A comprehensive dimensional reduction framework to learn single-cell phenotypic topology uncovers T cell diversity

Davi Sidarta-Oliveira[1,2,3] and Licio A. Velloso[2,3]

1. Physician-Scientist Graduate Program, School of Medical Sciences, University of Campinas, Brazil
2. Laboratory of Cell Signaling, Obesity and Comorbidities Research Center, University of Campinas, Brazil
3. School of Medical Sciences, University of Campinas, Brazil

Correspondence:

D. Sidarta-Oliveira - davisidarta@gmail.com

L. A. Velloso - lavellos@unicamp.br

Obesity and Comorbidities Research Center, University of Campinas

13083-864, Campinas, SP, Brazil

Abstract

The core task when analyzing single-cell data (SCD) is dimensional reduction (DR), which aims to find latent signals encoding biological heterogeneity. Here, we dissected DR steps to build TopOMetry, a machine learning framework that learns latent data topology to perform DR in a modular fashion and show that current analysis practices are biased due to the non-uniformity and non-linearity of the geometry underlying SCD. We used TopOMetry to analyze SCD from peripheral blood mononuclear cells (PBMC), and consistently found a plethora of unreported T CD4 subpopulations that are missed by current workflows across datasets and diseases. We also found comparable T CD4 diversity in the human cerebrospinal fluid. The proposed framework scalates to millions of cells, excels at discovering fine-grained local structure, and holds natural connections to clustering and trajectory inference. We hope this powerful tool will accelerate biomedical discovery and inspire new methods to learn and explore phenotypic topology.

Introduction

Finding meaningful, latent information from high-dimensional data (HDD) is a central concern in several research areas and particularly critical when analyzing single-cell data (SCD). Single-cell assays have inaugurated a new era in the unbiased molecular sampling of the cellular diversity present in biological systems, from cell lines to whole organisms[1,2]. However, the complexity and vastitude of the data generated by such experiments require sophisticated computational frameworks for its analysis[3,4]. In such data, information regarding cellular phenotype (e.g. gene expression, chromatin accessibility, etc.) is individually gathered for every cell, resulting in matrices containing up to millions of cells and dozens of thousands of observations. Thus, a core task in the analysis of SCD is dimensional reduction (DR), which aims to embed these cell-specific observations into a latent representation lying on a lower-dimensional space (usually dozens of latent dimensions, or in 2- or 3-D for visualization) while preserving as much original information as possible[4,5]. Essentially, this attempts to find latent signals encoding phenotypic diversity. Furthermore, DR is indispensable for downstream tasks in SCD analysis[6], such as clustering, RNA velocity[7,8], batch-correction[9,10], lineage inference, and pseudotime estimation[11,12]. Thus, it is crucial to obtain representations that faithfully represent the complex topologies underlying phenotypic diversity in SCD.

The concept of finding latent signals representative of cellular identity is intrinsically related to defining a finite number of cell lineages and types within a given dataset. There is also broad interest in the somewhat familiar property of DR of frequently revealing the dynamics underlying cellular differentiation when applied to SCD, as numerous cells asynchronously

undergoing differentiation trajectories are sampled simultaneously. Given adequate low-dimensional mappings, moving through continuous topologies in the form of connected populations often corresponds to differentiation or stimulus-response processes, a feature explored by trajectory inference (TI) algorithms[11,13]. This property is particularly noticeable in Diffusion Maps (DM), a DR method broadly employed for pseudotime estimation, due to its intrinsic relation to the heat kernel and Markov chains, and is extensively used to model cellular development[6,12,14,15]. The essence of TI through DR is also intrinsically related to a theoretical epigenetic landscape where cell states co-exist through development trajectories (i.e. an extended Waddington's landscape[16]).

Numerous DR methods have been developed, and many have been applied to SCD. The seminal Principal Component Analysis (PCA)[17,18] algorithm performs linear transformations to find hyperplanes of maximum covariance to denoise redundantly correlated features that are named Principal Components, and is a linear method. More recent and sophisticated algorithms most often obtain non-linear embeddings by assuming that data exists in or close to an underlying lower-dimensional manifold (i.e. the manifold hypothesis). In practice, such methods rely on first building k-nearest-neighbors graphs by computing distances (i.e., dissimilarities) between cells in the high-dimensional space, although random projections and neural-network methods can also be employed. A similarity weighting function (i.e., a *kernel*) can also be used to balance initially measured distances to their neighborhoods, usually to account for non-linearity, such as in kernel-PCA. The eigendecomposition of such similarity matrices or their graph Laplacian can then be used to explore latent aspects of data geometry, such as in Laplacian

Eigenmaps (LE)[19] and Diffusion Maps (DM)[20–22]. We refer to these methods as embedding algorithms.

Methods best suited for visualization compute such similarity scores to learn new graphs with some desired properties and then optimize cost functions that minimize differences between such graph and a lower-dimensional initialization, given certain constraints - usually with 2- or 3-D for visualization. Examples are t-SNE[23,24] (t-Distributed Stochastic Neighborhood Embedding); UMAP[25,26] (Uniform Manifold Approximation and Projection);[27] and TriMAP (triplet-based)[28]. Some methods focus on preserving distances rather than local information, such as Isomap[29] and MDS[30], although this could be biased when dealing with high-dimensional data, commonly leading to loss of local topology[19,31,32]. This trade-off between conserving either distances or topology (i.e., global or local information) is remarkable, has been approached by recent reports[33,34]. Yet, PaCMAP (Pairwise Controlled Manifold Approximation and Projection)[31] recently emerged as a balanced solution to achieve both. We collectively refer to these algorithms as graph layout optimization methods. Most techniques involve using a number of iterations for stochastic gradient-descent algorithms to optimize an embedding under given constraints. More recently, a generalization of DR through graph layout optimization was proposed with Minimum Distortion Embedding (MDE)[35], which broadly established the layout process as combinations of specific constraints, losses, and initializations, and PCA, LE, and UMAP as special cases of MDE[31,34].

Some exceptions to the exposed methods include tailored combinations of embedding and layout optimization, such as scVI(single-cell Variational Inference)[36], which employs neural

networks to find latent signals, and UMAP to optimize layouts. Another popular method, PHATE (Potential of Heat Affinity and Transition Estimation)[6], uses an adaptation of DM internal Markov chains to learn similarities with a diffusion operator and then a modified version of MDS to optimize a 2- or 3-D layout. Further exceptions are ZIFA[37] and ZINB-WAVE[38], which are tailored to deal with the dropout phenomenon observed in single-cell RNA sequencing (scRNAseq) - albeit others claim these could be rather informational[39], thus distorted by these methods. Other alternatives encompass advanced matrix factorization methods, such as non-negative matrix factorization (NMF)[40], which can also be helpful for transfer learning and data integration. A systematic review of existing methods surpasses the scope of this manuscript; however, we note that most DR methods fall into two broad categories depending on their focus: embedding and layout algorithms, used respectively for finding latent embeddings or optimizing a layout (usually into 2- or 3-D for visualization) from an initialization. UMAP and MDE are corner cases that can also learn latent embeddings through layout optimization to an arbitrary number of dimensions. Embedding methods tend to hold a greater focus on similarity learning, a feature generally overseen in layout methods in favor of sophisticated optimization functions tailored for visualization.

A particularity of methods grounded on discrete differential geometry such as LE, DM, and UMAP is their ability to independently achieve discrete approximations of the Laplace-Beltrami Operator (LBO)[19,22,25,41], a generalization of the graph Laplacian and an optimal tool to recover information about data intrinsic structure[19,42]. Methods that approximate the LBO are advantageous due to their natural connection to spectral theory and clustering and are intuitively

related to the graph smoothing process employed by these methods[19,31,32,41,43], such as in continuous k-nearest-neighbors (CkNN), which can also approximate higher-order operators (the 0-th Laplace-de Rham Operator is the Laplace-Beltrami Operator)[32].

Some work has been done in comparing the performance of distinct methods for DR in single-cell data[44–46] - yet, these are limited to quality metrics in benchmarking pipelines and lack both the needed tuning of each algorithm to the analyzed data sets and the qualitative assessment of each algorithm's topology preservation of actual ground-truth biological landmarks, such as the cell cycle as a closed logic loop. Even more complex is validating the faithfulness of the learned topologies themselves. Even when downstream hypotheses are experimentally validated, these can hardly be extended as direct proofs of the learned representations *per se*[33]. Each algorithm's assumptions regarding data structure and distribution are also disturbingly challenging to assess in practice. The universally used PCA holds the relatively strong assumption that hyperplanes (the PCs) can approximate data geometry and can reach misleading results when faced with non-linear structure, missing local information, and introducing linear distortions. The popular UMAP, while more general by loosely assuming that data is distributed in or near a locally connected Riemannian manifold, assumes manifold uniformity and a locally constant metric. An alternative to assuming data topological properties would be to hold only the manifold hypothesis by seeking approximations of the LBO[32]. It would only be needed to assume that data lies in or close to a set of topological manifolds, possibly with non-linear and non-uniform data structure and unknown sampling distribution. In practice, this is true when the relation of neighborhood size S to the total number N of cells approaches zero, that is, when it is

small enough for one to reasonably explore differential calculus (i.e. this ratio is 3×10^{-4}, using ~45,000 cells and 15 k-nearest neighbors). Assuming such naturally-occurring phenotypic manifolds exist within SCD holds strong connections to the concept of cell types, identities, and cellular differentiation along phenotypically-defined trajectories in the high-dimensional observation-derived topological space. In such an informational space, continuous submanifolds can naturally represent the concept of cellular trajectories, while discrete submanifolds with little to no edges between their graph representation would represent discrete, unrelated cellular populations. Multiple continuous and discrete subspaces must exist in this space to allocate the plethora of cellular diversity of unknown shapes. In essence, such subspaces represent an extension of the Waddington's landscape, as cells progress along multiple discrete differentiation trees[16]. We name this property *phenotypic topology*, i.e., the geometric structure underlying cellular identities, and use approximations of the LBO to extract it from scRNAseq data. Multiple algorithms have been developed for this task, and it would be naive to elect a single one for each and every data set. A solution would be to use different algorithms to approximate the LBO, which could naturally converge to a common goal. Qualitatively evaluating how these algorithms perform in representing single-cell data through clustering and visualization and quantitatively scoring global and local structure preservation would allow one to select the best method for their particular data, rather than naively electing a single method *a priori*.

Grounded on these insights, we dissected the steps of DR (similarity learning, learning a new latent orthogonal basis, graph learning, and layout initialization and optimization) to design

TopOMetry(Topologically Optimized geoMetry), a general and modular framework for topological data analysis. Each step independently approximates the LBO except for layout optimization, leading to dozens of possible combinations that converge to the same mathematical operator. These combinations include numerous methods unused in SCD, such as similarity learning with CkNN, building a new orthogonal basis with multiscale diffusion maps, and graph layout optimization with PaCMAP. This framework was implemented in the python package *TopOMetry* for high-dimensional data analysis. Its companion single-cell package, *CellTOMetry, was* designed to be user-friendly and function seamlessly within the python single-cell analysis environment. *CellTOMetry* systematically runs several possible algorithmic combinations for DR, including the default PCA-primed UMAP workflow, and quantitatively evaluates all results, allowing users to interpret which models to best trust and if different high-scoring models converge to the same topological representation. Here, we use it to show how single-cell data is most commonly non-linear and non-uniform by revealing dozens of previously hidden T CD4 subtypes in various public scRNAseq datasets of circulating blood cells from healthy donors and patients with systemic lupus erythematosus, dengue fever, and multiple sclerosis. We also show that such T CD4 diversity can be partly found in the cerebrospinal fluid. This approach escalates to whole-organism data composed of millions of cells, revealing fine-grained differentiation trajectories that would otherwise remain hidden. Quantitatively, we show that TopOMetry's topological models outperform the standard PCA-centered approach for single-cell analysis, defining new directions for developing computational biology tools.

Results

A modular framework for learning topological metrics, graphs, and layouts from high-dimensional single-cell data.

Unlike most methods that assume unknown aspects of data topology, we built a framework to explore sequential approximations of the Laplace-Beltrami Operator (LBO) (Figure 1A). The LBO represents latent topology when the data is or lies near a manifold. In other words, knowledge about the LBO means knowledge about data topology[19,22,32] - nevertheless, its approximation for discrete, finite, real-world datasets can be challenging. Diffusion Maps (DM)[22], Laplacian Eigenmaps (LE)[19], continuous k-Nearest-Neighbors (CkNN), and UMAP's construction of fuzzy simplicial sets (FSS) have explored LBO approximations with distinct - yet related - mathematical and computational means, thus being useful for different tasks, such as metric-learning (CkNN, diffusion harmonics, FSS), denoising (DM, LE) and projection (spectral decomposition of the graph Laplacian, FSS cross-entropy optimization). To recover high-dimensional topology with high resolution without preference for a particular model, we adapted these previous approaches into a comprehensive framework that aims to obtain discrete approximations of the LBO at each step of the dimensionality reduction procedure with different algorithms (Figure 1B).

From a given high-dimensional data matrix, *TopOMetry* first learns a topological metric with either multiscale self-adaptive diffusion harmonics (DM), CkNN, or FSS (Fig. 1B) to build a topological graph from data that approximates the LBO; then, the learned metric is used to build a first, non-linear topology-preserving orthogonal basis with an eigendecomposition (Fig. 1C). A

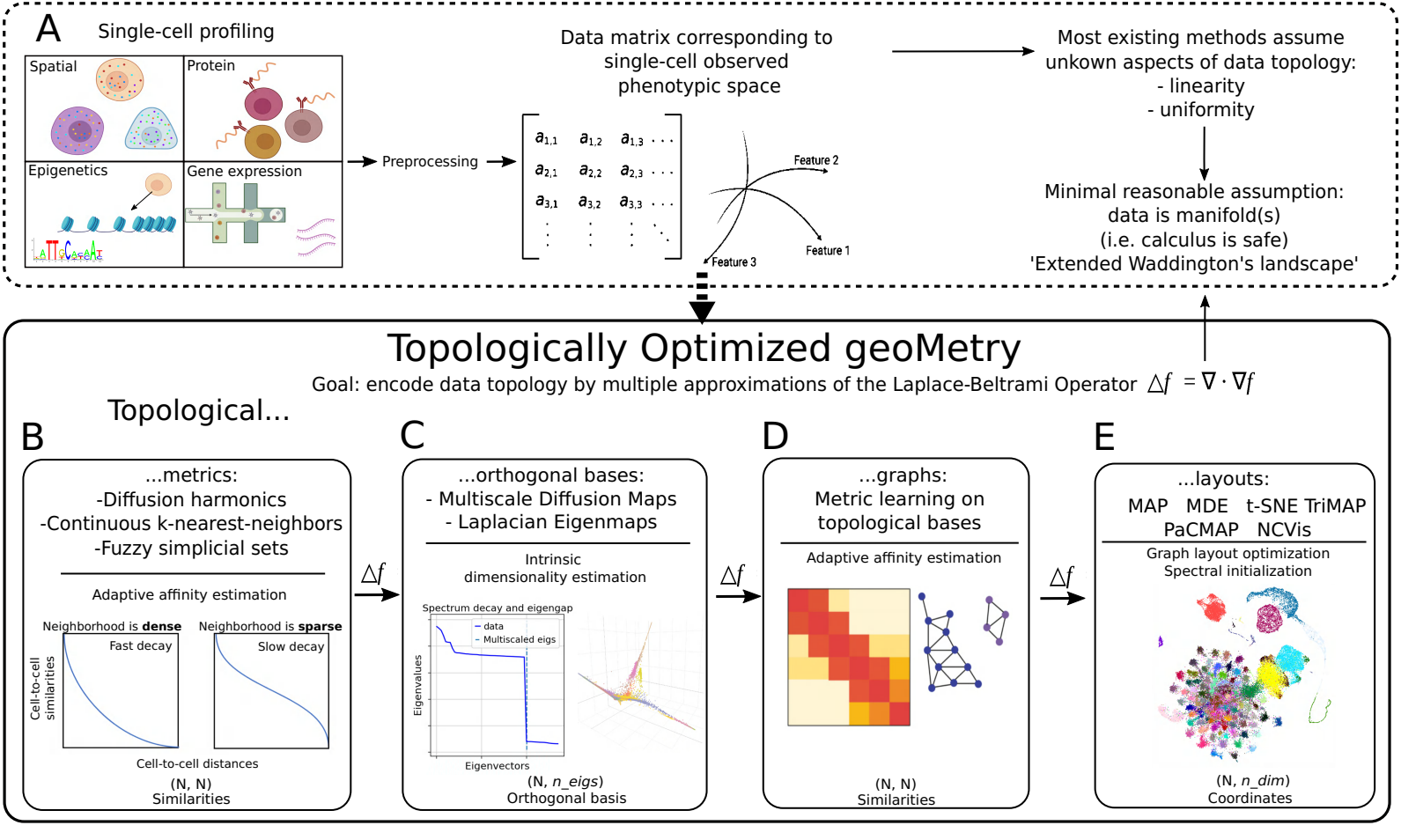

A
Single-cell profiling
Spatial
Protein
Epigenetics
Gene expression
Preprocessing
Data matrix corresponding to single-cell observed phenotypic space
Feature 1
Feature 2
Feature 3
Most existing methods assume unkown aspects of data topology:
- linearity
- uniformity
Minimal reasonable assumption:
data is manifold(s)
(i.e. calculus is safe)
'Extended Waddington's landscape'
Topologically Optimized geoMetry
Goal: encode data topology by multiple approximations of the Laplace-Beltrami Operator $\Delta f = \nabla \cdot \nabla f$
Topological...
B
...metrics:
-Diffusion harmonics
-Continuous k-nearest-neighbors
-Fuzzy simplicial sets
Adaptive affinity estimation
Neighborhood is dense
Fast decay
Neighborhood is sparse
Slow decay
Cell-to-cell similarities
Cell-to-cell distances
(N, N)
Similarities
Δf
C
...orthogonal bases:
- Multiscale Diffusion Maps
- Laplacian Eigenmaps
Intrinsic dimensionality estimation
Spectrum decay and eigengap
data
Multiscaled eigs
Eigenvalues
Eigenvectors
(N, n_eigs)
Orthogonal basis
Δf
D
...graphs:
Metric learning on topological bases
Adaptive affinity estimation
(N, N)
Similarities
Δf
E
...layouts:
MAP MDE t-SNE TriMAP
PaCMAP NCVis
Graph layout optimization
Spectral initialization
(N, n_dim)
Coordinates

