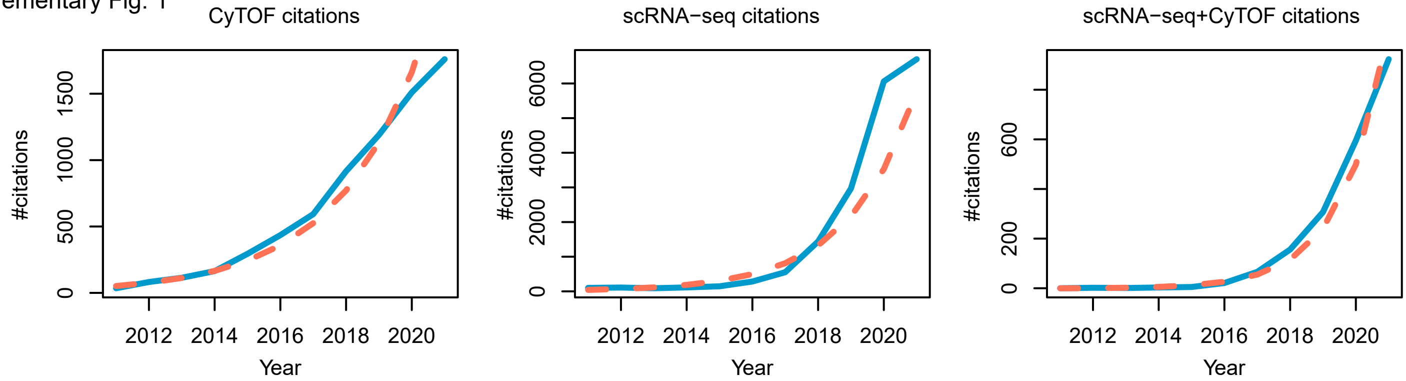
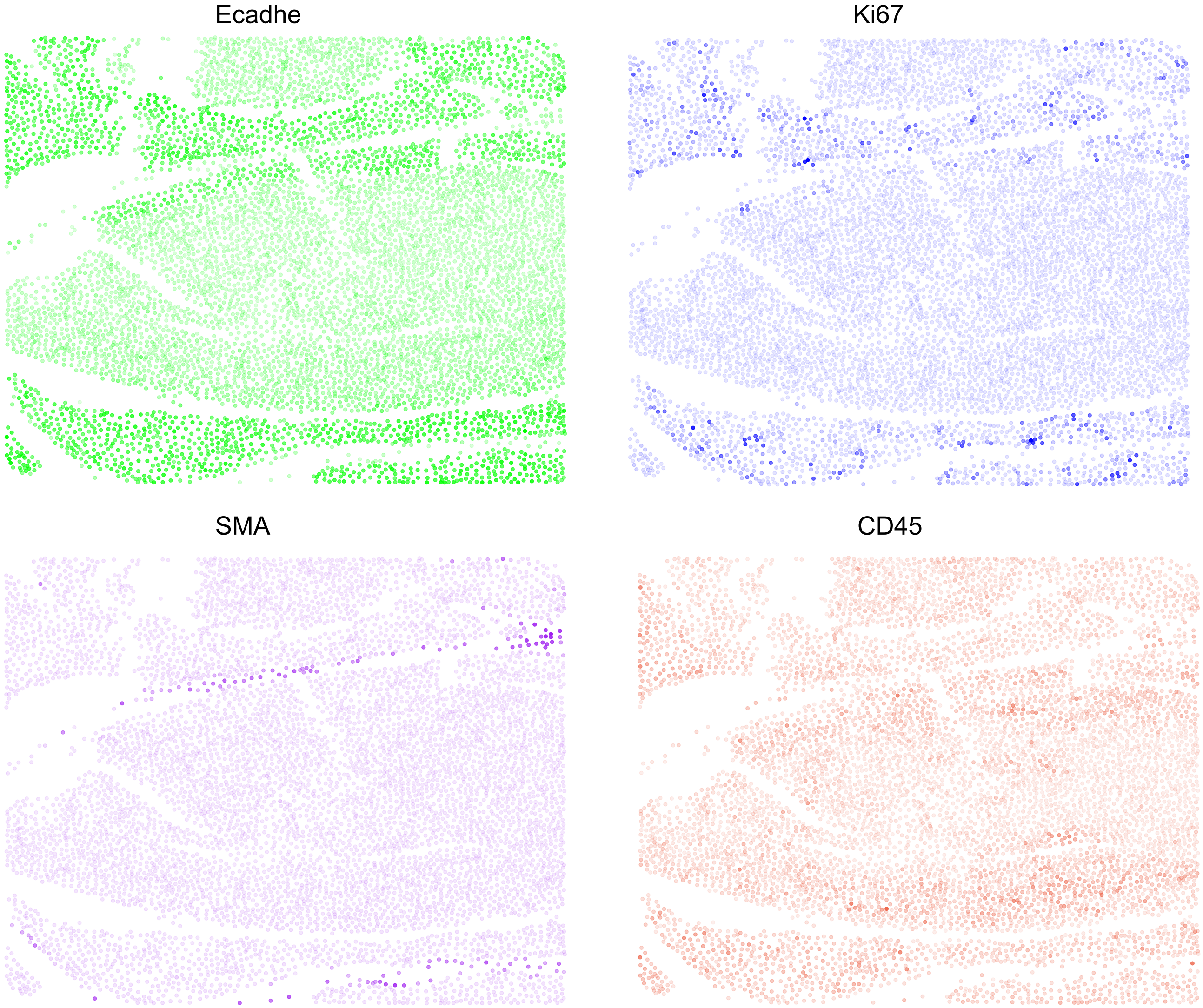
Supplementary Fig. 1



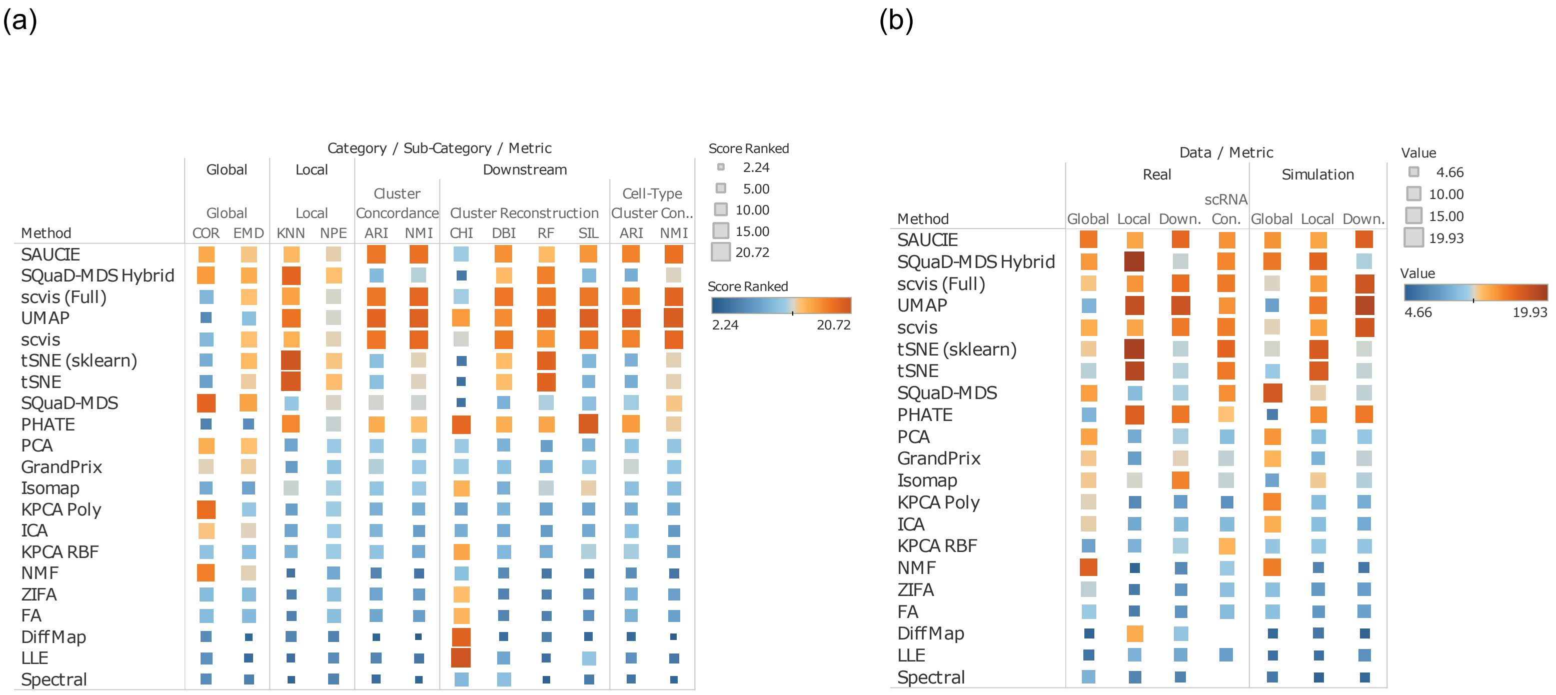
Supplementary Fig. 1 Number of yearly citations returned by Google Scholar, by searching for "CyTOF", "scRNA-seq", and "CyTOF & scRNA-seq". The blue curves show the true citations counts, and the orange curves show the exponential curves fitted to the true citation counts. Source data are provided as a Source Data file.

