ARTICLES
https://doi.org/10.1038/s41587-019-0336-3

nature biotechnology

Corrected: Author Correction

Visualizing structure and transitions in high-dimensional biological data

Kevin R. Moon[1,14], David van Dijk[2,3,14], Zheng Wang[4,5,14], Scott Gigante[6,14], Daniel B. Burkhardt[7], William S. Chen[7], Kristina Yim[7], Antonia van den Elzen[7], Matthew J. Hirn[8,9], Ronald R. Coifman[10], Natalia B. Ivanova[11,15*], Guy Wolf[12,13,15*] and Smita Krishnaswamy[3,7,15*]

The high-dimensional data created by high-throughput technologies require visualization tools that reveal data structure and patterns in an intuitive form. We present PHATE, a visualization method that captures both local and global nonlinear structure using an information-geometric distance between data points. We compare PHATE to other tools on a variety of artificial and biological datasets, and find that it consistently preserves a range of patterns in data, including continual progressions, branches and clusters, better than other tools. We define a manifold preservation metric, which we call denoised embedding manifold preservation (DEMaP), and show that PHATE produces lower-dimensional embeddings that are quantitatively better denoised as compared to existing visualization methods. An analysis of a newly generated single-cell RNA sequencing dataset on human germ-layer differentiation demonstrates how PHATE reveals unique biological insight into the main developmental branches, including identification of three previously undescribed subpopulations. We also show that PHATE is applicable to a wide variety of data types, including mass cytometry, single-cell RNA sequencing, Hi-C and gut microbiome data.

High-dimensional, high-throughput data are accumulating at a staggering rate, especially of biological systems measured using single-cell transcriptomics and other genomic and epigenetic assays. Because humans are visual learners, it is important that these datasets are presented to researchers in intuitive ways to understand both the overall shape and the fine granular structure of the data. This is especially important in biological systems, where structure exists at many different scales and a faithful visualization can lead to hypothesis generation.

There are many dimensionality-reduction methods for visualization[1–11], of which the most commonly used are principal-component analysis (PCA)[11] and *t*-distributed stochastic neighbor embedding (*t*-SNE)[1–3]. However, these methods are suboptimal for exploring high-dimensional biological data. First, such methods tend to be sensitive to noise. Biomedical data is generally very noisy, and methods like PCA and Isomap[4] fail to explicitly remove this noise for visualization, rendering fine-grained local structure impossible to recognize. Second, nonlinear visualization methods such as *t*-SNE often scramble the global structure in data. Third, many dimensionality-reduction methods (for example, PCA and diffusion maps) fail to optimize for two-dimensional (2D) visualization as they are not specifically designed for visualization.

Furthermore, common implementations of dimensionality-reduction methods often lack computational scalability. The volume of biomedical data being generated is growing at a scale that far outpaces Moore's law. State-of-the-art methods such as multidimensional scaling (MDS) and *t*-SNE were originally presented (see refs. [1,7]) as proofs-of-concept with somewhat naive implementations, which do not scale well to datasets with hundreds of thousands, let alone millions, of data points owing to speed or memory constraints. Although some heuristic improvements may be made[3,8], most available packages still follow the original implementation and thus cannot run on big data, which severely limits the usability of these methods in the medium-to-long term.

Finally, we note that some methods try to alleviate visualization challenges by directly imposing a fixed geometry or intrinsic structure on the data. However, methods that impose a structure on the data generally have no way of alerting the user whether the structural assumption is correct. For example, any data will be transformed to fit a tree with Monocle2[12] or clusters with *t*-SNE. While such methods are useful for data that fit their prior assumptions, they can generate misleading results otherwise, and are often ill suited for hypothesis generation or data exploration.

To address the above concerns, we have designed a dimensionality-reduction method for visualization named potential of heat diffusion for affinity-based transition embedding (PHATE). PHATE generates a low-dimensional embedding specific for visualization, which provides an accurate, denoised representation of both local and global structure of a dataset in the required number of dimensions without imposing any strong assumptions on the structure of the data, and is highly scalable both in memory and runtime. To achieve this, we combine ideas from manifold learning, information geometry and data-driven diffusion geometry, and integrate them with current state-of-the-art methods. The result is that

[1]Department of Mathematics and Statistics, Utah State University, Logan, UT, USA. [2]Cardiovascular Research Center, section Cardiology, Department of Internal Medicine, Yale University, New Haven, CT, USA. [3]Department of Computer Science, Yale University, New Haven, CT, USA. [4]School of Basic Medicine, Qingdao University, Qingdao, China. [5]Yale Stem Cell Center, Department of Genetics, Yale University, New Haven, CT, USA. [6]Computational Biology and Bioinformatics Program, Yale University, New Haven, CT, USA. [7]Department of Genetics, Yale University, New Haven, CT, USA. [8]Department of Computational Mathematics, Science and Engineering, Michigan State University, East Lansing, MI, USA. [9]Department of Mathematics, Michigan State University, East Lansing, MI, USA. [10]Applied Mathematics Program, Yale University, New Haven, CT, USA. [11]Department of Genetics, Center for Molecular Medicine, University of Georgia, Athens, GA, USA. [12]Department of Mathematics and Statistics, Université de Montréal, Montréal, Quebec, Canada. [13]Mila—Quebec Artificial Intelligence Institute, Montréal, Quebec, Canada. [14]These authors contributed equally: Kevin R. Moon, David van Dijk, Zheng Wang, Scott Gigante. [15]These authors jointly supervised this work: Natalia B. Ivanova, Guy Wolf, Smita Krishnaswamy. *e-mail: natalia.ivanova@uga.edu; guy.wolf@umontreal.ca; smita.krishnaswamy@yale.edu

high-dimensional and nonlinear structures, such as clusters, nonlinear progressions and branches, become apparent in two or three dimensions and can be extracted for further analysis (Fig. 1a).

We develop a new metric called DEMaP to quantify the ability of an embedding to preserve denoised manifold distances, we show that PHATE consistently outperforms 11 other methods on synthetically generated data with known ground truth. We also use PHATE to visualize several biological and non-biological real-world datasets, showing the capacity of PHATE to visualize datasets with many different underlying structures including trajectories, clusters, disconnected and intersecting manifolds, and more (Fig. 1). To demonstrate the ability of PHATE to reveal new biological insights, we apply PHATE to a newly generated single-cell RNA sequencing (scRNA-seq) dataset of human embryonic stem cells grown as embryoid bodies over a period of 27 d to observe differentiation into diverse cell lineages. PHATE successfully captures all known branches of development within this system as well as differentiation pathways, and enables the isolation of rare populations on the basis of surface markers, which we validate experimentally.

Results

Visualizing complex, high-dimensional data in a way that is both easy to understand and faithful to the data is a difficult task. Such a visualization method needs to preserve local and global structure in the high-dimensional data, denoise the data so that the underlying structure is clearly visible and preserve as much information as possible in low (two to three) dimensions. Additionally, a visualization method should be robust in the sense that the revealed structure of the data is insensitive to user configurations of the algorithm and scalable to the large sizes of modern data.

Popular dimensionality-reduction methods are deficient in one or more of these attributes. For example, *t*-SNE[1] focuses on preserving local structure, often at the expense of the global structure (Fig. 1b,c), while PCA focuses on preserving global structure at the expense of the local structure (Fig. 1b,c). Although PCA is often used for denoising as a preprocessing step, both PCA and *t*-SNE provide noisy visualizations when the data is noisy, which can obscure the structure of the data (Fig. 1b,c). By contrast, diffusion maps[13] effectively denoise data and learn the local and global structure. However, diffusion maps typically encode this information in higher dimensions[14], which are not amenable to visualization, and can introduce distortions in the visualization under certain conditions (Supplementary Figs. 1 and 2).

PHATE is designed to overcome these weaknesses and provide a visualization that preserves the local and global structure of the data, denoises the data and presents as much information as possible into low dimensions. There are three major steps in the PHATE algorithm (Fig. 2):

1. Encode local data information via local similarities (Fig. 2a–c). For some data types, such as Hi-C chromatin conformation maps[15], the local relationships are encoded directly in the measurements. However, for most data types, the local similarities must be learned. We assume that component-wise, the data are well-modeled as lying on a manifold. Effectively this means that local relationships between data points, even when noisy, are meaningful with respect to the overall structure of the data, as they can be chained together to learn global relationships along the manifold. We apply a kernel function that we developed (called the α-decay kernel) to Euclidean distances to accurately encode the local structure of the data even when the data are not uniformly sampled along the underlying manifold structure.
2. Encode global relationships in data using the potential distance (Fig. 2d,e). Diffusing through data is a concept that was popularized in the derivation of diffusion maps[13]. Diffusion is performed by first transforming the local similarities into probabilities that measure the probability of transitioning from one data point to another in a single step of a random walk and then powering this operator to t steps to give t-step walk probabilities. Thus, both the local and global manifold distances are represented in the newly-calculated multistep transition probabilities, which are referred to as the diffusion probabilities. For example, two points that have multiple potential short paths that connect them will have a higher diffusion probability than two points that either have only long paths or relatively few paths connecting them. By considering all possible random walks, the diffusion process also denoises the data by down-weighting spurious paths created by noise. However, directly embedding the diffusion probabilities into two and three dimensions via eigenvalue decomposition results in either a loss of information (Supplementary Fig. 1) or an unstable embedding (Supplementary Figs. 2a and 3d, respectively). In PHATE we interpret the diffusion probability of each point to all other points as the 'global context of the data point', and derive an information-theoretic potential distance between each pair of cells that compares the entire global context. Potential distance is computed as a divergence between the associated diffusion probability distributions of the two cells to all other cells. Thus the relationship of each cell to both near neighbors and distant points is accounted for in this distance. Notably, many divergences use a sublinear transformation of probability distributions (such as a log-scale transformation), which prevents nearest neighbors from dominating the distance.
3. Embed potential distance information into low dimensions for visualization (Fig. 2e–f). The information in the potential distances are then squeezed into low dimensions for visualization via metric MDS, which creates an embedding by matching the distances in the low-dimensional space to the input distances. Unlike PCA, this ensures that all variability is squeezed into two dimensions for a maximally informative embedding.

These steps are outlined in Table 1. All of these steps are necessary to create a good visualization that preserves local and global structure in the high-dimensional data, denoises the data and presents as much information as possible in low dimensions. Further details on all of the steps of PHATE are included in the Methods, Supplementary Table 1 and Supplementary Note 1. PHATE is also robust to the choice of parameters (Methods; Supplementary Fig. 4) and produces the same results every time it is run, regardless of random seed (Supplementary Fig. 5).

In addition to the exact computation of PHATE, we developed an efficient and scalable version of PHATE that produces near-identical results. In this version, PHATE uses landmark subsampling, sparse matrices and randomized matrix decompositions. For more details on the scalability of PHATE see the Methods, Supplementary Table 2 and Supplementary Fig. 6, which shows the fast runtime of PHATE on datasets of different sizes, including a dataset of 1.3 million cells (2.5 h) and a network of 1.8 million nodes (12 min).

Extracting information from PHATE. PHATE embeddings contain a large amount of information on the structure of the data, namely, local transitions, progressions, branches or splits in progressions and end states of progression. Here we present new methods that provide suggested end points, branch points and branches on the basis of the information from higher-dimensional PHATE embeddings (Fig. 3). These may not always correspond to real decision points, but provide an annotation to aid the user in interpreting the PHATE visual.

- Branch-point identification with local intrinsic dimensionality. In biological data, branch points often encapsulate switch-like decisions where cells sharply veer towards one of a small number

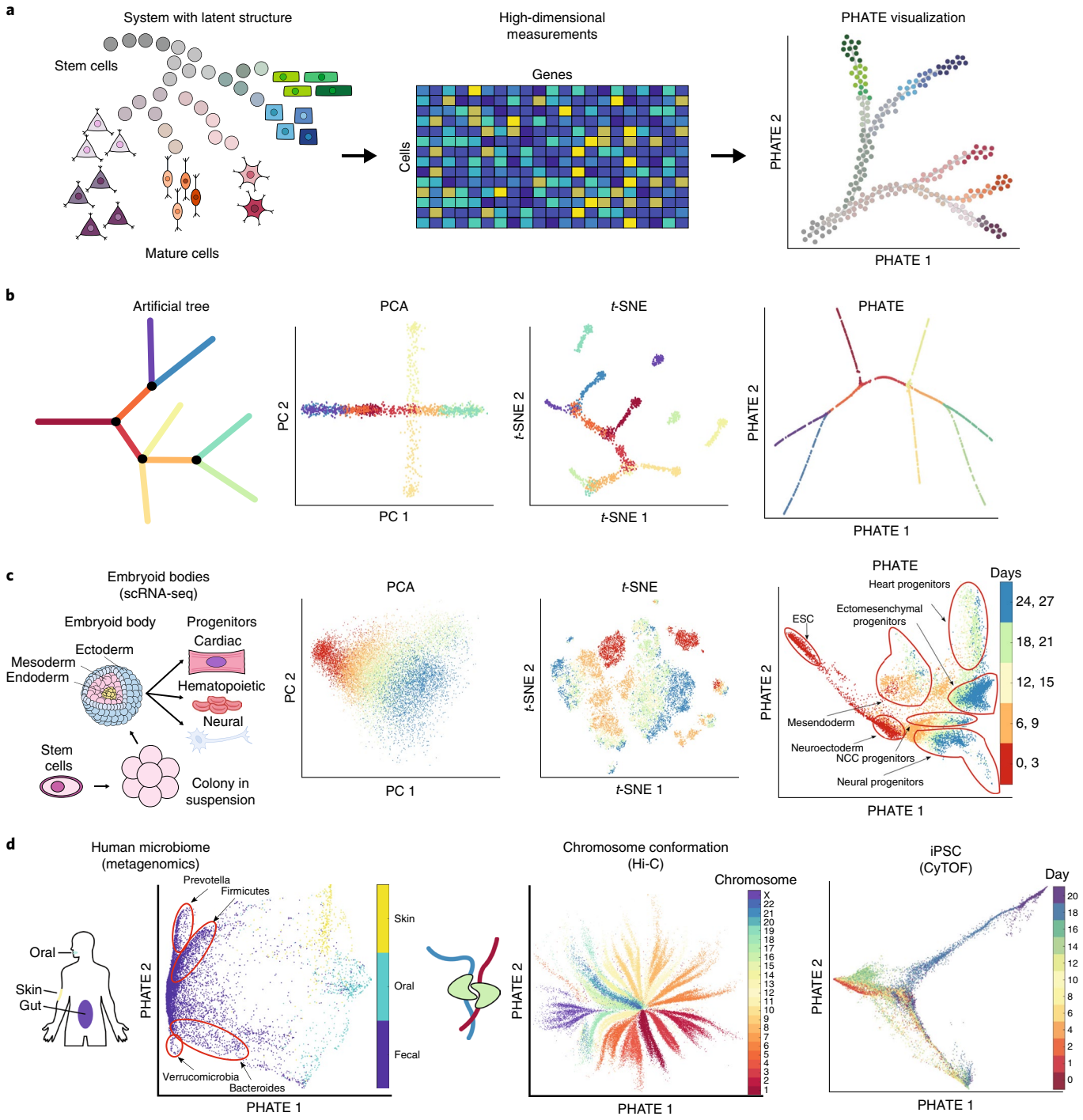

Fig. 1 | Overview of PHATE and its ability to reveal structure in data. a, Conceptual figure demonstrating the progression of stem cells into different cell types and the corresponding high-dimensional single-cell measurements rendered as a visualization by PHATE. **b**, Left, a 2D drawing of an artificial tree with color-coded branches. Data is uniformly sampled from each branch in 60 dimensions with Gaussian noise added (Methods). Right, comparison of PCA, *t*-SNE and the PHATE visualizations for the high-dimensional artificial tree data. PHATE is best at revealing global and branching structure in the data. In particular, PCA cannot reveal fine-grained local features such as branches while *t*-SNE breaks the structure apart and shuffles the broken pieces within the visualization. See Supplementary Fig. 3 for more comparisons of artificial data. **c**, Comparison of PCA, *t*-SNE and the PHATE visualizations for new EB data showing similar trends as in **b**. **d**, PHATE applied to various datatypes. Left, PHATE on human microbiome data shows clear distinctions between skin, oral and fecal samples, as well as different enterotypes within the fecal samples. Middle, PHATE on Hi-C chromatin conformation data shows the global structure of chromatin[57]. The embedding is colored by the different chromosomes. Right, PHATE on iPSC CyTOF data. The embedding is colored by time after induction. See Figs. 5 and Supplementary Figs. 8, 10 and 11 for more applications to real data.

of fates (Supplementary Fig. 7a). Identifying branch points is of critical importance for analyzing such decisions. We make a key observation that most points in PHATE plots of biological data lie on low-dimensional progressions with some noise as demonstrated in Fig. 3a. As branch points lie at the intersections of such progressions, they have higher local intrinsic dimensionality

