, G., Grün, D., Groen, N., Dielen, T., Jansen, E., van Gurp, L., Engelse, M.A., Carlotti, F., de Koning, E.J.P., van Oudenaarden, A.: A Single-Cell Transcriptome Atlas of the Human Pancreas. Cell Systems **3**(4), 385–3943 (2016) https://doi.org/10.1016/j.cels.2016.09.002

[91] Xin, Y., Kim, J., Ni, M., Wei, Y., Okamoto, H., Lee, J., Adler, C., Cavino,

K., Murphy, A.J., Yancopoulos, G.D., Lin, H.C., Gromada, J.: Use of the Fluidigm C1 platform for RNA sequencing of single mouse pancreatic islet cells. Proceedings of the National Academy of Sciences **113**(12), 3293–3298 (2016) https://doi.org/10.1073/pnas.1602306113

[92] Zappia, L., Phipson, B., Oshlack, A.: Splatter: Simulation of single-cell RNA sequencing data. Genome Biology **18**(1), 174 (2017) https://doi.org/10.1186/s13059-017-1305-0

[93] Holik, A.Z., Law, C.W., Liu, R., Wang, Z., Wang, W., Ahn, J., Asselin-Labat, M.-L., Smyth, G.K., Ritchie, M.E.: Rna-seq mixology: designing realistic control experiments to compare protocols and analysis methods. Nucleic acids research **45**(5), 30–30 (2017)

[94] The ENCODE Project Consortium: The ENCODE (ENCyclopedia Of DNA Elements) Project. Science **306**(5696), 636–640 (2004) https://doi.org/10.1126/science.1105136

[95] Corces, M.R., Buenrostro, J.D., Wu, B., Greenside, P.G., Chan, S.M., Koenig, J.L., Snyder, M.P., Pritchard, J.K., Kundaje, A., Greenleaf, W.J., *et al.*: Lineage-specific and single-cell chromatin accessibility charts human hematopoiesis and leukemia evolution. Nature genetics **48**(10), 1193–1203 (2016)

[96] Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum Likelihood from Incomplete Data Via the EM Algorithm. Journal of the Royal Statistical Society: Series B (Methodological) **39**(1), 1–22 (1977) https://doi.org/10.1111/j.2517-6161.1977.tb01600.x

[97] Grover, A., Leskovec, J.: Node2vec: Scalable Feature Learning for Networks. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. KDD '16, pp. 855–864. Association for Computing Machinery, New York, NY, USA (2016). https://doi.org/10.1145/2939672.2939754

[98] McDonald, G.C.: Ridge regression. WIREs Computational Statistics **1**(1), 93–100 (2009) https://doi.org/10.1002/wics.14

[99] Pearson, K.: LIII. On lines and planes of closest fit to systems of points in space. The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science **2**(11), 559–572 (1901)

[100] Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., Ganguli, S.: Deep Unsupervised Learning using Nonequilibrium Thermodynamics. In: Proceedings of the 32nd International Conference on Machine Learning, pp. 2256–2265. PMLR, Lille, France (2015)

[101] Ho, J., Jain, A., Abbeel, P.: Denoising Diffusion Probabilistic Models. In: Advances in Neural Information Processing Systems, vol. 33, pp. 6840–6851. Curran Associates, Inc., Virtual (2020)

[102] Peebles, W., Xie, S.: Scalable Diffusion Models with Transformers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 4195–4205 (2023)

[103] Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative Adversarial Nets. In: Advances in Neural Information Processing Systems, vol. 27. Curran Associates, Inc., Montréal Canada (2014)

[104] Ronneberger, O., Fischer, P., Brox, T.: U-Net: Convolutional Networks for Biomedical Image Segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015, pp. 234–241. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24574-4_28

[105] Scarselli, F., Gori, M., Tsoi, A.C., Hagenbuchner, M., Monfardini, G.: The Graph Neural Network Model. IEEE Transactions on Neural Networks **20**(1), 61–80 (2009) https://doi.org/10.1109/TNN.2008.2005605

[106] Wan, C., Chang, W., Zhang, Y., Shah, F., Lu, X., Zang, Y., Zhang, A., Cao, S., Fishel, M.L., Ma, Q., Zhang, C.: LTMG: A novel statistical modeling of transcriptional expression states in single-cell RNA-Seq data. Nucleic Acids Research **47**(18), 111 (2019) https://doi.org/10.1093/nar/gkz655

[107] Hinton, G.E., Salakhutdinov, R.R.: Reducing the Dimensionality of Data with Neural Networks. Science **313**(5786), 504–507 (2006) https://doi.org/10.1126/science.1127647

[108] Kingma, D.P., Welling, M.: Auto-Encoding Variational Bayes. arXiv (2022). https://doi.org/10.48550/arXiv.1312.6114