Figure 1 - Overview of TopOMetry workflow for single-cell data analysis. (A) Processed single-cell data from a given modality in the form of a data matrix composed of features per cell (i.e., gene expressions per cell) is fed to the algorithm. Its main advantages are looser assumptions regarding data topology than current standards, focusing on approximating the Laplace-Beltrami Operator (LBO), which can be done with different algorithms at each step. After learning latent topology (i.e., the extended Waddington's landscape), it can be visualized with any existing algorithms (t-SNE, MAP, PaCMAP, TriMAP, NCVis, and MDE are the current options, but more can be easily done externally due to TopOMetry modular design). **(B)** First, cell-cell similarity metrics are learned with algorithms that extract data topology; currently available algorithms are diffusion harmonics, continuous k-nearest-neighbors, and fuzzy simplicial sets. The advantage of these methods is adaptively estimating affinities based on neighborhood sparsity. It has been independently shown that these metrics approximate the LBO. **(C)** Next, spectral decompositions of these graph Laplacians are used to obtain a latent lower-dimensional embedding - the dataset intrinsic dimensionality can also be indirectly estimated by the number of non-negative eigenvalues (up to the 64-bit precision limit). **(D)** A second topological graph is obtained by applying the metrics used in **(B)** to the topological basis learned in **(C)** - note that different metrics can be combined. **(E)** The topological graph is used to obtain a spectral layout initialization, which is then optimized to match the graph by a layout optimization algorithm.

particular advantage of TopOMetry models is that they frequently have discrete eigenspectra: as the precision of eigenvalues and eigenvectors reach increasingly high values, the bit limit is frequently hit, which causes null or negative values to emerge as one reaches computational precision. This property and visualization of the eigenspectra can be used to obtain an indirect, empirical estimation of data intrinsic dimensionality (Fig. 1C). This is a significant advantage over current approaches in which users need to select a certain number of dimensions to use in downstream tasks. Next, a further refined topological graph approximates the LBO of this new orthogonal topological basis (Fig. 1D). A lower-dimensional initialization that preserves global structure[47] is then generated with a spectral decomposition of this topological graph. Finally, the layout can then be optimized using either an adapted version of UMAP (MAP - Manifold Approximation Projection) that minimizes the cross-entropy between the topological graph and the embedding or any other graph layout optimization algorithms; TopOMetry built-ins are tSNE, MAP, MDE, TriMAP, NCVis and PaCMAP (Fig. 1E). All steps were designed to approximate the LBO independently, except the graph layout optimization, which minimizes the differences between the learned topological graph and its lower-dimensional layout and should ideally be executed with multiple algorithms for visual comparisons. TopOMetry latent orthogonal bases can also be used in downstream analysis of SCD to alleviate the computational burden (instead of PCA) while providing denoised phenotypical signals, meaning they can be used to enhance the results of existing methods that rely on latent, denoised signals for clustering, RNA velocity estimation, batch correction, and multi-omics data integration.

In summary, by investigating data heterogeneity at multiple levels with sequential approximations of the LBO with topological metrics, graphs, and spectral layout initialization followed by optimization with state-of-the-art algorithms, TopOMetry wraps previous advances in DR into a unified framework focused on preserving data topology. These steps are all illustrated in Figure 1, and the differences to the current workflow are highlighted in Suppl. Fig.1. TopOMetry's single-cell companion, CellTOMetry, enables users to run and evaluate all possible models with a single line of code, being a powerful interface between this complex framework and the computational environment for single-cell analysis in python.

Identifying the non-linear and non-uniform topology of circulating T CD4 cells.

We first explored our algorithm in a famous real-world scRNAseq experiment: the *pbmc68k* dataset from 10X Genomics, which contains the transcriptome from ~68,000 peripheral blood mononuclear cells (PBMCs) from healthy human donors. *pbmc68k* is an exhaustively explored dataset, commonly used for teaching single-cell analysis and developing new tools, and its clustering results, PCA embeddings, and t-SNE and UMAP layouts are considered well known (Fig. 2A-D, G; Suppl. Fig. 2A-F). However, despite such prior knowledge, we show that TopOMetry models can identify over a hundred new T CD4 subtypes (Fig. 2E-F, h; Suppl. Fig. 2G-R) that remained hidden when using PCA, PCA-preprocessed UMAP, or only UMAP (Fig.

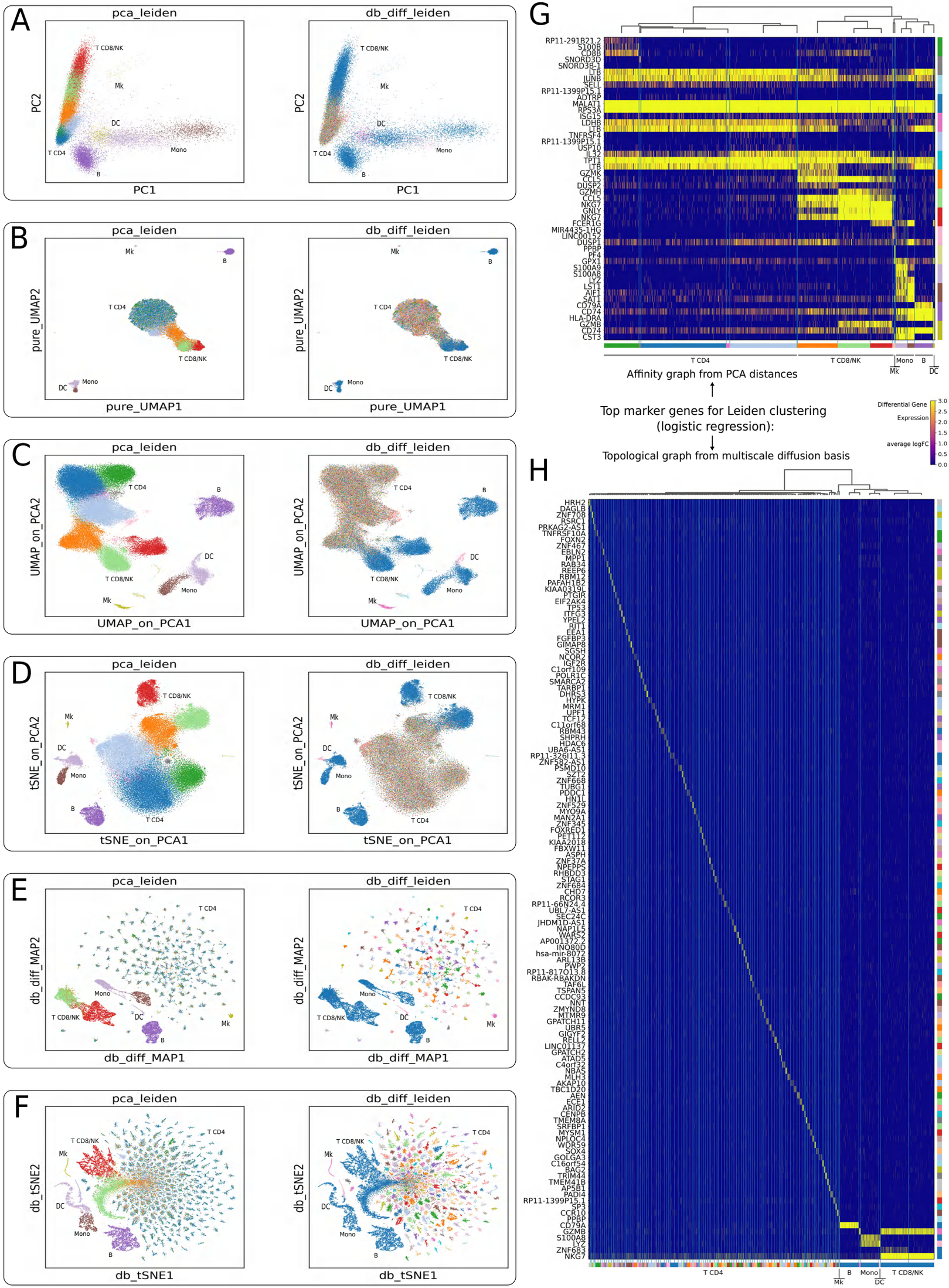

A
pca_leiden
db_diff_leiden
T CD8/NK
Mk
DC
T CD4
Mono
B
PC1
PC2
B
pure_UMAP1
pure_UMAP2
C
UMAP_on_PCA1
UMAP_on_PCA2
D
tSNE_on_PCA1
tSNE_on_PCA2
E
db_diff_MAP1
db_diff_MAP2
F
db_tSNE1
db_tSNE2
G
Affinity graph from PCA distances
Top marker genes for Leiden clustering
(logistic regression):
Topological graph from multiscale diffusion basis
Differential Gene
Expression
average logFC
H

Figure 2 - TopOMetry uncovers non-linear and non-uniform T CD4 diversity in peripheral blood mononuclear cells (PBMCs). Single-cell RNA sequencing (scRNAseq) data from 66,834 PBMCs from a single healthy donor (10X Genomics). Data were preprocessed according to the current standard workflow. Clustering was performed with the Leiden algorithm on either the graph obtained with fuzzy simplicial sets from the results of PCA (left column, upper heatmap) or the diffusion graph from the diffusion basis (right column, lower heatmap). Please note blue clusters in the right column are distinct clusters. **(A)** PCA embedding of the data (first two PCs). **(B)** UMAP embedding of the data without PCA priming (embedding was computed directly from data rather than from PCs). **(C)** UMAP embedding of the data with PCA priming. Note the appearance of artificial clusters not found with a pure UMAP analysis. **(D)** t-SNE embedding with PCA priming. **(E)** TopOMetry embedding using the diffusion basis (*db*), diffusion graph (*diff*) and MAP layout optimization (*db_diff_MAP*). **(F)** TopOMetry embedding using the diffusion basis (*db*) and tSNE for metric learning and layout optimization (*db_tSNE*). **(G)** Heatmap showing top 3 marker genes per clusters found with the standard Leiden clustering on the graph obtained with fuzzy simplicial sets from PCs (*pca_leiden*), shown in the left columns from **(A)** to **(F)**. **(H)** Heatmap showing top marker genes per clusters found with Leiden clustering on the diffusion graph from the diffusion basis (*db_diff_leiden*). Marker genes were found with the logistic regression algorithm. Notably, T CD4 clusters learned with TopOMetry are much more numerous and have a rather specific gene signature; conversely, T CD4 clusters learned with the standard analysis have non-specific marker genes.

2A-C), a feature we associate with the assumptions of data linearity and uniformity being broken for this system. Furthermore, we show that layout algorithms such as t-SNE, PHATE, PaCMAP, and TriMAP fail to reveal such heterogeneity when performed on PCs or directly on the data matrix (Suppl. Fig. 2A-F). Disturbingly, priming embeddings with PCA seems to introduce linear distortion, creating artificial clusters that do not have any specific marker genes (Fig 2B-C, G). However, these algorithms perform similarly in representing T CD4 heterogeneity when primed with either the diffusion or fuzzy orthogonal bases and their topological graphs (Suppl. Fig. 2G-R). Likewise, Leiden clustering on either four combinations of topological graphs shows similar results (Suppl. Fig. 2G-R), pointing towards the convergence of these two models to the LBO. As will be discussed further on, the current implementation of the continuous model is computationally inefficient for large datasets (>15,000 cells) and therefore was excluded from most analyses.

An empirical explanation for the distortion exhibited by current approaches is the lack of meaningful latent information extracted by PCA after a dozen or so dimensions (Suppl. Fig. 3A, B), a feature also related to data non-uniformity and non-linearity. Conversely, TopOMetry diffusion and fuzzy models lead to uncorrelated orthogonal components that keep extracting latent information until a numerical bit limit is reached at relatively similar number of dimensions (Suppl. Fig. 3A, D). Jointly, this points to topology non-linearity. Furthermore, computing layouts directly from data leads to severe underappreciation of the heterogeneity of T CD4 substates, which is also related to most algorithms' assumption of topology uniformity, as,

in this scenario, the linearity and uniformity assumptions are violated by a classic biological hallmark encoded within the data geometry: T CD4 diversity.

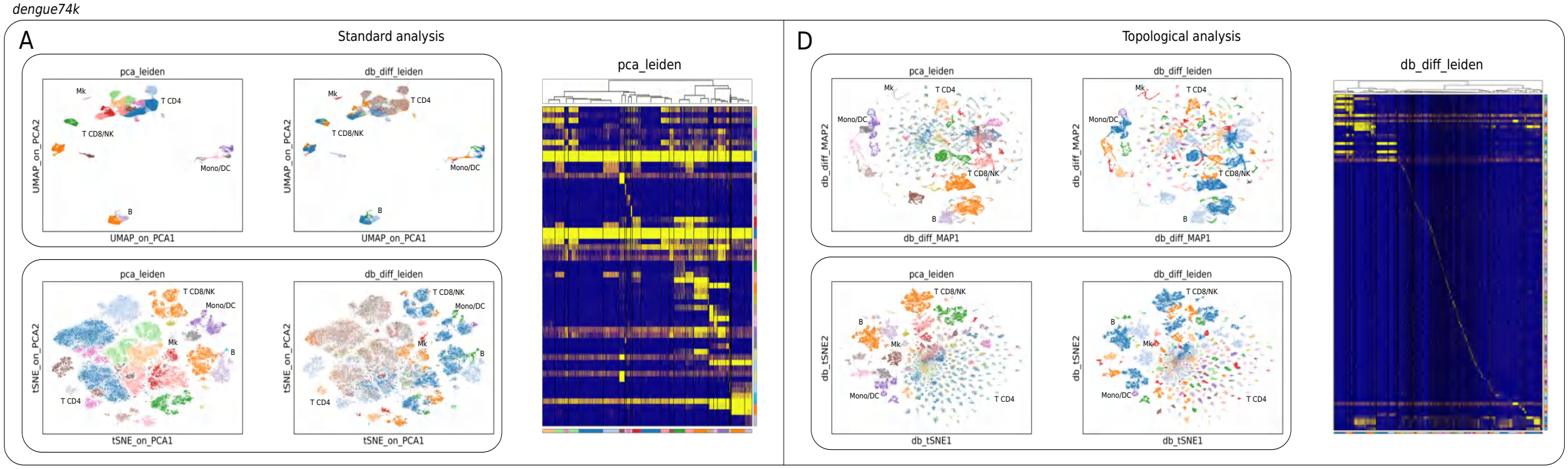
T CD4 cells are considered uniquely diverse and heterogeneous and are central for the maintenance of the immune repertoire and the regulation of the immune response. This prior-known ground-truth biological diversity is encoded in data by numerous sharp, non-uniform, and non-linear variations in data structure learned by TopOMetry and can be directly visualized as an intricate 'blooming tree' of cellular subpopulations. To assess if these new models effectively led to the identification of cell clusters defined by exclusive, specific marker genes, we harnessed the clusters found with the Leiden community detection algorithm from SCANPY default affinity graph (PCA-primed fuzzy simplicial sets) and with the four topological graphs. Clusters' marker genes were then surveyed with logistic regression[48]. Coincidently with the visualization and clustering results, we show that the standard PCA-derived approach fails to identify exclusive, specific marker genes across distinct T CD4 subpopulations (Fig. 2G, Suppl. Fig. 4A-B). In contrast, CellTOMetry successfully identifies around a hundred novel T CD4 subpopulations defined by very specific marker genes (Fig 2H, Suppl. Fig. 4A, C-F). Collectively, these observations provide first-of-its-kind evidence that phenotypic topology (i.e., SCD underlying structure representing biological information) is potentially non-linear and non-uniform - assuming otherwise *a priori* could represent a dangerously overseen bias in existing approaches. It also sheds light on a previously unappreciated degree of T CD4 cellular heterogeneity.

