DensityPath: an algorithm to visualize and reconstruct cell state-transition path on density landscape for single-cell RNA sequencing data

Supplementary Information

Ziwei Chen[1,3*] Shaokun An[1,3*] Xiangqi Bai[1,3] Fuzhou Gong[1,3] Liang Ma[2†]
Lin Wan[1,3†]

[1]NCMIS, Academy of Mathematics and Systems Science,
Chinese Academy of Sciences, Beijing 100190, China

[2]Beijing Institute of Genomics, Chinese Academy of Sciences, Beijing 100101, China

[3]University of Chinese Academy of Sciences, Beijing 100049, China

*These authors contributed equally to this work.

†To whom correspondence may be addressed. Email: lwan@amss.ac.cn (Lin Wan) and mal@big.ac.cn (Liang Ma).

Contents

Supplementary Methods

Supplementary Note 1: Calculation of the shortest distance path (geodesics) on density landscape using Dijkstra's algorithm

DensityPath calculates the shortest distance path (geodesics) of the peak points on the surface of density landscape by applying Dijkstra's algorithm as follows:

1. **Divide the grid.** The surface of density landscape is represented by the z-coordinates (density values) of points (nodes) above a 2-d square grid in the x-y plane with uniformly spaced EE1-coordinate and EE2-coordinate.

2. **Construct the king graph.** A king's graph is constructed with the nodes of the grid points and edges connected by points with their eight Moore neighborhoods (including 4 orthogonal and 4 diagonal nearest neighbors).

3. **Assign the weight between each pair of nodes.** The weight between each pair of nodes is set as (i) the reciprocal of the averaged value of the two nodes' densities if they are in Moore neighborhood and connected by one of the edge on the king's graph, that is, $\frac{1}{\frac{1}{2}(f(x_i)+f(x_j))}$, where f is the density function, and the x_i and x_j are the EE-coordinates of the edge's nodes i and j, or (ii) $+\infty$ otherwise.

4. **Cacualate shortest distance path (geodesics) of two points on the density landscape.** Given the coordinates of two points (RCSs) on the surface of density landscape, their shortest distance path (geodesics) is calculated as: finding their nearest nodes on the king's graph, and finding the shortest distance path between the two nodes on the king's graph by Dijkstra's algorithm, with the edges' weights of the king's graph assigned as step 3 above. The distortion caused by the difference of distance between the orthogonal connection and diagonal connection on the king's graph is corrected.

DensityPath implements the above steps mainly based on the R Package "gdistance" (see van Etten, 2017 for details). The embedded cell state-transition path is then constructed by finding the MST of the peak points of RCSs.

Supplementary Note 2: Data Availability

We perform DensityPath analysis on three real scRNA-seq datasets. (1) The scRNA-seq dataset of mouse bone marrow cells is from Paul *et al.*, 2015. The processed read count profile of 2730 cells and 8716 genes was provided by authors of Haghverdi *et al.*, 2016 in a personal communication,

and the data were processed into the reads per kilobase per million mapped reads (RPKM) values. We select out the 3461 informative genes which were provided by authors of Haghverdi *et al.*, 2016. (2) The scRNA-seq dataset of mouse hematopoietic stem and progenitor cells (HSPCs) bifurcating to myeloid and erythroid precursors is also from Paul *et al.*, 2015. We directly download the processed expression profile of these data by Wishbone (Setty *et al.*, 2016), containing 4423 single cells with 2312 informative genes, at https://github.com/ManuSetty/wishbone, and we follow the same normalization procedures as those of Wishbone using their code (Setty *et al.*, 2016). (3) The scRNA-seq dataset of human preimplantation embryos is from Petropoulos *et al.*, 2016. The expression profile of 1529 single cells with 26178 genes in RPKM values was downloaded from https://www.ebi.ac.uk/arrayexpress/experiments/E-MTAB-3929/.

We also perform DensityPath analysis on two simulated scRNA-seq datasets. The PHATE dataset of multi-branching trajectories was simulated based on the codes in `https://github.com/KrishnaswamyLab/PHATE`, which were provided by Moon *et al.*, 2017, containing 1440 cells with 60 genes. The SLS3279 dataset of the bifurcating trajectory is discussed in the supplementary material of Zwiessele and Lawrence, 2017, and the dataset utilized here is from the supplementary material of Guo and Zheng, 2017, which contains 490 cells with 48 genes.

Supplementary Note 3: Method Evaluations

To evaluate pseudotime calculation and branch assignment performance, we adopt two indexes, Pearson's correlation coefficient (PCC) and the adjusted rand index (ARI) (Rand, 1971; Qiu *et al.*, 2017), and compare the results obtained by different methods on datasets with known branch assignment or real-time information of cells. The PCC of the calculated pseudotime and real-time information, either experimental time or simulation time, is used to evaluate the accuracy or robustness of pseudotime. We also provide the PCCs of pseudotimes calculated by DensityPath between different parameter values and the default parameter value (we denote it as PCC') in Fig. S17, Fig. S19-S23, Fig. S26 for comparing the visualizations.