Supplementary Fig. 2



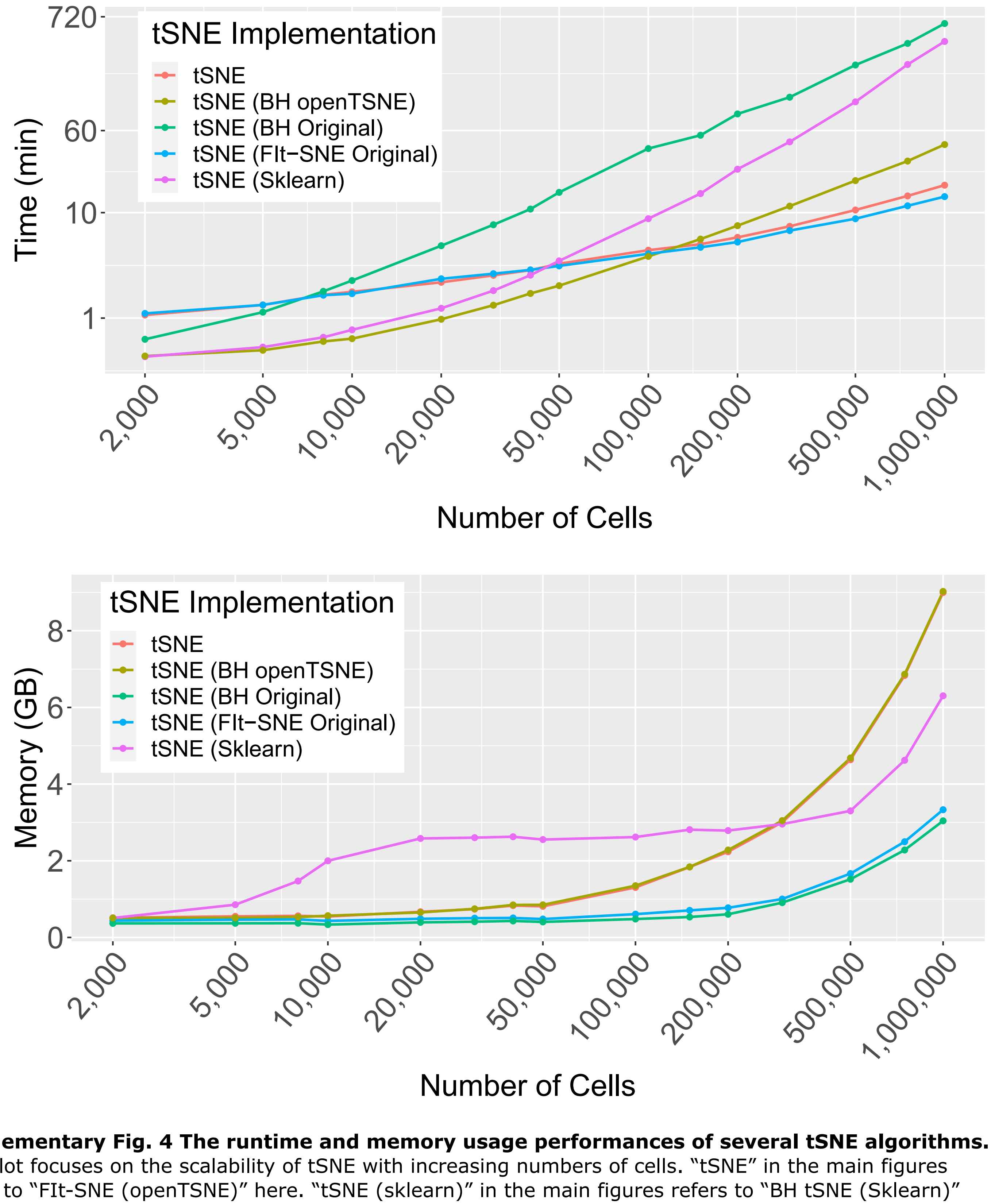
Supplementary Fig. 2 One example breast cancer Imaging CyTOF dataset. The per-cell staining of the Ecadhe (E-cadherin), Ki67, SMA (α-smooth muscle actin), and CD45 channels were shown in the spatial context. Source data are provided as a Source Data file.