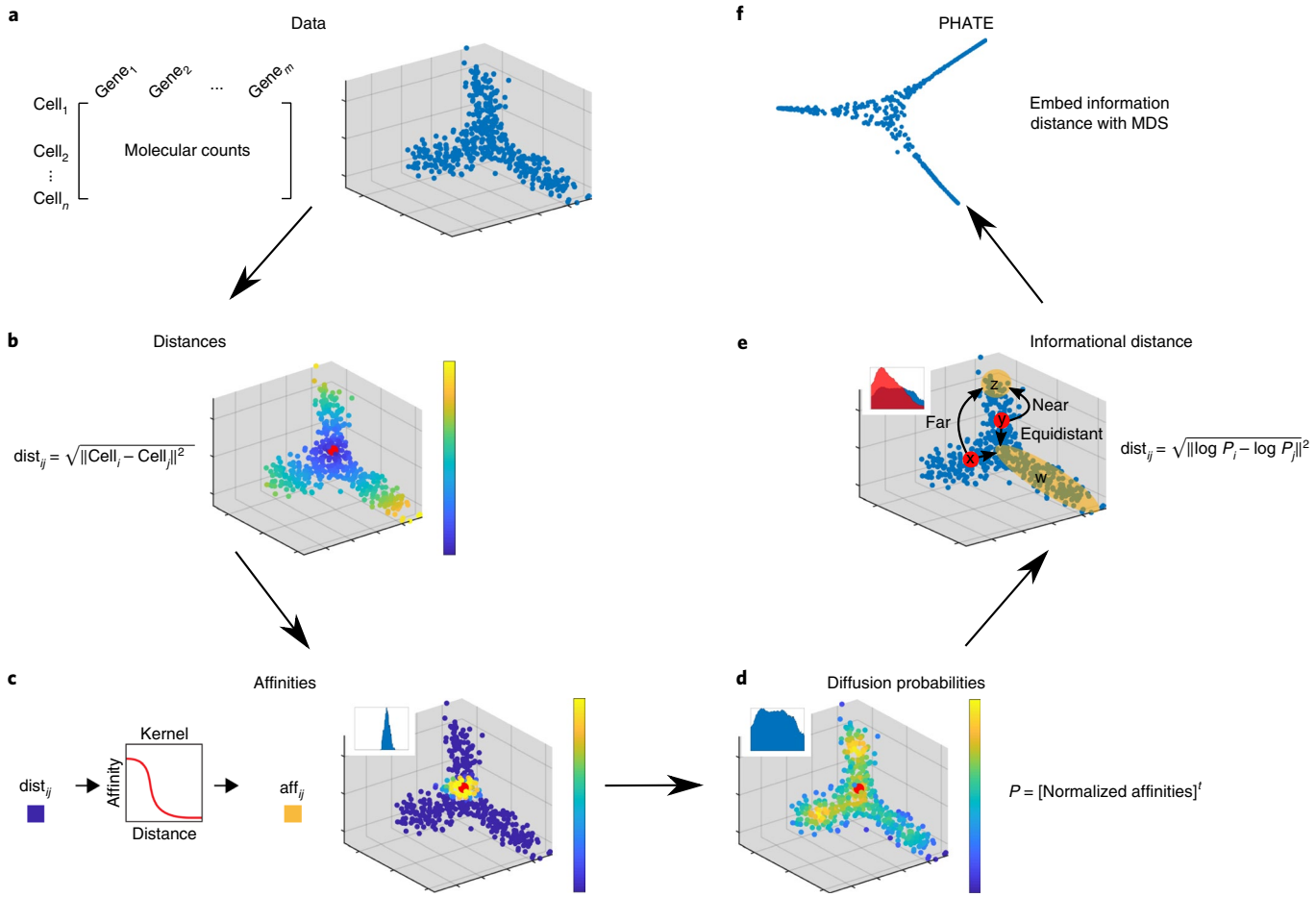

Fig. 2 | a,**b**, Data (**a**) and Euclidean distances (**b**); data points are colored by their Euclidean distance to the highlighted point. **c**, Markov-normalized affinity matrix; distances are transformed to local affinities via a kernel function and then normalized to a probability distribution. Data points are colored by the probability of transitioning from the highlighted point in a single-step random walk. **d**, Diffusion probabilities; the normalized affinities are diffused to denoise the data and learn long-range relationships between points. Data points are colored by the probability of transitioning from the highlighted point in a *t*-step random walk. **e**, Informational distance; an informational distance (for example, the potential distance) that measures the dissimilarity between the diffused probabilities is computed. The informational distance is better suited for computing differences between probabilities than the Euclidean distance. **f**, The final PHATE embedding; the informational distances are embedded into low dimensions using MDS. Note that distances or affinities can be directly input to the appropriate step in cases of connectivity data. Therefore, the Euclidean distance or our constructed affinities can be replaced with distances or affinities that best describe the data. For example, in Supplementary Fig. 11d we replace our affinity matrix with the Facebook connectivity matrix.

and can thus be identified by estimating the local intrinsic dimension. Figure 3a shows that points of intersection in the artificial tree data indeed have higher local intrinsic dimensionality than points on branches.

- End-point identification with diffusion extrema. We identify end points in the PHATE embedding as those that are least central and most distinct by computing the eigenvector centrality and the distinctness of a cellular state relative to the general data by considering the minima and maxima of diffusion eigenvectors (Fig. 3a) as motivated by ref. [16]. After identifying branch points and end points, the remaining points are assigned to branches between two branch points or between a branch point and an end point using an approach that is based on a previously developed branch-point-detection method[14], which compares the correlation and anticorrelation of neighborhood distances. Figure 3a gives a visual demonstration of this approach and details are given in the Methods. Figure 3b shows the results of our approach to identifying branch points, end points and branches on an artificial tree dataset, an scRNA-seq dataset of bone marrow[17] and an induced pluripotent stem cell (iPSC) cytometry by time of flight (CyTOF) dataset[18]. Our procedure identifies the branches on the artificial tree perfectly and defines biologically meaningful branches on the other two datasets, which we will use for data exploration.

Table 1 | General steps in the PHATE algorithm

Input: Data matrix, algorithm parameters (Methods)
Output: The PHATE visualization
(1) Compute the pairwise distances from the data matrix.
(2) Transform the distances to affinities to encode local information.
(3) Learn global relationships via the diffusion process.
(4) Encode the learned relationships using the potential distance.
(5) Embed the potential distance information into low dimensions for visualization.

Comparison of PHATE to other methods. Here we compare PHATE to multiple dimensionality-reduction methods. We provide

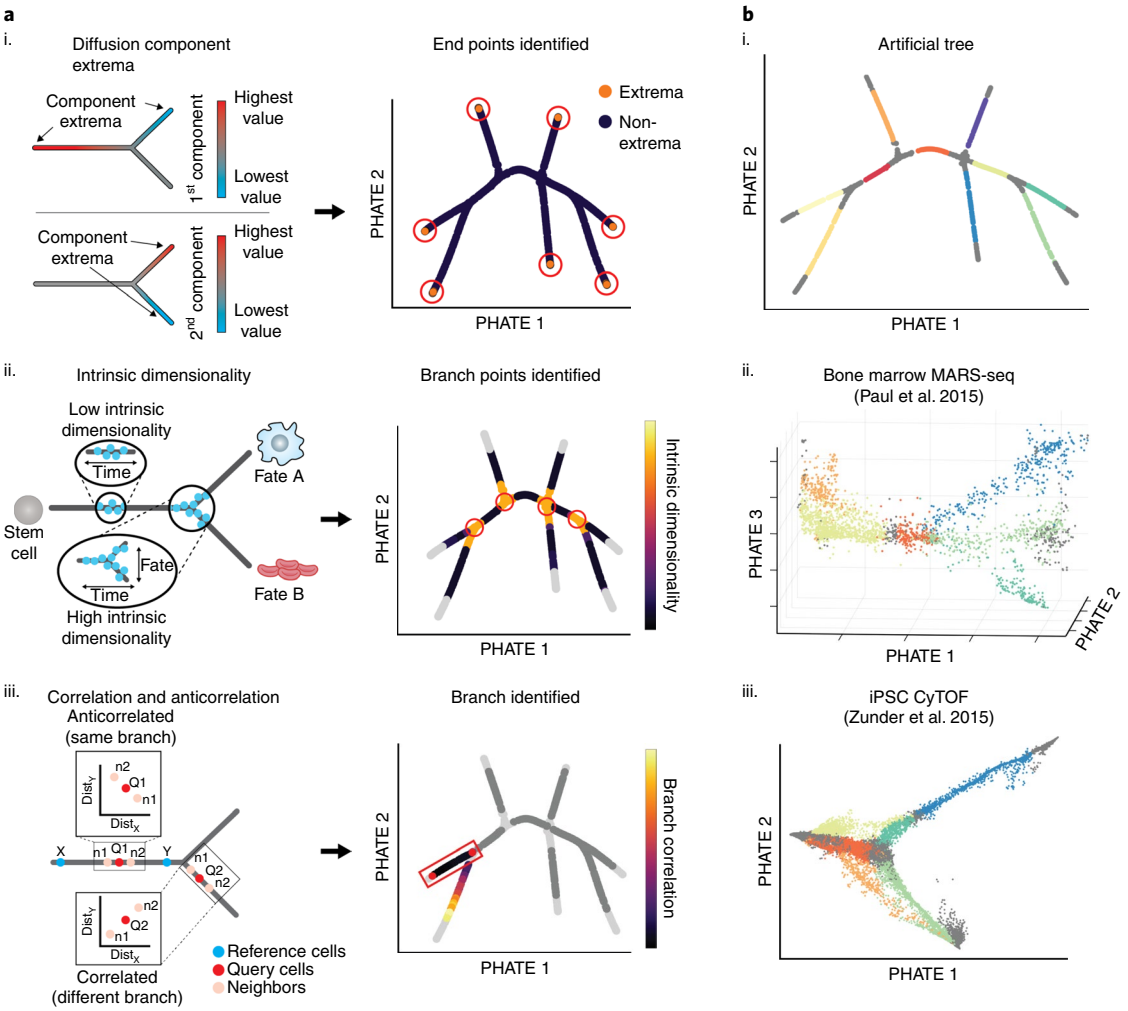

Fig. 3 | Extracting branches and branchpoints from PHATE. a, Methods for identifying suggested end points, branch points and branches. i, PHATE computes a specialized diffusion operator as an intermediate step (Fig. 2d). We use this diffusion operator to find end points. Specifically, we use the the extrema of the corresponding diffusion components (eigenvectors of the diffusion operator) to identify end points[16]. ii, Local intrinsic dimensionality is used to find branchpoints in a PHATE visual. As there are more degrees of freedom at branch points, the local intrinsic dimension is higher than through the rest of a branch. iii, Cells in the PHATE embedding can be assigned to branches by considering the correlation between distances of neighbors to reference cells (for example, branch points or end points). **b**, Detected branches in artificial tree data (i), bone-marrow scRNA-seq data[17] (ii) and iPSC CyTOF data[18] (iii). MARS-seq, massively parallel single-cell RNA seqeuncing.

quantitative comparisons on simulated data where the ground truth is known, and provide a qualitative comparison using both simulated and real biological data.

Quantitative comparisons. Quantifying the accuracy of a dimensionality reduction for visualization is an open problem in machine learning[19–21] as it is generally impossible to greatly reduce the dimensionality of a dataset without loss of information. To quantify the quality of a visualization, we needed a metric that judged whether a method preserves the information that is necessary for visual understanding. Previous work has focused on preserving pairwise distances or local neighborhoods[5,22,23]. However, these quantifications are not strictly desirable. For example, classical MDS is analytically the optimal solution to pairwise distance preservation in n dimensions[7]. However, MDS, as is visible in Supplementary Figs. 3 and 8, often does not produce clear or insightful visualizations for complex, nonlinear data. On the other hand, preserving local neighborhoods is the basis of the objective function for t-SNE[1], which fails to incorporate global structure and is hence insufficient for our purposes (Supplementary Fig. 3).

Previous work has also emphasized the utility of geodesic distances in computing both dimensionality reductions[4] and associated metrics[20]. Similar computations have been used to compare the output of trajectory-inference algorithms[24]. However, this metric is insufficient for our use for two reasons: (1) unlike in trajectory inference, the raw data is noisy, and we wish to quantify the ability of a visualization method to denoise the data; and (2) geodesic distances on low-dimensional visualizations fail to capture the inherent meaning of curvature. As visualizations do not suffer from the curse of dimensionality, we are able instead to use Euclidean distances, which capture the difference between straight and curved lines and which are also meaningful to the human eye.

Hence, to quantitatively compare PHATE to other visualization methods, we formulated the DEMaP metric. DEMaP is designed to encapsulate the desirable properties of a dimensionality-reduction method that is intended for visualization. These include: (1) the preservation of relationships in the data such that cells close together on the manifold are close together in the embedded space and cells that are far apart on the manifold are far apart in the embedding, including disconnected manifolds (for example, clusters), which

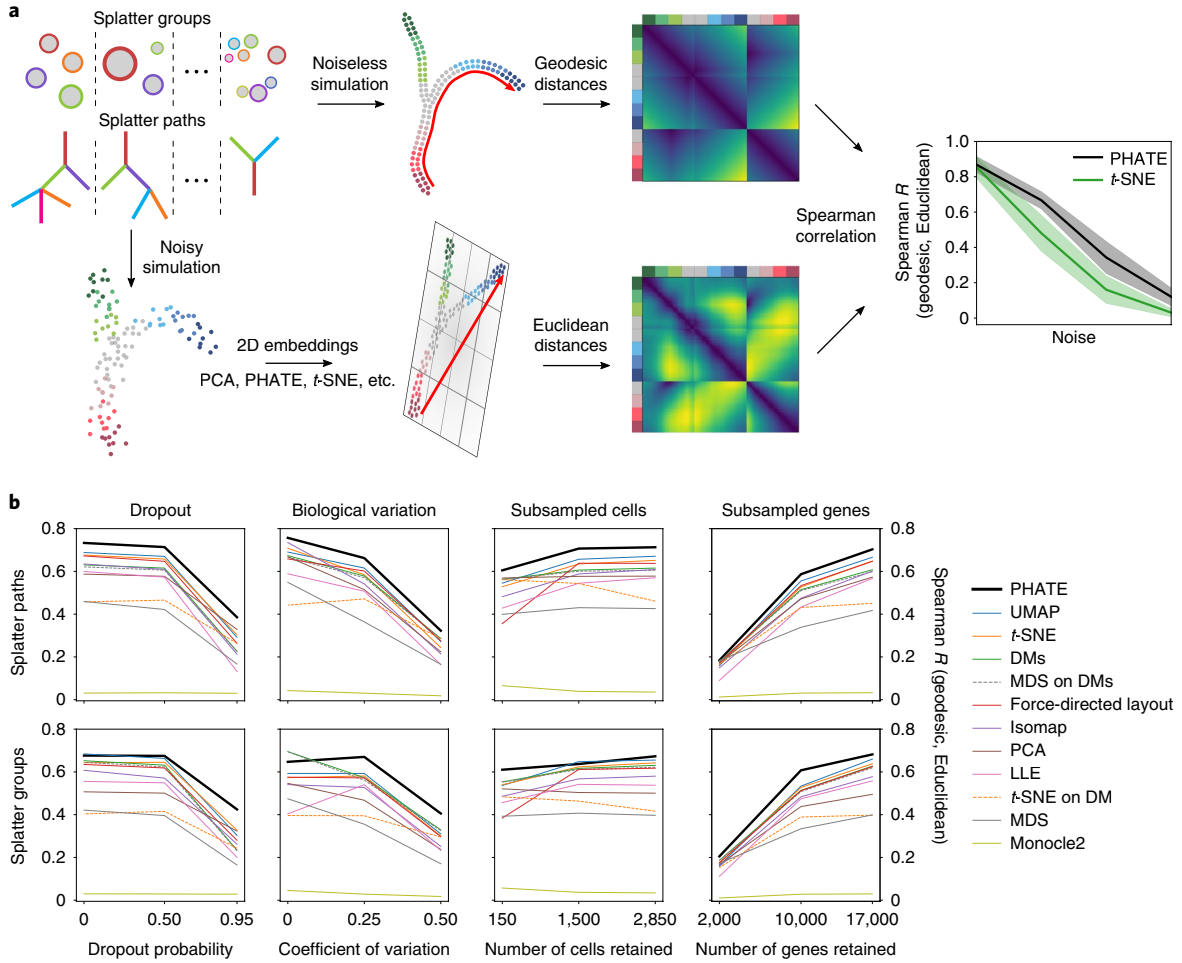

Fig. 4 | PHATE most accurately represents manifold distances in a 2D embedding. a, Schematic of the performance comparison. For each method and each type of corruption, Euclidean distances in the 2D embedding are compared to geodesic distances in an equivalent noiseless simulation using Spearman correlation. **b**, Performance of 12 different methods, such as UMAP, *t*-SNE, and local linear embedding (LLE)[6], across varying levels of corruption by dropout, decreased signal-to-noise ratio (BCV), randomly subsampled cells (subsample) and randomly subsampled genes (n_genes). The mean correlation of 20 runs for each configuration is shown. For further details see Supplementary Table 3. DMs, diffusion maps.

should be as well separated as possible; and (2) denoising, such that the low-dimensional embedding accurately represents the ground-truth data and is as invariant as possible to biological and technical noise. DEMaP encapsulates each of these properties by comparing the geodesic distances on the noiseless data to the Euclidean distances of the embedding extracted from noisy data. An overview of DEMaP is presented in Fig. 4a (Methods).