Circulating T CD4 diversity is present across diseases and in the cerebrospinal fluid

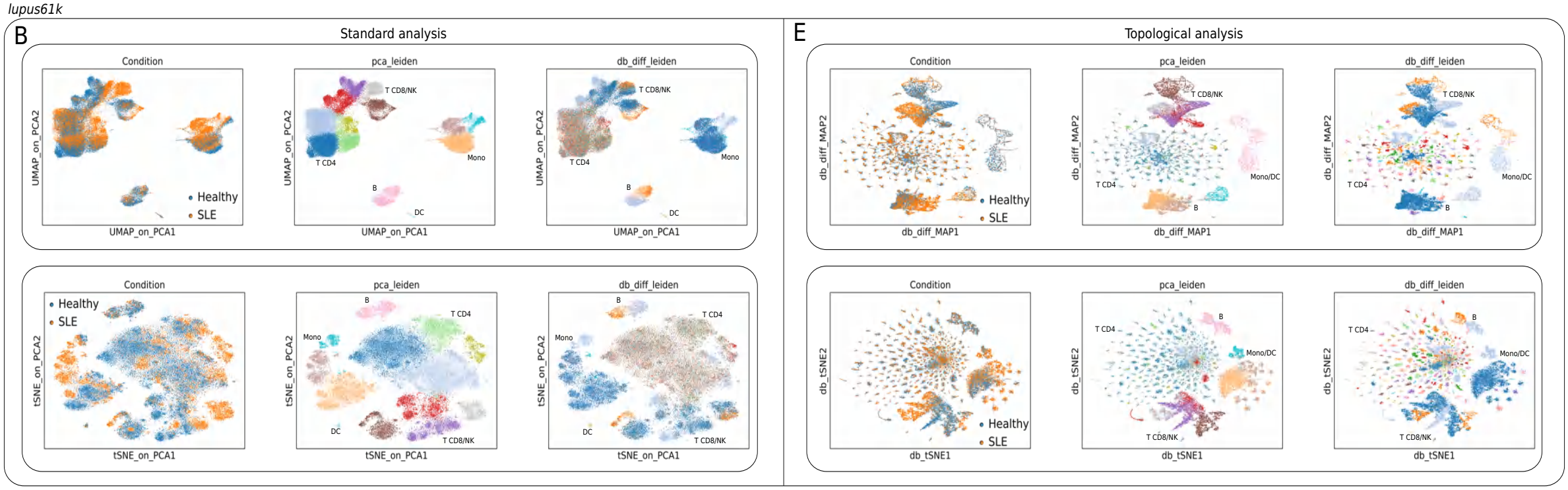
To evaluate whether the reported T CD4 diversity is reproducible and if such diversity could be detected in human disease, we harnessed three additional PBMC datasets from donors with 1) dengue-virus (DENV) infection (*dengue72k*)[49], 2) systemic lupus erythematosus (SLE; *lupus61k*)[50] and 3) multiple sclerosis (MS; *ms47k*)[51]. For the *lupus61k* and the *ms47k* datasets, data from healthy controls were also available, and data was available both for PBMCs and for mononuclear cells obtained from the cerebrospinal fluid (CSF) for the *ms47k* dataset. For all three datasets, we performed both the standard analysis (Figure 3A-C) and the topological analysis (Figure 3D-F) and show a similar degree of T CD4 diversity that can only be appreciated when using topological models, including in the CSF (Figure 3C, Figure 3F). Furthermore, diffusion and fuzzy models and all four downstream topological clustering results showed remarkably similar results across datasets (Suppl. Fig. 5, Suppl. Fig. 6, Suppl. Fig. 7). Additionally, we show that this new level of definition allows identifying highly-specific marker genes across models and datasets (Suppl. Fig. 8, Suppl. Fig. 9, Suppl. Fig 10).

Interestingly, for the *dengue72k* dataset, T CD4 marker genes corresponded to unique TCR genes, effectively demonstrating that individual clusters can be related to specific clonal lineages. To the best of our knowledge, this is the first analysis to identify T CD4 clones from gene expression data alone, without considering V(D)J sequences or TCR-sequencing data. These findings confirm a strikingly higher transcriptional diversity of circulating T CD4 cells than previously thought (approximately by an order of magnitude, with a total of approximately a hundred conserved T CD4 states across datasets). The similarity between the general structure

dengue74k



lupus61k



ms47k - T cells only

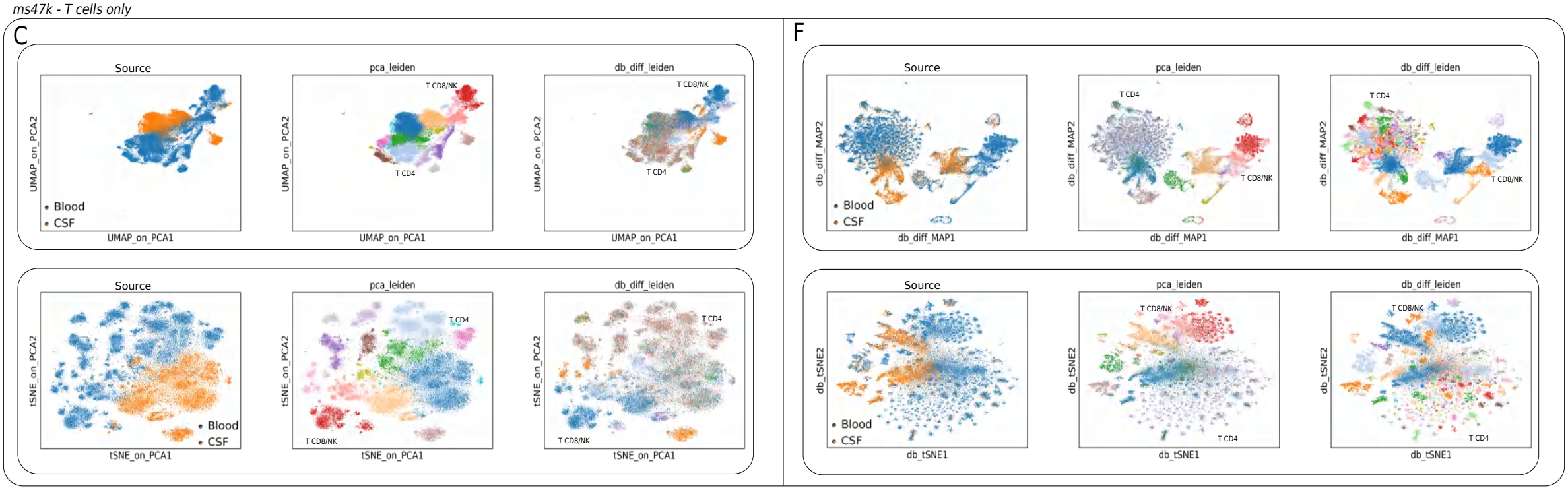


Figure 3 - T CD4 diversity is consistent across diseases and partly present in the CSF. Single-cell RNA sequencing (scRNAseq) data from the three discussed datasets: *dengue72k*, *lupus61k,* and *ms47k*. Data were preprocessed accordingly to the current standard workflow, and clustering was performed with the Leiden algorithm on either the graph obtained with fuzzy simplicial sets from the results of PCA (left columns, standard approach) or the diffusion graph from the diffusion basis (right columns, topological approach). *ms47k* was filtered to contain only clusters annotated as T cells prior to analysis. **(A-C)** default workflow for all three datasets with UMAP and t-SNE with PCA priming. **(D-F)** TopOMetry embeddings using the diffusion basis, diffusion graph, and t-SNE and UMAP for layout. **(A and D):** standard (A) and topological analysis (D) of the *dengue72k* dataset. **(B and E):** standard (B) and topological analysis (E) of the *lupus61k* dataset. **(C and F**): standard (C) and topological analysis (F). Heatmaps were only included for the *dengue72k* dataset in **A and D** for illustration; all heatmaps from clustering results are available as Supplementary Figures.

results between models and datasets is remarkable, and suggests that the most if not all T CD4 clusters identified are clonal lineages. Additionally, we show that such striking diversity is also present in the cerebrospinal fluid of healthy donors and MS patients. These results highlight the significance of T CD4 transcriptional heterogeneity to homeostasis and disease, and critically exemplify the limitations imposed by the currently held assumptions.

Phenotypic topology representations enable high-resolution trajectory inference and visualization of the cell cycle.

To further explore representations of ground-truth biology within single-cell data, we harnessed ungated bone marrow scRNAseq data from the Human Cell Atlas project, in which both PBMCs and hematopoietic stem cells (HSC) are classically described (Fig. 4A-C). When exploring these data with *CellTOMetry*, a similarly unreported transcriptional heterogeneity of T CD4+ cells was found (Fig. 4D-F), holding a strong graphic similarity to those observed in the *pbmc68k, dengue72k, sle61k,* and *ms47k* datasets across models (Suppl. Fig. 11). Consistently with these results, we found that the topological models could identify significantly more T CD4 and HSC subtypes than the standard approach (Fig. 4G-H). Similar to previous results, TopOMetry led to the identification of a more specific gene signature for discrete T CD4 subpopulations across models (Suppl. Fig. 12), a feature exemplified with several marker genes (Suppl. Fig. 13).

Interestingly, the topological approach was able to map clusters expressing MKI67 (a mitotic marker) and CD34 (an HSC marker) to closed-loop trajectories that could arguably represent the

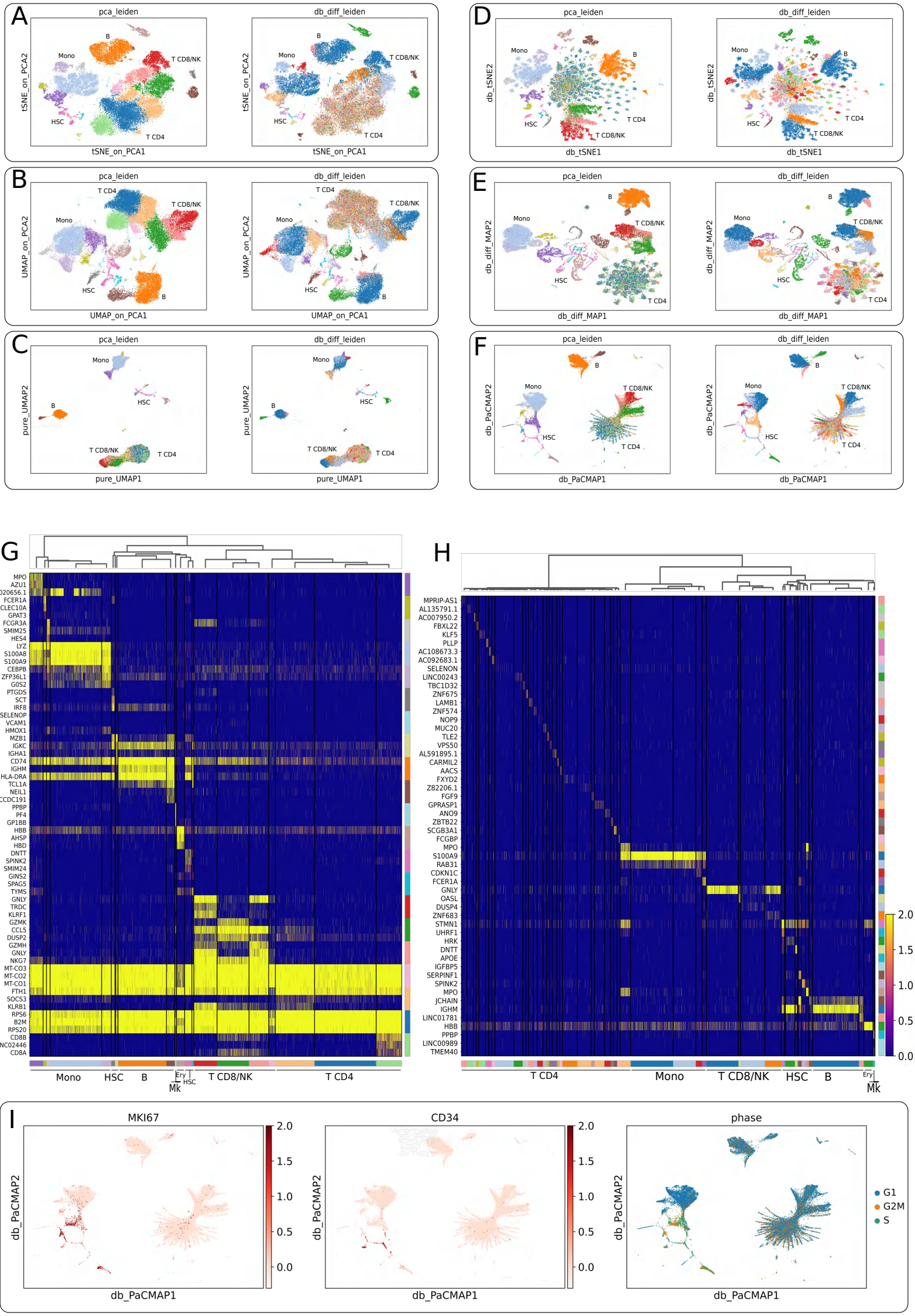

A
pca_leiden
db_diff_leiden
tSNE_on_PCA2
tSNE_on_PCA1
Mono
B
T CD8/NK
HSC
T CD4
B
UMAP_on_PCA2
UMAP_on_PCA1
C
pure_UMAP2
pure_UMAP1
D
db_tSNE2
db_tSNE1
E
db_diff_MAP2
db_diff_MAP1
F
db_PaCMAP2
db_PaCMAP1
G
MPO
AZU1
AC020656.1
FCER1A
CLEC10A
GPAT3
FCGR3A
SMIM25
HES4
LYZ
S100A8
S100A9
CEBPB
ZFP36L1
G0S2
PTGDS
SCT
IRF8
SELENOP
VCAM1
HMOX1
MZB1
IGKC
IGHA1
CD74
IGHM
HLA-DRA
TCL1A
NEIL1
CCDC191
PPBP
PF4
GP1BB
HBB
AHSP
HBD
DNTT
SPINK2
SMIM24
GINS2
SPAG5
TYMS
GNLY
TRDC
KLRF1
GZMK
CCL5
DUSP2
GZMH
GNLY
NKG7
MT-CO3
MT-CO2
MT-CO1
FTH1
SOCS3
KLRB1
RPS6
B2M
RPS20
CD8B
LINC02446
CD8A
Mono
HSC
B
Ery
HSC
Mk
T CD8/NK
T CD4
H
MPRIP-AS1
AL135791.1
AC007950.2
FBXL22
KLF5
PLLP
AC108673.3
AC092683.1
SELENON
LINC00243
TBC1D32
ZNF675
LAMB1
ZNF574
NOP9
MUC20
TLE2
VP550
AL591895.1
CARMIL2
AACS
FXYD2
Z82206.1
FGF9
GPRASP1
ANO9
ZBTB22
SCGB3A1
FCGBP
MPO
S100A9
RAB31
CDKN1C
FCER1A
GNLY
OASL
DUSP4
ZNF683
STMN1
UHRF1
HRK
DNTT
APOE
IGFBP5
SERPINF1
SPINK2
MPO
JCHAIN
IGHM
LINC01781
HBB
PPBP
LINC00989
TMEM40
T CD4
Mono
T CD8/NK
HSC
B
Ery
Mk
2.0
1.5
1.0
0.5
0.0
I
MKI67
CD34
phase
db_PaCMAP2
db_PaCMAP1
G1
G2M
S

Figure 4 - T CD4 diversity and cell cycle topology in the human bone marrow. Single-cell RNA sequencing (scRNAseq) data from the human bone marrow was retrieved from the Human Cell Atlas. Data were preprocessed accordingly to the current standard workflow, and clustering was performed with the Leiden algorithm on either the graph obtained with fuzzy simplicial sets from the results of PCA **(A-C, G)** or the diffusion graph from the diffusion basis **(D-F, H)**. **(A)** PCA-primed t-SNE. **(B)** PCA-primed UMAP. **(C)** Pure UMAP representation. **(D)** t-SNE on the diffusion basis. **(E)** MAP of the diffusion graph learned from the diffusion basis. **(F)** PaCMAP on the diffusion basis. **(G)** Heatmap of top 3 marker genes per cluster found with the standard PCA-primed workflow. **(H)** Heatmap of top marker genes per cluster found with the diffusion basis and diffusion graph clustering. **(I)** PaCMAP on the diffusion basis representations showing the expression of *MKI67* (left, a mitosis marker) and *CD34* (middle, an HSC marker). The predicted cell cycle phase is also shown (right).

structure of the cell cycle itself (Fig. 4I). To validate this hypothesis, we scored cells along the cell cycle using a standard reference[52] to show that variations in a cell's learned position within the cell cycle correspond to smooth topological transitions in the detected loops and trajectories (Fig. 4I). These results agree with recent work on cell-cycle topology[53,54]; with canonical[55] views on the very concept of cellular types (i.e., topology discreteness), identities (i.e., topological branching), and states (i.e., topological neighborhoods); and with the proposed unifying concept of phenotypic topology. By noticing known biological hallmarks are missing from standard embeddings and yet present in some topological layouts, we argue that TopOMetry models and layouts effectively recover phenotypic topology. This association between canonical gene signatures, dynamic biological processes, and the represented phenotypic topology is fundamental during the inspection of results and their interpretation to generate new hypotheses.

TopOMetry scales to whole-organism data and reveals fine-grained developmental hierarchies during mouse organogenesis.

The introduction of new assays that can profile hundreds of thousands of single-cells drives future analysis to ever-growing datasets, often comprising whole organisms and millions of cells - thus, proposed algorithms must be scalable to such massive information. To evaluate TopOMetry scalability to large data and its usefulness to represent these as detailed atlases of cellular hierarchies, we harnessed 1.3 million high-quality pre-filtered single-cell transcriptomes from the mouse embryo during organogenesis[56]. As previously described, standard data analysis

was performed with PCA and UMAP (Fig. 5A-C), and topological analysis was performed with the diffusion basis, UMAP on the diffusion basis (Fig. 5D), and the diffusion (Fig. 5E) and fuzzy (Fig. 5F) topological graphs. Similar to the previous examples, the topological analysis revealed extensive additional fine-grained detail of cellular hierarchy than the default PCA-preprocessed UMAP. Furthermore, by applying the Leiden clustering algorithm with default parameters to the diffusion and fuzzy topological graphs, we found approximately 380 groups of subpopulations, while the default approach (clustering of a fuzzy simplicial graph on 300 PCs) led to multiple discrete subpopulations and lineages being merged into the same clusters - in this context, the originally 56 described subclusters are only rough approximations of the major cell types and insensitive to the revealed fine-grained hierarchy. Interestingly, the clusters found with the topological analysis corresponded to apparent finer-grained structure on the standard PCA-primed embedding (Fig. 5B-C), pointing that even UMAP attempts to find such latent structure, despite its limitations. Hence, it is shown that TopOMetry excels at faithfully representing vast datasets comprising hundreds of coexisting lineages, being ideal for representing whole-organism data and performing lineage inference for its very reliance on the concept of phenotypic topology.