Supplementary Fig. 3



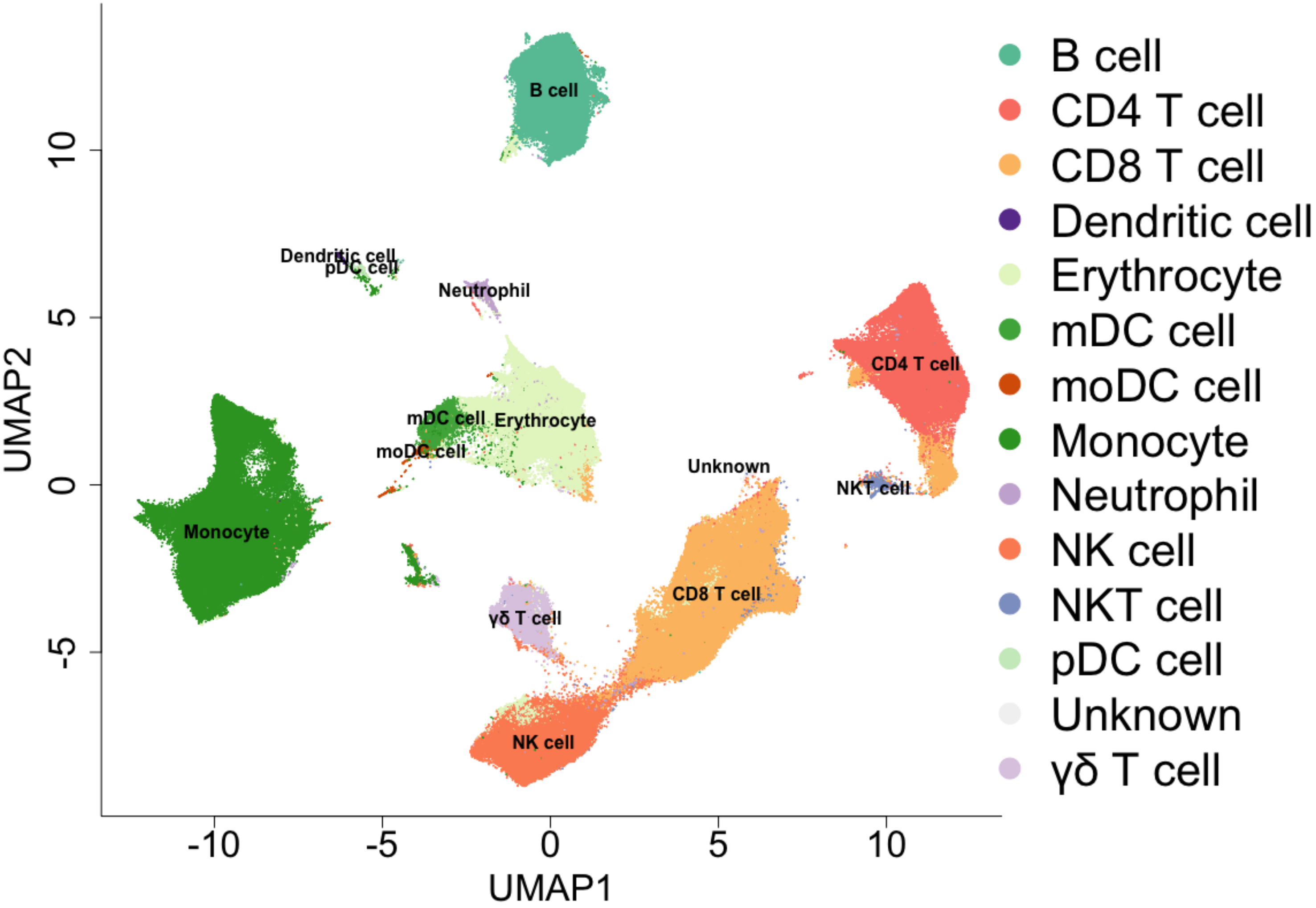
Supplementary Fig. 3 Performances of the DR methods on the simulation data. (a) Performances of the DR methods on the simulation data for each specific category of the accuracy metrics. (b) Performances of the DR methods on the simulation and real data for each major category of the accuracy metrics (scores of the specific categories averaged for each major category). Source data are provided as a Source Data file.

Supplementary Fig. 4



Supplementary Fig. 4 The runtime and memory usage performances of several tSNE algorithms. This plot focuses on the scalability of tSNE with increasing numbers of cells. “tSNE” in the main figures refers to “FIt-SNE (openTSNE)” here. “tSNE (sklearn)” in the main figures refers to “BH tSNE (Sklearn)” here. Source data are provided as a Source Data file.

Supplementary Figure 5

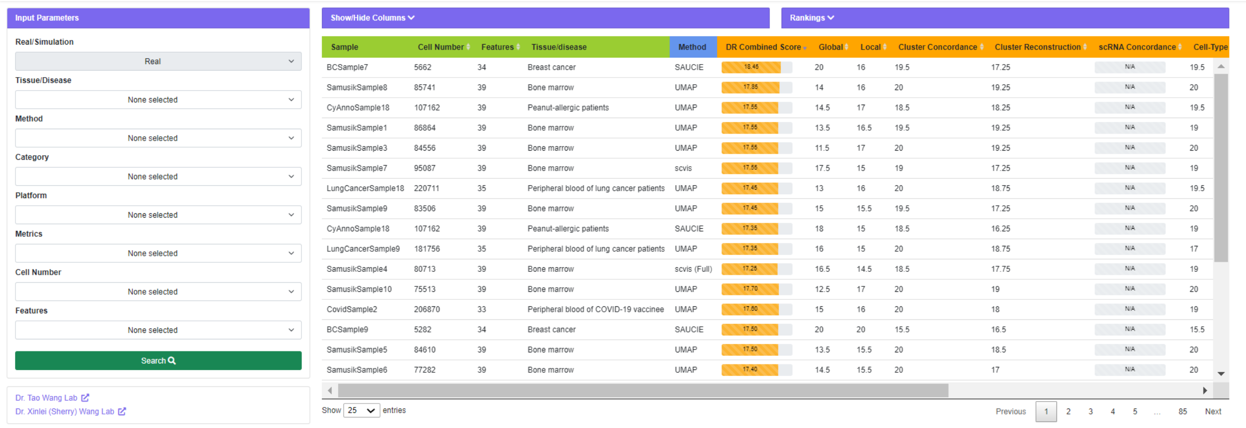


Supplementary Fig. 5 Manual cell type assignment results of the CyTOF data from the Lung cancer cohort. All cell types are visualized by a colored UMAP embedding. Source data are provided as a Source Data file.

Supplementary Note 1: Additional analyses and discussions

Web resources we generated for this CyTOF DR review study

We generated three resources, a webserver for displaying the results for benchmarking CyTOF DR methods (**Supplementary Fig. 6**), a webserver for providing our easy-to-deploy implementations of the DR methods and our evaluation metrics (**Supplementary Fig. 7**), and the *Cytomulate* algorithm for simulating CyTOF data (**Supplementary Fig. 8**).



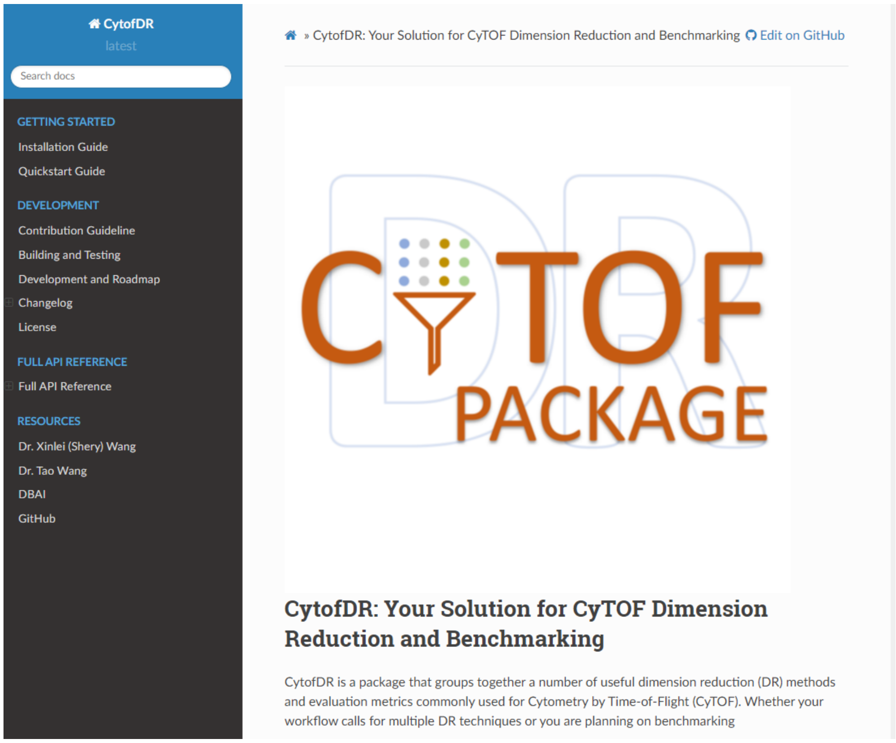
Real/Simulation: Real
Tissue/Disease: None selected
Method: None selected
Category: None selected
Platform: None selected
Metrics: None selected
Cell Number: None selected
Features: None selected