To compare the performance of PHATE to 12 dimensionality-reduction methods, we simulated scRNA-seq data from Splatter[25]. Splatter uses a parametric model to generate data with various structures, such as branches or clusters. This simulated data provides a ground-truth reference to which we can add various types of noise. We then use this noisy data as input for each dimensionality-reduction algorithm, and quantify the degree to which each representation preserves local and global structures and denoises the data using DEMaP. To generate a diverse set of ground-truth references, we simulated 50 datasets containing clusters and 50 datasets containing branches (Methods).

For each method, we used the default parameters and calculated DEMaP on each simulated dataset using different noise settings. The results are presented in Fig. 4b and Supplementary Table 3. We found that PHATE had the highest DEMaP score in 22 of 24 comparisons and was the top-performing method overall. Uniform manifold approximation and projection (UMAP) was the second best performing method overall but had the highest DEMaP score in only two of the comparisons, one of which is equal with PHATE. We ran further tests on cluster data using the adjusted Rand index[26] and found that on average PHATE preserves local cluster structure as well as, or better than, *t*-SNE, UMAP and PCA (Supplementary Fig. 9). From all of these results, we conclude that PHATE captures the true structure of high-dimensional data more accurately than existing visualization methods.

Qualitative comparisons. In addition to the quantitative comparison, we can visually compare the embeddings provided by different methods. Figure 5 shows a comparison of the PHATE visualization to seven other methods on five single-cell datasets with known trajectory (Fig. 5a,d,e) and cluster (Fig. 5b,c) structures. We see that PHATE provides a clean and relatively denoised visualization of the data that highlights both the local and global structure: local clusters or branches are visually connected to each other in a global structure in each of the PHATE visualizations. Many of these branches are consistent with cell types or clusters validated by the authors[17,18,27,28] and are also present in other visualizations such as force-directed layout and *t*-SNE, suggesting that the structures in the PHATE embedding reflect true structure in the dataset.

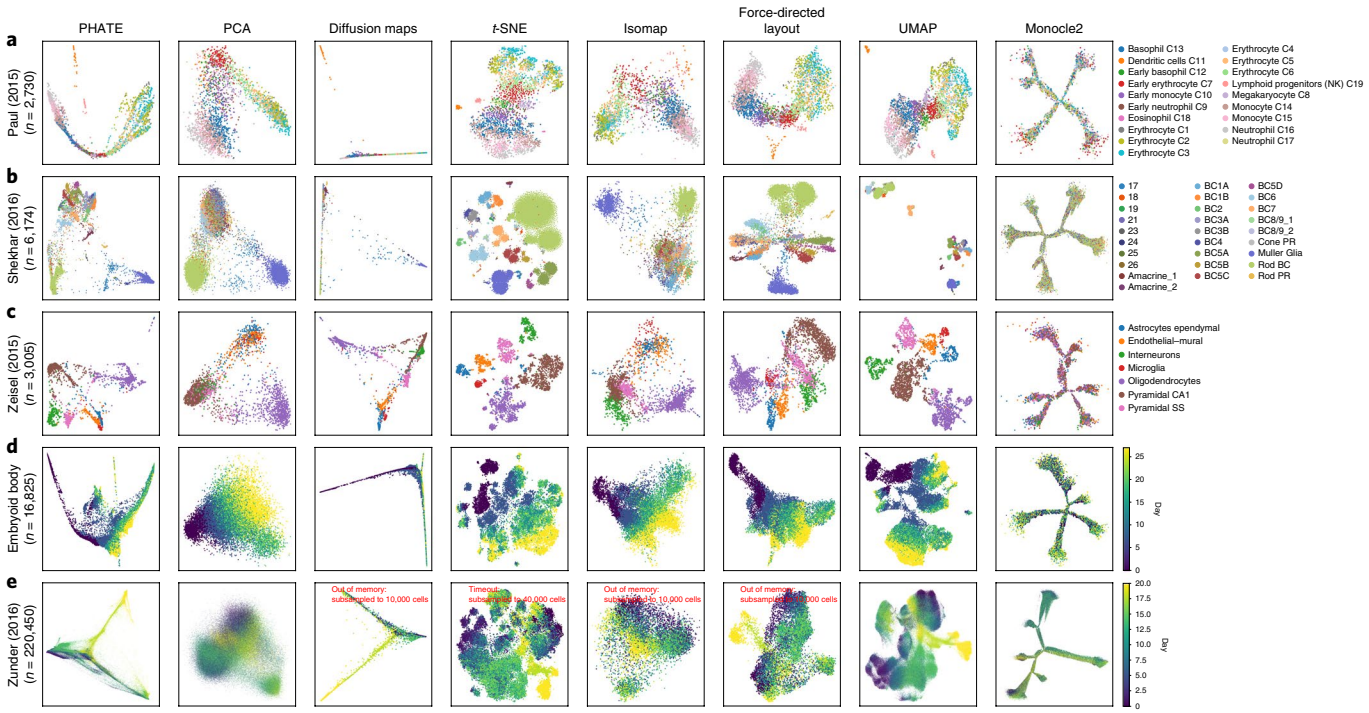

Fig. 5 | Comparison of PHATE to other visualization methods on biological datasets. Columns represent different visualization methods, rows different datasets.

However, force-directed layouts tends to give a noisier visualization with fewer clear branches. Additionally, *t*-SNE[22] tends to shatter trajectories into clusters, creating the false impression that the data contain natural clusters. We characterize each of these visualizations in detail in Supplementary Note 2.

We obtained similar results by comparing PHATE to eleven methods on nine non-biological datasets, including four artificial datasets where the ground truth was known (Supplementary Fig. 3). Expanded comparisons on single-cell data, including additional datasets and visualization methods, are also included in Supplementary Fig. 8. See Supplementary Note 2 for a full discussion of each method in all of these comparisons.

Data exploration with PHATE. PHATE can reveal the underlying structure of the data for a variety of data types. Supplementary Note 3 discusses PHATE applied to multiple different datasets, including single-nucleotide polymorphism data, microbiome data, Facebook network data, Hi-C chromatin conformation data and facial images (Supplementary Figs. 10 and 11). In this section, however, we show the insights gained through PHATE visualization of this structure for single-cell data (see Methods for details on preprocessing steps).

We show that the identifiable trajectories in the PHATE visualization have biological meaning that can be discerned from the patterns of gene expression and the mutual information between gene expression and the ordering of cells along the trajectories. We analyzed the mouse-bone-marrow scRNA-seq[17] and iPSC CyTOF[18] datasets described previously. Our analysis of the iPSC CyTOF data is presented here while the analysis of the mouse-bone-marrow data is presented in Supplementary Note 3. For both of these datasets, we used our new methods for detecting branches and branch points. We then ordered the cells within each trajectory using Wanderlust[29] applied to higher-dimensional PHATE coordinates. We note that ordering could also be based on other pseudotime-ordering software[14,30–33]. To estimate the strength of the relationship between gene expression and cell ordering along branches, we estimated the DREMI score (a weighted mutual information that eliminates biases to reveal shape-agnostic relationships between two variables[34]) between gene expression and the Wanderlust-based ordering within each branch. Genes with a high DREMI score within a branch are changing along the branch. We also used PHATE to analyze the transcriptional heterogeneity in rod bipolar cells to demonstrate the ability of PHATE to preserve cluster structure (Supplementary Note 3 and Supplementary Fig. 12a).

Supplementary Figure 7c shows the mass-cytometry dataset from ref. [18] that shows cellular reprogramming of mouse embryonic fibroblasts to iPSCs with Oct4–GFP at a single-cell resolution. The protein markers measure pluripotency, differentiation, cell-cycle and signaling status. The cellular embedding (with combined timepoints) by PHATE shows a unified embedding that contains five main branches, further segmented in our visualization, each corresponding to the identified biology[18]. Branch 2 contains early reprogramming intermediates with the correct set of reprogramming factors $Sox2^{+}Oct4^{+}Klf4^{+}Nanog^{+}$ and with relatively low CD73 at the beginning of the branch. Branch 2 splits into two additional branches. Branches 4 and 6 (Supplementary Fig. 7) show the successful reprogramming to embryonic stem (ES)-cell-like lineages expressing markers such as Nanog, Oct4, Lin28, Ssea1 and Epcam that are associated with the transition to pluripotency[35]. Branch 5 shows a lineage that is refractory to reprogramming, does not express pluripotency markers and is referred to as "mesoderm-like"[18].

The other branches are similarly analyzed in Supplementary Note 3. In addition, the data features can be reweighted to obtain specific 'views' of the data (Supplementary Note 3 and Supplementary Fig. 13).

PHATE analysis of human ES cell differentiation data. To test the ability of PHATE to provide novel insights in a complex biological system, we generated and analyzed scRNA-seq data from human ES cells differentiating as embryoid bodies (EBs)[36], a system that has never before been extensively analyzed at the single-cell level. EB differentiation is thought to recapitulate key aspects of early

embryogenesis and has been successfully used as the first step in differentiation protocols for certain types of neurons, astrocytes and oligodendrocytes[37–40], hematopoietic, endothelial and muscle cells[41–49], hepatocytes and pancreatic cells[50,51], and germ cells[52,53]. However, the developmental trajectories through which these early lineage precursors emerge from human ES cells, as well as their cellular and molecular identities, remain largely unknown, particularly in human models.

We measured approximately 31,000 cells, equally distributed over a 27-d differentiation time course (Supplementary Fig. 14a; Methods). Samples were collected at 3-d intervals and pooled for measurement on the 10x Chromium platform. The PHATE embedding of the EB data revealed a highly ordered and clean cellular structure dominated by continuous progressions (Figs. 1c and 6a), unlike other methods such as PCA or *t*-SNE (Supplementary Fig. 8). Exploratory analysis of this system using PHATE uncovered a comprehensive map of four major germ layers with both known and new differentiation intermediates that were not captured with other visualization methods.

A comprehensive lineage map of embryoid bodies from PHATE. Importantly, PHATE retained global structure and organization of the data as evidenced by the retention of a strong time trend in the embedding, although sample time was not included in creating the embedding. Furthermore, PHATE revealed greater phenotypic diversity at later time points as seen by the larger space encompassed by the embedding at days 18 to 27 (Fig. 1c).

This phenotypic heterogeneity was further analyzed by both an automated analysis (Supplementary Note 4, Fig. 6a and Supplementary Tables 4 and 5) and by manual examination of the embedding in conjunction with the established literature on germ-layer development (Supplementary Fig. 14b). For the manual analyses, we used 80 markers from the literature to identify populations along the PHATE map, which gave rise to a detailed germ-layer specification map (Fig. 6b and Supplementary Videos 1, 2 and 3). These populations are shown on the PHATE visualization in Fig. 6c. In the lineage tree, the dots are the populations and the arrows represent transitions between the populations. Our map shows in detail how human ES cells give rise to germ-layer derivatives via a continuum of defined intermediate states.

Novel transitional populations in embryoid bodies. The comprehensive nature of the lineage map generated from the PHATE embedding allowed us to identify novel transitional populations that have not yet been characterized. Three new precursor states were identified in both manual and automated analyses: a bipotent neural crest and neural progenitor precursor, an endodermal precursor and a cardiac precursor.

Within the ectodermal lineage, differentiation begins with the induction of preneuroectoderm state characterized by downregulaton of *POU5F1* and induction of *OTX2*. This state is resolved into two precursors, neuroectoderm 1 (expressing *GBX2*, *ZIC2* and *ZIC5*) and neuroectoderm 2 (expressing *GBX2*, *OLIG2* and *HOXD1*). While neuroectoderm 1 appeared to develop along the canonical neuroectoderm specification route and expressed a set of well-established anterior neuroectoderm markers (*ZIC2*, *ZIC5*, *PAX6*, *GLI3*, *SIX3* and *SIX6*), neuroectoderm 2 gave rise to a bipotent precursor expressing *HOXA2* and *HOXB1* that subsequently separated into the neural crest and neural progenitor branches. Given its potential to generate both neuroectoderm and neural crest cell types, the precursor expressing *HOXA2* and *HOXB1* could represent the equivalent of the neural plate border cells that have been defined in model organisms[54,55].

Within the endoderm branch, the canonical precusors expressing *EOMES*, *FOXA2* and *SOX17* was clustered together with a new precusor that expressed *GATA3*, *SATB1* and *KLF8* but did not express *EOMES* or *FOXA2*, which further differentiated into cells expressing the posterior endoderm markers *NKX2-1*, *CDX2*, *ASCL2* and *KLF5*. Finally, a new cardiac precursor cell expressing *T (TBXT)*, *GATA4*, *CER1* and *PROX1* was identified within the mesoderm lineage that gave rise to cells expressing *TNNT2* via a differentiation intermediate that expressed *GATA6* and *HAND1*.

A more detailed analysis of the new and canonical cell types derived from the PHATE embedding is given in Supplementary Note 4.

Experimental validation of PHATE-identified lineages. We next used the ability of PHATE to extract data on specific regions within the visualization to define a set of surface markers for the isolation and molecular characterization of specific cell populations within the EB differentiation process.

We focused on two specific regions that correspond to the neural crest branch (sub-branch iii; Fig. 6a) and cardiac precursor sub-branch within the mesoderm branch (sub-branch vii; Fig. 6a). Differential expression analysis identified a set of candidate markers for each region (Figs. 6d,e). We focused on markers with a high Earth mover's distance (EMD; Methods)[56] score in the targeted sub-branch and low EMD scores in all other sub-branches. On the basis of these analyses and the availability of antibodies, *ITGA4* (also known as *CD49D*) was chosen for the neural crest (the highest scoring surface marker for sub-branch iii) while *F3* (also known as *CD142*) and *CD82* were chosen for cardiac precursors (among the top 6% of surface markers and the top 3% of all genes by EMD). We FACS purified $CD49d^{+}CD63^{-}$ and $CD82^{+}CD142^{+}$ cells and performed bulk RNA-seq (Supplementary Fig. 14f) on these sorted populations.

To verify that we isolated the correct regions, we calculated the Spearman correlation between the gene-expression pattern of each cell and the bulk RNA-seq data from the $CD49d^{+}CD63^{-}$ sorted cells (Figs. 6f and Supplementary Fig. 14d). The correlation coefficient was highest in the neural crest branch (sub-branch iii), which corresponded to the highest expression of CD49d. Similar results were obtained for the cardiac precursor cells (Figs. 6f and Supplementary Fig. 14e).

Taken together, our analyses show that PHATE has the potential to greatly accelerate the pace of biological discovery by suggesting hypotheses in the form of finely grained populations and identifying markers with which to isolate populations. These populations can be probed further using alternative measurements such as epigenetic or protein-expression assays.

Discussion

With large amounts of high-dimensional, high-throughput biological data being generated in many types of biological systems, there is a growing need for interpretable visualizations that can represent structures in data without strong prior assumptions. However, most existing methods are highly deficient at retaining structures of interest in biology. These include clusters, trajectories or progressions of various dimensionality, hybrids of the two, as well as local and global nonlinear relations in data. Furthermore, existing methods have trouble contending with the sizes of modern datasets and the high degree of noise inherent to biological datasets. PHATE provides a unique solution to these problems by creating a diffusion-based informational geometry from the data, and by preserving a divergence metric between data points that is sensitive to near and far manifold-intrinsic distances in the data space. Additionally, PHATE is able to offer clean and denoised visualizations because the information geometry created in PHATE is based on data diffusion dynamics, which are robust to noise. Thus, PHATE reveals intricate local as well as global structure in a denoised way.