When considering computation time, TopOMetry models perform very differently. When considering up to 15,000 cells, the continuous model (using CkNN) emerges as the single most inefficient approach (Suppl. Fig. 14A) - for this reason, it was excluded from most analyses presented in this manuscript. When analyzing a number of cells more similar to those present in most studies (between 10,000 and 50,000), the fuzzy model emerges as more expensive than the

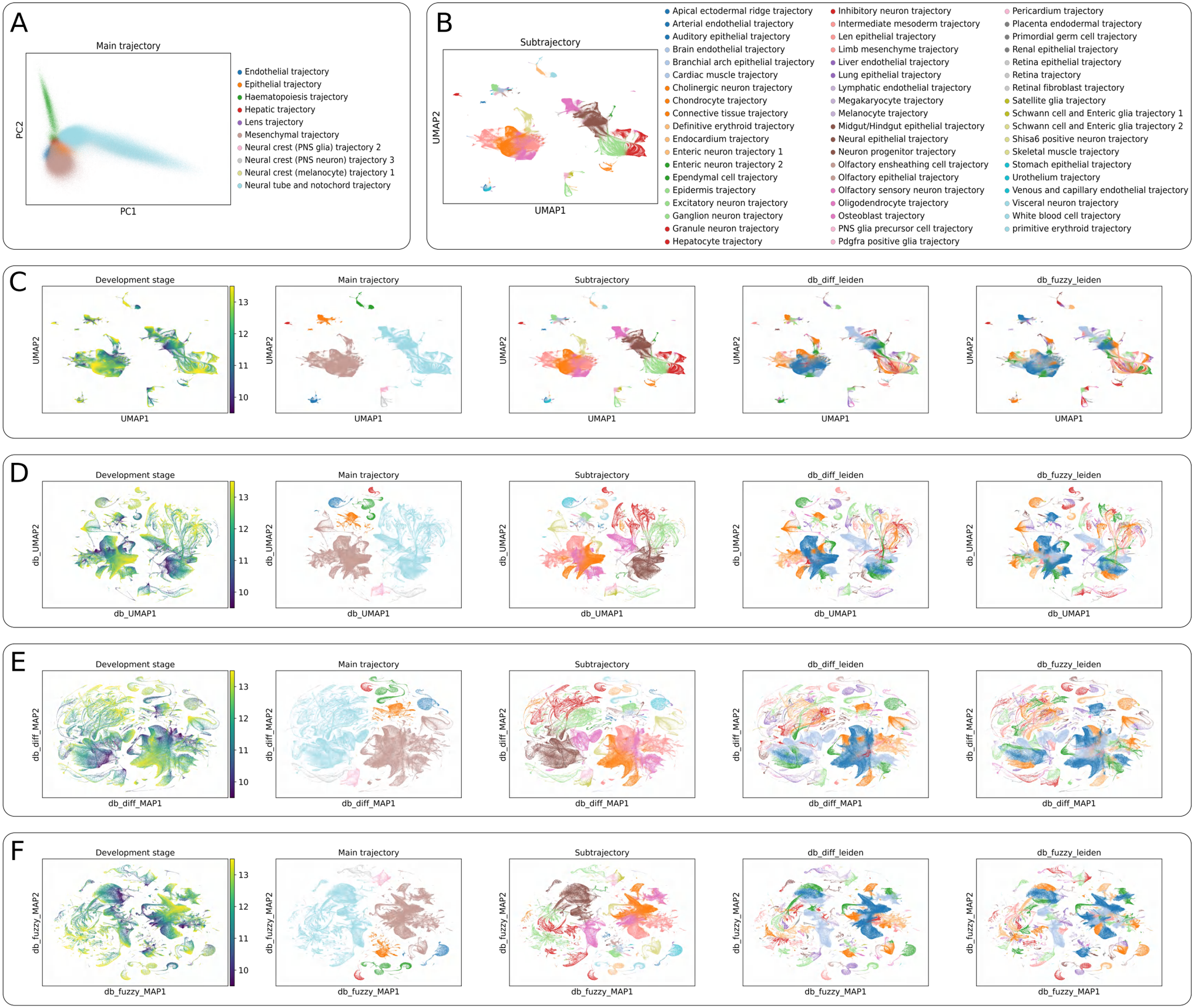

A
Main trajectory
PC2
PC1
Endothelial trajectory
Epithelial trajectory
Haematopoiesis trajectory
Hepatic trajectory
Lens trajectory
Mesenchymal trajectory
Neural crest (PNS glia) trajectory 2
Neural crest (PNS neuron) trajectory 3
Neural crest (melanocyte) trajectory 1
Neural tube and notochord trajectory
B
Subtrajectory
UMAP2
UMAP1
Apical ectodermal ridge trajectory
Arterial endothelial trajectory
Auditory epithelial trajectory
Brain endothelial trajectory
Branchial arch epithelial trajectory
Cardiac muscle trajectory
Cholinergic neuron trajectory
Chondrocyte trajectory
Connective tissue trajectory
Definitive erythroid trajectory
Endocardium trajectory
Enteric neuron trajectory 1
Enteric neuron trajectory 2
Ependymal cell trajectory
Epidermis trajectory
Excitatory neuron trajectory
Ganglion neuron trajectory
Granule neuron trajectory
Hepatocyte trajectory
Inhibitory neuron trajectory
Intermediate mesoderm trajectory
Len epithelial trajectory
Limb mesenchyme trajectory
Liver endothelial trajectory
Lung epithelial trajectory
Lymphatic endothelial trajectory
Megakaryocyte trajectory
Melanocyte trajectory
Midgut/Hindgut epithelial trajectory
Neural epithelial trajectory
Neuron progenitor trajectory
Olfactory ensheathing cell trajectory
Olfactory epithelial trajectory
Olfactory sensory neuron trajectory
Oligodendrocyte trajectory
Osteoblast trajectory
PNS glia precursor cell trajectory
Pdgfra positive glia trajectory
Pericardium trajectory
Placenta endodermal trajectory
Primordial germ cell trajectory
Renal epithelial trajectory
Retina epithelial trajectory
Retina trajectory
Retinal fibroblast trajectory
Satellite glia trajectory
Schwann cell and Enteric glia trajectory 1
Schwann cell and Enteric glia trajectory 2
Shisa6 positive neuron trajectory
Skeletal muscle trajectory
Stomach epithelial trajectory
Urothelium trajectory
Venous and capillary endothelial trajectory
Visceral neuron trajectory
White blood cell trajectory
primitive erythroid trajectory
C
Development stage
Main trajectory
Subtrajectory
db_diff_leiden
db_fuzzy_leiden
UMAP2
UMAP1
13
12
11
10
D
db_UMAP2
db_UMAP1
E
db_diff_MAP2
db_diff_MAP1
F
db_fuzzy_MAP2
db_fuzzy_MAP1

Figure 5 - Uncovering the fine-grained phenotypic topology of whole-organism data. scRNAseq data from 1.3 million cells of the mouse organogenesis (MOCA) was collected during five time-points between 9.5 and 13.5 days of gestation. Annotated labels are the same as the original publication. **(A)** PCA embedding of MOCA data, colored by main trajectories. **(B)** UMAP embedding of the first 300 PCs of the MOCA data, colored by subtrajectories. **(C)** PCA-primed UMAP; and three options of topological analysis: **(D)** UMAP on the diffusion basis, **(E)** MAP of the diffusion graph learned from the diffusion basis, and MAP of the fuzzy graph learned from the diffusion basis. From left to right, cells are colored by i) the development stage during thy were sampled, ii) main trajectories and iii) subtrajectories from the original MOCA publication, iv) results from the Leiden community detection algorithm on either the v) fuzzy or vi) diffusion graphs learned from the diffusion basis.

diffusion model, which performs similarly to state-of-the-art PCA and SVD (Single-Value Decomposition) implementations (Suppl. Fig. 14B). These results elect the diffusion model as the most computationally efficient TopOMetry approach for approximating the LBO. It is also shown that the diffusion model computation time escalates linearly with the number of analyzed samples (cells), even when considering up to a million cells (Suppl. Fig. 14C). It still performs slower than PCA and SVD, but this was largely expected, given the higher complexity of operating on large kNN graphs than performing simple linear operations. We note that this difference is primarily due to the computation of multiple kNN graphs, thus dependent on the performance of nearest-neighbors (NN) algorithms, a prolific subfield of study[57]. The presented version of TopOMetry uses Hierarchical Navigable Small Worlds (HNSW)[58,59] , a current top performer on speed and accuracy among approximate NN methods, and also offers a flexible and user-friendly module for its use and prediction of its accuracy. Despite the accounted limitations, we note TopOMetry runtime and memory usage was amenable to commercial laptops when analyzing up to ~200,000 cells, thus being accessible to most researchers.

Quantitative assessment of embedding quality.

Objectively evaluating the performance of DR algorithms is a problem of its own due to the inevitable need for assumptions on underlying data structure and distribution (i.e., what is the reference if the data structure is unknown?). Nevertheless, guided by recent studies[28,46], we used general quantitative metrics to assess the preservation of global and local structures (Fig. 6A). To evaluate the preservation of the global structure, we propose scoring bases and layouts against

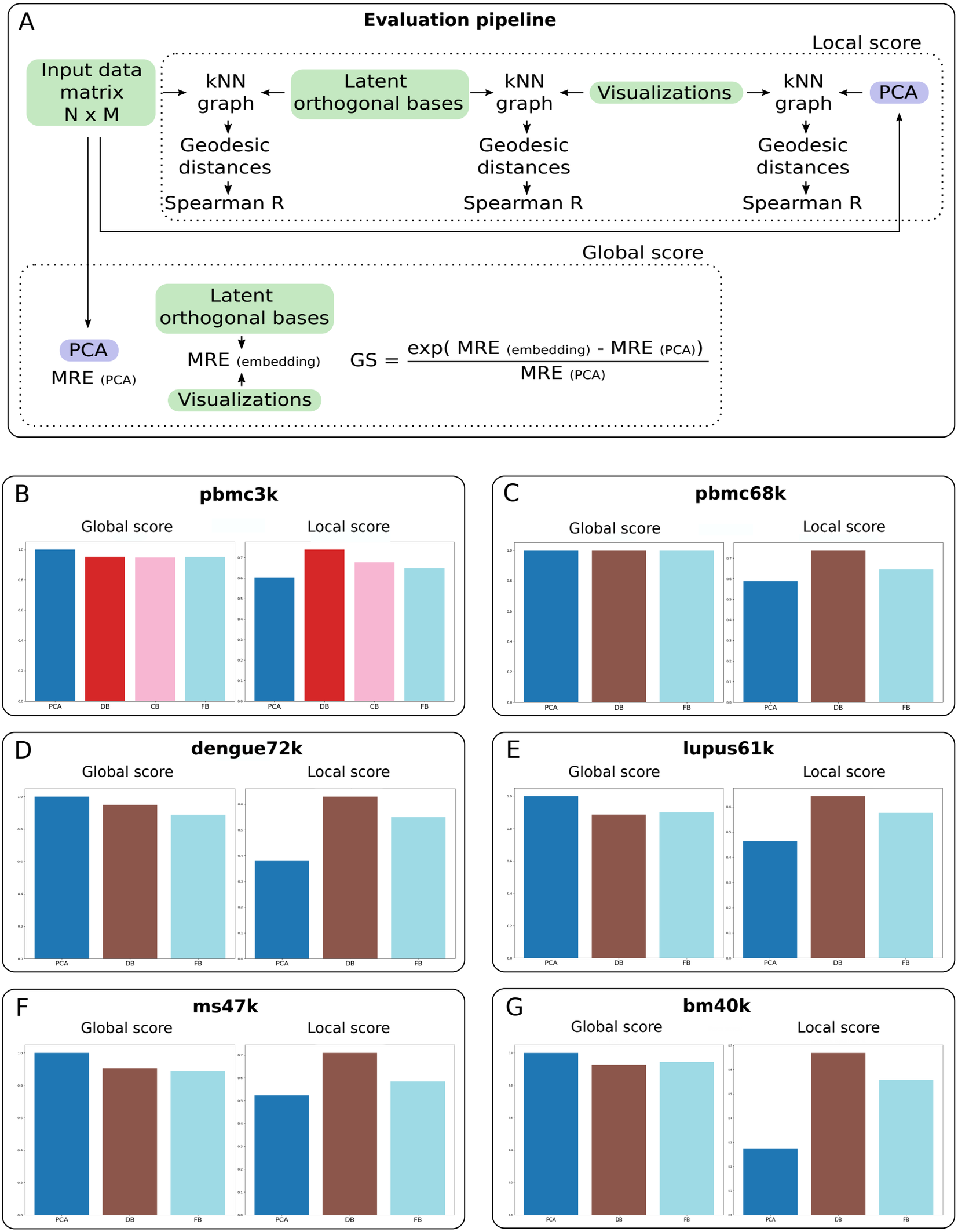

A
Evaluation pipeline
Local score
Input data matrix N x M
kNN graph
Latent orthogonal bases
kNN graph
Visualizations
kNN graph
PCA
Geodesic distances
Geodesic distances
Geodesic distances
Spearman R
Spearman R
Spearman R
Global score
Latent orthogonal bases
PCA
MRE (PCA)
MRE (embedding)
Visualizations
GS = exp(MRE (embedding) - MRE (PCA)) / MRE (PCA)
B
pbmc3k
Global score
Local score
PCA
DB
CB
FB
C
pbmc68k
Global score
Local score
PCA
DB
FB
D
dengue72k
Global score
Local score
PCA
DB
FB
E
lupus61k
Global score
Local score
PCA
DB
FB
F
ms47k
Global score
Local score
PCA
DB
FB
G
bm40k
Global score
Local score
PCA
DB
FB

Figure 6 - Quantitative assessment of embedding quality. (A) Overview of the evaluation pipeline. For the local score (upper), neighborhood graphs are built from the original high-dimensional data matrix and the orthogonal bases. The Spearman correlation between geodesic distances in the high- and low-dimensional spaces is then used to evaluate local structure preservation. Visualizations were scored against their input (topological bases, PCA) for fair comparisons. For the global score (lower), visualizations and orthogonal bases are scored against PCA's minimum reconstruction error (MRE). **(B-G)** Evaluation results for PCA and the three TopOMetry models for learning latent orthogonal bases for the 10X Genomics *pbmc3k* dataset and remaining datasets used in the manuscript. MOCA was not used for evaluation due to its large size, which made the evaluation computations unfeasible.

PCA, based on previous work[28], as it intrinsically preserves most of the global structure by design. To evaluate local structure preservation, we designed a score based on Spearman's correlations between the pairwise geodesic distances on the high-dimensional ambient space and the low-dimensional latent orthogonal bases and layouts. This scoring system is independent of TopOMetry models. It is important to note that, in this system, scores are computed using the input data for each algorithm (e.g., PCs for standard layouts such as UMAP, the diffusion basis for diffusion-based layouts) to isolate the preservation of the local structure of the orthogonal basis from that obtained by the layout technique (Fig. 6A). It is also important to note that the computations involved in evaluating the results of DR are far more expensive than performing DR itself. The evaluation pipeline included in TopOMetry demands less memory, but is only slightly faster than current methods. Computing scores for MOCA, for instance, was unfeasible in our computational settings.

We applied this scoring system to all exposed datasets, except for the MOCA dataset. Regarding the global score, we found that all orthogonal bases preserve nearly all global structure present in the high-dimensional data (Fig. 6B-G) and that all layout methods also perform similarly given a PCA or spectral initialization (Suppl. Fig. 15). The main differences between methods were related to changes in the local score, for which the diffusion basis was the top performer, and PCA was the worst method across all datasets (the fuzzy and continuous basis, when computed, varied). This was expected given the locality-preserving nature of TopOMetry's models and quantitatively confirms their superiority to the standard PCA approach. The graphs obtained from these bases were also evaluated for local structure preservation (Suppl.

Fig. 16), and these scores are useful when choosing which clustering results to trust most when they do not converge.

Regarding the visualization (layout) algorithms, we found MAP, t-SNE, and PaCMAP to be the top performers when used with topological models, and MDE and TriMAP to be the best alternatives when used with PCA, although this varied across datasets (Suppl. Fig. 15). The high variability between the performance of different layout algorithms across datasets and orthogonal bases highlights the need to obtain, inspect and compare multiple visualization results before interpretation and hypothesis generation. This is opposite to the current practice in single-cell analysis, during which a single orthogonal basis is used (PCA), and then only one or two methods are used for layout - usually t-SNE or UMAP, thus reinforcing the need to carefully reassess currently held assumptions and standards.

Discussion

Here, we employed existing state-of-the-art algorithms to construct a framework that efficiently extracts meaningful information from data through the approximation of the Laplace-Beltrami Operator (LBO). Despite its known power to describe high-dimensional data, the approximation of LBO has remained unexplored in single-cell data analysis. We show that by encoding data's phenotypic topology through different metrics, bases, and graphs through sequential LBO approximations, it is possible to identify several more cellular subtypes in single-cell data that are defined by the selective expression of marker genes, as seen in the previously undescribed and rather prodigious T CD4 transcriptional heterogeneity we herein

report. Furthermore, LBO's geometrical properties are harnessed throughout the exposed examples to represent known biological hallmarks faithfully and intuitively into graphic topological structures, such as the cell cycle as a closed-loop. The underlying geometry of such biological hallmarks is referred to as the *phenotypic topology* of single-cells.

From a knowledge representation perspective, these remarkable results point to the qualitative superiority of dissecting the steps of dimensional reduction (i.e., metric and graph learning, layout optimization) to approximate the LBO at each step. We acknowledge that only a fraction of the employed theory is novel - yet, by standing on the shoulders of researchers who developed earlier methods, we show that a carefully designed framework that aims to obtain sequential approximations of the LBO can result in striking new biological insights. The fact that visual representations carry properties from the graph layout optimization algorithms is also a notable feature and an argument contrary to the consistent superiority of a given optimization method over others and to the choice of a single method throughout all analysis (i.e., UMAP vs. tSNE) - instead, the choice of which similarity metrics, orthogonal bases, and neighborhood graphs to feed these optimization methods with appears to be far more critical. For example, PCA seemingly introduces linear distortions in data throughout the analyses, which can be problematic as most available toolkits perform it by default. Additionally, PCA also fails to denoise biological signals as linearly uncorrelated latent dimensions, leading to poorly-performing neighborhood graphs compared to the LBO-approximating orthogonal bases and graphs, which preserve most of the global and significantly more of the local information in high-dimensional spaces. By allowing control of each step of DR, TopOMetry empowers

scientists to obtain biological insights from the visualization of common patterns and structures across different models and visualizations. We hope that these insights may prove helpful in the design of future machine-learning frameworks.