Dr. Tao Wang Lab
Dr. Xinlei (Sherry) Wang Lab

Sample	Cell Number	Features	Tissue/disease	Method	DR Combined Score	Global	Local	Cluster Concordance	Cluster Reconstruction	scRNA Concordance	Cell-Type
BCSample7	5662	34	Breast cancer	SAUCIE	18.45	20	16	19.5	17.25	N/A	19.5
SamusikSample8	85741	39	Bone marrow	UMAP	17.85	14	16	20	19.25	N/A	20
CyAnnoSample18	107162	39	Peanut-allergic patients	UMAP	17.55	14.5	17	18.5	18.25	N/A	19.5
SamusikSample1	86864	39	Bone marrow	UMAP	17.55	13.5	16.5	19.5	19.25	N/A	19
SamusikSample3	84556	39	Bone marrow	UMAP	17.55	11.5	17	20	19.25	N/A	20
SamusikSample7	95087	39	Bone marrow	scvis	17.55	17.5	15	19	17.25	N/A	19
LungCancerSample18	220711	35	Peripheral blood of lung cancer patients	UMAP	17.45	13	16	20	18.75	N/A	19.5
SamusikSample9	83506	39	Bone marrow	UMAP	17.45	15	15.5	19.5	17.25	N/A	20
CyAnnoSample18	107162	39	Peanut-allergic patients	SAUCIE	17.35	18	15	18.5	16.25	N/A	19
LungCancerSample9	181756	35	Peripheral blood of lung cancer patients	UMAP	17.35	16	15	20	18.75	N/A	17
SamusikSample4	80713	39	Bone marrow	scvis (Full)	17.25	16.5	14.5	18.5	17.75	N/A	19
SamusikSample10	75513	39	Bone marrow	UMAP	17.70	12.5	17	20	19	N/A	20
CovidSample2	206870	33	Peripheral blood of COVID-19 vaccinee	UMAP	17.60	15	16	20	18	N/A	19
BCSample9	5282	34	Breast cancer	SAUCIE	17.50	20	20	15.5	16.5	N/A	15.5
SamusikSample5	84610	39	Bone marrow	UMAP	17.50	13.5	15.5	20	18.5	N/A	20
SamusikSample6	77282	39	Bone marrow	UMAP	17.40	14.5	15.5	20	17	N/A	20

Show 25 entries

Supplementary Fig. 6 Screenshot of CyTOF DR playground. The user interface of the webserver.



GETTING STARTED
Installation Guide
Quickstart Guide

DEVELOPMENT
Contribution Guideline
Building and Testing
Development and Roadmap
Changelog
License

FULL API REFERENCE
Full API Reference

RESOURCES
Dr. Xinlei (Shery) Wang
Dr. Tao Wang
DBAI
GitHub

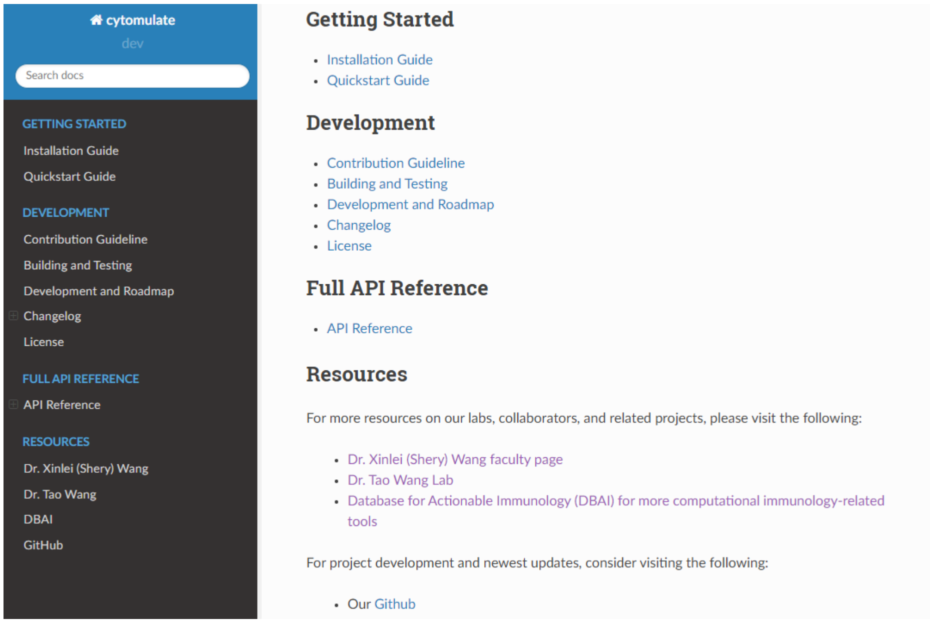
» CytofDR: Your Solution for CyTOF Dimension Reduction and Benchmarking   Edit on GitHub

CytofDR: Your Solution for CyTOF Dimension Reduction and Benchmarking

CytofDR is a package that groups together a number of useful dimension reduction (DR) methods and evaluation metrics commonly used for Cytometry by Time-of-Flight (CyTOF). Whether your workflow calls for multiple DR techniques or you are planning on benchmarking

Supplementary Fig. 7 Screenshot of CyTOF DR package. The landing page of CyTOF DR Package's documentation.

To facilitate open and user-friendly deployment and evaluation of DR methods, we established the CyTOF DR Package website (CytofDR.readthedocs.io, **Supplementary Fig. 7**) to provide implementation of the DR methods and our evaluation metrics, which is streamlined as much as possible. As we illustrated in **Fig. 1f**, a new DR user of CyTOF data can take a prior knowledge-driven approach of examining our DR benchmark results of the datasets evaluated in this study, shared conveniently through "CyTOF DR Playground", and then run their favorite DR methods using their own installation or the implementations that we provide on "CyTOF DR Package". The user can also take an empirical approach of deploying the evaluation metrics (or a subset of our evaluation metrics according to preference or data availability) on their own CyTOF datasets, using the evaluation and DR code implementation that we offered through "CyTOF DR Package", for picking the best approach. We also envision our system to be quite useful to future developers of DR methods for CyTOF data or other types of data in general. The extensive collection of evaluation metrics will allow future developers to craft their own benchmark recipes for DR in any field. At the same time, developers can easily extend the framework and add functionalities by simply deriving from our Python classes. We hope such a unified and streamlined evaluation-execution paradigm will become more widely adopted for various bioinformatics development scenarios.