ARI, which is a measure of the similarity between two partitions of data and has been utilized by Monocle2 (Qiu *et al.*, 2017), is used to measure the accuracy or robustness of trajectory branch assignment of cells. Given a dataset of n single cells and two assignments $\mathcal{X} = \{X_1, X_2, .., X_r\}$ and $\mathcal{Y} = \{Y_1, Y_2, .., Y_s\}$ of n cells into r and s branches, respectively, the overlap between $\mathcal{X}$ and $\mathcal{Y}$ can be characterized by a set of counts n_{ij} which represents the overlap number in X_i and Y_j, i.e., $n_{ij} = |X_i \cap Y_j|$. We then define the number of cells within segment i from the former clustering result, as $a_i = \sum_{j=1}^{s} n_{ij}$, and the number of cells within segment j from the latter is $b_j = \sum_{i=1}^{r} n_{ij}$.

The ARI value is then formulated as

$$\mathrm{ARI}(\mathcal{X},\mathcal{Y}) = \frac{\sum_{ij}\binom{n_{ij}}{2} - [\sum_i \binom{a_i}{2}\sum_j \binom{b_j}{2}]/\binom{n}{2}}{\frac{1}{2}[\sum_i \binom{a_i}{2} + \sum_j \binom{b_j}{2}] - [\sum_i \binom{a_i}{2}\sum_j \binom{b_j}{2}]/\binom{n}{2}}.$$

The value of ARI is in the region of [0,1]. A larger value indicates better assignment. In our analysis, each cluster is defined as the cells mapped on the trajectory segment from one branching or initial peak to the next branching peak point or the end peak.

Supplementary Results

Supplementary Note 4: DensityPath reveals the clusters and developmental structures of bone marrow cells

We test DensityPath on the scRNA-seq dataset of bone marrow cells from Paul *et al.*, 2015 (hereinafter denoted as Paul data). Paul data are combined with index Fluorescence Assisted Cell Sorting (FACS) and have been widely used to study heterogeneity in myeloid progenitors. The processed data of the sorted c-Kit$^+$Scal$^-$lineage(Lin$^-$) bone marrow cells, which contain 2730 single cells with 3461 informative genes, have been adopted. In their study, Paul *et al.*, 2015 identified 19 distinct progenitor clusters at different degrees of differentiation through the EM-based clustering approach. Accordingly, clusters 1 to 6 represent erythroid (Ery) fate, clusters 7 to 10 represent the common myeloid progenitor (CMP), cluster 11 corresponds to dendritic cell fate (DC), and clusters 12 to 18 reflect granulocyte/macrophage progenitor (GMP) fate. Cluster 19, which is constituted by the missorted population of lymphoid progenitors, is denoted as "Outlier" in this study.

DensityPath extracts 10 separate high-density clusters of RCSs (Fig. S2(a,b)), picking up a total of 865 cells out of the 2730, with each cluster ranging in size from 16 to 276 cells. To reveal the structure of the cell development process, an MST connecting the peak points of RCSs is constructed on the surface of the density landscape, showing a cell state-transition path constituted of two bifurcating events, the first occurring at the rightmost RCS where the start cell (root) is mapped (Fig. S2(b)).

In contrast to the previously defined clusters by Paul *et al.*, 2015, which constitute a partition of all cells, it is worth noting that DensityPath picks up the high-density clusters and constructs the cell state-transition-path connecting density peaks of density clusters. The extracted RCSs show more homologous representation, as characterized by the cell clusters defined in Paul *et al.*, 2015. As shown in Fig. S2(f), among the extracted 10 RCSs, 4 RCSs (C7,C8, C9 and C10) are completely in the GMP clusters, and 4 RCSs (C2 (99.2%), C3, C4 and C5) are almost completely contained by Ery lineage (Fig. S2(c)). For the remaining 2 RCSs, C1 contains cells mainly from CMP (68.1%) and small portions from Ery (16.7%) and GMP (14.9%), showing the occurrence of a bifurcating event of the transition states from CMP to GMP lineage, as well as to Ery lineage. The RCS (C6) contains cells evenly from CMP (44%) and GMP (56%), indicating the transition state progressing from CMP to GMP (Fig. S2(c,f)). Owing to the scarcity of dendritic cells (30 cells) and Outlier (31 cells) in the Paul data, no cells in the RCSs correspond to DC, and only one cell in C1 corresponds to Outlier, showing that only representative cells in high-density clusters are extracted by DensityPath.