Our results also impact the life sciences as a whole. A fundamental definition in biology is what ***is*** a cell type. By introducing the concept of phenotypic topology, we tie the concepts of cell types, identities, and states to differential geometry operators that mathematically describe discontinuous, contiguous, or fine-grained cellular populations in the form of latent topological signals across a set of phenotypic manifolds. This idea makes the task of trajectory inference (TI) as simple as walking through topology-oriented graphs and embeddings, as had been widely suggested. Even though the graph-walking concept is in accordance with current TI algorithms and reported trajectories, most approaches rely on PCA and UMAP representations to generate results. Thus, we point out that these could also be further enhanced by using topological representations. It is also noticeable that most TI methods struggle with the coexistence of discrete, continuous, tree-shaped, and cyclic topologies within the same dataset and often hold strong assumptions on the data structure (e.g., Monocle2[60] assumes data is entirely tree-shaped). The contribution of the proposed models to clustering is even more direct and leads to the efficient identification of cell types, developing identities, and states. Although not yet implemented, combinations of TopOMetry with virtually any batch-correction methods are possible, as most techniques use PCA as an orthogonal basis. The same is true for RNA velocity and a wide array of currently implemented bioinformatics tools.

In contrast with the power of these concepts, we acknowledge that the depth of exploration of the biological conclusions we report is limited by the sheer complexity of each biological hallmark represented in the examples. T CD4 diversity, for instance, is a core concept in immunology, and systematic, comprehensive annotation of each of the hundreds of novel cellular states in health and disease lies well beyond the scope and length of this manuscript. The gene expression dynamics that determine the molecular organization of the cell cycle are also profound and are only beginning to be understood. Similarly, the systematic annotation of cellular differentiation trajectories during mouse organogenesis is a daunting task that only community effort may accomplish. Despite their uniqueness, these phenomena have the shared characteristic of embedding the cellular identity topological landscape to specific graphic shapes related to their underlying mechanisms. By providing open-source and user-friendly software, we hope to enable the community to collectively obtain further insights regarding the phenotypic topology of these and other biological phenomena.

Methods

Data acquisition, availability, and processing

pbmc68k

Single-cell RNA sequencing (scRNAseq) data from peripheral blood mononuclear cells (PBMCs) was obtained from https://www.10xgenomics.com/resources/datasets/. After filtering, 66,834 cells were retained, and the top 2,318 highly variable genes were selected for downstream analysis. The selection of highly variable genes was performed with default parameters. Data were processed and analyzed accordingly to the standard workflow described in the subsection 'Standard data processing and analysis'. TopOMetry was executed with the parameter *n_eigs* set to 150, and the remaining parameters were set to defaults.

dengue72k

scRNAseq data from PBMCs from two dengue-infected subjects were obtained from a previously published study[49]. After filtering, 72,412 cells were retained, and the top 1,924 highly variable genes were selected for downstream analysis. The selection of highly variable genes was performed with default parameters. Data were processed and analyzed according to the standard workflow described further in the 'Standard data processing and analysis' subsection. TopOMetry was executed with the parameter *n_eigs* set to 300, and the remaining parameters were set to defaults.

lupus61k

scRNAseq data from PBMCs from five healthy donors and six SLE subjects was obtained from a published study[50]. After filtering, 61,472 cells were retained, and the top 2,058 highly variable genes were selected for downstream analysis. The selection of highly variable genes was performed with default parameters. Data were processed and analyzed according to the standard workflow described further in the 'Standard data processing and analysis' subsection. TopOMetry was executed with the parameter *n_eigs* set to 150, and the remaining parameters were set to defaults.

ms47k

scRNAseq data from PBMCs and mononuclear cells isolated from the cerebrospinal fluid from six donors with idiopathic intracranial hypertension (IIH) who were otherwise healthy and therefore used as controls and six multiple sclerosis subjects were obtained from a previously published study[51]. Before preprocessing, cells were filtered based on the labels assigned by the authors of the original study: clusters labeled as invalid were removed, and data was filtered to include only cells originally labeled as T cells. Such selection was performed in this dataset to provide a greater focus on T cells. After filtering, 47,018 cells were retained, and the top 2,526 highly variable genes were selected for downstream analysis. The selection of highly variable genes was performed with default parameters. Data were processed and analyzed accordingly to the standard workflow described in the subsection 'Standard data processing and analysis'.

TopOMetry was executed with the parameter *n_eigs* set to 200, and the remaining parameters were set to defaults.

bm40k

A preprocessed and subsampled dataset containing 48,219 cells from the bone marrow of human donors was obtained from the Immune Cell Atlas of the Human Cell Atlas through the Pegasus analysis toolkit[61]. After conversion to SCANPY and filtering, 41,803 cells were retained, and 3,464 genes were selected for downstream analysis. The selection of highly variable genes was performed with default parameters. Data were processed and analyzed according to the standard workflow described further in the 'Standard data processing and analysis' subsection. TopOMetry was executed with the parameter *n_eigs* set to 150, and the remaining parameters were set to defaults. Seventeen singletons were excluded from the analysis after topological clustering to simplify visualizations.

MOCA

A filtered gene expression matrix containing 1,331,984 cells and 26,183 genes and associated metadata from the mouse organogenesis cell atlas (MOCA) were obtained from https://oncoscape.v3.sttrcancer.org/atlas.gs.washington.edu.mouse.rna/downloads. Then, data were processed and analyzed accordingly to the standard workflow described further in the subsection 'Standard data processing and analysis', and the top 2,358 highly variable genes were selected for downstream analysis. The standard workflow was performed with the following

modifications: 1) the number of PCs was set to 300, and UMAP parameters were customized (*min_dist* was set to 0.3, *spread* was set to 2, *alpha* was set to 2 and *maxiter* was set to 1000). This choice of hyperparameters tends to make UMAP represent more details and spread clusters more evenly across the embedding when analyzing a large dataset such as this. Similarly, for fairness of comparison, TopOMetry was executed with the following modifications: 1) *n_eigs* was set to 600; 2) *base_knn* and *graph_knn* were both set to 10; and 3) the hyperparameters of MAP layout optimization were modified to exactly match the UMAP hyperparameters used in the modified standard workflow (*min_dist* was set to 0.3, *spread* was set to 2, *alpha* was set to 2 and *maxiter* was set to 1000).

Standard data analysis

Count matrices were used to create AnnData objects within SCANPY[62], which were filtered to recommended thresholds. Data were then processed with the following workflow: 1) first, data were normalized and variance stabilized by log normalization; 2) next, highly variable genes (HVG) were identified and kept for downstream analysis; 3) HVG-data was then centered and scaled and; 4) dimensionality reduced to 100 dimensions with PCA; 5) neighborhoods were computed using UMAP fuzzy simplicial sets as per SCANPY's defaults; 6) the neighborhood graph was projected with UMAP and t-SNE; 7) the neighborhood graph was clustered with the Leiden community detection algorithm. These steps correspond to the workflow applied in most manuscripts in which single-cell data is analyzed, being the norm in the field and regarded as the

current gold-standard[63]. Thus, we refer to these as the standard workflow throughout the manuscript. Unless otherwise stated, this was performed with default parameters within the SCANPY toolkit for single-cell analysis.

Cluster marker genes were found with logistic regression[48] by using *scanpy's* built-in functions. The maximum number of iterations was increased until convergence, i.e., until error messages claiming results did not converge with the maximum number of iterations stopped being triggered. Anecdotally, we noticed that the number of iterations required for convergence increased with the number of cells and detected clusters, which is rather intuitive. The maximum number of iterations used for each dataset is explicitly defined in our analysis code that can be freely reproduced.

Topologically Optimized geoMetry

Fundamentally, TopOMetry is based on the discrete approximation of the Laplace-Beltrami Operator (LBO), formally defined as the divergence of the gradient:

$$\triangle f = \nabla \cdot \nabla f$$

The LBO is a powerful descriptor of data geometry and has been approximated by some previous approaches, namely Diffusion Maps (DM), Laplacian Eigenmaps (LE), continuous-k-nearest-neighbors (CkNN), and fuzzy simplicial sets used for metric learning in UMAP. TopOMetry explores the power of each algorithm by combining them into a comprehensive framework that iteratively approximates the LBO during metric learning and affinity estimation, decomposition of a lower-dimensional basis, and building a new graph from

such a basis. Layout optimization algorithms do not intrinsically harness the ability to approximate the LBO, but they can be offered a LE initialization, which powers them with such information. Likewise, most algorithms operate on iteratively constructing a layout that minimizes some differences to the informed high-dimensional graph, corresponding to an LBO approximation in TopOMetry's models. It is of note that this carries the manifold assumption, as on a Riemannian manifold, the LBO entirely determines the Riemannian metric - and therefore, all topological properties of the manifold.

In an overview, TopOMetry dissects the problem of graph-based dimensionality reduction (DR) into four steps, aiming to approximate the LBO at each step. The first consists of learning k-nearest-neighbor (kNN) similarities from distances (i.e., metric learning) and can be performed with diffusion harmonics (DH; customizable between multiple kernels), CkNN, and fuzzy simplicial sets (FSS), rendering graphs represented by symmetric similarities matrices. The second is to denoise this learned graph to extract latent information through weighted decompositions with DM and LE's of CkNN and fuzzy graphs. This approach often can be used to indirectly estimate the intrinsic dimensionality of data by keeping only eigenvectors with positively-valued eigenvalues (reaching the bit limit is represented with a sign change in linear algebra libraries), thus reducing the impact of the user's choice of the number of dimensions to be computed. This denoised latent high-dimensional orthogonal basis is then explored with a new neighborhood graph in the third step; once again, DH, CkNN, or FSS can be used to learn a similarity metric from this topological basis. The learned topological basis and graph are then used to obtain a two or three-dimensional layout initialized with a LE and optimized with five

possible algorithms: t-SNE, MAP, MDE, PaCMAP, and TriMAP. Through this framework, TopOMetry harbors a high-level API that allows users to compute and evaluate the results of a total of 30 layouts with a single line of code. We argue this empowers single-cell analysis, as one cannot be sure whether a particular computational algorithm will perform well in any given set of data: extending the conclusions obtained from even dozens of datasets to the vast and exponentially growing corpora of single-cell data would be a dangerous assumption. Instead of relying on the presumed supremacy of any given algorithm or assuming data structures, users are endowed with the capacity of testing on their data different approaches that pursue the same unbiased mathematical goal and decide which to adopt based on evaluation results and on qualitative inspection of the learned clusters and layouts (and its association to known biology).

The widespread use of DM in single-cell biology associates the proposed concept of phenotypic topology with existing trajectory inference and pseudotime estimation frameworks. DM has been shown to describe cellular state dynamics in scRNAseq data and is useful for pseudotime estimation and dynamical modeling of cellular differentiation. However, their direct use in visualization is unsuitable for complex high-dimensional data such as those generated by single-cell assays since its results remain high-dimensional, rendering poor visualizations (similarly to PCA). Accordingly, it has been previously shown that multiple diffusion components can be needed to encode data heterogeneity[6,12,15], each encoding a different aspect of data structure, i.e., distinct cellular trajectories, and dozens or hundreds of diffusion components could be needed to describe large-scale, whole-organism single-cell data. We noticed that these

high-dimensional, phenotypic encoding components that approximate the LBO can be seen as lying in a *phenotypic manifold* of their own, being thus suitable for learning new graphs and optimizing layouts. Because of this property, lower-dimensional orthogonal bases that encode phenotypic topology can be obtained with DM or with Laplacian Eigenmaps built from CkNN or FSS similarity graphs, which have remained unexplored in the context of single-cell analyses. In CkNN, an unweighted adjacency graph is harnessed to encode the Laplace-de Rham Operator (LRO; 0-the LRO acts on 0-forms and corresponds to the LBO); which arguably makes it an even more unbiased choice than DM. However, it is computationally more expensive than DM and FSS, and can be challenging to apply in practice. In UMAP's FSS construction, an independent metric space is defined for each sample based on the sparseness of its neighborhood - a practical approach for the uniformization of the sample distribution on the learned manifold - and then a fuzzy set union is performed to aggregate all local individual metrics into a global view. As with DM (i.e., eigendecomposition of weighted Laplacian from the diffusion harmonics), the metrics learned with CkNN and FSS are used to form a new lower-dimensional basis - with CkNN and FSS, we employ Laplacian Eigenmaps (LE) to perform the decomposition step, noting that LE also provides an approximation of the LBO given a metric space. For clarity, we named these *diffusion, continuous* and *fuzzy bases* (db, cb, fb) and use them as a prefix to the downstream graphs and the layouts names.

Similar to PCA preprocessing, a new similarity graph needs to be constructed from the learned topological bases to perform clustering and optimizing layouts (all layout optimization algorithms operate on similarity graphs). This graph is also needed to obtain a further LE of the

learned basis that can be used as initialization for layout algorithms[47]. For this step, the three previously used metric-learning algorithms are further employed, rendering nine possible combinations of topological models, referred to by their basis prefix and graph suffix (e.g., fb_diff, db_cknn).

Noticeably, the final, layout optimization step is independent from the adopted upstream topological approach; thus, established (tSNE, MAP) and recent (densMAP, PaCMAP, TriMAP, MDE) methods can also be employed by the user to explore different visualizations of the same data structure, resulting in a family of topology-preserving layouts with different metric models for obtaining lower-dimensional basis and graphs. Within TopOMetry, the layout optimization method is added as a suffix to graph names (e.g. db_diff_MAP).

Distances estimation

One of the most fundamental aspect of machine-learning algorithms dealing with high dimensional data is the curse of dimensionality: in higher-dimensional spaces (HDS), the capacity of distance metrics to discriminate dissimilarity decays, i.e. the measured distance of the most similar samples tends to approach that of the most dissimilar ones as dimensionality increases[47]. The task of accurately and efficiently computing distances between samples in a large HDS is a challenge by itself, and several algorithms and tools have been developed to address it[64]. The most commonly used metric in single-cell data analysis is the Euclidean distance, followed by the cosine and the manhattan distances - yet, in principle, any metric distance could be applied. Recent benchmarks[65,66] have achieved initial insight on the suitability of different metrics to estimate distances between single-cells, pointing towards correlation

metrics as a favorable choice, although these were limited to only some single-cell RNA sequencing datasets in PCA-based workflows, and further studies are needed. This guided the choice of the cosine distance metric as a default in TopOMetry and in all presented analysis. Of note, the distance metric used in learning an orthogonal basis and in learning a new graph from it can differ, and assessing which particular combinations work best to recover latent information is a task yet to be done.

Guided by a recent benchmark between distance estimation algorithms[57], we employed the Hierarchical Navigable Small Worlds (HNSW)[58,59] algorithm to power the computational efficiency of finding k-nearest-neighbors of large single-cell datasets. HNSW has been shown to achieve fast and near-optimal results while being useful to compute a variety of distance metrics (e.g. euclidean, cosine, manhattan). Similarly to other non-linear algorithms, the first step of the diffusion basis is to find a k-nearest-neighbors graph (kNN) graph. As previous studies have shown that using only highly variable features (HVF) for computations increases the power of single-cell analysis[63,67], we estimated distances from the cell by HVF matrix for the shown experiments. As a reference, ~2,000 genes are generally considered a reasonable number of HVFs in single-cell RNA sequencing data analysis.

Affinity estimation

Diffusion Harmonics

The implemented diffusion operator includes many recent theoretical and empirical advances in diffusion harmonics. Of note, 1) an anisotropic diffusion kernel is used to

individualize the diffusion metric to a cell-by-cell basis[21,22]; 2) the anisotropic kernel is adapted by each cell neighborhood density, being normalized by each cell's distance to its median *k*-nearest-neighbor[14,15,68]; 3) neighborhood search expansion, a process in which the original *k*NN graph is expanded to additional neighbors using a data-derived estimate on neighborhood sparseness[15,32]. Additionally, we introduce an adaptive kernel decay rate, in which the rate of similarity learning is adaptively adjusted by each cell neighborhood density instead of by a fixed hyperparameter as otherwise proposed[6].