We applied PHATE to a wide variety of datasets, including single-cell CyTOF and RNA-seq data, as well as gut microbiome and single-nucleotide polymorphism data, where the data points are

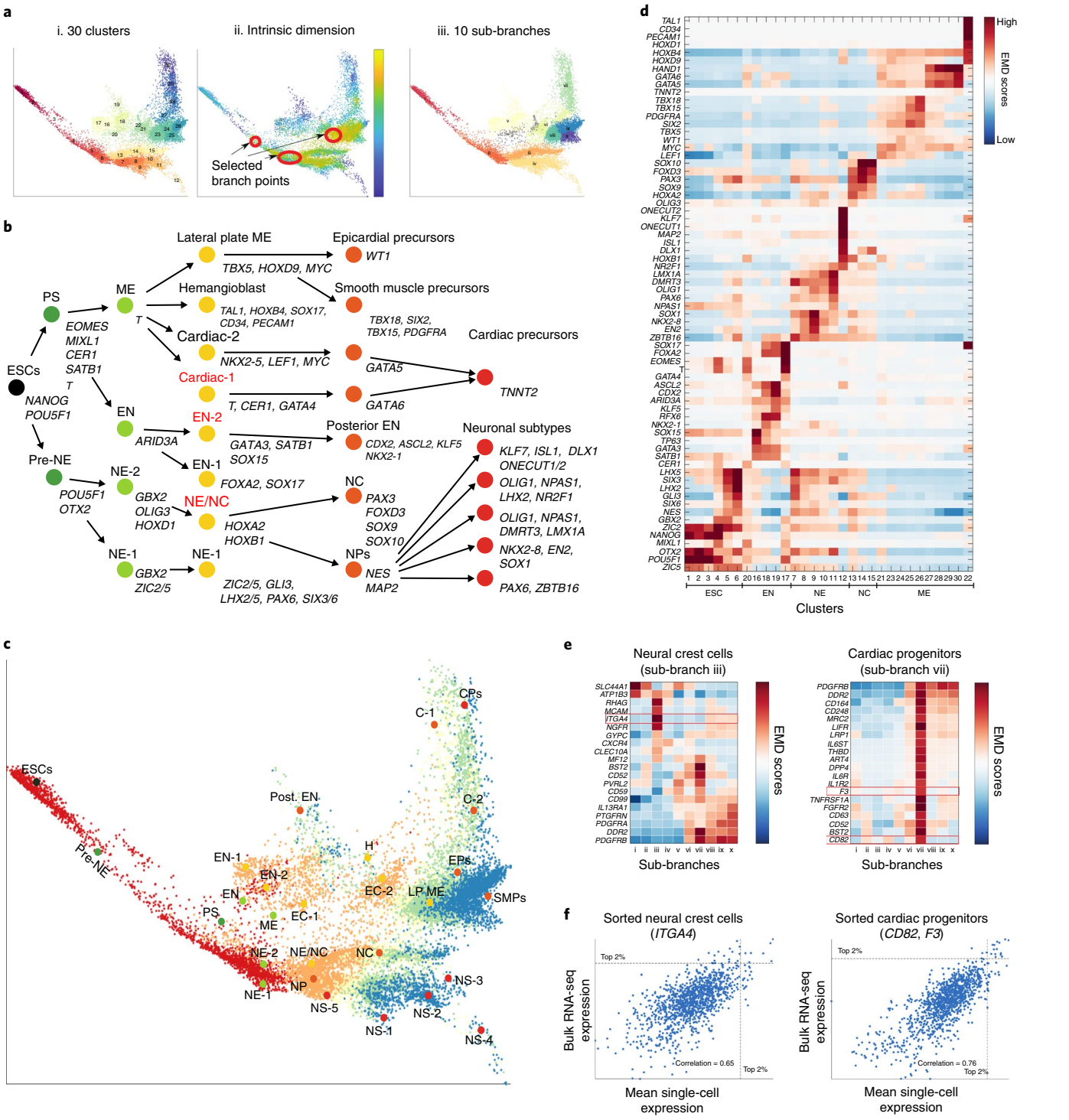

Fig. 6 | PHATE analysis of embryoid body scRNA-seq data with $n = 16,825$ cells. a, Intrinsic dimension of PHATE embedding identifes branches. i, The PHATE visualization colored by clusters;. Clustering is done on a ten-dimensional PHATE embedding. The number of cells in each cluster is given in Supplementary Table 5. ii) The PHATE visualization colored by estimated local intrinsic dimensionality with selected branch points highlighted. iii) Branches and sub-branches chosen from contiguous clusters for analysis. **b**, Lineage tree of the EB system determined from the PHATE analysis showing ES cells (ESCs), the primitive streak (PS), mesoderm (ME), endoderm (EN), neuroectoderm (NE), neural crest (NC), neural progenitors (NPs) and others. Red font indicates new cell precursors. See Supplementary Videos 1, 2 and 3 for 3D PHATE visualizations of each stage in the tree. **c**, PHATE embedding overlaid with each of the populations in the lineage tree. Other abbreviations include lateral plate mesoderm (LP ME), hemangioblast (H), cardiac (C), epicardial precursors (EPs), smooth muscle precursors (SMPs), cardiac precursors (CPs) and neuronal subtypes (NS). **d**, Heat map showing the EMD score between the cluster distribution and the background distribution for each gene. Relevant genes for identifying the main lineages were manually identified. Genes are organized according to their maximum EMD score. The number of cells in each cluster is given in Supplementary Table 5. **e**, The EMD scores of the top scoring surface markers in the targeted sub-branches (sub-branches iii and vii). **f**, Scatter plots of the bulk transcription factor expression versus the mean single-cell transcription factor expression in sub-branches iii (left, $n = 2,537$ cells) and vii (right, $n = 1,314$ cells). The Spearman correlation coefficients are calculated for $n = 1,213$ transcription factors.

subjects rather than cells. We also tested PHATE on network data, such as Hi-C and Facebook networks. In each case, PHATE was able to reveal structures of visual interest to humans that other methods entirely miss. Moreover, we have implemented PHATE in a scalable way that enables it to process millions of data points in a matter of hours. Hence, PHATE can efficiently handle the datasets that are now being produced using scRNA-seq technologies.

To showcase the ability of PHATE to explore data generated in new systems, we applied PHATE to our newly generated human EB differentiation dataset consisting of roughly 31,000 cells sampled over a differentiation time course. We found that PHATE successfully resolves cellular heterogeneity and correctly maps all germ-layer lineages and branches on the basis of scRNA-seq data alone, without any additional assumptions on the data. Through detailed subpopulation and gene-expression analysis along these branches we identified both canonical and new differentiation intermediates. The insights obtained with PHATE in this system will be a valuable resource for researchers working on early human development, human ES cells and their regenerative medicine applications.

We expect numerous biological, but also non-biological, data types to benefit from PHATE, including applications in high-throughput genomics, phenotyping and many other fields. As such, we believe that PHATE will revolutionize biomedical data exploration by offering a new way of visualizing, exploring and extracting information from large-scale high-dimensional data.

Online content

Any methods, additional references, Nature Research reporting summaries, source data, extended data, supplementary information, acknowledgements, peer review information; details of author contributions and competing interests; and statements of data and code availability are available at https://doi.org/10.1038/s41587-019-0336-3.

Received: 2 October 2018; Accepted: 29 October 2019;
Published online: 3 December 2019

References

1. van der Maaten, L. & Hinton, G. Visualizing data using *t*-SNE. *J. Mach. Learn. Res.* **9**, 2579–2605 (2008).
2. Amir, E. D. et al. viSNE enables visualization of high dimensional single-cell data and reveals phenotypic heterogeneity of leukemia. *Nat. Biotechnol.* **31**, 545–552 (2013).
3. Linderman, G. C., Rachh, M., Hoskins, J. G., Steinerberger, S. & Kluger, Y. Fast interpolation-based *t*-SNE for improved visualization of single-cell RNA-seq data. *Nat. Methods* **16**, 243–245 (2019).
4. Tenenbaum, J. B., De Silva, V. & Langford, J. C. A global geometric framework for nonlinear dimensionality reduction. *Science* **290**, 2319–2323 (2000).
5. Becht, E. et al. Dimensionality reduction for visualizing single-cell data using UMAP. *Nat. Biotechnol.* **37**, 38–44 (2019).
6. Roweis, S. T. & Saul, L. K. Nonlinear dimensionality reduction by locally linear embedding. *Science* **290**, 2323–2326 (2000).
7. Cox, T. F. & Cox, M. A. A. *Multidimensional Scaling* 2nd edn (Chapman & Hall/CRC, 2001).
8. De Silva, V. & Tenenbaum J. B. *Sparse Multidimensional Scaling Using Landmark Points* (Stanford University, 2004).
9. Unen, V. et al. Visual analysis of mass cytometry data by hierarchical stochastic neighbour embedding reveals rare cell types. *Nat. Commun.* **8**, 1740 (2017).
10. Chen, L. & Buja, A. Local multidimensional scaling for nonlinear dimension reduction, graph drawing, and proximity analysis. *J. Am. Stat. Assoc.* **104**, 209–219 (2009).
11. Moon, T. K. & Stirling, W. C. *Mathematical Methods and Algorithms for Signal Processing* (Prentice Hall, 2000).
12. Qiu, X. et al. Reversed graph embedding resolves complex single-cell trajectories. *Nat. Methods* **14**, 979–982 (2017).
13. Coifman, R. R. & Lafon, S. Diffusion maps. *Appl. Comput. Harmon. Anal.* **21**, 5–30 (2006).
14. Haghverdi, L., Buettner, M., Wolf, F. A., Buettner, F. & Theis, F. J. Diffusion pseudotime robustly reconstructs lineage branching. *Nat. Methods* **13**, 845–848 (2016).
15. Darrow, E. M. et al. Deletion of *DXZ4* on the human inactive X chromosome alters higher-order genome architecture. *Proc. Natl Acad. Sci. USA* **113**, E4504–E4512 (2016).
16. Cheng, X., Rachh, M. & Steinerberger, S. On the diffusion geometry of graph Laplacians and applications. *Appl. Comput. Harmon. Anal.* **46**, 674–688 (2019).
17. Paul, F. et al. Transcriptional heterogeneity and lineage commitment in myeloid progenitors. *Cell* **163**, 1663–1677 (2015).
18. Zunder, E. R., Lujan, E., Goltsev, Y., Wernig, M. & Nolan, G. P. A continuous molecular roadmap to iPSC reprogramming through progression analysis of single-cell mass cytometry. *Cell Stem Cell* **16**, 323–337 (2015).
19. Lui, K., Ding, G. W., Huang, R. & McCann, R. Dimensionality reduction has quantifiable imperfections: two geometric bounds. In *Proc. 32nd International Conference on Neural Information Processing Systems* (Eds. Bengio, S. et al.) 8453–8463 (Curran Associates, 2018).
20. Tsai, F. S. A visualization metric for dimensionality reduction. *Expert Syst. Appl.* **39**, 1747–1752 (2012).
21. Bertini, E., Tatu, A. & Keim, D. Quality metrics in high-dimensional data visualization: an overview and systematization. *IEEE Trans. Vis. Comput. Graph.* **17**, 2203–2212 (2011).
22. Maaten, Lvd, Postma, E. & Herik, Jvd Dimensionality reduction: a comparative review. *J. Mach. Learn. Res.* **10**, 66–71 (2009).
23. Vankadara, L. C. & von Luxburg, U. Measures of distortion for machine learning. In *Proc. 32nd International Conference on Neural Information Processing Systems* (Eds. Bengio, S. et al.) 4886–4895 (Curran Associates, 2018).
24. Saelens, W., Cannoodt, R., Todorov, H. & Saeys, Y. A comparison of single-cell trajectory inference methods. *Nat. Biotechnol.* **37**, 547–554 (2019).
25. Zappia, L., Phipson, B. & Oshlack, A. Splatter: simulation of single-cell RNA sequencing data. *Genome Biol.* **18**, 174 (2017).
26. Rand, W. M. Objective criteria for the evaluation of clustering methods. *J. Am. Stat. Assoc.* **66**, 846–850 (1971).
27. Shekhar, K. et al. Comprehensive classification of retinal bipolar neurons by single-cell transcriptomics. *Cell* **166**, 1308–1323 (2016).
28. Zeisel, A. et al. Cell types in the mouse cortex and hippocampus revealed by single-cell RNA-seq. *Science* **347**, 1138–1142 (2015).
29. Bendall, S. C. et al. Single-cell trajectory detection uncovers progression and regulatory coordination in human B cell development. *Cell* **157**, 714–725 (2014).
30. Setty, M. et al. Wishbone identifies bifurcating developmental trajectories from single-cell data. *Nat. Biotechnol.* **34**, 637–645 (2016).
31. Liiv, I. Seriation and matrix reordering methods: an historical overview. *Stat. Anal. Data Min.* **3**, 70–91 (2010).
32. Hahsler, M., Hornik, K. & Buchta, C. Getting things in order: an introduction to the R package seriation. *J. Stat. Soft.* **25**, 1–34 (2008).
33. Wolf, F. A. et al. PAGA: graph abstraction reconciles clustering with trajectory inference through a topology preserving map of single cells. *Genome Biol.* **20**, 59 (2019).
34. Krishnaswamy, S. et al. Conditional density-based analysis of T cell signaling in single-cell data. *Science* **346**, 1250689 (2014).
35. Polo, J. M. et al. A molecular roadmap of reprogramming somatic cells into iPS cells. *Cell* **151**, 1617–1632 (2012).
36. Martin, G. R. & Evans, M. J. Differentiation of clonal lines of teratocarcinoma cells: formation of embryoid bodies in vitro. *Proc. Natl Acad. Sci. USA* **72**, 1441–1445 (1975).
37. Bibel, M., Richter, J., Lacroix, E. & Barde, Y.-A. Generation of a defined and uniform population of CNS progenitors and neurons from mouse embryonic stem cells. *Nat. Protocols* **2**, 1034–1043 (2007).
38. Kang, S.-M. et al. Efficient induction of oligodendrocytes from human embryonic stem cells. *Stem Cells* **25**, 419–424 (2007).
39. Zhao, X., Liu, J. & Ahmad, I. Differentiation of embryonic stem cells to retinal cells in vitro. In *Embryonic Stem Cell Protocols: Differentiation Models* Vol. 2 (Ed. Turksen, K.) 401–416 (Humana Press, 2006).
40. Liour, S. S. et al. Further characterization of embryonic stem cell-derived radial glial cells. *Glia* **53**, 43–56 (2006).
41. Nakano, T., Kodama, H. & Honjo, T. In vitro development of primitive and definitive erythrocytes from different precursors. *Science* **272**, 722 (1996).
42. Nishikawa, S.-I., Nishikawa, S., Hirashima, M., Matsuyoshi, N. & Kodama, H. Progressive lineage analysis by cell sorting and culture identifies FLK1$^+$ VE-cadherin$^+$ cells at a diverging point of endothelial and hemopoietic lineages. *Development* **125**, 1747–1757 (1998).
43. Wiles, M. V. & Keller, G. Multiple hematopoietic lineages develop from embryonic stem (ES) cells in culture. *Development* **111**, 259–267 (1991).
44. Potocnik, A. J., Nielsen, P. J. & Eichmann, K. In vitro generation of lymphoid precursors from embryonic stem cells. *EMBO J.* **13**, 5274 (1994).
45. Tsai, M. et al. In vivo immunological function of mast cells derived from embryonic stem cells: an approach for the rapid analysis of even embryonic lethal mutations in adult mice in vivo. *Proc. Natl Acad. Sci. USA* **97**, 9186–9190 (2000).

46. Fairchild, P. et al. Directed differentiation of dendritic cells from mouse embryonic stem cells. *Curr. Biol.* **10**, 1515–1518 (2000).
47. Yamashita, J. et al. Flk1-positive cells derived from embryonic stem cells serve as vascular progenitors. *Nature* **408**, 92–96 (2000).
48. Maltsev, V. A., Rohwedel, J., Hescheler, J. & Wobus, A. M. Embryonic stem cells differentiate in vitro into cardiomyocytes representing sinusnodal, atrial and ventricular cell types. *Mech. Dev.* **44**, 41–50 (1993).
49. Rohwedel, J. et al. Muscle cell differentiation of embryonic stem cells reflects myogenesis in vivo: developmentally regulated expression of myogenic determination genes and functional expression of ionic currents. *Dev. Biol.* **164**, 87–101 (1994).
50. Kania, G., Blyszczuk, P., Jochheim, A., Ott, M. & Wobus, A. M. Generation of glycogen- and albumin-producing hepatocyte-like cells from embryonic stem cells. *Biol. Chem.* **385**, 943–953 (2004).
51. Schroeder, I. S., Rolletschek, A., Blyszczuk, P., Kania, G. & Wobus, A. M. Differentiation of mouse embryonic stem cells to insulin-producing cells. *Nat. Protocols* **1**, 495–507 (2006).
52. Geijsen, N. et al. Derivation of embryonic germ cells and male gametes from embryonic stem cells. *Nature* **427**, 148–154 (2004).
53. Kehler, J., Hübner, K., Garrett, S. & Schöler, H. R. Generating oocytes and sperm from embryonic stem cells. *Semin. Reprod. Med.* **23**, 222–233 (2005).
54. Betancur, P., Bronner-Fraser, M. & Sauka-Spengler, T. Assembling neural crest regulatory circuits into a gene regulatory network. *Annu. Rev. Cell Dev. Biol.* **26**, 581–603 (2010).
55. Barembaum, M. & Bronner-Fraser, M. Early steps in neural crest specification. *Semin. Cell Dev. Biol.* **16**, 642–646 (2005).
56. Treleaven, K. & Frazzoli, E. An explicit formulation of the earth movers distance with continuous road map distances. Preprint at *arXiv* https://arxiv.org/abs/1309.7098 (2013).
57. Lieberman-Aiden, E. et al. Comprehensive mapping of long-range interactions reveals folding principles of the human genome. *Science* **326**, 289–293 (2009).