[109] Dunn, J.C.: A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters. Journal of Cybernetics **3**(3), 32–57 (1973) https://doi.org/10.1080/01969727308546046

[110] Maaten, L., Hinton, G.: Visualizing Data using t-SNE. Journal of Machine Learning Research **9**(86), 2579–2605 (2008)

[111] McInnes, L., Healy, J., Melville, J.: UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. arXiv (2020). https://doi.org/10.48550/arXiv.1802.03426

[112] Love, M.I., Huber, W., Anders, S.: Moderated estimation of fold change and

dispersion for RNA-seq data with DESeq2. Genome Biology **15**(12), 550 (2014) https://doi.org/10.1186/s13059-014-0550-8

[113] Wolf, F.A., Hamey, F.K., Plass, M., Solana, J., Dahlin, J.S., Göttgens, B., Rajewsky, N., Simon, L., Theis, F.J.: PAGA: Graph abstraction reconciles clustering with trajectory inference through a topology preserving map of single cells. Genome Biology **20**(1), 59 (2019) https://doi.org/10.1186/s13059-019-1663-x

[114] Ji, Z., Ji, H.: TSCAN: Pseudo-time reconstruction and evaluation in single-cell RNA-seq analysis. Nucleic Acids Research **44**(13), 117 (2016) https://doi.org/10.1093/nar/gkw430

[115] Domínguez Conde, C., Xu, C., Jarvis, L.B., Rainbow, D.B., Wells, S.B., Gomes, T., Howlett, S.K., Suchanek, O., Polanski, K., King, H.W., Mamanova, L., Huang, N., Szabo, P.A., Richardson, L., Bolt, L., Fasouli, E.S., Mahbubani, K.T., Prete, M., Tuck, L., Richoz, N., Tuong, Z.K., Campos, L., Mousa, H.S., Needham, E.J., Pritchard, S., Li, T., Elmentaite, R., Park, J., Rahmani, E., Chen, D., Menon, D.K., Bayraktar, O.A., James, L.K., Meyer, K.B., Yosef, N., Clatworthy, M.R., Sims, P.A., Farber, D.L., Saeb-Parsy, K., Jones, J.L., Teichmann, S.A.: Cross-tissue immune cell analysis reveals tissue-specific features in humans. Science **376**(6594), 5197 (2022) https://doi.org/10.1126/science.abl5197

[116] Spearman, C.: The Proof and Measurement of Association between Two Things. The American Journal of Psychology **15**(1), 72–101 (1904) https://doi.org/10.2307/1412159 1412159

[117] Hubert, L., Arabie, P.: Comparing partitions. Journal of Classification **2**(1), 193–218 (1985) https://doi.org/10.1007/BF01908075

[118] Kendall, M.G.: A New Measure of Rank Correlation. Biometrika **30**(1-2), 81–93 (1938) https://doi.org/10.1093/biomet/30.1-2.81

[119] Virshup, I., Rybakov, S., Theis, F.J., Angerer, P., Wolf, F.A.: Anndata: Access and store annotated data matrices. Journal of Open Source Software **9**(101), 4371 (2024) https://doi.org/10.21105/joss.04371

[120] Ansel, J., Yang, E., He, H., Gimelshein, N., Jain, A., Voznesensky, M., Bao, B., Bell, P., Berard, D., Burovski, E., Chauhan, G., Chourdia, A., Constable, W., Desmaison, A., DeVito, Z., Ellison, E., Feng, W., Gong, J., Gschwind, M., Hirsh, B., Huang, S., Kalambarkar, K., Kirsch, L., Lazos, M., Lezcano, M., Liang, Y., Liang, J., Lu, Y., Luk, C.K., Maher, B., Pan, Y., Puhrsch, C., Reso, M., Saroufim, M., Siraichi, M.Y., Suk, H., Zhang, S., Suo, M., Tillet, P., Zhao, X., Wang, E., Zhou, K., Zou, R., Wang, X., Mathews, A., Wen, W., Chanan, G., Wu, P., Chintala, S.: PyTorch 2: Faster Machine Learning Through Dynamic Python Bytecode Transformation and Graph Compilation. In: Proceedings of

the 29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2. ASPLOS '24, vol. 2, pp. 929–947. Association for Computing Machinery, New York, NY, USA (2024). https://doi.org/10.1145/3620665.3640366

[121] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, É.: Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research **12**(85), 2825–2830 (2011)

[122] Ginestet, C.: Ggplot2: Elegant Graphics for Data Analysis. Journal of the Royal Statistical Society Series A: Statistics in Society **174**(1), 245–246 (2011) https://doi.org/10.1111/j.1467-985X.2010.00676_9.x