Getting Started

- Installation Guide
- Quickstart Guide

Development

- Contribution Guideline
- Building and Testing
- Development and Roadmap
- Changelog
- License

Full API Reference

- API Reference

Resources

For more resources on our labs, collaborators, and related projects, please visit the following:

- Dr. Xinlei (Shery) Wang faculty page
- Dr. Tao Wang Lab
- Database for Actionable Immunology (DBAI) for more computational immunology-related tools

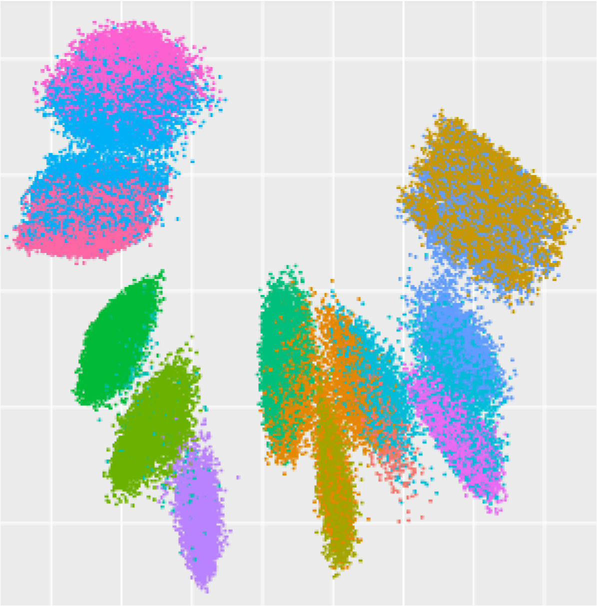
For project development and newest updates, consider visiting the following:

- Our Github

Supplementary Fig. 8 Screenshot of Cytomulate. The landing page of Cytomulate's documentation website.

An example simulated CyTOF dataset generated by *Cytomulate*

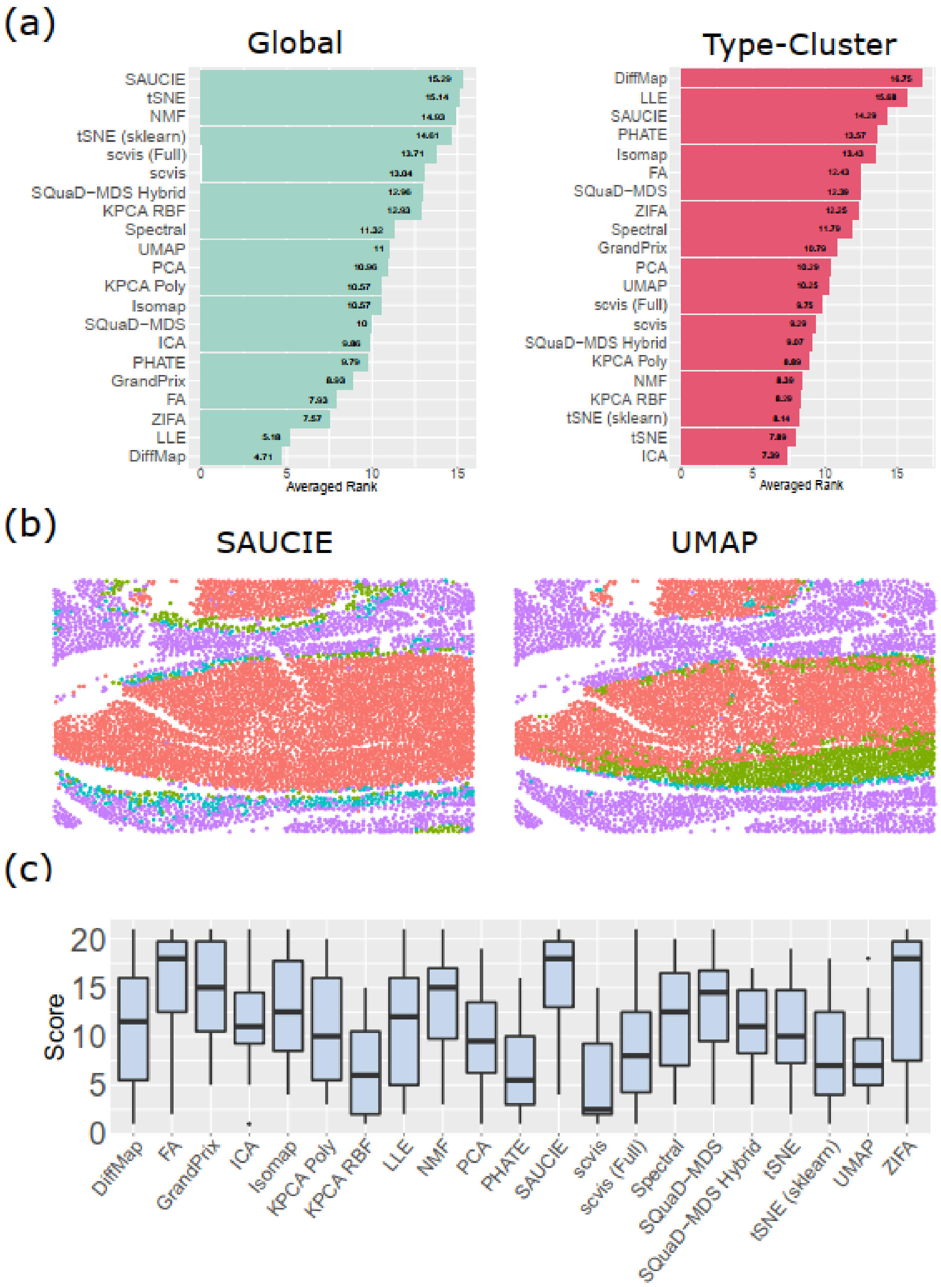
In **Supplementary Fig. 9**, we showcase a simulated CyTOF dataset generated by *Cytomulate*. Multivariate Gaussians were used to generate the protein expressions of cells of various cell types. SAUCIE was used to visualize the single cells in a two dimensional space. According to the overall pattern of distribution of this dataset and those of real CyTOF data (**Fig. 4a**), we can see that *Cytomulate* closely mimics real CyTOF data. Compare the pattern of distribution of cells (such as level of dispersion with respect to the clusters) in this figure with the SAUCIE figure of **Fig. 4a**. More analyses and results validating the performance of *Cytomulate* in simulation of CyTOF data can be found in our *Cytomulate* work [1].



Supplementary Fig. 9 One simulated CyTOF dataset visualized in the SAUCIE space. Source data are provided as a Source Data file.