Methods

Here we present an expanded explanation of our computational methods, experimental methods and data-processing steps. For the computational details, we first provide a detailed overview of the PHATE algorithm followed by a robustness analysis of PHATE with respect to the parameters and the number of data points. We then provide details on the scalable version of PHATE, identifying branch points and branches and the EMD score analysis.

The embedding provided by PHATE is designed for visualizing global and local structure in the data in exploratory settings with the following properties in mind: (1) the visualization should capture the relevant structure in low (two to three) dimensions; (2) the visualization should preserve and emphasize global and local structure including transitions and clusters; (3) the visualization is denoised to enable data exploration; and (4) the visualization is robust in the sense that the revealed structure is insensitive to user configurations.

The mathematical steps of PHATE are provided in Supplementary Table 1. We now provide further details about each of the steps in the PHATE algorithm and explain how these steps ensure that PHATE meets the four properties described above. For more mathematical details of the algorithm, see Supplementary Note 1.

Distance preservation. Consider the common approach of linearly embedding the raw data matrix itself, for example, with PCA, to preserve the global structure of the data. PCA finds the directions of the data that capture the largest global variance. However, in most cases local transitions are noisy and global transitions are nonlinear. Therefore, linear notions such as global variance maximization are insufficient to capture latent patterns in the data, and they typically result in a noisy visualization (Supplementary Fig. 3). To provide reliable structure preservation that emphasizes transitions in the data, we need to consider the intrinsic structure of the data. This implies and motivates preserving distances between data points (for example, cells) that consider gradual changes between them along these nonlinear transitions (Fig. 2a,b).

Local affinities and the diffusion operator. A standard choice for a distance metric is the Euclidean distance. However, global Euclidean distances are not reflective of transitions in the data, especially in biological datasets that have nonlinear and noisy structures. For instance, cells sampled from a developmental system, such as hematopoiesis or ES cell differentiation, show gradual changes where adjacent cells are only slightly different from each other. But these changes quickly aggregate into nonlinear transitions in marker expression along each developmental path. Therefore, we transform the global Euclidean distances into local affinities that quantify the similarities between nearby (in the Euclidean space) data points (Fig. 2c).

A common approach to transforming global (for example, Euclidean) distances to local similarities is to apply a kernel function to all pairs of points. A popular kernel function is the Gaussian kernel $k_\varepsilon(x,y)=\exp(-\|x-y\|^2/\varepsilon)$ that quantifies the similarity between the two points x and y on the basis of their Euclidean distance. The bandwidth ε determines the radius (or spread) of neighborhoods captured by this kernel. Let $\mathcal{X}\subset\mathbb{R}^d$ be a dataset with N independent and identically distributed points sampled from a probability distribution $p:\mathbb{R}^d\to[0,\infty)$ (with $\int p(\mathbf{x})d\mathbf{x}=1$) that is essentially supported on a low dimensional manifold $\mathcal{M}^m\subseteq\mathbb{R}^d$, where m is the dimension of $\mathcal{M}$ and $m\ll d$. A kernel matrix that includes all pairwise measures of local affinity is constructed by computing the kernel function between all pairs of points in $\mathcal{X}$.

Embedding local affinities directly can result in a loss of global structure as is evident in t-SNE (Figs. 1 and 5, and Supplementary Figs. 3 and 8) or kernel PCA embeddings. For example, t-SNE only preserves data clusters, but not transitions between clusters, as it does not enforce any preservation of global structure. By contrast, a faithful structure-preserving embedding (and visualization) needs to go beyond local affinities (or distances), and also consider global relations between parts of the data. To accomplish this, PHATE is based on constructing a diffusion geometry to learn and represent the shape of the data[13,58,59]. This construction is based on computing local similarities between data points, and then walking or diffusing through the data using a Markovian random-walk diffusion process to infer more global relations (Fig. 2d).

The initial probabilities in this random walk are calculated by normalizing the row-sums of the kernel matrix. In the case of the Gaussian kernel described above, we obtain the following:

$$\nu_\varepsilon(x)=\|k_\varepsilon(\mathbf{x},\cdot)\|_1=\sum_{\mathbf{z}\in\mathcal{X}}k_\varepsilon(\mathbf{x},\mathbf{z}) \tag{1}$$

resulting in a $N\times N$ row-stochastic matrix

$$[\mathbf{P}_\varepsilon]_{(\mathbf{x},\mathbf{y})}=\frac{k_\varepsilon(\mathbf{x},\mathbf{y})}{\nu_\varepsilon(\mathbf{x})},\qquad \mathbf{x},\mathbf{y}\in\mathcal{X} \tag{2}$$

The matrix $\mathbf{P}_\varepsilon$ is a Markov transition matrix where the probability of moving from x to y in a single time step is given by $\Pr[\mathbf{x}\to\mathbf{y}]=[\mathbf{P}_\varepsilon]_{(\mathbf{x},\mathbf{y})}$. This matrix is also referred to as the diffusion operator.

The α-decaying kernel and adaptive bandwidth. When applying the diffusion map framework to data, the choice of the kernel K and bandwidth ε plays a key role in the results. In particular, choosing the bandwidth corresponds to a tradeoff between encoding global and local information in the probability matrix $\mathbf{P}_\varepsilon$. If the bandwidth is small, then single-step transitions in the random walk using $\mathbf{P}_\varepsilon$ are largely confined to the nearest neighbors of each data point. In biological data, trajectories between major cell types may be relatively sparsely sampled. Thus, if the bandwidth is too small, then the neighbors of points in sparsely sampled regions may be excluded entirely and the trajectory structure in the probability matrix $\mathbf{P}_\varepsilon$ will not be encoded. Conversely, if the bandwidth is too large, then the resulting probability matrix $\mathbf{P}_\varepsilon$ loses local information as $[\mathbf{P}_\varepsilon]_{(\mathbf{x},\cdot)}$ becomes more uniform for all $\mathbf{x}\in\mathcal{X}$, which may result in an inability to resolve different trajectories. Here we use an adaptive bandwidth that changes with each point to be equal to its kth-nearest-neighbor distance, along with an α-decaying kernel that controls the rate of decay of the kernel.

The original heuristic proposed[13] suggests setting ε to be the smallest distance that still keeps the diffusion process connected. In other words, it is chosen to be the maximal 1-nearest-neighbor distance in the dataset. While this approach is useful in some cases, it is greatly affected by outliers and sparse data regions. Furthermore, it relies on a single manifold with constant dimension as the underlying data geometry, which may not be the case when the data is sampled from specific trajectories rather than uniformly from a manifold. Indeed, the intrinsic dimensionality in such cases differs between midbranch points that mostly capture one-dimensional trajectory geometry, and branching points that capture multiple trajectories crossing each other.

This issue can be mitigated by using a locally adaptive bandwidth that varies on the basis of the local density of the data. A common method for choosing a locally adaptive bandwidth is to use the k-nearest-neighbor (k-NN) distance of each point as the bandwidth. A point x that is within a densely sampled region will have a small k-NN distance. Thus, local information in these regions is still preserved. By contrast, if x is on a sparsely sampled trajectory, the k-NN distance will be greater and will encode the trajectory structure. We denote the k-NN distance of x as $\varepsilon_k(x)$ and the corresponding diffusion operator as P_k.

A weakness of using locally adaptive bandwidths alongside kernels with exponential tails (for example, the Gaussian kernel) is that the tails become heavier (that is, decay more slowly) as the bandwidth increases. Thus for a point x in a sparsely sampled region where the k-NN distance is large, $[\mathbf{P}_k]_{(x,\cdot)}$ may be close to a fully-supported uniform distribution owing to the heavy tails, resulting in a high affinity with many points that are far away. This can be mitigated by using the following kernel

$$K_{k,\alpha}(\mathbf{x},\mathbf{y})=\frac{1}{2}\exp\left(-\left(\frac{\|\mathbf{x}-\mathbf{y}\|_2}{\varepsilon_k(\mathbf{x})}\right)^\alpha\right)+\frac{1}{2}\exp\left(-\left(\frac{\|\mathbf{x}-\mathbf{y}\|_2}{\varepsilon_k(\mathbf{y})}\right)^\alpha\right) \tag{3}$$

that we call the α-decaying kernel. The exponent α controls the rate of decay of the tails in the kernel $K_{k,\alpha}$. Increasing α increases the decay rate while decreasing α decreases the decay rate. As $\alpha=2$ for the Gaussian kernel, choosing $\alpha>2$ will result in lighter tails in the kernel $K_{k,\alpha}$ as compared to the Gaussian kernel. We denote the resulting diffusion operator as $P_{k,\alpha}$. This is similar to common utilizations of Butterworth filters in signal-processing applications[60]. See Supplementary Fig. 2b for a visualization of the effect of different values of α on this kernel function.

Our use of a locally adaptive bandwidth and the kernel $K_{k,\alpha}$ requires the choice of two tuning parameters: k and α. k should be sufficiently small to preserve local information, that is, to ensure that $[\mathbf{P}_{k,\alpha}]_{(\mathbf{x},\cdot)}$ is not a fully-supported uniform distribution. However, k should also be sufficiently large to ensure that the underlying graph represented by $P_{k,\alpha}$ is sufficiently connected, that is, the probability that we can walk from one point to another within the same trajectory in a finite number of steps is nonzero.

The parameter α should also be chosen with k. α should be sufficiently large so that the tails of the kernel $K_{k,\alpha}$ are not too heavy, especially in sparse regions of the data. However, if k is small when α is large, then the underlying graph represented by $P_{k,\alpha}$ may be too sparsely connected, making it difficult to learn long-range connections. Thus we recommend that α be fixed at a large number (for example, $\alpha\geq 10$) and then k can be sufficiently large to ensure that points are locally connected. In practice, we found that choosing k to be around 5 and α to be about 10 works well for all the datasets presented in this work. However, the PHATE embedding is robust to the choice of these parameters as discussed later.

In addition to progression or trajectory structures, the recommendations provided in this section work well for visualizing data that naturally separate into distinct clusters. In particular, the α-decay kernel ensures that relationships are preserved between distinct clusters that are relatively close to each other.

Propagating affinities via diffusion. Here we discuss diffusion, that is, raising the diffusion operator to its t-th power as shown in Supplementary Table 1 (Fig. 2d). To simplify the discussion we use the notation $\mathbf{P}$ for the diffusion operator, whether defined with a fixed-bandwidth Gaussian kernel or our adaptive kernel. This matrix is referred to as the diffusion operator, since it defines a Markovian diffusion process that essentially only allows single-step transitions within local

data neighborhoods whose sizes depend on the kernel parameters (ε or k and α). In particular, let $\mathbf{x} \in \mathcal{X}$ and let $\boldsymbol{\delta}_x$ be a Dirac at x, that is, a row vector of length N with a one at the entry corresponding to x and zeros everywhere else. The t-step distribution of x is the row in $\mathbf{P}_\varepsilon^t$ corresponding to x

$$p_{\mathbf{x}}^t \triangleq \delta_{\mathbf{x}} \mathbf{P}^t = [\mathbf{P}^t]_{(\mathbf{x},\cdot)} \tag{4}$$

These distributions capture multiscale (where t serves as the scale) local neighborhoods of data points, where locality is considered via random walks that propagate over the intrinsic manifold geometry of the data. This provides a global and robust intrinsic data distance that preserves the overall structure of the data. In addition to learning the global structure, powering the diffusion operator has the effect of low-pass filtering the data such that the main pathways in it are emphasized and small noise dimensions are diminished, thus achieving the denoising objective of our method as well.

Choosing the diffusion time scale *t* with von Neumann entropy. The diffusion time scale t is an important parameter that affects the embedding. The parameter t determines the number of steps taken in a random walk. A larger t corresponds to more steps as compared to a smaller t. Thus, t provides a tradeoff between encoding local and global information in the embedding. The diffusion process can also be viewed as a low-pass filter where local noise is smoothed out on the basis of more global structures. The parameter t determines the level of smoothing. If t is chosen to be too small, then the embedding may be too noisy. On the other hand, if t is chosen to be too large, then some of the signal may be smoothed away.

We formulate a new algorithm for choosing the timescale t. Our algorithm quantifies the information in the powered diffusion operator with various values of t. This is accomplished by computing the spectral or von Neumann entropy (VNE)[61,62] of the powered diffusion operator. The amount of variability explained by each dimension is equal to its eigenvalue in the eigendecomposition of the related (non-Markov) affinity matrix that is conjugate to the Markov diffusion operator. The VNE is calculated by computing the Shannon entropy on the normalized eigenvalues of this matrix. Owing to noise in the data, this value is artificially high for low values of t, and rapidly decreases as one powers the matrix. Thus, we choose values that are around the 'knee' of this decrease.

More formally, to choose t, we first note that its impact on the diffusion geometry can be determined by considering the eigenvalues of the diffusion operator, as the corresponding eigenvectors are not impacted by the time scale. To facilitate spectral considerations and for computational ease, we use a symmetric conjugate

$$[A]_{(\mathbf{x},\mathbf{y})} = \sqrt{\nu(\mathbf{x})}\,[\mathbf{P}]_{(\mathbf{x},\mathbf{y})} / \sqrt{\nu(\mathbf{y})}$$

of the diffusion operator $\mathbf{P}$ with the row-sums ν. This symmetric matrix is often called the diffusion affinity matrix. The VNE of this diffusion affinity is used to quantify the amount of variability. It can be verified that the eigenvalues of A^t are the same as those of $\mathbf{P}^t$, and furthermore these eigenvalues are given by the powers $\{\lambda_i^t\}_{i=1}^{N-1}$ of the spectrum of $\mathbf{P}$. Let $\eta(t)$ be a probability distribution defined by normalizing these (non-negative) eigenvalues as $[\eta(t)]_i = \lambda_i^t / \sum_{j=0}^{N-1} \lambda_j^t$. Then, the VNE $H(t)$ of A^t (and equivalently of $\mathbf{P}^t$) is given by the entropy of $\eta(t)$, that is,

$$H(t) = -\sum_{i=1}^{N} [\eta(t)]_i \log[\eta(t)]_i \tag{5}$$

where we use the convention of $0\log(0) \triangleq 0$. The VNE $H(t)$ is dominated by the relatively large eigenvalues, while eigenvalues that are relatively small contribute little. Therefore, it provides a measure of the number of the relatively significant eigenvalues.

The VNE generally decreases as t increases. As mentioned previously, the initial decrease is primarily due to a denoising of the data as less significant eigenvalues (likely corresponding to noise) decrease rapidly to zero. The more significant eigenvalues (likely corresponding to signal) decrease much more slowly. Thus the overall rate of decrease in $H(t)$ is high initially as the data is denoised but then low for larger values of t as the signal is smoothed. As $t \to \infty$, eventually all but the first eigenvalue decrease to zero and so $H(t) \to 0$.

To choose t, we plot $H(t)$ as a function of t as in the first plot of Supplementary Fig. 2c. Choosing t from among the values where $H(t)$ is decreasing rapidly generally results in noisy visualizations and embeddings (Supplementary Fig. 2c). Very large values of t result in a visualization where some of the branches or trajectories are combined together and some of the signal is lost (fourth plot in Supplementary Fig. 2c). Good PHATE visualizations can be obtained by choosing t from among the values where the decrease in $H(t)$ is relatively slow, that is, the set of values around the 'knee' in the plot of $H(t)$ (Supplementary Fig. 2c and Fig. 1). This is the set of values for which much of the noise in the data has been smoothed away and most of the signal is still intact. The PHATE visualization is fairly robust to the choice of t in this range, as discussed later.

In the code, we include an automatic method for selecting t on the basis of a knee-point detection algorithm that finds the knee by fitting two lines to the VNE curve (https://www.mathworks.com/matlabcentral/fileexchange/35094-knee-point). This algorithm calculates the error between the VNE plot and two lines fitted to the data. The first line has end points at the first VNE value and the suggested knee point. The second line has end points at the suggested knee point and the last VNE value. The suggested knee point with the minimum error is selected.

Potential distances. To resolve instabilities in diffusion distances and embed the global structure captured by the diffusion geometry in low (two or three) dimensions, we use a new diffusion-based informational distance, which we call potential distance (Fig. 2e). It is calculated by computing the distance between log-transformed transition probabilities from the powered diffusion operator. The key insight in formulating the potential distance is that an informational distance between probability distributions is more sensitive to global relationships (between far-away points) and more stable at boundaries of manifolds than straight pointwise comparisons of probabilities (that is, diffusion distances). This is because the diffusion distance is sensitive to differences between the main modes of the diffused probabilities and is largely insensitive to differences in the tails. By contrast, the potential distance, or more generally informational distances, use a submodular function (such as a log) to render distances sensitive to differences in both the main modes and the tails. This gives PHATE the ability to preserve both local and manifold-intrinsic global distances in a way that is optimized for visualization. The resulting metric space also quantifies differences between energy potentials that dominate 'heat' propagation along diffusion pathways (that is, on the basis of the heat-equation diffusion model) between data points, instead of simply considering transition probabilities along them.