Guided by previous approaches, we first construct a k-nearest-neighbors graph for a given dataset, with N samples (cells) and M measures (genes), using a given distance metric as a local similarity measure. Let $\mathcal{M} \subset \mathbb{R}^n$ be a Riemannian manifold with unknown sampling distribution q (not necessarily uniformly distributed) in ambient dimension n. For a given sample $x_i \in X_{n \times m} \subset \mathcal{M}$ (the manifold hypothesis - data is assumed to be contained in the proximities of the manifold), consider $nbrs\ (x_i)$ to be the ordered set of its k-nearest-neighbors under some distance metric $d(x_i, y_i)$:

$$nbrs\ (x_i) \ = \ \{y_{i\,(1)},\ y_{i\,(2)}, \ldots,\ y_{i\,(\frac{k}{2})},\ \ldots, y_{i\,(k)}\} \mid k < N,\ k \subset \mathbb{N}$$

Given that the vector of sample observations x_i, a scaling factor σ_i corresponding to the distance of sample i to its median nearest-neighboring sample(s) in the high-dimensional space is defined:

$$\sigma_i \ \equiv \ d(x_i,\ y_{i\,(\frac{k}{2})})$$

Drawing from previous work on adaptive bandwidth diffusion kernels, the following kernel function could be initially used to convert distances into affinities:

$$K\ (x_i,\ x_j)\ =\ exp(\frac{(x_i - x_j)(x_j - x_i)}{\sigma_i + \sigma_j})\ .\ \sqrt{\left|2\pi(\sigma_i\ +\ \sigma_j)\right|}$$

To minimize the impact of the choice of the number of k neighbors hyperparameter on the learned graph structure, we harness the potential of the scaling factor σ_i to carry information regarding the neighborhood density of x_i. For this, a new indirect, continuous metric named $pseudomedian(\Omega)$ is defined for each sample x_i as the interpolation of all σ_i into the new interval:

$$\Omega_i\ = \frac{(k-1)(\sigma_i - argmin(\sigma))}{argmax(\sigma) - argmin(\sigma)} + 1$$

Ω_i represents a simple, indirect, and yet intrinsic reference from the distribution of $nbrs\ (x_i)$ at a given fixed k, thus providing information on the neighborhood density of each individual sample and about the discrete graph-building process. A sample with a sparser neighborhood will have a larger value of Ω than one with a dense neighborhood, indicating the relative need for search expansion in the sparser region. This estimate is implemented in the proposed diffusion process as a way to obtain enriched information about x_i neighborhood density, and also as a measure to alleviate the impact of a fixed k hyperparameter, as the graph

search can be extended as a function of maximum Ω to help avoid insufficient sampling at sparser neighborhoods:

$$k' = k + (k - argmax(\Omega_i)) \mid k' > k,\ k' \subset \mathbb{N}$$

It is of note that PHATE[6] introduced a kernel with variable bandwidth εequipped with a novel alpha-decay, in which α is a hyperparameter that controls the kernel decay rate. This was a first attempt to handle cases in which a sparsely sampled neighborhood can present with distant neighbors that are weighted with a high affinity. Note that $\alpha = 2$ leads to the classic Gaussian kernel, and ε is a fixed bandwidth.

$$K_{\alpha}(x_i,\ x_j) = \frac{1}{2}exp(-\left(\frac{d(x_i, x_j)}{\varepsilon_i}\right)^{\alpha}) + \frac{1}{2}exp(-\left(\frac{d(x_i, x_j)}{\varepsilon_j}\right)^{\alpha})$$

We harness the concept of the alpha-decaying kernel to implement an *adaptively-decaying* kernel, in which the rate of decay is related to the neighborhood sampling density distributionΩ:

$$K_{\alpha,k'}(x_i,\ x_j) = \frac{1}{2}exp(-\left(\frac{d(x_i, x_j)}{\sigma_i}\right)^{\alpha_i}) + \frac{1}{2}exp(-\left(\frac{d(x_i, x_j)}{\sigma_j}\right)^{\alpha_j}) \mid \alpha_i = 2^{\frac{k'-\Omega_i}{\Omega_i}}$$

The construction of a diffusion process on a graph from a given kernel function is a classical topic in spectral graph theory. We proceed with the weighted graph Laplacian normalization by denoting the distribution probabilities as q(x) so that:

$$\int q(x)dx = 1 \mid q: \mathbb{R}^n \rightarrow [0, \infty)$$

and then defining the transition probabilities $p(x)$:

$$p(x) = \int_{x_i} K_{\alpha,k'}(x,y)q(y)dy$$

To gain further insight regarding the structure of $X_{n \times m} \subset \mathcal{M}$, the kernel $K_{\alpha,k'}$is normalized by the to generate the anisotropic transition kernel $K'_{\alpha,k'}$:

$$K'_{\alpha,k'}(x_i, x_j) = \frac{K_{\alpha,k'}(x_i, x_j)}{p(x_i)^{a} p(x_j)^{a}} \mid a > 0$$

In which setting $a = 1$yields a differential operator Tf that encodes the probabilities of reaching a given sample from another given sample and approximates the Laplacian-Beltrami Operator independently of the data sampling distribution:

$$Tf = \Delta f - 2(1 - a)\nabla f \cdot \frac{\nabla p(x_i)\, p(x_j)}{p(x_i)\, p(x_j)}$$

Continuous k-nearest neighbors

Continuous k-nearest neighbors (CkNN) is a recently introduced[32] method that constructs a single unweighted graph from which the underlying manifold's structure (i.e. topological information) can be retrieved, given any compact Riemannian manifold. In its introductory manuscript, it was also shown that CkNN is the only unweighted graph construction strategy for which the graph Laplacian converges to the Laplace-Beltrami Operator (LBO). Conceptually, CkNN points to the superiority of *consistent homology* over *persistent homology*. However, differently from diffusion harmonics, LE and fuzzy simplicial sets, this is achieved through an unweighted graph rather than through a weighted graph.

CkNN's goal is to create an unweighted undirected graph from a given set of samples with distances given by a metric d. This translates to deciding whether to connect samples with edges. The two traditional approaches for constructing such a graph are a) fixed ϵ-balls (i.e. connect samples i, j if $d(i,j) < \epsilon$); and b) k-Nearest Neighbors (kNN), in which a fixed integer k given as a user defined hyperparameter is used to define edges (i.e. connect samples i, j if $d(i,j) \leq d(i,i_k)$ or if $d(i,j) \leq d(j,j_k)$; i_k and j_k being i and j k-th nearest neighbor, respectively). As shown in CkNN's manuscript and in its top-performant scores within TopOMetry, a less discrete and rather continuous version of kNN circumvents several limitations imposed by these traditional strategies.

CkNN connects samples i, j with edges if $d(i,j) < \delta(\sqrt{d(i,i_k)\, d(j,j_k)})$. Here, δ is a unitless scale parameter allowed to continuously vary. Given a finite set of data, there is only a finite set of possible values for δ, which is set within TopOMetry as a user-provided hyperparameter. There are two main advantages to this approach: first, it allows one to interpret the CkNN graph construction as a kernel method, enabling interpreting the graph Laplacian in terms of δ; second, it allows fixing the k hyperparameter, enabling the use of $d(i,i_k)$ as a local density estimate, similarly to the approximation of geodesics with fuzzy simplicial sets and diffusion harmonics. In practice, this captures the natural multiscale topology of data, wielding smooth geometries where data is sparsely sampled and fine-grained details where data is densely sampled.

Fuzzy simplicial sets

The application of 1-skeleton fuzzy simplicial sets as topological *metrics* on TopOMetry draws *entirely* from previous work, particularly from UMAP[25] theory and the work of David Spivak[69]. Our innovation is represented by the use of these metrics to construct new orthogonal bases that capture latent topology, and subsequently to construct topological graphs prior to layout optimization. When using this method to learn metrics from some data $X_{n \times m}$, it is assumed $X$is uniformly distributed on a locally connected $\mathcal{M}$ Riemannian manifold with a locally constant metric g. UMAP originally addresses the uniformity assumption creating a custom distance for each sample by normalizing distances with respect to their k-th nearest-neighbor to effectively approximate geodesics. To merge each of the discrete metric spaces generated per sample into a global informational space, the metric spaces are converted into fuzzy simplicial sets. A consensus metric can then be obtained by taking a fuzzy union across all metric spaces encoded as fuzzy simplicial sets, rendering a single fuzzy simplicial set which is granted to capture relevant topological metrics on $\mathcal{M}$. The Laplace-Beltrami Operator acting on $\mathcal{M}$ can then be approximated from the normalized laplacian of this fuzzy graph representation.

The aforementioned approach is grounded on solid category theory and no major modifications to the fuzzy simplicial set algorithms defined and implemented in UMAP were made, aside from improving its computational performance with approximate-nearest-neighbors with HNSW. For brevity, we direct the reader to UMAP's[25] and David Spivak's[69] manuscripts for in-depth descriptions of the involved category theory. A summarized description follows.

Simplicial sets are generalizations of simplicial complexes that provide a combinatorial approach to the study of topological spaces. Let Δbe a category with the finite order sets $[n] = \{1, \ldots, n\}$ as objects, and morphisms given by order-preserving maps. Δ^{op}denotes a category with the same objects as Δ and morphisms equivalent to those of Δ with reversed direction. Then, a simplicial set is defined as a functor from Δ^{op} to $Sets$(the category of sets). Given a simplicial set $X : \Delta^{op} \to Sets$, we refer to the set $X([n])$ as X_n , and to the elements of the set as the n-simplices of X. In the category of simplicial sets, there's a direct correspondence between the n-simplices of $X$and morphisms of the standard simplices Δ^n ($\Delta^n \to X$). For each $x \in X_n$there's a corresponding morphism x: $\Delta^n \to X$, and therefore $colim_{x \in X_n} \Delta^n \approx X$.

The standard covariant functor $|\cdot| : \Delta \to Top$(the category of topological spaces) that sends $[n]$to the standard n-simplex $\left|\Delta^n\right| \subset \mathbb{R}^{n+1}$ with standard subspace topology is defined as:

$$\left|\Delta^n\right| \equiv \left\{(t_0, \ldots, t_n) \in \mathbb{R}^{n+1} \mid \sum_{i=0}^{n} t_i = 1,\ t_i \geq 0\right\}$$

And because $X : \Delta^{op} \to Sets$ is a simplicial set, the realization of X ($|X|$) is $|X| = colim_{x \in X_n} |\Delta^n|$, which associates the topological space with the simplicial set. On the other hand, given a topological space Y, the associated simplicial set $S(Y)$(the singular set of Y) is defined as $S(Y) : [n] \to hom_{Top}(|\Delta^n|, Y)$. The theoretical contributions of David Spivak and UMAP's authors involve exploiting the fact that the realization functor and singular set functors

form an adjunction, translating between topological spaces and simplicial sets - for this, a categorical presentation of fuzzy sets was outlined.

A fuzzy set can be defined as a set for which membership is not binary - rather, it is a fuzzy property represented by continuous values in the unit interval I (with $I = (0, 1] \subseteq \mathbb{R}$, and with topology given by intervals of the form $[0, a)\ for\ a \in (0, 1]$). This allows one to define presheafs: a presheaf $\mathcal{P}$ on I is a functor from $\Delta^{op}$to $Sets$. A fuzzy set is a presheaf on I such that all maps $\mathcal{P}(a \leq b)$ are injections. $\mathcal{P}([0, a))$is the set of all elements with at least a membership strength. With this concept in mind, the category $Fuzz$of fuzzy sets is defined as the full subcategory of sheaves on I spanned by fuzzy sets. Consequently, defining fuzzy simplicial sets (FSS) corresponds to considering presheaves of $\Delta$to be valued in $Fuzz$ instead of $Sets$. The objects of the category of FSS $sFuzz$are thus functors from $\Delta^{op}$to $Fuzz$, and its morphisms are given by natural transformations. Conversely, FSS can be viewed as sheafs over the product topology $\Delta \times I$, allowing one to apply them to a broad category of extended-pseudo-metric spaces. This will be useful as the initial geodesic normalization step involves creating a unique metric space for each sample.

Extended-pseudo-metric-spaces (X, d) are defined as sets $X$and maps $d : X \times X \to \mathbb{R}_{\geq 0} \cup \{\infty\}$, being objects of the category $EPMet$, which has non-expansive maps as morphisms. A further subcategory of finite extended-pseudo-metric-spaces is defined as $FinEPMet$. Previously[69], the realization and singular set functors were extended to FSS by the construction of natural extension functors, $Real$and $Sing$, between the categories $sFuzz$and

$EPMet$. $Real$is defined in terms of standard fuzzy simplices ($\Delta^n_{<a}$), similarly to the the classic realization functor $|\cdot|$:

$$Real(\Delta^n_{<a}) \equiv \left\{(t_0, \dots, t_n) \in \mathbb{R}^{n+1} \mid \sum_{i=0}^{n} t_i = -\log(a),\ t_i \geq 0\right\}$$

A morphism $\Delta^n_{<a} \to \Delta^m_{<b}$ exists only if $a \leq b$, and is determined by a Δ morphism $\sigma: [n] \to [m]$. The action of Real on such a morphism is then given by the non-expansive map

$$(x_0, x_1, \dots, x_n) \to \frac{\log(b)}{\log(a)}\left(\sum_{i_0 \in \sigma^{-1}(0)} x_{i_0}, \sum_{i_0 \in \sigma^{-1}(1)} x_{i_0}, \dots, \sum_{i_0 \in \sigma^{-1}(m)} x_{i_0}\right).$$

This map is then extended to a more general simplicial set X by defining

$$Real(X) \equiv colim_{\Delta^n_{<a} \to X} Real(\Delta^n_{<a}).$$

Once again, similarly to the classical case, the right adjoint functor $Sing$is then defined for a given extended-pseudo-metric space Yin terms of its action on the category of product topology $\Delta \times I$:

$$Sing(Y): ([n], [0, a)) \to hom_{EPMet}(Real(\Delta^n_{<a}), Y)$$

As only finite metric spaces are of interest in this case, the associated FSS $Fin - sFuzz$ is considered and the correspondent $FinReal$and $FinSing$are used. Formally, the functor $FinReal: Fin - sFuzz \to FinEPMet$ is defined by:

$$FinReal(\Delta^n_{<a}) \equiv (\{(x_0, x_1, \dots, x_n\}, d_a)$$

where $d_a(x_i, x_j)$ is equivalent to$-\log(a)$ if $i \neq j$ and to 0 otherwise, allowing

$$FinReal(X) \equiv colim_{\Delta^n_{<a} \to X} FinReal(\Delta^n_{<a})$$

to be defined. Analogously, $FinSing: FinEPMet \to Fin - sFuzz$can be defined by:

$$FinSing(Y): ([n], [0, a)) \to hom_{FinEPMet}(FinReal(\Delta^n_{<a}), Y).$$

By construction, we have that the functors $FinReal: Fin - sFuzz \to FinEPMet$ and $FinSing: FinEPMet \to Fin - sFuzz$form an adjunction in which $FinReal$is the left adjoint and $FinSing$is the right adjoint.

After the trouble of defining fuzzy simplicial sets and their functors, one can merge the rather incompatible metric spaces created during geodesic approximation to preserve topological information from each metric by first translating each to a fuzzy simplicial set with the $FinSing$ functor, and then taking a union across all FSS, resulting in a single FSS that is ought to capture the underlying structure of the manifold $\mathcal{M}$. Given a dataset $X = \{X_1, \ldots, X_n\} \in \mathbb{R}^n$and its associated family of extended-pseudo-metric spaces $\{(X, d_i)\}_{i=1\ldots N}$ with common carrier set X, it follows that $d_i(X_i, X_k) = d_{\mathcal{M}}(X_i, X_k) - \rho$if $i = j$or $i = k$, and that $d_i(X_i, X_k) = \infty$if otherwise. $\rho$is the distance between $X_i$and its nearest neighbor, and $d_{\mathcal{M}}$is the approximate geodesic distance on the manifold $\mathcal{M}$. Then, the fuzzy topological representation of X corresponds to the union:

$$\bigcup_{i=1}^{n} = FinSing((X, d_i)).$$

Learning a latent basis

Once an adequate affinity graph has been constructed, obtaining it's weighted graph Laplacian is trivial, albeit extremely useful. The idea of using the spectral decomposition of graph Laplacians is not new in machine learning[19], and has been thoroughly used across different applications, including single-cell genomics. The novelty regarding such spectral embeddings in our work is the realization that approximations of the LBO are possible with spectral decompositions of the Laplacian of any given adjacency graph, weighted (diffusion harmonics, fuzzy simplicial sets) or not (CkNN). The eigenvectors obtained through this process form a new orthogonal basis with a reduced number of dimensions, in which each low-dimensional component encodes some aspect of data topology. The eigenvalues of such decomposition are useful to perform an empiric estimation of the data intrinsic dimensionality during computations, as the numerical values hit the limits of floating-point precision, which is encoded either by a sign-flip or by the decomposition algorithm outputting null outputs for the eigenvalues that pass this limit. We found that this is a useful way to set the key hyperparameter *n_eigs*, which defines a fixed number of latent dimensions to be computed, and recommend putative users to increase this parameter up to a point where such a spectral gap can be observed in scree plots. It is also important to note different decomposition strategies are used: for the diffusion harmonics model, we obtain multiscale diffusion maps, while for the FSS and CkNN models we used classic spectral decompositions of the normalized graph Laplacian.

Multiscale Diffusion Maps

Besides providing adaptations to the diffusion operator described in the *Diffusion Harmonics* subsection, we adapted the diffusion maps algorithm for better resolution through multiscaling. This includes component multiscaling, which avoids setting a fixed number of *t* diffusion steps hyperparameter, rendering multiscale diffusion maps. Instead, diffusion distances are weighted across all possible timescales and jointly considered to build multiscale distances [15,41].. Although our adaptations add computational performance and discriminative power to the DM algorithm, core steps remained unaltered in order to preserve this key property.

Given a symmetric T differential operator with a defined number of t steps and a given estimate L of the intrinsic dimensionality (number of non-trivial eigenvalues) of $\mathcal{M}$, one can achieve the spectral decomposition:

$$T^t = \sum_{i \geq 0} \lambda_i^{2t} \phi_i \phi_j$$

In which T^t encodes the diffusion distance between samples $x_i$and x_j, $\phi_i^{(l)} = \phi(x_j)$ along the diffusion component ϕ:

$$DD^t(\phi_i, \phi_j)^2 = \sum_{t=1}^{L} \lambda_l^{2t} (\phi_i^{(l)} - \phi_j^{(l)})^2$$

We next multiscale distances to account for all possible t scales, rendering multiscale diffusion distances:

$$MSD^t(\phi_i, \phi_j)^2 = \sum_{t=1}^{\infty} \sum_{t=1}^{L} \lambda_l^{2t} (\phi_i^{(l)} - \phi_j^{(l)})^2$$

Which can be reformulated as:

$$MSD^t(\phi_i, \phi_j)^2 = \sum_{t=1}^{L} \left(\frac{\lambda_l}{1-\lambda_l}\right)^2 \left(\phi_i^{(l)} - \phi_j^{(l)}\right)^2 \mid 1 > \lambda_1 > \lambda_2 > \ldots > \lambda_L > 0$$

Resulting in an orthogonal basis that is robust to the choice of L components and density differences.

Laplacian Eigenmaps

Laplacian Eigenmaps (LE) is a seminal geometrically-inspired method to perform DR through spectral decomposition that drew attention due to its natural connections to clustering and locality-preserving properties[19]. In their manuscript introducing LE[19], Belkin and Niyogi explore the concept of computing a neighborhood graph and harnessing its Laplacian to obtain a lower-dimensional representation of data that preserves local information. They explicitly approximate the LBO by using the weighted Laplacian of the neighborhood adjacency graph. To appropriately choose weights, the connection between the LBO and the heat kernel[70] was originally explored to learn a similarity metric in a geometrically principled fashion. As a consequence, the new orthogonal basis provided by solving the sparse eigenvalue problem approximates the eigenmaps of the LBO. Interestingly, they also address the connection of preserving local information to clustering, whereas global methods such as Isomap[29] that try to preserve pairwise geodesic distances do not wield similar results to that of clustering algorithms.