Detailed inspection of the DR results for the BC dataset

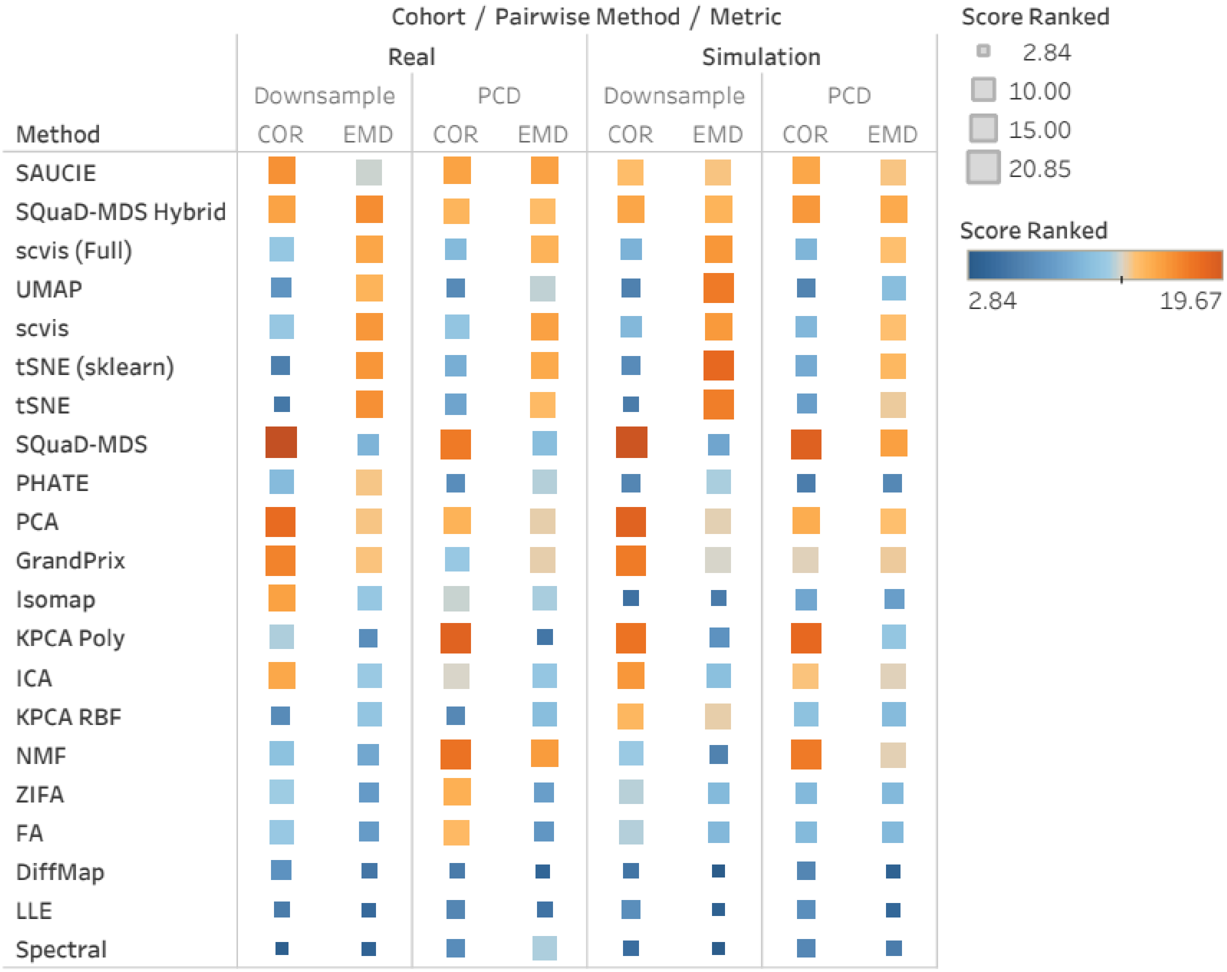
SAUCIE is the top performer in terms of Global structure preservation. Further, SAUCIE significantly outperforms UMAP and scvis in the cell-type cluster concordance subcategory of Downstream performance (**Supplementary Fig. 10a**). These two results suggest that SAUCIE is the best at detecting and distinguishing the main types of cells that exist in the samples. We classified the cells in the BC CyTOF samples into four clusters in the DR space as guided by the original BC publication, which reported four cell types (tumor, immune, vessel, stroma). Then, we visualized the clusters in the spatial context (**Supplementary Fig. 10b**). The red cells in both panels and the green cells in the right panel from UMAP denote tumor cells. This shows that SAUCIE tends to be better at clustering all tumor cells together as a whole.



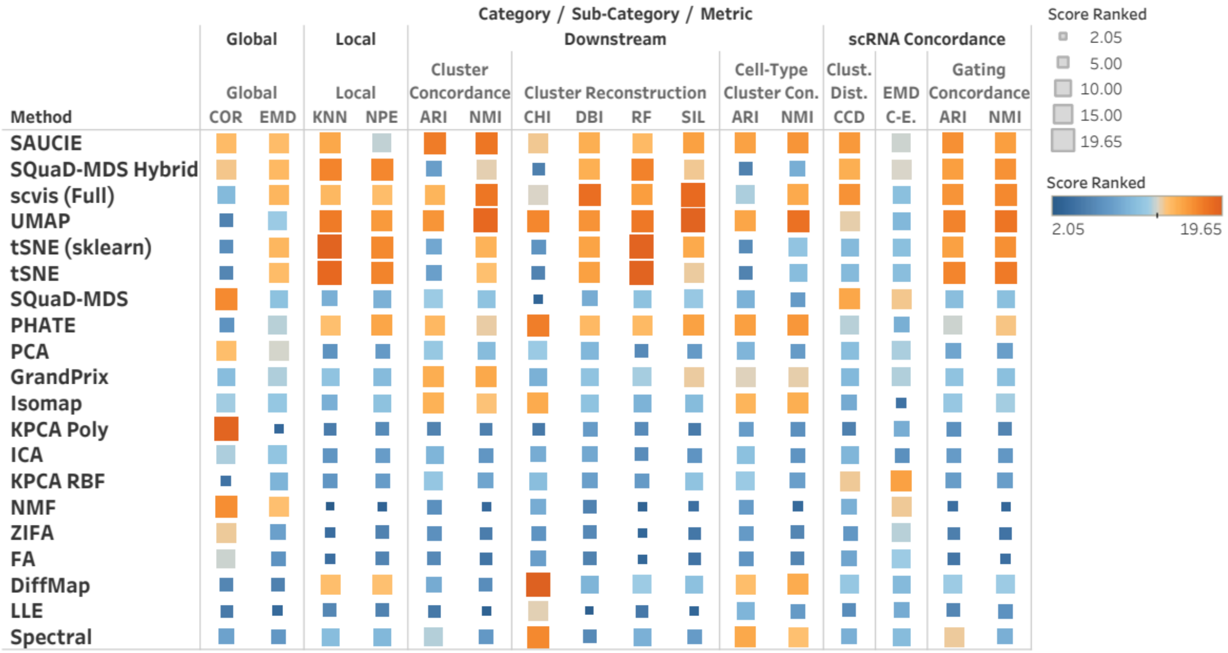
Supplementary Fig. 10 Closer examination of the DR results for the BC dataset. (a) The performances of the DR methods in terms of global accuracy metric and cell type-cluster concordance subcategory of Downstream performance. (b) The cells in the BC dataset (sample #1) were classified into four types, according to clustering of the cells in the DR space. The cells and these clusters were visualized in the cells' spatial context. Left: SAUCIE; right: UMAP. (c) The correlation between spatial distances and the distances in gene expression between pairs of cells in the same "clusters" that were determined in (b). Box boundaries represent interquartile ranges, whiskers extend to the most extreme data point which is no more than 1.5 times the interquartile range, and the line in the middle of the box represents the median. For each method, the sample size is $N = 14$, which corresponds to the 14 samples from the BC cohort. Source data are provided as a Source Data file.

A comparison between downsampling-based and Point Cluster Distance (PCD)-based benchmark metrics