The potential distance is inspired by information theory and stochastic dynamics, which are both fields where probability distributions are compared for different purposes. First, in information theory literature, information divergences are used to measure discrepancies between probability distributions in the information space rather than the probability space, as they are more sensitive to differences between the tails of the distributions as described above. Second, when analyzing dynamical systems of moving particles, it is not the pointwise difference between absolute particle counts that is used to compare states, but rather the ratio between these counts. Indeed, in the latter case the Boltzmann distribution law directly relates these ratios to differences in the energy of a state in the system. Therefore, similar to the information theory case, dynamical states are differentiated in energy terms, rather than probability terms. We employ the same reasoning in our case by defining our potential distance using localized diffusion energy potentials, rather than diffusion transition probabilities.

To go from the probability space to the energy (or information) space, we log transform the probabilities in the powered diffusion operator and consider an L^2 distance between these localized energy potentials in the data as our intrinsic data distance, which forms an M-divergence between the diffusion probability distributions[63,64]. Mathematically, if $U_{\mathbf{x}}^t = -\log(\mathbf{p}_{\mathbf{x}}^t)$ for $\mathbf{x} \in \mathcal{X}$, then the t-step potential distance is defined as

$$\mathfrak{V}^t(\mathbf{x},\mathbf{y}) = ||U_{\mathbf{x}}^t - U_{\mathbf{y}}^t||_2, \mathbf{x},\mathbf{y} \in \mathcal{X} \tag{6}$$

To give a more intuitive view, consider two points x and y that are on different sides of a line of points $W = \{w_1, w_2, \ldots, w_n\}$ (Fig. 2e), suppose that there is a small set of distant points $Z = \{z_1, z_2, \ldots, z_n\}$ that are on the same side of W as y but opposite side as x such that they are twice as far from x as from y. The representation of each point x is as its t-step diffusion probability to all other points. So to compute the potential distance between x and y we compare these probabilities. It is then necessary to determine which is the right type of distance to measure the distinction between these two probability distributions. One solution has been the diffusion distance, which is simply the Euclidean distance between these probability distributions. However, in the example mentioned above the diffusion distance would be dominated by larger probabilities and the probabilities to the Z points would not affect the distance from x to y perhaps making them seem close. But instead, we take a divergence between the probabilities from x and y by first log-scale transforming the probabilities and then taking their Euclidean distance, which makes the distance sensitive to fold-change. Thus, if a probability of 0.01 from x to a point z_i is changed to 0.02 from y then this has the same effect as if the probabilities had been 0.1 and 0.2. Thus, PHATE is sensitive to small differences in probability distribution corresponding to differences in long-range global structure, which allows PHATE to preserve global manifold relationships using this potential distance.

We note that the potential distance is a particular case of a wider family of diffusion-based informational distances that view the diffusion geometry as a statistical manifold in information geometry. See Supplementary Note 1 for details on this family of distances.

Embedding the potential distances in low dimensions. A popular approach for embedding diffusion geometries is to use the eigendecomposition of the diffusion operator to build a diffusion map of the data. However, this approach tends to isolate progression trajectories into numerous diffusion coordinates (that is, eigenvectors of the diffusion operator; see Supplementary Fig. 1). In fact, this specific property was used as a heuristic for ordering cells along specific

developmental tracks[14]. Therefore, while diffusion maps preserve global structure and denoise the data, their higher intrinsic dimensionality is not amenable for visualization. Instead, we squeeze the variability into low dimensions using metric MDS, a distance embedding method (Fig. 2f).

There are multiple approaches to MDS. Classical MDS[7] takes a distance matrix as input and embeds the data into a lower-dimensional space as follows. The squared potential distance matrix is double centered:

$$\mathbf{B} = -\frac{1}{2}\mathbf{J}\mathfrak{V}^{t(2)}\mathbf{J} \tag{7}$$

where $\mathfrak{V}^{t(2)}$ is the squared potential distance matrix (that is, each entry is squared) and $\mathbf{J} = \mathbf{I} - \frac{1}{N}\mathbf{1}\mathbf{1}^T$, with $\mathbf{1}$ a vector of ones with length N. The classical MDS coordinates are then obtained by an eigendecomposition of the matrix $\mathbf{B}$. This is equivalent to minimizing the following 'strain' function:

$$\text{Strain}(\hat{\mathbf{x}}_1, \ldots, \hat{\mathbf{x}}_N) = \sqrt{\sum\nolimits_{i,j} \left(\mathrm{B}_{ij} - \left\langle \hat{\mathbf{x}}_i, \hat{\mathbf{x}}_j \right\rangle\right)^2 / \sum\nolimits_{i,j} \mathrm{B}_{ij}^2} \tag{8}$$

over embedded m-dimensional coordinates $\hat{\mathbf{x}}_i \in \mathbb{R}^m$ of data points in $\mathcal{X}$. We apply clasical MDS to the potential distances of the data to obtain an initial configuration of the data in low dimension m.

While classical MDS is computationally efficient relative to other MDS approaches, it assumes that the input distances directly correspond to low-dimensional Euclidean distances, which is overly restrictive in our setting. Metric MDS relaxes this assumption by only requiring the input distances to be a distance metric. Metric MDS then embeds the data into lower dimensions by minimizing the following 'stress' function:

$$\text{Stress}(\hat{\mathbf{x}}_1, \ldots, \hat{\mathbf{x}}_N) = \sqrt{\sum\nolimits_{i,j} \left(\mathfrak{V}^t_{(\mathbf{x}_i,\mathbf{x}_j)} - \|\hat{\mathbf{x}}_i - \hat{\mathbf{x}}_j\|\right)^2 / \sum\nolimits_{i,j} \left(\mathfrak{V}^t_{\mathbf{x}_i,\mathbf{x}_j}\right)^2} \tag{9}$$

over embedded m-dimensional coordinates $\hat{\mathbf{x}}_i \in \mathbb{R}^m$ of data points in $\mathcal{X}$.

If the stress of the embedded points is zero, then the input data is faithfully represented in the MDS embedding. The stress may be nonzero owing to noise or if the embedded dimension m is too small to represent the data without distortion. Thus, by choosing the number of MDS dimensions to be $m = 2$ (or $m = 3$) for visualization purposes, we may trade off distortion in exchange for readily visualizable coordinates. However, some distortion of the distances and dissimilarities is tolerable in many of our applications as precise dissimilarities between points on two different trajectories are not important as long as the trajectories are visually distinguishable. By using metric MDS, we find an embedding of the data with the desired dimension for visualization and the minimum amount of distortion as measured by the stress. When analyzing the PHATE coordinates (for example, for clustering or branch detection), we use metric MDS with m chosen to explain most of the variance in the data as determined by the eigenvalues of the diffusion operator (as is done for VME). In this case, minimal distortion is introduced into the analysis.

A naive approach toward obtaining a truly low-dimensional embedding of diffusion geometries is to directly apply metric MDS, from the diffusion map space to a 2D space. However, as seen in Supplementary Figs. 3 and Supplementary Fig. 8, direct embedding of these distances produces distorted visualizations. Embedding the potential distances (defined in eq. 6) is more stable at boundary conditions near end points compared to diffusion maps, even in the case of simple curves that contain no branching points. Supplementary Figure 2a shows a half circle embedding with diffusion distances versus distances between log-scaled diffusion. We see that points are compressed towards the boundaries of the figure in the former. Additionally, this figure demonstrates that in the case of a full circle (that is, with no end points or boundary conditions), our potential embedding (PHATE) yields the same representation as diffusion maps.

PHATE achieves an embedding that satisfies all four properties delineated previously: PHATE preserves and emphasizes the global and local structure of the data by: (1) a localized affinity that is chained via diffusion to form global affinities through the intrinsic geometry of the data; (2) denoising of the data by low-pass filtering through diffusion; (3) providing a distance that accounts for local and global relationships in the data and has robust boundary conditions for purposes of visualization; and (4) capturing the data in low dimensions, using MDS, for visualization.

We have shown by demonstration in Supplementary Figs. 3 and 8 that all of the steps of PHATE, including the potential transform and MDS, are necessary, as diffusion maps, t-SNE on diffusion maps and MDS on diffusion maps fail to provide an adequate visualization in several benchmark test cases with known ground truth (even when using the same customized α-decaying kernel we developed for PHATE). We have also shown that PHATE is robust to the choice of parameters.

Robustness analysis of PHATE. Here we show that the PHATE embedding is robust to subsampling and the choice of parameters. We demonstrate this both qualitatively and quantitatively. For the quantitative demonstrations, we simulated scRNA-seq data using the Splatter package[25] as described below. We first calculated the geodesic pairwise distances for the noiseless data. Then for each setting, we calculated the pairwise Euclidean distances in the 2D embedding. We then compared the geodesic distances with the embedded distances via the Spearman correlation coefficient to compute DEMaP. We used both the paths and groups options of the Splatter package.

Supplementary Table 3 shows that PHATE is robust to subsampling on the Splatter datasets. For the paths dataset, the average Spearman correlation is the same when 95% and 50% of the data points are retained. For the groups dataset, the correlation drops slightly when going from 95% retention to 50% retention. Additionally, the correlation coefficient is still quite high (and better than all other methods) when only 5% of the data points are retained. Thus, quantitatively, PHATE is robust to subsampling.

We also demonstrate this visually. We ran PHATE on the iPSC mass cytometry dataset[18] with varying subsample sizes N. Supplementary Figure 4a shows the PHATE embedding for $N = 1{,}000$, 2,000, 5,000 and 10,000. Note that the primary branches or trajectories that are visible when $N = 50{,}000$ (Supplementary Fig. 7c) are still visible for all subsamples. Thus, PHATE is robust to the subsampling size. Similar results can be obtained on other datasets.

We also show that the PHATE embedding is robust to the choice of t, k, and α. Supplementary Figure 4b shows the PHATE embedding on the iPSC mass cytometry dataset[18] with varying scale parameter t. This figure shows that the embeddings for $50 \leq t \leq 200$ are nearly identical. Thus, PHATE is very visually robust to the scale parameter t. Similar results can be obtained on other datasets and with the k and α parameters.

The embedding is also quantitatively robust to the parameter choices. Supplementary Fig. 4c,d shows heat maps of the Spearman correlation coefficient between geodesic distances of the ground-truth data and the Euclidean distances of the PHATE visualization applied to the simulated Splatter datasets for different values of k, t and α. For $\alpha \geq 10$, the correlation coefficients are very similar for all values of k, t and α. This demonstrates that PHATE is robust to the choices of these parameters.

Scalability of PHATE. The native form of PHATE is limited in scalability owing to the computationally intensive steps of computing potential distances between all pairs of points, computing metric MDS and storing the diffused operator in memory the diffused operator. Thus, we describe here, and in Supplementary Table 2, an alternative way to compute a PHATE embedding that is highly scalable and provides a good approximation of the native PHATE described previously. The scalable version of PHATE uses a slight difference in computing t-step diffusion probabilities between points. It requires that every other step that the diffusion takes goes through one of a small number of 'landmarks'. Each landmark is selected to be a central point that is representative of a portion of the manifold, selected by spectrally clustering manifold dimensions.

First, we construct the α-decaying kernel on the entire dataset. This can be calculated efficiently and stored as a sparse matrix by using radius-based nearest-neighbor searches and thresholding (that is, setting to zero) connections between points below a specified value (for example, 0.0001), as we regard them numerically insignificant for the constructed diffusion process. The resulting affinity matrix $\mathbf{K}_{k,\alpha}$ will be sparse as long as α is sufficiently large (for example, $\alpha \geq 10$) to enforce sharp decay of the captured local affinities. The full diffusion operator $\mathbf{P}$ is constructed from $\mathbf{K}_{k,\alpha}$ by normalizing by row-sums as described previously.

However, powering the sparse diffusion operator would result in a dense matrix, causing memory issues. To avoid this, we instead perform diffusion between points via a series of M landmarks where $M < N$. We select the landmarks by first applying PCA to the diffusion operator and then using k-means clustering on the principal components to partition the data into M clusters. This is a variation on spectral clustering. We then calculate the probability of transitioning in a single step from the ith point in $\mathcal{X}$ to any point in the jth cluster for all pairs of points and clusters. Mathematically, we can write this as

$$\mathbf{P}_{NM}(i,j) = \sum\nolimits_{\xi \in C_j} \mathbf{P}(i,\xi) \tag{10}$$

where C_j is the set of points in the jth cluster. Thus, we can view each cluster as being represented by a landmark and the (i,j)-th entry in $\mathbf{P}_{\mathrm{NM}}$ gives the probability of transitioning from the ith point in $\mathcal{X}$ to the jth landmark in a single step. Similarly, we construct the matrix $\mathbf{P}_{\mathrm{MN}}$ where the (j,i)-th entry contains the probability of transitioning from the jth landmark to the ith point in $\mathcal{X}$. In this case, we cannot simply sum the transition probabilities $\mathrm{P}(\xi,i), \xi \in C_j$, as we also have to consider the prior probability $\mathbf{Q}(j,\xi)$ of the ξ-th point (with $\xi \in C_j$) being the source of a transition from a cluster C_j. For this purpose we use a previously proposed prior[65] and write

$$\mathbf{P}_{MN}(j,i) = \sum_{\xi \in C_j} \mathbf{Q}(j,\xi)\mathbf{P}(\xi,i) \tag{11}$$

with $\mathbf{Q}(j,\xi) = \sum_i \mathbf{K}_{k,\alpha}(\xi,i) / \sum_{\zeta \in C_j} \sum_i \mathbf{K}_{k,\alpha}(\zeta,i)$.

We use the two constructed transition matrices to compute $\mathbf{P}_{MM} = \mathbf{P}_{MN}\mathbf{P}_{NM}$, which provides the probability of transitioning from landmark to landmark in a random walk by walking through the full point space. Diffusion is then performed by powering the matrix $\mathbf{P}_{MM}$. This can be written as

$$\mathbf{P}_{MM}^{t} = \mathbf{P}_{MN}\mathbf{P}_{NM}\mathbf{P}_{MN}\mathbf{P}_{NM}\ldots\mathbf{P}_{MN}\mathbf{P}_{NM}. \quad (12)$$

From this expression, we see that powering the matrix $\mathbf{P}_{MM}$ is equivalent to taking a random walk between landmarks by walking from landmarks to points and then back to landmarks t times.

We then embed the landmarks into the PHATE space by calculating the potential distances between landmarks and applying metric MDS to the potential distances. Denote the resulting embedding as $\mathbf{Y}_{\text{landmarks}}$. We then perform an out of sample extension to all points from the landmarks by multiplying the point to landmark transition matrix $\mathbf{P}_{NM}$ by $\mathbf{Y}_{\text{landmarks}}$ to get

$$\mathbf{Y}_{\text{points}} = \mathbf{P}_{NM}\mathbf{Y}_{\text{landmarks}} \quad (13)$$

As M is chosen to be vastly less than N, the memory requirements and computational demands of powering the diffusion operator and embedding the potential distances are much lower.

The described steps are summarized in Supplementary Table 2. In Supplementary Fig. 6a–e we show that this constrained diffusion preserves distances between data points in the final PHATE embedding, with the scalable version giving near-identical results to the exact computation of PHATE. Furthermore, in Supplementary Fig. 6b we show that the embedding achieved by this approach is robust to the number of landmarks chosen.

We note that if the only computational bottleneck were in computing MDS, scalable versions of MDS could be used[8,66,67]. However, as storing the entries of the powered diffusion operator in memory is also an issue, we employ the use of landmarks earlier in the process. It has also been shown that 'compressing' the process of diffusion through landmarks in the fashion described here performs better than simply applying Nystrom extension (which includes landmark MDS[66]) to diffusion maps[68].

The fast version of PHATE was used in Fig. 5 and Supplementary Figs. 2d, 3, 6a–e, 8, 12 and 13. All other plots were generated using the exact version of PHATE.

To demonstrate the scalability of PHATE for data exploration on large datasets, we applied PHATE to the 1.3 million mouse brain cell dataset from 10x (https://community.10xgenomics.com/t5/10x-Blog/Our-1-3-million-single-cell-dataset-is-ready-to-download/ba-p/276). Supplementary Fig. 6c shows a comparison of PHATE to t-SNE, colored by 10 of the 60 clusters provided by 10x. We see that PHATE retains cluster coherence while t-SNE shatters some of the cluster structure.

Branch identification. Here we describe the methods we developed for identifying branches in a PHATE visualization and selecting representative branch points and end points.