In summary, given a nearest neighbors graph $G_{n\,x\,n}$ built from n samples, a given similarity metric is learned (originally, LE similarities are learned with a standard diffusion kernel on a k-NN graph; here, we extend it to those learned with CkNN and fuzzy simplicial sets). This leads to an affinity matrix $W_{n\,x\,n}$where each element$W_{i,j}$corresponds to a weighted edge or zero, if the samples $i, j$are not connected. The diagonal weight matrix D ($D_{ii} = \sum_{j} W_{ji}$) is then exploited to obtain the graph Laplacian $(L = D - W)$. Finally, the generalized eigenvector problem $Lf = \lambda Df$ is solved to obtain a number of $m$eigenvectors, and these are used to map the data into a m-dimensional embedding ($x_i \rightarrow (f_1(i), \ldots, f_m(i))$).

Topological Graphs

The construction of new, denoised orthogonal bases from which one can learn new similarity graphs is a hallmark of TopOMetry. These bases are still highly-dimensional, and ideally, each component is associated with a specific cellular identity topology. This property leads to the poor performance of DM and LE when used exclusively for visualization: these algorithms are optimized to encode latent information from high-dimensional data in the form of denoised dimensions; however, the number of dimensions on which data may be located is unknown and definitely far beyond only two or three. Conceptually, we consider these bases as high-dimensional spaces encoding phenotypic topology, and aim to encode them into new, refined graphs. For that task, the aforementioned manifold learning techniques (diffusion

harmonics, fuzzy simplicial sets, continuous k-nearest neighbors) are harnessed to learn new similarity metrics on top of the extracted latent information. These methods are employed within TopOMetry to learn topological graphs identically to when learning topological metrics directly from data; however, in this step, computations are performed on top of the denoised bases. This naturally leads to the efficiency shortcoming of needing to compute additional neighborhood graphs; however, the computational cost of this second neighborhood search is negligible compared to the first, full-data neighborhood search. From a practical point of view, this step corresponds to the neighborhood search employed in several algorithms and pipelines (i.e. UMAP, PaCMAP, Scanpy, Seurat), with the difference that similarity metrics are learned with multiple topological algorithms and that these metrics are learned from a topological orthogonal basis rather than naively from Principal Components.

Graph layout optimization

Graph layout optimization techniques are methods that aim to optimize a lower-dimensional representation (usually in two or three dimensions, for visualization) that preserves most of some properties from the original data by minimizing a given cost function. Differences between these methods are mostly ough to the different methods to learn similarities from distances and to the number and design of cost functions[31]. The seminal t-SNE method, for instance, employs t-Student scores to estimate similarities between samples, and minimizes the Kullback–Leibler divergence between the high-dimensional and lower-dimensional representations. UMAP, which followed t-SNE as the *de facto* gold standard for visualization,

learns similarities with fuzzy simplicial sets (FSS), and then minimizes the (fuzzy) cross-entropy between the FSS representations of the high and lower-dimensional topologies. PaCMAP has shed additional light on how these methods work by analyzing the asymptotics of cost functions and empirically testing and defining the characteristics of efficiently designed cost functions; it also introduced the use of more than one cost function during the layout optimization process (i.e. initial iterations follow one cost function, intermediate steps another cost function, final steps a third one). MDE generalizes these concepts by situating layout optimization methods as special cases of convex optimization. With the exception of MAP, TopOMetry harnesses pre-existing implementations of layout algorithms in the python programming language, such as multicoreTSNE, TriMAP, PaCMAP and MDE. A detailed description of each layout algorithm used in the framework follows:

t-Distributed Stochastic Neighborhood Embedding (t-SNE)

t-SNE[23] is an improved version of SNE[71] (Stochastic Neighborhood Embedding) that offers significantly superior performance when applied to real-world data. SNE first converts high-dimensional distances into conditional probabilities that represent similarities: the similarity between samples x_i and x_j corresponds to the probability $p_{i|j}$ that x_j would have x_i as neighbor if neighbors were chosen according to the probability density under a Gaussian centered at x_j.

$$p_{i|j} = \frac{exp(-||x_i - x_j||^2 / 2\sigma_i^2)}{\sum_{k \neq i} exp(-||x_i - x_k||^2 / 2\sigma_i^2)}$$

A similar conditional probability $q_{i|j}$ can be computed for the points in the low-dimensional embedding. Traditionally, the gaussian employed in $q_{i|j}$ is set to $\frac{1}{\sqrt{2}}$, and these similarities are defined by:

$$q_{i|j} = \frac{exp(-||x_i - x_j||^2)}{\sum_{k \neq i} exp(-||x_i - x_k||^2)}$$

SNE aims to find a low-dimensional representation that minimizes the difference between these two probability distributions by considering the sum of all Kullback-Leibler (KL) divergences between the high- and low-dimensional representation of each data point. Considering P_i as the conditional probability distribution of samples in their ambient high-dimensional space and Q_i its low-dimensional counterpart, the sum of KL divergences for all samples is then minimized through gradient descent, with a cost function C defined by:

$$C = \sum_i KL(P_i||Q_i) = \sum_i \sum_j p_{j|i} log \frac{p_{j|i}}{q_{j|i}} .$$

The gradient of of this cost function has the form:

$$\frac{\delta C}{\delta \gamma_i} = 2\sum_j (p_{j|i} - q_{j|i} + p_{i|j} - q_{i|j})(\gamma_i - \gamma_j).$$

And its update update with a momentum term (to avoid local minima) is given by:

$$\gamma^{(t)} = \gamma^{(t-1)} + \eta \frac{\delta C}{\delta \gamma} + \alpha^{(t)}(\gamma^{(t-1)} - \gamma^{(t-2)})$$

t-SNE's superiority to SNE is due to two differences: first, the implementation of a symmetrized version of the SNE cost function; second, the use of heavy-tailed distributions to

address the crowding problem. Significant advances have also been made regarding the optimization of the t-SNE cost function to avoid local minima and increase performance. To symmetrize the cost function, instead of minimizing the sum of all KL divergences between high-dimensional and low-dimensional probability distributions, a single KL divergence between the joint probabilities distributions P (high-dimensional) and Q (low-dimensional) is used:

$$C = KL(P||Q) = \sum_i \sum_j p_{j|i} log \frac{p_{ji}}{q_{ji}}$$

In t-SNE, a Student t-distribution with one degree of freedom (the same as a Cauchy distribution) is used as a heavy-tailed distribution for the low-dimensional mapping. This defines the joint probabilities $q_{i\,j}$:

$$q_{i\,j} = \frac{(1 + ||y_i - y_j||^2)^{-1}}{\sum_{k \neq l} (1 + ||y_k - y_l||^2)^{-1}}$$

The gradient of the KL divergence between P and Q (computed using the above equation) is given by:

$$\frac{\delta C}{\delta \gamma_i} = 2 \sum_j (p_{j|i} - q_{j|i} + p_{i|j} - q_{i|j})(\gamma_i - \gamma_j)$$

With an update with momentum identical to that of SNE.

There are numerous modifications and adaptations of the t-SNE algorithm (e.g. heavy tailing, density preservation, perplexity combination), and these exceed the scope of this manuscript. The version of t-SNE included in TopOMetry relies on the MulticoreTSNE[72,73] package.

Uniform Manifold Approximation and Projection (UMAP)

The similarity metric when using the 'fuzzy' algorithm in TopOMetry corresponds to this 1-skeleton of the fuzzy simplicial sets, that is, a fuzzy graph (i.e. a fuzzy set of edges). In essence, UMAP's algorithm consists of only a couple more steps to learn a low-dimensional representation from X by constructing fuzzy topological representations of X and its putative low-dimensional representation Y. First, the fuzzy set of edges is represented by a reference set A and a membership strength function $\mu : A \rightarrow [0, 1]$. Using a presheaf representation $\mathcal{P}$, it's possible to define $A = \bigcup_{a \in (0,1]} \mathcal{P}([0, a))$, and $\mu(x) = sup\{a \in (0, 1] \mid x \in \mathcal{P}([0, a))\}$.Then, to generate visualizations, the graph layout is optimized by minimizing the cross-entropy C between the high-dimensional and low-dimensional fuzzy sets (A, μ) and (A, v)in terms of the reference set A:

$$C((A, \mu), (A, v)) \equiv \sum_{a \in A} \left(\mu(a) \, log\left(\frac{\mu(a)}{\nu(a)}\right) + (1 - \mu(a)) \, log\left(\frac{1 - \mu(a)}{1 - v(a)}\right) \right)$$

The MAP approach included in TopOMetry corresponds to a near-identical implementation of UMAP's optimization procedure, but using arbitrary graphs that are directly used for layout optimization (i.e. the manifold has already been approximated), thus only assuming the cross-entropy optimization finds global near-optima solutions.

TriMAP

TriMAP is a graph layout optimization method focused on preserving global structure. Differently than nearly all other methods, instead of relying on pairwise (dis)similarities between samples, TriMAP incorporates triplets to encode global structure. Saying that $(i,\ j,\ k)$ is a triplet means that point i is closer to point j than point k. In TriMAP, given a low-dimensional initialization, triplets from the original high-dimensional space are used to further optimize the low-dimensional graph layout. The original TriMAP implementation is initialized with PCA - in TopOMetry, TriMAP is initialized with a LE from the chosen topological basis. Such basis is used to sample the triplets $T = \{(i, j, k)\}$ used during optimization, instead of the original data.

For optimization, each triplet is assigned a weight $w_{ijk} \geq 0$ that implies the relative distances between pairs $(i,\ k)$ and (i, j). The loss L of the triplet $(i, j, k)$is a similarity function between $y_i$and $y_j$that harness the Student t-distribution (s):

$$L_{ijk} \equiv w_{ijk} \frac{s(y_i, y_k)}{s(y_i, y_j) + s(y_i, y_k)}, \text{ with } s(y_i,\ y_j) = (1 + ||y_i - y_j||^2)^{-1}$$

The weight $w_{ijk}$is defined first in the unnormalized form u_{ijk}, which is a function of distance measures between samples:

$$u_{ijk} = exp(d_{ik}^2 - d_{ij}^2) \geq 0$$

In the TriMAP algorithm, the distance measure d_{ij} is normalized by a scaling factor σ_{ij}defined by the average distances between a target sample and its 4-th to 6-th nearest neighbors:

$$d_{ij}^2 = \frac{||x_i - x_j||^2}{\sigma_{ij}}$$

This normalization adaptively adjusts scaling based on neighborhood density, similarly to the neighborhood normalization employed in diffusion harmonics, fuzzy simplicial sets and continuous k-nearest-neighbors, albeit being considerably simpler and less robust, particularly for smaller weights. This motivates a further normalization step in which the γ-scaled log-transformation emphasizes these smaller weights:

$$w_{ijk} = \zeta_\gamma\left(\frac{u_{ijk}}{\varpi} + \delta\right) \quad | \quad \zeta_\gamma(m) \equiv log(1 + \gamma m)$$

ϖ being $max_{(i',j',k') \in T} u_{i',j',k'}$, $\gamma > 0$being a scaling factor and δ a small constant. By default, $\gamma = 500$and $\delta = 10^{-4}$.

To optimize an embedding for each sample, a given number of triplets (*n_outliers*, default 5) are sampled from its *k*- nearest-neighbors (*n_inliers*, default 10). Additionally, some randomly sampled triplets containing the target sample are considered (*n_random*, default 5), which arguably helps preserve global structure. The final loss is defined as the sum of the losses of each sampled triplet in T:

$$L_{TriMAP} = \sum_{(i,j,k) \in T} (L_{ijk})$$

When used in TopOMetry for graph-layout, the TriMAP embedding is initialized with LE, and a topological orthogonal basis is used for sampling triplets. Because of its peculiar sampling process, providing it with a downstream topological graph could be counterproductive.

Pairwise Controlled Manifold Approximation and Projection (PaCMAP)

PaCMAP holds significant similarities with UMAP and TriMAP, but introduces its own unique insights. The core idea of PaCMAP is establishing robust loss functions that take into account not only a sample's local neighborhood, but how that neighborhood relates to closely located neighborhoods and with far away points. First, the neighborhood affinity graph from data is built in a manner identical to TriMAP triplet weighting procedure, in which distances between samples are scaled by the distance between the target sample and its 4th to 6th nearest neighbor:

$$d_{ij}^2 = \frac{||x_i - x_j||^2}{\sigma_{ij}}$$

Next, pairs of samples are selected as follows:

- Near pairs - a sample's *k*-nearest-neighbors, based on the scaled distance.
- Mid-near pairs - for each sample *i*, 6 other samples are randomly selected, and the second-closest to *i* is paired to it in a mid-near pair. The number of mid-near pairs to be sampled is *k* times a hyperparameter (*MN_ratio, default 0.5)*.
- Further pairs - uniformly sample non-neighbors of *i*. The number of further pairs to be sampled is *k* times a hyperparameter (*FP_ratio, default 2)*.

Given an initialization (PCA in default PaCMAP, LE in TopOMetry), the embedding is then optimized in three different stages, each with different weights for near-pairs, mid-near pairs and further pairs. Consider a sample *i,* its neighbor *j*, its mid-near pair *k* and its further pair *l.* Given the low-dimensional distance $d'_{ab} = || y_a - y_b ||^2 + 1$between samples *a* and *b*, PaCMAP minimizes the loss:

$$L_{PacMAP} = (w_{nb} \times \sum \frac{d'_{ij}}{10+d'_{ij}}) + (w_{mn} \times \sum \frac{d'_{ik}}{10^4+d'_{ik}}) + (w_{fp} \times \sum \frac{1}{10+d'_{il}})$$

Where $w_{nb}$is a weight given to *i* neighbors, w_{mn}a weight given to its mid-near pairs and w_{fp}a weight given to its further pairs. The optimization is performed in three stages, each with a number of iterations *t* that can be controlled by the user as hyperparameters. In the first stage (i.e. first 100 *t* iterations), $w_{nb} = 2$, $w_{mn} = 1000 \times (1 - \frac{t-1}{100}) + 3 \times (\frac{t-1}{100})$, and $w_{fp} = 1$. In the second stage, $w_{nb} = 3$, $w_{mn} = 3$, and $w_{fp} = 1$. In the third and last stage, $w_{nb} = 1$, $w_{mn} = 0$, and $w_{fp} = 1$.

The uniqueness and novelty of PaCMAP lies in the fact that local and global structure are balanced throughout the optimization process by the choice of sampling neighbors, mid-near pairs and further pairs simultaneously. When employed in TopOMetry, PaCMAP uses a denoised topological orthogonal basis instead of the raw data; similarly to TriMAP, feeding it the

downstream topological graph may be counterproductive due to its peculiar neighborhood sampling process.

Minimum Distortion Embedding

Recently, the problem of mapping high-dimensional information to a lower-dimensional embedding under a set of constraints (i.e. preserving distances, preserving neighborhoods, etc) was approached in a more generalized way with the introduction of Minimum Distortion Embedding (MDE). Embedding constraints are encoded by distortion functions associated with embedding distances d_{ij}, which are assumed to be differentiable. These functions derive from affinity (or dissimilarity) measurements between samples, or from their deviations, or from graphs on samples. For example, given nonzero weights $w_{ij} \in \Re$, graph edges can be partitioned into similar items (positive weights) and dissimilar items (negative weights), and a distortion function derived from these weights have the form:

$$f_{ij}(d_{ij}) = w_{ij}p(d_{ij}),$$

$p(d_{ij})$being a penalty function. Formally, negative and positive weights can have different penalty functions. The simplest function of this form is the quadratic, $f_{ij}(d_{ij}) = w_{ij}d_{ij}^2$.

Similar distortion functions can be derived from deviations or distances between samples *i, j*, with nonnegative numbers δ_{ij}indicating their dissimilarity. Distortion functions derived from this information have the form:

$$f_{ij}(d_{ij}) = L(\delta_{ij}, d_{ij}),$$

L being a nonnegative loss function. The simplest function of this form is the square loss, $f_{ij}(d_{ij}) = (\delta_{ij} - d_{ij})^2$.

The main goal of MDE is to minimize an embedding average distortion, defined as the sum of all pairwise distortions:

$$E(X) = \frac{1}{|\varepsilon|} \sum_{(i,j)\in\varepsilon} f_{ij}(d_{ij})$$

As it can be promptly noticed, this framework is extremely flexible and unifies various concepts involving dimensional reduction, manifold learning and metric embedding. Classic methods such as PCA, Laplacian Eigenmaps and UMAP can be seen as special cases of MDE. Nevertheless, it can still be challenged by big, non-linear, non-uniform data, and therefore benefit from TopOMetry topological analysis for data denoising and similarity learning. Given a topological graph and an orthogonal basis, TopOMetry harnesses MDE to learn layouts that can be representative of the underlying data structure. Noticeably, these layouts are highly customizable, and comparisons between various distortion functions and constraints that can be used for their generation exceed the scope of the present work.

Benchmarks

Global score

To evaluate preservation of global structure by embeddings and layouts, we elected the PCA resemblance score. It was first introduced in the TriMAP manuscript[28] and roughly captures how much a given embedding $\boldsymbol{Y}$ performs compared to PCA when it comes to preserving the global structure of the original data $\boldsymbol{X}$. By design, PCA has the lowest possible *Minimum Reconstruction Error* (MRE), defined as:

$$\varepsilon(Y|X) \equiv min_{A \in \Re^{m \times n}} ||X - AY||_F^2$$

Where $|| \cdot ||_F$ denotes the Frobenius norm. A global score (GS) is then defined by normalizing a given MRE by PCA's MRE (which by construction is 1):

$$GS(Y|X) \equiv exp(-\frac{\varepsilon(Y|X) - \varepsilon_{PCA}}{\varepsilon_{PCA}}) \in [0, 1].$$

Interestingly, as exposed in the results, the global score remains rather stable and close to 1.0 across all embedding and layout algorithms that were tested, given a PCA or spectral initialization. We empirically show that this score is approximately the same using either PCA or a spectral layout (LE) and implement functions to perform such tests within TopOMetry.