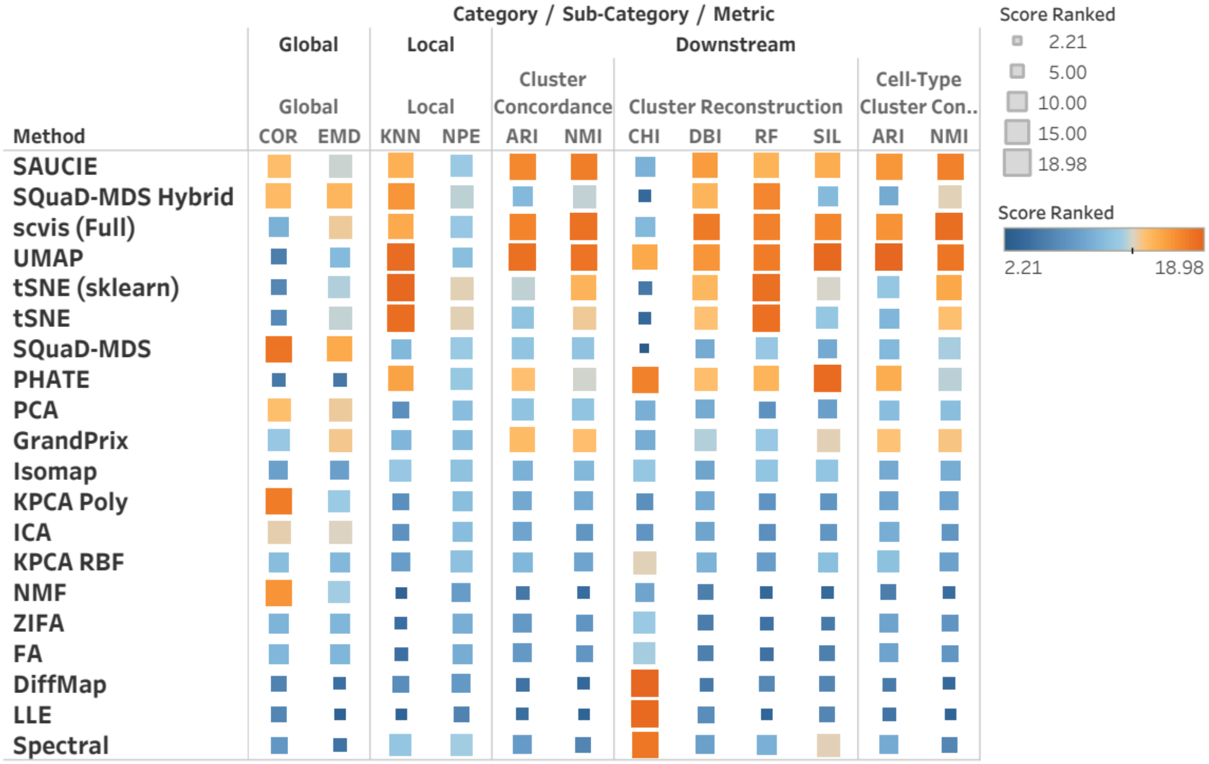
In our accuracy metrics, EMD and COR of the Global Category utilized PCD, which basically means that they used cluster centroids instead of actual data points and their embeddings to assess performance, in parts of the accuracy computation process. We chose this approach to avoid the quadratic computational complexity of certain metrics. However, one concern is that collapsing cell clusters to their centroid will obfuscate instances where two clusters overlap in the low dimensional embedding, and that clustering methods are not perfect, which means that a suboptimal clustering will lead to a suboptimal gold standard. To remove the concern regarding the negative impact of this potential caveat of PCD-based approaches, we also tested an alternative approach. We computed the pairwise distance in the embedding and DR space in a randomly sub-sampled set of 10,000 cells (without replacement) for each CyTOF dataset and then computed the full pairwise Euclidean distances within this subset. Other ranking and analysis methods remained the same, as they did not involve PCD. We compared the results obtained from both approaches in **Supplementary Fig. 11** and found that both PCD- and subsampling-based methods yield very similar results in real and simulated CyTOF samples. Critically, the results are nearly identical for the top-ranking methods.



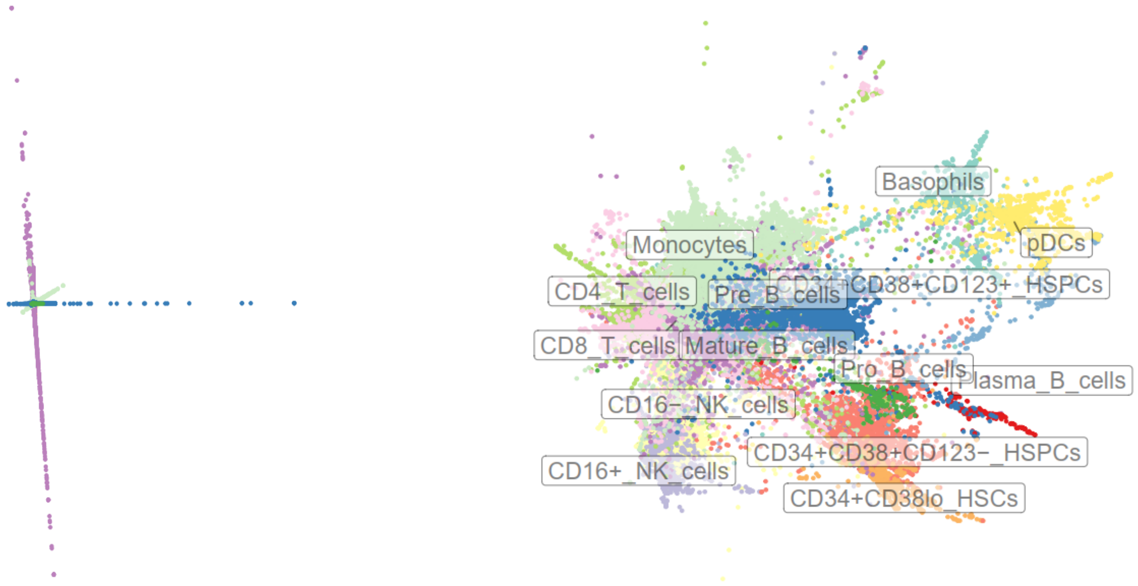
Supplementary Fig. 11 Global accuracy results from both downsampling-based and PCD-based approaches. Orange color indicates better performance, while the blue color indicates worse performance. All sub-metrics of the accuracy criterion were averaged. Source data are provided as a Source Data file.



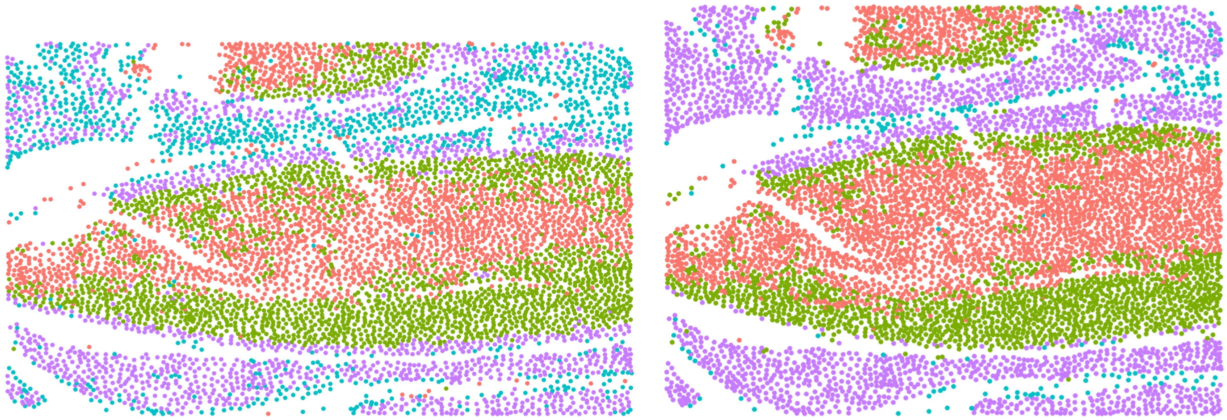
Supplementary Fig. 12 Accuracy performance on all 10% downsample real datasets. This plot corresponds to the results displayed in **Fig. 3b**, which does not downsample for evaluation. The performance of DR methods is similar to that of the main figure. Orange color indicates better performance, while the blue color indicates worse performance. All sub-metrics of the accuracy criterion were averaged. Source data are provided as a Source Data file.



Supplementary Fig. 13 Accuracy performance on all 10% downsample simulation datasets. This plot corresponds to the results displayed in **Supplementary Fig. 3a**, which does not downsample for evaluation. As with real datasets, the performance shown here is again comparable. Orange color indicates better performance, while the blue color indicates worse performance. All sub-metrics of the accuracy criterion were averaged. Source data are provided as a Source Data file.