We use the estimated local intrinsic dimensionality to identify branch points. We can regard intrinsic dimensionality in terms of degrees of freedom in the progression modeled by PHATE. If there is only one fate possible for a cell (that is, a cell lies on a branch as in Fig. 3a) then there are only two directions of transition between data points—forward or backward—and the local intrinsic dimension is low. If on the other hand, there are multiple fates possible, then there are at least three directions of transition possible—a single direction backwards and at least two forward. This cannot be captured by a one-dimensional curve and will require a higher-dimensional structure such as a plane, as shown in Fig. 3a. Thus, we can use the concept of local intrinsic dimensionality for identifying branch points.

We used the local intrinsic dimension estimation method derived in refs. [69,70] to provide suggested branch points. This method uses the relationship between the radius and volume of a d-dimensional ball. The volume increases exponentially with the dimensionality of the data. So as the radius increases by δ, the volume increases by δ^d where d is the dimensionality of the data. Thus the intrinsic dimension can be estimated via the growth rate of a k-NN ball with radius equal to the k-NN distance of a point. The procedure is as follows. Let $\mathbf{Z}_n = \{\mathbf{z}_1, \ldots, \mathbf{z}_n\}$ be a set of independent and identically distributed random vectors with values in a compact subset of $\mathbb{R}^d$. Let $\mathcal{N}_{k,j}$ be the k nearest neighbors of $\mathbf{z}_j$; that is, $\mathcal{N}_{k,j} = \{\mathbf{z} \in \mathbf{Z}_n \setminus \{\mathbf{z}_j\} : ||\mathbf{z} - \mathbf{z}_j|| \leq \epsilon_k(\mathbf{z}_j)\}$. The k-NN graph is formed by assigning edges between a point in $\mathbf{Z}_n$ and its k nearest neighbors. The power-weighted total edge length of the k-NN graph is related to the intrinsic dimension of the data and is defined as

$$L_{\gamma,k}(\mathbf{Z}_n) = \sum_{i=1}^{n} \sum_{\mathbf{z} \in \mathcal{N}_{k,i}} ||\mathbf{z} - \mathbf{z}_i||^{\gamma} \quad (14)$$

where $\gamma > 0$ is a power weighting constant. Let m be the global intrinsic dimension of all the data points in $\mathbf{Z}_n$. It can be shown that, for large n,

$$L_{\gamma,k}(\mathbf{Z}_n) = n^{\beta(m)}c + \epsilon_n \quad (15)$$

where $\beta(m) = (m - \gamma)/m$, ϵ_n is an error term that decreases to 0 as $n \to \infty$ and c is a constant with respect to $\beta(m)$ (ref. [69]). A global intrinsic dimension estimator $\hat{\mathbf{m}}$ can be defined on the basis of this relationship using nonlinear least squares regression over different values of n (refs. [69,70]).

A local estimator of intrinsic dimension $\widetilde{\mathbf{m}}(i)$ at a point $\mathbf{z}_i$ can be defined by running the above procedure in a smaller neighborhood about $\mathbf{z}_i$. This approach is demonstrated in Fig. 3a, where a k-NN graph is grown locally at each point in the data. However, this estimator can have high variance within a neighborhood. To reduce this variance, majority voting within a neighborhood of $\mathbf{z}_i$ can be performed:

$$\hat{\mathbf{m}}(i) = \text{argmax}_{\ell} \sum_{\mathbf{z}_j \in \mathcal{N}_{k,i}} \mathbb{1}(\widetilde{m}(j) = \ell) \quad (16)$$

where $\mathbb{1}(\cdot)$ is the indicator function[70].

We note that other local intrinsic dimension estimation methods could be used such as the maximum likelihood estimator in ref. [71].

We also identify end points in the PHATE embedding. These points can correspond to the beginning or end-states of differentiation processes. For example, Supplementary Fig. 7a shows the PHATE visualization of the iPSC CyTOF dataset[18] with highlighted end points, or end-states, of the reprogrammed and refractory branches. While many major end points can be identified by inspecting the PHATE visualization, we provide a method for identifying other end points or end-states that may be present in the higher-dimensional PHATE embedding. We identify these states using the centrality and distinctness of data points as described below.

First, we compute the centrality of a data point by quantifying the impact of its removal on the connectivity of the graph representation of the data (as defined using the local affinity matrix $\mathbf{K}_{k,\alpha}$). Removing a point that is on a one-dimensional progression pathway, either branching point or not, breaks the graph into multiple parts and reduces the overall connectivity. However, removing an end point does not result in any breaks in the graph. Therefore we expect end points to have low centrality, as estimated using the eigenvector centrality measure of $\mathbf{K}_{k,\alpha}$.

Second, we quantify the distinctness of a cellular state relative to the general data. We expect the beginning or end-states of differentiation processes to have the most distinctive cellular profiles. As shown in ref. [16], we quantify this distinctness by considering the minima and the maxima of diffusion eigenvectors (Fig. 3a). Thus we identify end points in the embedding as those that are most distinct and least central.

After identifying branch points and end points, the remaining points can be assigned to branches between two branch points or between a branch point and end point. Owing to the smoothly varying nature of centrality and local intrinsic dimension, the previously described procedures identify regions of points as branch points or end points rather than individual points. However, it can be useful to reduce these regions to representative points for analysis such as branch detection and cell ordering. To do this, we reduce these regions to representative points using a 'shake and bake' procedure similar to that proposed in ref. [72]. This approach groups collections of branch points or end points together into representative points on the basis of their proximity.

Let $\mathcal{V}_n = \{\mathbf{v}_1, \ldots, \mathbf{v}_n\}$ be the set of branch points and end points in the high-dimensional PHATE coordinates that we wish to reduce. We create a Voronoi partitioning of these points as follows. We first permute the order of $\mathcal{V}_n$, which we denote as $\mathcal{V}' = \{\mathbf{v}_{1'}, \ldots, \mathbf{v}_{n'}\}$. We then take the first point $\mathbf{v}_{1'}$ and find all the points in $\mathcal{V}'$ that are within a distance of h, where h is a scale parameter provided by the user. These points (including $\mathbf{v}_{1'}$) are assigned to the first component of the partition and removed from the set $\mathcal{V}'$. This process is then repeated until all points in $\mathcal{V}_n$ are assigned to the partition. To ensure that each point is assigned to the nearest component of the partition (as measured by proximity to the centroid), we next calculate the distance of each point to all centroids of the partition, and reassign the point to the component with the nearest centroid. This reassignment process is repeated until a stable partition is achieved. This completes the process of constructing the Voronoi partition.

The Voronoi partition constructed from this process may be sensitive to the ordering of the points in $\mathcal{V}'$. To reduce this sensitivity, we repeat this process multiple times (for example, 40–100) to create multiple Voronoi partitions. We then construct a distance between points by estimating the probability that two points are not in the same component from this partitioning process. This provides a notion of distance that is robust to noise, random permutations and the scale parameter h. We then partition the data again using the above procedure except we use these probability-based distances. The representative points are then selected from the resulting centroids of this final partition.

A representative point is labeled an end point if the corresponding collection of points contains one or more end points as identified using centrality and distinctness. Otherwise, the representative point is labeled a branch point.

After representative points have been selected, the remaining points can be assigned to corresponding branches. We use an approach that is based on the branch-point-detection method in ref. [14], which compares the correlation and anticorrelation of neighborhood distances. However, we use higher-dimensional

PHATE coordinates instead of the diffusion maps coordinates. Figure 3a gives a visual demonstration of this approach. Here we consider two reference cells X and Y. We wish to determine if cells $Q1$ and $Q2$ belong to the branch between X and Y or not. Consider $Q1$ first, which does belong to this branch. If we move from $Q1$ towards X, we also move farther away from Y. Thus the distances to X and Y of a neighborhood of points around $Q1$ (which will be located on the branch) are negatively correlated with each other. Now consider $Q2$ which does not belong to the branch between X and Y. In this case, if we move from $Q2$ towards Y, we also move closer to X. Thus the distances to X and Y of a neighborhood of points around $Q2$ are positively correlated with each other. In practice, these distance-based correlations are computed for each possible branch and the point is assigned to the branch with the largest anticorrelation (that is, the most negative correlation coefficient).

EMD score analysis. The EMD is measure of dissimilarity between two probability distributions that is particularly popular in computer vision[73]. The EMD was chosen to perform differential expression analysis in the EB scRNA-seq data owing to its stability in estimation as compared to other divergence measures. Intuitively, if each distribution is viewed as a pile of dirt, the EMD can be thought of as the minimum cost of converting one pile of dirt into the other. If the distributions are identical, then the cost is zero. When comparing univariate distributions (as we do, that is, we only consider a single gene at a time), the EMD simplifies to the L^1 distance between the cumulative distribution functions[56]. That is, if P and Q are the cumulative distributions of densities p and q, respectively, then the EMD between p and q is $\int |P(x) - Q(x)| dx$. While the EMD is non-negative, we assign a sign to the EMD score on the basis of the difference between the medians of the distributions.

Biological methods. The processes for generating the EB data and for preprocessing the biological data are described here.

Generation of human embryoid body data. These experiments were approved by the Yale Embryonic Stem Cell Research Oversight (ESCRO) committee. Low-passage H1 hESCs were maintained on Matrigel-coated dishes in DMEM/F12-N2B27 medium supplemented with FGF2. For EB formation, cells were treated with Dispase, dissociated into small clumps and plated in non-adherent plates in medium supplemented with 20% FBS, which was prescreened for EB differentiation. Samples were collected during 3-d intervals during a 27-d-long differentiation timecourse. An undifferentiated hESC sample was also included (Supplementary Fig. 14a). Induction of key germ-layer markers in these EB cultures was validated by quantitative PCR (data not shown). For single-cell analyses, EB cultures were dissociated, FACS sorted to remove doublets and dead cells and processed on a 10x genomics instrument to generate cDNA libraries, which were then sequenced. Small-scale sequencing determined that we had successfully collected data on 31,161 cells distributed throughout the timecourse. After preprocessing the data as described below, we are left with 16,825 cell measurements for data analysis. See also the Life Sciences Reporting Summary for further details.

Data preprocessing. Here we discuss methods we used to preprocess the various datasets.

Data subsampling. The full PHATE implementation scales well for sample sizes up to approximately $N = 50{,}000$. For N much larger than 50,000, computational complexity can become an issue owing to the multiple matrix operations required. All of the scRNA-seq datasets considered in this paper have $N < 50{,}000$. Thus, we used the full data and did not subsample these datasets. However, the mass cytometry datasets have much larger sample sizes. To aid in branch analysis, we randomly subsampled these datasets for analysis using uniform subsampling. For the comparison figures (Fig. 5 and Supplementary Figs. 3 and 8), scalable PHATE was used and subsampling was not performed except as indicated in the figures. The PHATE embedding is robust to the number of samples chosen, which we demonstrated in Supplementary Fig. 4.

Mass cytometry data preprocessing. We processed the mass cytometry datasets as previously described[74].

Single-cell RNA sequencing data preprocessing. This data was processed from raw reads to molecule counts using the Cell Ranger pipeline[75]. Additionally, to minimize the effects of experimental artifacts on our analysis, we preprocessed the scRNA-seq data. We first filtered out dead cells by removing cells that had high expression levels in mitochondrial DNA. In the case of the EB data, which had a wide variation in library size, we then removed cells that were either below the 20th percentile or above the 80th percentile in library size. scRNA-seq data have large cell-to-cell variations in the number of observed molecules in each cell or library size. Some cells are highly sampled with many transcripts, while other cells are sampled with fewer. This variation is often caused by technical variations owing to enzymatic steps including lysis efficiency, mRNA capture efficiency and the efficiency of multiple amplification rounds[76]. Removing cells with extreme library size values helped to correct for these technical variations. We then dropped genes that were only expressed in a few cells and then perform library size normalization. Normalization was accomplished by dividing the expression level of each gene in a cell by the library size of the corresponding cell.

After normalizing by the library size, we took the square-root transform of the data and then performed PCA to improve the robustness and reliability of the constructed affinity matrix $\mathbf{K}_{k,a}$. We chose the number of principal components to retain approximately 70% of the variance in the data, which resulted in 20–50 principal components.

Gut microbiome data preprocessing. We used the cleaned L6 American Gut data and removed samples that were near duplicates of other samples. We then preprocessed the data using a similar approach for scRNA-seq data. We first performed 'library size' normalization to account for technical variations in different samples. We then log transformed the data and used PCA to reduce the data to 30 dimensions.

Applying PHATE to this data revealed several outlier samples that were very far from the rest of the data. We removed these samples and then reapplied PHATE to the log-transformed data to obtain the results that are shown in Fig. 1d.

Chromatin immunoprecipitation–sequencing processing for Hi-C visualization. We used narrow peak bed files and took the average peak intensity for each bin at a resolution of 10 kilobases. For visualization, we smoothed the average peak intensity values on the basis of location using a 25-bin moving average.

DEMaP. To quantitatively compare each dimensionality-reduction tool, we wished to calculate the degree to which each method preserves the underlying structure of the reference dataset and removes noise. As scRNA-seq and other biological types of data are highly noisy, visual renderings of the data that can offer denoised embeddings that reveal the underlying structure of the data are desirable. Therefore, the goal of our accuracy metric was to quantify the correspondence between distances in the low-dimensional embedding and manifold distances in the ground-truth reference.

To define a quantitative notion of manifold distance we use geodesic distances. Geodesic distances are shortest-path distances on a nearest-neighbor graph of the data weighted by the Euclidean distances between connected points[4]. In cases where points are sampled noiselessly from a manifold, such as in our ground-truth reference, geodesic distances converge exactly to distances along the manifold of the data[4,77]. Thus we reason that if geodesic distances between points on the noiseless manifold are preserved by an embedding computed on the noisy data then the data are sufficiently denoised and the manifold structure is also preserved.

We take this approach to formulate our ground-truth manifold distance as a quantification of the degree to which each dimensionality-reduction method preserves the pairwise geodesic distances of the noiseless data after low-dimensional embedding of the corresponding noisy data. As the low-dimensional embedding is often a result of a nonlinear dimensionality reduction, curves and major paths in the data are 'straightened' such that Euclidean distances in the embedding space correspond to manifold distance in the high-dimensional space[7]. Thus we quantify the preservation of manifold distances as the correlation between geodesic distance in the noiseless reference dataset and Euclidean distances in the embedding space as a measure of structure preservation which we call DEMaP (Fig. 4a).

Construction of the artificial tree test case. The artificial tree data shown in Fig. 1b was constructed as follows. The first branch consists of 100 linearly spaced points that progress in the first four dimensions. All other dimensions were set to zero. The 100 points in the second branch are constant in the first four dimensions with a constant value equal to the end point of the first branch. The next four dimensions then progress linearly in this branch while all other dimensions were set to zero. The third branch was constructed similarly except the progression occurs in dimensions 9–12 instead of dimensions 5–8. All remaining branches were constructed similarly with some variation in the length of the branches. We then added 40 points at each end point and branch point and added zero mean Gaussian noise with a s.d. of 7. This construction models a system where progression along a branch corresponds to an increase in gene expression in several genes. Before adding noise, we also constructed a small gap between the first branch point and the orange branch that splits into a blue and purple branch (see the top set of branches in the left part of Fig. 1b). This simulates gaps that are often present in measured biological data. We also added additional noise dimensions, bringing the total dimensionality of the data to 60.

Splatter simulation details. Splatter is an scRNA-seq simulation package that uses a parametric model to generate data with various structures, such as branches or clusters[25]. We use Splatter to simulate multiple ground-truth datasets for multiple experiments. To select parameters for the simulation, we fit the Splatter simulation to the EB data, and then modified the resulting dataset from both the Splatter 'paths' and the Splatter 'groups' simulations as described in "Comparison of PHATE to other methods." Note that we do not make use of Splatter's built-in dropout function, as it uses a zero-inflated model and multiple studies have shown that an undersampling (binomial) model is more appropriate[78–82]. Each simulation is performed with 3,000 simulated cells. The mean correlation coefficient and s.d. were calculated from 20 trials.

To generate a diverse set of ground-truth references, we simulated 50 datasets containing clusters and 50 datasets containing branches. In each of these simulated datasets, the number and size of the clusters of branches, as well as the global position of the clusters or branches with respect to each other, is random. Furthermore, the local relationships between individual cells on these structures is random. Finally, the changes in gene expression within clusters or along branches is random. The output of this simulation is the ground-truth reference.

Next, we added biological and technical noise to the reference data. First, to simulate stochastic gene expression we used Splatter's biological coefficient of variation (BCV) parameter, which controls the level of gene expression in each cell following an inverse γ-distribution. Second, to simulate the inefficient capture of mRNA in single cells, we undersampled from the true counts using the default BCV. Third, to demonstrate robustness to varying of total genes measured, we randomly removed genes from the data matrix. Finally, to demonstrate robustness to the number of cells captured, we randomly removed cells from each dataset. We varied each of these parameters, including by default some degree of biological variation and mRNA undersampling to each simulation.