Local scores

For evaluating the preservation of local structure, we built upon previous approaches that explored geodesic distances and the Spearman correlation between original high-dimensional and downstream low-dimensional distances[6,46]. This task is very computationally intensive. The

included evaluation pipeline demands less memory, but is only slightly faster than current methods. Computing scores for MOCA, for instance, were unfeasible in our computational settings. As depicted in Suppl. Fig. 1B, kNN graphs are built from both the original high-dimensional data matrix and the representations to be scored. Next, the geodesic distances (i.e. the weighted shortest-path pairwise distances) are obtained for each of these graphs, and their distributions are then scored accordingly to the Spearman correlation (R). Cells can then be ranked accordingly to their neighborhood indices - let n be the number of cells and d_i be the difference between ranks for cell i in the original high-dimensional space and in the latent low-dimensional space. Then, we can calculate the Spearman correlation as:

$$R = 1 - \frac{6\sum d_i^2}{n(n^2-1)}$$

The Spearman rank-order correlation coefficient (R) is a nonparametric measure of association between two sets of data, and superior to the Pearson correlation in this case because it assumes only monotonicity, not linearity. Intuitively, this score tells whether increasing distances in the low-dimensional embedding is related to increasing distances in the high-dimensional data: a score of 0 implies no association at all, and a score of 1.0 implies perfect association. As shown in the results section, real-world applications usually lie somewhere between these two extremes.

Marker gene scores

To evaluate how marker genes found for each clustering result (i.e. leiden clustering on the PCA graph or TopOMetry's graphs) perform at predicting a cluster identity, we developed a simple R package, *markeRscore*. Given clustering labels and a list of marker genes for each cluster, *markeRscore* computes the sensibility, specificity and positive likelihood ratio of the top *k* marker genes for each cluster, for each clustering result. The aggregated values are then visualized using violin plots to assess how well each set of marker genes perform. Naturally, this can be also used to compare methods that find marker genes given a single clustering result; however, here we opted to implement it to compare different clustering results using a single method to find marker genes.

Runtime

The MOCA dataset was randomly sampled into three groups of datasets, each with increasing number of cells. The datasets from the first group ranged from 50,00 to 15,000 total cells, while the datasets from the second group ranged from 1,000 to 50,000 total cells and those from the third group ranged from 1,000 to 1 million total cells. All models were performed for the first group, while only the diffusion and fuzzy models were used in the second group, and only the diffusion model was used in the third group. The PCA and the truncated single-value decomposition (SVD) implementations used were the default *scikit-learn*[74] algorithms *sklearn.decomposition.PCA()* and *sklearn.decomposition.TruncatedSVD()*. The number of eigencomponents was fixed to 100 for all comparisons.

Computational Environment

All analyses were performed on a six-core Lenovo ThinkPad P52 Intel® Xeon(R) E-2176M CPU @ 2.70GHz × 12 threads, with a total of 128GB of ERCC DDR SDRAM running the 64-bit 21.10 Pop!OS distribution of the Linux operating system.

Code availability

TopOMetry is freely accessible as a python library at https://github.com/davisidarta/topometry and extensively documented at https://topometry.readthedocs.io . Any additional information can be requested and will be promptly made available by the authors.

Funding

DS-O was supported by the grant #2020/04074-2, São Paulo Research Foundation (FAPESP), and LAV was supported by the grant #2013/07607-8, São Paulo Research Foundation (FAPESP).

Acknowledgements

We thank Gustavo B. Bellini, Valcir C. Vargas and Marcos A. Miachiro at the Institute of Biology from the University of Campinas for their most kind support with computational infrastructure. We thank Leland McInness, Dmitry Kobak and Akshay Agrawal for their seminal work, and for their time to provide their valuable comments regarding some of the ideas presented in the manuscript. We thank Bruno Loyola Barbosa for his kind assistance in testing the early implementations of TopOMetry. We thank Ebru Erbay and Helder Nakaya for their generous feedback during the development of TopOMetry.

Author Contributions

DS-O and LAV jointly conceived the idea of harnessing phenotypic topology to visually represent biological hallmarks from single-cell data. DS-O designed and built TopOMetry and CellTOMetry, and performed all bioinformatic analyses. Both authors wrote the manuscript and agreed to the final submitted version.

Competing interests

The authors have no competing interests.

References

1. Shapiro, E., Biezuner, T. & Linnarsson, S. Single-cell sequencing-based technologies will revolutionize whole-organism science. *Nat. Rev. Genet.* **14**, 618–630 (2013).
2. Potter, S. S. Single-cell RNA sequencing for the study of development, physiology and disease. *Nat. Rev. Nephrol.* **14**, 479–492 (2018).
3. Linnarsson, S. & Teichmann, S. A. Single-cell genomics: coming of age. *Genome Biol.* **17**, 97 (2016).
4. Bacher, R. & Kendziorski, C. Design and computational analysis of single-cell RNA-sequencing experiments. *Genome Biol.* **17**, 63 (2016).
5. Ma, Y. & Zhu, L. A review on dimension reduction. *Int. Stat. Rev.* **81**, 134–150 (2013).
6. Moon, K. R. *et al.* Visualizing structure and transitions in high-dimensional biological data. *Nat. Biotechnol.* **37**, 1482–1492 (2019).
7. La Manno, G. *et al.* RNA velocity of single cells. *Nature* **560**, 494–498 (2018).
8. Bergen, V., Lange, M., Peidli, S., Wolf, F. A. & Theis, F. J. Generalizing RNA velocity to transient cell states through dynamical modeling. *Nat. Biotechnol.* **38**, 1408–1414 (2020).
9. Stuart, T. *et al.* Comprehensive Integration of Single-Cell Data. *Cell* **177**, 1888-1902.e21 (2019).
10. Polański, K. *et al.* BBKNN: fast batch alignment of single cell transcriptomes. *Bioinformatics* btz625 (2019) doi:10.1093/bioinformatics/btz625.
11. Trapnell, C. *et al.* The dynamics and regulators of cell fate decisions are revealed by

pseudotemporal ordering of single cells. *Nat. Biotechnol.* **32**, 381–386 (2014).

12. Haghverdi, L., Büttner, M., Wolf, F. A., Buettner, F. & Theis, F. J. Diffusion pseudotime robustly reconstructs lineage branching. *Nat. Methods* **13**, 845–848 (2016).
13. A comparison of single-cell trajectory inference methods | Nature Biotechnology. https://www.nature.com/articles/s41587-019-0071-9.
14. Haghverdi, L., Buettner, F. & Theis, F. J. Diffusion maps for high-dimensional single-cell analysis of differentiation data. *Bioinformatics* **31**, 2989–2998 (2015).
15. Setty, M. *et al.* Characterization of cell fate probabilities in single-cell data with Palantir. *Nat. Biotechnol.* **37**, 451–460 (2019).
16. Huang, S. The molecular and mathematical basis of Waddington's epigenetic landscape: A framework for post-Darwinian biology? *BioEssays* **34**, 149–157 (2012).
17. F.R.S, K. P. LIII. On lines and planes of closest fit to systems of points in space. *Lond. Edinb. Dublin Philos. Mag. J. Sci.* **2**, 559–572 (1901).
18. Jolliffe, I. T. Principal Component Analysis and Factor Analysis. in *Principal Component Analysis* (ed. Jolliffe, I. T.) 115–128 (Springer, 1986). doi:10.1007/978-1-4757-1904-8_7.
19. Belkin, M. & Niyogi, P. Laplacian Eigenmaps for Dimensionality Reduction and Data Representation. *Neural Comput.* **15**, 1373–1396 (2003).
20. Coifman, R. R. & Lafon, S. Diffusion maps. *Appl. Comput. Harmon. Anal.* **21**, 5–30 (2006).
21. Coifman, R. R., Lafon, S., Kevrekidis, I. G., Maggioni, M. & Nadler, B. Diffusion maps, reduction coordinates, and low dimensional representation of stochastic systems. *Multiscale Model. Simul.* **7**, 842–864 (2008).

22. Coifman, R. R. *et al.* Geometric diffusions as a tool for harmonic analysis and structure definition of data: Diffusion maps. *Proc. Natl. Acad. Sci.* **102**, 7426–7431 (2005).

23. Maaten, L. van der & Hinton, G. Visualizing Data using t-SNE. *J. Mach. Learn. Res.* **9**, 2579–2605 (2008).

24. Kobak, D. & Berens, P. The art of using t-SNE for single-cell transcriptomics. *Nat. Commun.* **10**, 5416 (2019).

25. McInnes, L., Healy, J. & Melville, J. UMAP: Uniform manifold approximation and projection for dimension reduction. *arXiv* (2018).

26. Becht, E. *et al.* Dimensionality reduction for visualizing single-cell data using UMAP. *Nat. Biotechnol.* **37**, 38–44 (2019).

27. Artemenkov, A. & Panov, M. NCVis: Noise Contrastive Approach for Scalable Visualization. in *Proceedings of The Web Conference 2020* 2941–2947 (Association for Computing Machinery, 2020).

28. Amid, E. & Warmuth, M. K. TriMap: large-scale dimensionality reduction using triplets. *arXiv* (2019).

29. Tenenbaum, J. B., Silva, V. de & Langford, J. C. A Global Geometric Framework for Nonlinear Dimensionality Reduction. *Science* **290**, 2319–2323 (2000).

30. Kruskal, J. B. Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika* **29**, 1–27 (1964).

31. Wang, Y., Huang, H., Rudin, C. & Shaposhnik, Y. *Understanding How Dimension Reduction Tools Work: An Empirical Approach to Deciphering t-SNE, UMAP, TriMAP, and PaCMAP*

for Data Visualization. https://github.com/YingfanWang/PaCMAP.

32. Berry, T. & Sauer, T. Consistent Manifold Representation for Topological Data Analysis. *ArXiv160602353 Math* (2019).
33. Chari, T., Banerjee, J. & Pachter, L. *The Specious Art of Single-Cell Genomics*. 2021.08.25.457696 https://www.biorxiv.org/content/10.1101/2021.08.25.457696v1 (2021) doi:10.1101/2021.08.25.457696.
34. Böhm, J. N., Berens, P. & Kobak, D. *A Unifying Perspective on Neighbor Embeddings along the Attraction-Repulsion Spectrum*. https://github.com/berenslab/ne-spectrum.
35. Agrawal, A., Ali, A. & Boyd, S. Minimum-Distortion Embedding. *ArXiv210302559 Cs Math Stat* (2021).
36. Lopez, R., Regier, J., Cole, M. B., Jordan, M. I. & Yosef, N. Deep generative modeling for single-cell transcriptomics. *Nat. Methods* **15**, 1053–1058 (2018).
37. Pierson, E. & Yau, C. ZIFA: Dimensionality reduction for zero-inflated single-cell gene expression analysis. *Genome Biol.* **16**, 241 (2015).
38. Risso, D., Perraudeau, F., Gribkova, S., Dudoit, S. & Vert, J.-P. A general and flexible method for signal extraction from single-cell RNA-seq data. *Nat. Commun.* **9**, 284 (2018).
39. Qiu, P. Embracing the dropouts in single-cell RNA-seq analysis. *Nat. Commun.* **11**, 1169 (2020).
40. Duren, Z. *et al.* Integrative analysis of single-cell genomics data by coupled nonnegative matrix factorizations. *Proc. Natl. Acad. Sci.* **115**, 7723–7728 (2018).
41. Nadler, B., Lafon, S., Coifman, R. R. & Kevrekidis, I. G. Diffusion maps, spectral clustering

and reaction coordinates of dynamical systems. *Appl. Comput. Harmon. Anal.* **21**, 113–127 (2006).

42. Kondor, R. I. & Lafferty, J. Diffusion kernels on graphs and other discrete structures. in *In Proceedings of the ICML* 315–322 (2002).
43. Korenblum, D. Laplacian mixture modeling for network analysis and unsupervised learning on graphs. *PLOS ONE* **13**, e0204096 (2018).
44. Xiang, R. *et al.* A Comparison for Dimensionality Reduction Methods of Single-Cell RNA-seq Data. *Front. Genet.* **12**, (2021).
45. Sun, S., Zhu, J., Ma, Y. & Zhou, X. Accuracy, robustness and scalability of dimensionality reduction methods for single-cell RNA-seq analysis. *Genome Biol.* **20**, 269 (2019).
46. Heiser, C. N. & Lau, K. S. A Quantitative Framework for Evaluating Single-Cell Data Structure Preservation by Dimensionality Reduction Techniques. *Cell Rep.* **31**, 107576 (2020).
47. Kobak, D. & Linderman, G. C. Initialization is critical for preserving global data structure in both t-SNE and UMAP. *Nat. Biotechnol.* **39**, 156–157 (2021).
48. Ntranos, V., Yi, L., Melsted, P. & Pachter, L. A discriminative learning approach to differential expression analysis for single-cell RNA-seq. *Nat. Methods* **16**, 163–166 (2019).
49. Waickman, A. T. *et al.* Temporally integrated single cell RNA sequencing analysis of PBMC from experimental and natural primary human DENV-1 infections. *PLOS Pathog.* **17**, e1009240 (2021).
50. Nehar-Belaid, D. *et al.* Mapping systemic lupus erythematosus heterogeneity at the

single-cell level. *Nat. Immunol.* **21**, 1094–1106 (2020).

51. Schafflick, D. *et al.* Integrated single cell analysis of blood and cerebrospinal fluid leukocytes in multiple sclerosis. *Nat. Commun.* **11**, 247 (2020).
52. Tirosh, I. *et al.* Dissecting the multicellular ecosystem of metastatic melanoma by single-cell RNA-seq. *Science* **352**, 189–196 (2016).
53. Zheng, S. C. *et al. Universal prediction of cell cycle position using transfer learning*. 2021.04.06.438463 https://www.biorxiv.org/content/10.1101/2021.04.06.438463v2 (2021) doi:10.1101/2021.04.06.438463.
54. Schwabe, D., Formichetti, S., Junker, J. P., Falcke, M. & Rajewsky, N. The transcriptome dynamics of single cells during the cell cycle. *Mol. Syst. Biol.* **16**, (2020).
55. Trapnell, C. Defining cell types and states with single-cell genomics. *Genome Res.* **25**, 1491–1498 (2015).
56. Cao, J. *et al.* The single-cell transcriptional landscape of mammalian organogenesis. *Nature* **566**, 496–502 (2019).
57. Aumüller, M., Bernhardsson, E. & Faithfull, A. ANN-Benchmarks: A benchmarking tool for approximate nearest neighbor algorithms. *Inf. Syst.* **87**, 101374 (2020).
58. Malkov, Y. A. & Yashunin, D. A. *Efficient and robust approximate nearest neighbor search using Hierarchical Navigable Small World graphs*. (2018).
59. Boytsov, L. & Naidan, B. Engineering Efficient and Effective Non-metric Space Library. in *Similarity Search and Applications - 6th International Conference, SISAP 2013, A Coruña, Spain, October 2-4, 2013, Proceedings* (eds. Brisaboa, N. R., Pedreira, O. & Zezula, P.) vol.

8199 280–293 (Springer, 2013).

60. Qiu, X. *et al.* Reversed graph embedding resolves complex single-cell trajectories. *Nat. Methods* **14**, 979–982 (2017).
61. Li, B. *et al.* Cumulus provides cloud-based data analysis for large-scale single-cell and single-nucleus RNA-seq. *Nat. Methods* **17**, 793–798 (2020).
62. Wolf, F. A., Angerer, P. & Theis, F. J. SCANPY: Large-scale single-cell gene expression data analysis. *Genome Biol.* **19**, 15–15 (2018).
63. Luecken, M. D. & Theis, F. J. Current best practices in single‐cell RNA‐seq analysis: a tutorial. *Mol. Syst. Biol.* **15**, e8746 (2019).
64. Yang, L. & Jin, R. Distance Metric Learning: A Comprehensive Survey. 51.
65. Skinnider, M. A., Squair, J. W. & Foster, L. J. Evaluating measures of association for single-cell transcriptomics. *Nat. Methods* **16**, 381–386 (2019).
66. Kim, T. *et al.* Impact of similarity metrics on single-cell RNA-seq data clustering. *Brief. Bioinform.* **20**, 2316–2326 (2019).
67. Yip, S. H., Sham, P. C. & Wang, J. Evaluation of tools for highly variable gene discovery from single-cell RNA-seq data. *Brief. Bioinform.* **20**, 1583–1589 (2018).
68. Berry, T. & Harlim, J. Variable bandwidth diffusion kernels. *Appl. Comput. Harmon. Anal.* **40**, 68–96 (2016).
69. Spivak, D. I. METRIC REALIZATION OF FUZZY SIMPLICIAL SETS. (2012).
70. Kondor, R. I. & Lafferty, J. Diffusion Kernels on Graphs and Other Discrete Input Spaces. 8.
71. Cook, J., Sutskever, I., Mnih, A. & Hinton, G. Visualizing Similarity Data with a Mixture of

Maps. in *Proceedings of the Eleventh International Conference on Artificial Intelligence and Statistics* 67–74 (PMLR, 2007).

72. Ulyanov, D. *Multicore t-SNE*. (2021).
73. Maaten, L. van der. Accelerating t-SNE using Tree-Based Algorithms. *J. Mach. Learn. Res.* **15**, 3221–3245 (2014).
74. Buitinck, L. *et al.* API design for machine learning software: experiences from the scikit-learn project. in *ECML PKDD Workshop: Languages for Data Mining and Machine Learning* 108–122 (2013).
75. Wang, Y., Huang, H., Rudin, C. & Shaposhnik, Y. Understanding How Dimension Reduction Tools Work: An Empirical Approach to Deciphering t-SNE, UMAP, TriMAP, and PaCMAP for Data Visualization. *ArXiv201204456 Cs Stat* (2021).