Supplementary Fig. 14 Examples of bad DR embeddings from the Levine32 dataset. The left panel is LLE, which fails to distinguish any cell types or clusters meaningfully. The right panel is the PHATE embedding. PHATE, which usually performs decently, produces a confusing embedding with cell types mixed with each other. Source data are provided as a Source Data file.



Supplementary Fig. 15 Examples of bad IMC plots from the BC cohort Sample 1. The left panel is clustering results from KPCA RBF and the right panel from scvis. As compared to the good clustering performance from SAUCIE embedding (**Supplementary Fig. 10b)**, these two DR methods' embeddings fail to cluster tumor cells in one cluster (red and green clusters). Further, KPCA RBF's blue and purple cells are noticeably mixed together as well. Source data are provided as a Source Data file.

Supplementary Discussion 1: Discussion of Scalability and Dimensionality

As discussed in the **Introduction** section of our paper, a unique differentiating factor between CyTOF and scRNA-seq is the dimensionality of the datasets. Namely, CyTOF has orders of magnitude less channels (features denoted as P) and much more cells (sample size denoted as N). The scalability benchmark clearly indicates that different methods scale vastly differently with respect to both N but not so much for P. Here, we discuss efficiency concerns of some of the prominent methods and their fit for CyTOF data analysis purposes.

SAUCIE

SAUCIE[2] is based on an autoencoder implemented in tensorflow. It uses an 8-layer neural network, and DR is achieved by extracting the middle bottleneck layer. While SAUCIE does not give a specific efficiency claim in terms of N or P, both its author and our study found that its efficiency for DR is top-tier. Surprisingly, the performance of SAUCIE, even without using GPUs, is on par with the most efficient methods tested. Given the nature of the autoencoder, only the input and output layers have to have the same size as P of the original training data. Thus, the impact of P should be small, unless larger intermediate layers are needed for higher dimensional data like scRNA-seq. Sample size N can potentially interact with the choice of batch size which may in turn impact scalability, but our results suggest that default settings already produce good embeddings for CyTOF. From our study, SAUCIE remains fast and accurate for CyTOF samples. Thus, CyTOF's dimensionality is not of concern for SAUCIE.

scvis

scvis[3] uses variational inference to find the latent variable to represent the latent variable representing underlying structure in the data. To aid the visualization of the latent space, it utilizes the idea of tSNE as a regularizer: specifically, it uses KL divergence to minimize distance as defined by a Gaussian kernel. While scvis is among the least efficient methods without memory bounds, we found that the number of features does not play a role in its runtime with typical CyTOF feature spaces. However, sample size N poses a significant concern here since a typical sample can take as much as 12 hours to run on a server even though runtime does tend to plateau. The use of scvis is thus a compromise between theoretical fit and practicality.

UMAP

UPAM falls under the second tier of methods in terms of efficiency. In UMAP's methodological paper[4], the authors claimed approximately $O(N^{1.14})$ for the optimization algorithm. Since UMAP falls in the same category of algorithms that utilize k-nearest neighbors graphs, its performance mostly depends on sample size P rather than N like Fit-SNE. However, due to the fact that it does not need to normalize the density in the optimization step, the authors claimed that UMAP can be much more efficient when reducing dimensions down to more than 2 or 3 dimensions. While higher dimensional DR embeddings are not the focus of this paper, it is a common workflow in scRNA-seq data analysis, which may help explain the advantage of UMAP in that setting.

PHATE

PHATE[5] is a method that focuses the differentiation path of cells. Namely, it calculates the local similarity and global relationship. Then, it uses potential distance to account for the overall structure and metric MDS for visualization. There is no explicit claim of efficiency, but we observed that it is among the least efficient methods for runtime. Like other methods, dimensionality does not play a large role for efficiency except for using larger vectors, but it scales at least quadratically with the number of cells. Like scvis, the original paper also used PHATE for scRNA-seq data. Therefore, dimensionality should not be much of a concern given that it is a distance-based method, but sample size again poses a challenge.

SQuaD-MDS

SQuaD-MDS[6] improves upon the original MDS by employing Stochastic Quartet Descent. CyTOF's large sample size N renders MDS unusable without downsampling because of $O(N^2)$ memory complexity. SQuaD-MDS vastly improves the efficiency by reducing runtime and memory complexity down to $O(N)$, which is on par with the second tier of methods. As with other force-directed methods, dimensionality does not play nearly as large of a role as N. SQuaD-MDS is one of the methods that is general-purpose. However, it was tested on both image data with high dimensions and some low-dimensional scRNA-seq benchmark datasets by its authors. We can thus reasonably expect that SQuaD-MDS will work for CyTOF and scRNA-seq datasets across a wide range of dimensionality.

tSNE (All major variants)

As tested extensively in this study, tSNE's implementation has important ramification in terms of scalability. While we included two implementations in main benchmarks and five in **Supplementary Fig. 4**, these implementations can be divided into the following categories:

1. Original tSNE [7]
2. Barnes-Hut tSNE [8]
3. Fit-SNE[9]

The original tSNE has $O(N^2)$ for computation time and memory efficiency. This is obviously unrealistic for large CyTOF samples: in fact, any samples with $N > 10,000$ can start to become problematic. As tSNE's optimization involves the pairwise forces exerted by each point in both the original and DR space, the main bottleneck is N rather than P.

The Barnes-Hut optimization for tSNE (BH tSNE) reduces the run time to $O(N\ logN)$ for computation and $O(N)$ for memory usage. In our benchmarks, BH tSNE can be realistically used for most CyTOF datasets, but significant time is needed when samples are large. Recently, Fit-SNE further reduces runtime to $O(2pN)$. Fit-SNE is by far the fastest implementation we tested while also being the most competitive with other methods (*e.g.* UMAP). The one caveat with Fit-SNE is its availability since it is still relatively new, but there are already implementations and wrappers available for use.

Supplementary References

1. Yang, Y., Wang, K., Lu, Z., Wang, T. & Wang, X. Cytomulate: Accurate and Efficient Simulation of CyTOF data. *BioRxiv* (2022) doi:10.1101/2022.06.14.496200.
2. Amodio, M. *et al.* Exploring single-cell data with deep multitasking neural networks. *Nat. Methods* **16**, 1139–1145 (2019).
3. Ding, J., Condon, A. & Shah, S. P. Interpretable dimensionality reduction of single cell transcriptome data with deep generative models. *Nat. Commun.* **9**, 2002 (2018).
4. McInnes, L., Healy, J. & Melville, J. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. *arXiv* (2018) doi:10.48550/arxiv.1802.03426.
5. Moon, K. R. *et al.* Visualizing structure and transitions in high-dimensional biological data. *Nat. Biotechnol.* **37**, 1482–1492 (2019).
6. Lambert, P., de Bodt, C., Verleysen, M. & Lee, J. A. SQuadMDS: A lean Stochastic Quartet MDS improving global structure preservation in neighbor embedding like t-SNE and UMAP. *Neurocomputing* **503**, 17–27 (2022).
7. Van der Maaten, L. & Hinton, G. Visualizing data using t-SNE. *Journal of machine learning research* **9**, (2008).
8. Van Der Maaten, L. Accelerating t-SNE using tree-based algorithms. *The Journal of Machine Learning Research* **15**, 3221–3245 (2014).
9. Linderman, G. C., Rachh, M., Hoskins, J. G., Steinerberger, S. & Kluger, Y. Fast interpolation-based t-SNE for improved visualization of single-cell RNA-seq data. *Nat. Methods* **16**, 243–245 (2019).