The default parameters used in the simulation were the following: 'batchCells=3000'; 'nGenes=17580'; 'mean.shape=6.6'; 'mean.rate=0.45'; 'lib.loc=9.1'; 'lib.scale=0.33'; 'out.prob=0.016'; 'out.facLoc=5.4'; 'out.facScale=0.90'; 'bcv.common=0.18'; 'bcv.df=21.6'; and 'de.prob=0.2'.

We also set 'dropout.type="none"', with a post-hoc binomial dropout of 50%. For the groups simulation we drew the number of groups n from a Poisson distribution with rate $\lambda = 10$, and then drew the 'group.prob' parameter from a Dirichlet distribution with n categories and a uniform concentration $\alpha_1 = \cdots = \alpha_n = 1$. For the paths simulation, we set 'group.prob' as above, and additionally set the ith entry in the parameter 'path.from' as a random integer between 0 and $i-1$, drew the parameter 'path.nonlinearProb' from a uniform distribution on the interval (0,1), and drew the parameter 'path.skew' from a β-distribution with shape $\alpha = 10$, $\beta = 10$. Note that here the library size was doubled from the fit value, since the EB data itself suffers from dropout. To reduce the number of genes for the n_genes simulation, we randomly removed genes post-hoc to avoid changing the state of the random number generator in building the simulation.

For the ground-truth simulations, we set bcv.common to 0, did not perform binomial dropout, and did not remove genes or cells. For the BCV simulation, we performed 50% post-hoc binomial dropout, did not remove genes or cells and set bcv.common to 0, 0.25 and 0.5. For the dropout simulation, we set bcv.common to 0.18, did not remove genes or cells and performed 0%, 50% and 95% post-hoc binomial dropout. For the subsample simulation, we set bcv.common to 0.18, performed 50% post-hoc binomial dropout, did not remove genes and subsampled rows of the matrix to retain 95%, 50% and 5% of the total cells. For the n_genes simulation, we set bcv.common to 0.18, performed 50% post-hoc binomial dropout, did not remove cells and subsampled columns of the matrix to retain 17,000, 10,000, and 2,000 genes.

PHATE experimental details. For all of the quantitative comparisons, we have used the default parameter settings for the PHATE plots. For the majority of the qualitative comparisons in Fig. 5 and Supplementary 3 and 8, we also used the default parameter settings for all methods. Exceptions to this are the artificial tree (Supplementary Fig. 3a), the intersecting circles (Supplementary Fig. 3d) and the MNIST dataset (Supplementary Fig. 3l). In these cases, the PHATE parameters have been tuned to give a clearer separation of the branches. However, in general, the default PHATE settings give good results on most datasets, especially those that are complex, high-dimensional and noisy as demonstrated empirically in "Robustness analysis of PHATE." The default settings are also used in Supplementary Figs. 2d, 6a–e, 12 and 13. For all other PHATE plots, the parameters were tuned slightly to better highlight the structure of the data.

Reporting Summary. Further information on research design is available in the Nature Research Reporting Summary linked to this article.

Data availability

The embryoid body scRNA-seq and bulk RNA-seq datasets generated and analyzed during the current study are available from the Mendeley Data repository at https://doi.org/10.17632/v6n743h5ng.1. Supplementary Figure 14a contains images of the raw single cells while Supplementary Fig. 14f contains scatter plots showing the gating procedure for fluorescence activated cell sorting populations for the bulk RNA-seq data.

Code availability

Python, R and Matlab implementations of PHATE are available on GitHub (https://github.com/KrishnaswamyLab/PHATE) for academic use.

References

58. Nadler, B., Lafon, S., Coifman, R. R. & Kevrekidis, I. Diffusion maps, spectral clustering and eigenfunctions of Fokker-Planck operators. In *Proc 18th International Conference on Neural Information Processing Systems* (Eds. Weiss, Y. et al.) 955–962 (MIT Press, 2005).
59. 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).
60. Butterworth, S. On the theory of filter amplifiers. *Wireless Engineer* **7**, 536–541 (1930).
61. Neumann, J. *Mathematische Grundlagen der Quantenmechanik*. (Springer, 1932).
62. Anand, K., Bianconi, G. & Severini, S. Shannon and von Neumann entropy of random networks with heterogeneous expected degree. *Phys. Rev. E* **83**, 036109 (2011).
63. Salicrú, M. & Pons, A. A. Sobre ciertas propiedades de la M-divergencia en análisis de datos. *Qüestiió* **9**, 251–256 (1985).
64. Salicrú, M., Sanchez, A., Conde, J. & Sanchez, P. Entropy measures associated with K and M divergences. *Soochow J. Math.* **21**, 291–298 (1995).
65. Wolf, G., Rotbart, A., David, G. & Averbuch, A. Coarse-grained localized diffusion. *Appl. Comput. Harm. Anal.* **33**, 388–400 (2012).
66. Platt, J. Fastmap, metricmap, and landmark mds are all Nystrom algorithms. In *Proc. 10th International Workshop on Artificial Intelligence and Statistics* (Eds. Cowell, R. & Ghahramani, Z.) (AI/Stats, 2005).
67. Yang, T., Liu, J., McMillan, L. & Wang, W. A fast approximation to multidimensional scaling. In *Proc. IEEE Workshop on Computation Intensive Methods for Computer Vision* (IEEE, 2006).
68. Gigante, S. et al. Compressed diffusion. In *The 13th International Conference on Sampling Theory and Applications (Bordeaux, France)*, sampta2019:267712 (2019).
69. Costa, J. A. & Hero, A. O. III Determining intrinsic dimension and entropy of high-dimensional shape spaces. In *Statistics and Analysis of Shapes* (Eds Hamid, K. & Yezzi Jr, A) 231–252 (Birkhäuser, 2006).
70. Carter, K. M., Raich, R. & Hero, A. O. III On local intrinsic dimension estimation and its applications. *IEEE Trans. Signal Process.* **58**, 650–663 (2010).
71. Levina, E. & Bickel, P. J. Maximum likelihood estimation of intrinsic dimension. In *Proc. 18th International Conference on Neural Information Processing Systems* (ed. Weiss, Y.) 777–784 (Curran Associates, 2005).
72. David, G. & Averbuch, A. Hierarchical data organization, clustering and denoising via localized diffusion folders. *Appl. Comput. Harmon. Anal.* **33**, 1–23 (2012).
73. Rubner, Y., Tomasi, C. & Guibas, L. J. A metric for distributions with applications to image databases. In *Proc. IEEE Sixth International Conference on Computer Vision* 59–66 (IEEE, 1998).
74. Bendall, S. C. et al. Single-cell mass cytometry of differential immune and drug responses across a human hematopoietic continuum. *Science* **332**, 687–696 (2011).
75. Zheng, G. X. et al. Massively parallel digital transcriptional profiling of single cells. *Nat. Commun.* **8**, 14049 (2017).
76. Grün, D., Kester, L. & Van Oudenaarden, A. Validation of noise models for single-cell transcriptomics. *Nat. Methods* **11**, 637–640 (2014).
77. Balasubramanian, M. & Schwartz, E. L. The isomap algorithm and topological stability. *Science* **295**, 7–7 (2002).
78. van Dijk, D. et al. Recovering gene interactions from single-cell data using data diffusion. *Cell* **174**, 716–729 (2018).
79. Vieth, B., Ziegenhain, C., Parekh, S., Enard, W. & Hellmann, I. powsimR: power analysis for bulk and single cell rna-seq experiments. *Bioinformatics* **33**, 3486–3488 (2017).
80. Brennecke, P. et al. Accounting for technical noise in single-cell RNA-seq experiments. *Nat. Methods* **10**, 1093 (2013).
81. Hwang, B., Lee, J. H. & Bang, D. Single-cell RNA sequencing technologies and bioinformatics pipelines. *Exp. Mol. Med.* **50**, 96 (2018).
82. Kim, J. K., Kolodziejczyk, A. A., Ilicic, T., Teichmann, S. A. & Marioni, J. C. Characterizing noise structure in single-cell RNA-seq distinguishes genuine from technical stochastic allelic expression. *Nat. Commun.* **6**, 8687 (2015).

Acknowledgements

This research was supported in part by the Gruber Foundation (to S.G.); the Eunice Kennedy Shriver National Institute of Child Health and Human Development of the National Institutes of Health (NIH) (award number F31HD097958) (to D.B.B.); an Alfred P. Sloan Fellowship (grant FG-2016-6607); a DARPA Young Faculty Award (grant D16AP00117); National Science Foundation (NSF) grants 1620216, 1912906; an NSF CAREER award (grant 1845856) (to M.J.H.); NIH grant 1R01HG008383-01A1 (to R.R.C.); NIH grant R01GM107092 (to N.B.I.); IVADO (Institut de valorisation des données) (to G.W.); the Chan–Zuckerberg Initiative (grant 182702); NIH grant R01GM130847; the State of Connecticut (grant 16-RMB-YALE-07) (to S.K.); and NIH grant R01GM135929 (to M.J.H., G.W. and S.K.). The content is solely the responsibility of the authors and does not necessarily represent the official views of the funding agencies.

Author contributions

K.R.M., S.K., G.W. and D.v.D. envisioned the project. K.R.M., D.v.D., S.G. and G.W. implemented the method. K.R.M., D.v.D., S.G., S.K. and N.B.I. performed the analyses. K.R.M., S.K., G.W., and N.B.I. wrote the paper. D.v.D., S.G. and D.B.B. assisted in

writing. D.B.B., W.S.C. and K.Y. assisted in the analysis. K.R.M., G.W., M.J.H. and R.R.C. developed the mathematical foundations of the method. Z.W., A.v.d.E. and N.B.I. were responsible for data acquisition and processing.

Competing interests

Smita Krishnaswamy serves on the scientific advisory board of AI Therapeutics.

Additional information

Supplementary information is available for this paper at https://doi.org/10.1038/s41587-019-0336-3.

Correspondence and requests for materials should be addressed to N.B.I., G.W. or S.K.

Reprints and permissions information is available at www.nature.com/reprints.

nature research

Corresponding author(s): Smita Krishnaswamy

Last updated by author(s): Sep 23, 2019

Reporting Summary

Nature Research wishes to improve the reproducibility of the work that we publish. This form provides structure for consistency and transparency in reporting. For further information on Nature Research policies, see Authors & Referees and the Editorial Policy Checklist.

Statistics

For all statistical analyses, confirm that the following items are present in the figure legend, table legend, main text, or Methods section.

n/a	Confirmed
☐	☒ The exact sample size (*n*) for each experimental group/condition, given as a discrete number and unit of measurement
☐	☒ A statement on whether measurements were taken from distinct samples or whether the same sample was measured repeatedly
☒	☐ The statistical test(s) used AND whether they are one- or two-sided *Only common tests should be described solely by name; describe more complex techniques in the Methods section.*
☒	☐ A description of all covariates tested
☒	☐ A description of any assumptions or corrections, such as tests of normality and adjustment for multiple comparisons
☒	☐ A full description of the statistical parameters including central tendency (e.g. means) or other basic estimates (e.g. regression coefficient) AND variation (e.g. standard deviation) or associated estimates of uncertainty (e.g. confidence intervals)
☒	☐ For null hypothesis testing, the test statistic (e.g. *F*, *t*, *r*) with confidence intervals, effect sizes, degrees of freedom and *P* value noted *Give P values as exact values whenever suitable.*
☒	☐ For Bayesian analysis, information on the choice of priors and Markov chain Monte Carlo settings
☒	☐ For hierarchical and complex designs, identification of the appropriate level for tests and full reporting of outcomes
☒	☐ Estimates of effect sizes (e.g. Cohen's *d*, Pearson's *r*), indicating how they were calculated

Our web collection on statistics for biologists contains articles on many of the points above.

Software and code

Policy information about availability of computer code

Data collection	The Cell Ranger pipeline, version 1.2.0, was used for cell separation. FACS data were collected using BD FACSDiva V6.0.
Data analysis	Data were analyzed using standard functions in MATLAB version 2018a, Python 3.7, and custom code developed by the authors available at https://github.com/krishnaswamylab/PHATE. Code for estimating intrinsic dimension can be found at http://tbayes.eecs.umich.edu/kmcarter/smoothing. FlowJo V10 was used for flow cytometry analysis. Wanderlust was run as part of the cyt3 MATLAB package which is available at https://github.com/dpeerlab/cyt3.

For manuscripts utilizing custom algorithms or software that are central to the research but not yet described in published literature, software must be made available to editors/reviewers. We strongly encourage code deposition in a community repository (e.g. GitHub). See the Nature Research guidelines for submitting code & software for further information.

Data

Policy information about availability of data

All manuscripts must include a data availability statement. This statement should provide the following information, where applicable:

- Accession codes, unique identifiers, or web links for publicly available datasets
- A list of figures that have associated raw data
- A description of any restrictions on data availability

The embryoid body scRNA-seq and bulk RNA-seq datasets generated and analyzed during the current study are available in the Mendeley Data repository at http://dx.doi.org/10.17632/v6n743h5ng.1. Figure 14SA contains images of the raw single cells while Figure S14F contains scatter plots showing the gating procedure for FACS sorting cell populations for the bulk RNA-seq data.

Field-specific reporting

Please select the one below that is the best fit for your research. If you are not sure, read the appropriate sections before making your selection.

☒ Life sciences ☐ Behavioural & social sciences ☐ Ecological, evolutionary & environmental sciences

For a reference copy of the document with all sections, see nature.com/documents/nr-reporting-summary-flat.pdf

Life sciences study design

All studies must disclose on these points even when the disclosure is negative.

Sample size	Flow cytometry analysis of H1 derived embryoid bodies were conducted on n=3 stained wells.
Data exclusions	No exclusion was done prior to loading to 10X single cell sorting. After obtaining the sgRNA-seq data, dead cells were excluded based on high mitochondrial DNA expression. Then cells were excluded based on high or low library size. These exclusion criteria were not predetermined. See Online Methods for details.
Replication	For flow cytometry analysis, all replication attempts were successful.
Randomization	All derived embryoid bodies were pooled, and the sample was randomly split into two tubes prior to dissociation, staining (CD49d/CD63; CD92/CD142) and FACS procedures
Blinding	None

Reporting for specific materials, systems and methods

We require information from authors about some types of materials, experimental systems and methods used in many studies. Here, indicate whether each material, system or method listed is relevant to your study. If you are not sure if a list item applies to your research, read the appropriate section before selecting a response.

Materials & experimental systems

n/a	Involved in the study
☐	☒ Antibodies
☐	☒ Eukaryotic cell lines
☒	☐ Palaeontology
☒	☐ Animals and other organisms
☒	☐ Human research participants
☒	☐ Clinical data

Methods

n/a	Involved in the study
☒	☐ ChIP-seq
☐	☒ Flow cytometry
☒	☐ MRI-based neuroimaging

Antibodies

Antibodies used	FITC Mouse Anti-Human CD63 (supplier: BD; catalog number: 561924; clone name: Clone H5C6; dilution: 20 µl per million cells in 100 µl staining volume); Alexa Fluor® 647 Mouse Anti-Human Tspan-27 (CD82) (supplier: BD; catalog number: 564341; clone name: Clone 423524; dilution: 5 µl per million cells in 100 µl staining volume); PE anti-human CD142 (supplier: BIOLEGEND; catalog number: 365203; clone name: Clone NY2; dilution: 5 µl per million cells in 100 µl staining volume); PE Mouse Anti-Human CD49d (supplier: BD; catalog number: 560972; clone name: Clone 9F10; dilution: 20 µl per million cells in 100 µl staining volume)
Validation	The antibodies used are all commercialized monoclonal, react to human, flow cytometry antibodies. The manufacturers' websites contain the validation information.

Eukaryotic cell lines

Policy information about cell lines

Cell line source(s)	H1 (WA01) (WiCell Research Institute, Inc.)
Authentication	H1 (WA01) is authenticated by Karyotype analysis and staining for pluripotent stem cell markers
Mycoplasma contamination	H1 (WA01) is free of Myoplasma and tested by Yale Pathology Department Lab
Commonly misidentified lines (See ICLAC register)	none

Flow Cytometry

Plots

Confirm that:

☒ The axis labels state the marker and fluorochrome used (e.g. CD4-FITC).

☒ The axis scales are clearly visible. Include numbers along axes only for bottom left plot of group (a 'group' is an analysis of identical markers).

☒ All plots are contour plots with outliers or pseudocolor plots.

☒ A numerical value for number of cells or percentage (with statistics) is provided.

Methodology

Sample preparation	H1 (WA01) formed embryoid bodies were dissociated to single cells by Accutase.
Instrument	BD LSR II
Software	FlowJo V10
Cell population abundance	Purity of samples was determined by re-sorting the samples. The purity for all the experiments was calculated to be between 90-95%.
Gating strategy	Single cells were selected based on SSC-A/FSC-A. Live cells were selected based on DAPI staining. Alexa 488, Alexa 647, and PE gates were defined based on negative controls (H1 cells in pluripotent state).

☒ Tick this box to confirm that a figure exemplifying the gating strategy is provided in the Supplementary Information.