It is noteworthy that DensityPath constructs 2 branches on the trajectory of the GMP lineage

(upper left of Fig. S2(f)), showing a second bifurcating event. We thus further divide the GMP cluster into 4 cellular subtypes/lineages, including monocytes (Mono), neutrophils (Neu), eosinophils (Eos), and basophils (Baso), according to Paul *et al.*, 2015. The 4 subtypes show clear transition patterns of developmental progression (Fig. S2(e)). When looking at the cells of the RCSs along the GMP lineage, the 4 sublineages maintain their continuity and the progression order of cell types such that Neu and Baso fates are clearly separated in RCSs from the two branches of the second bifurcating events, while Mono can be found on both branches (Fig. S2(g)).

We also identify a number of branch-specific expressed genes (Fig. S2(h,i,j)). DensityPath maps each cell onto the cell state-transition path at step A5. We call the mapped point as waypoint. Each waypoint may have multiple cells mapped onto it. For each given gene, we calculate the mean gene expression value of the cells mapped to the waypoint. For example, the marker gene *Sfpi1* of GMP shows a significant increment of expression level along the branches C1-C6-C7-C8 and C1-C6-C9-C10, but it is not expressed on the branch C1-C2-C3-C4-C5 of lineage Ery. The genes *Klf1* and *Zfpm1* of Ery display remarkably high expression on the branch C1-C2-C3-C4-C5, but low expression on GMP.

Supplementary Note 5: DensityPath reconstructs the branched trajectories of HSPCs bifurcating to myeloid and erythroid precursors

We also apply DensityPath to the scRNA-seq dataset generated by Paul *et al.*, 2015, consisting of single cells extracted from the process of mouse HSPCs bifurcating to myeloid and erythroid precursors (hereinafter denoted as HSPCs data). The Wishbone algorithm reconstructed the bifurcating development trajectories on the HSPCs data (Setty *et al.*, 2016). We directly download the preprocessed expression profile of HSPCs data in Setty *et al.*, 2016, which contains 4423 single cells with 2312 informative genes.We normalize the data as in Setty *et al.*, 2016.

DensityPath first maps the scRNA-seq expression profile of 2312 informative genes onto 6 principal components by PCA and then further maps the data onto the 2-d space of EE1 and EE2 by EE. DensityPath extracts 9 separate high-density clusters of RCSs with sizes ranging from 49 to 660 that add up to 2178 of the total 4423 cells (Fig. S3(a)). It is clear that the 9 RCSs are aligned along a branch-shaped trajectory coinciding with the actual bifurcation process from HSPCs to myeloid and erythroid precursors (Fig. S3(b)), as well as the result from the Wishbone algorithm (Setty *et al.*, 2016).

To calculate pseudotime, we first select the same start cell as in Setty *et al.*, 2016. We then calculate the geodesic distance between start cell and the other cells, all of which have been mapped to the MST path on the same density landscape and identified as pseudotime of cells. The blue points on the middle right regions are identified as trunk, while points with colors, ranging from

green to yellow on the upper left and lower left regions, reflect two branches in opposite directions (Fig. S3(c)). In addition to the similar bifurcating structure, the points in density clusters that are distributed on the trunk and branches mostly agreed with the result of Wishbone (Setty *et al.*, 2016). In detail, after mapping all cells to the cell state-transition path, we find that 952 cells, which account for 98.65% of 965 cells in the trunk of the DensityPath result, are located in the trunk defined in Wishbone, including 1253 cells in branch 1 obtained by Wishbone, taking an 87.81% share of 1427 cells in one of the branches of DensityPath, and 1915 cells in branch 2 of Wishbone, accounting for 94.29% of 2031 cells located in the other branch of DensityPath. The PCC between the pseudotime calculated by DensityPath and that calculated by Wishbone is 0.83.

The expression trends of the marker genes for the myeloid and erythroid lineages are shown along the reconstructed trajectory (Fig. S3(d,e)). The myeloid markers *Mpo* and *Cebpe*, as well as the erythroid markers *Klf1*, *Gata1* and *Gata2*, are selected and show a distinct trend along the two branches, indicating that the upper left branch corresponds to the erythroid lineage, while the other one corresponds to the myeloid lineage. Furthermore, *Gata2* shows an earlier activation with high expression along the erythroid lineage compared to that for *Gata1* (Fig. S3(d)), which is consistent with our current understanding (Setty *et al.*, 2016; Kaneko *et al.*, 2010).

Supplementary Note 6: DensityPath reconstructs the branched trajectory of simulated SLS3279 data

We test DensityPath on the simulated dataset with 490 single cells and 48 genes from Zwiessele and Lawrence, 2017 (hereinafter denoted as SLS3279 data), which models bifurcating trajectory data with two terminating destinations. DensityPath recovers a branching structure on the 2-d space of EE1 and EE2 by PCA and EE, and extracts a total of 287 cells into 14 RCSs (Fig. S5(a)). We also fix the initial cell (the first cell), which is selected according to the simulated time, as the start cell and calculate the pseudotime by DensityPath (Fig. S5(b)). The PCC of pseudotime, as calculated by DensityPath, and the time in simulated progression (Fig. S5(c)) is 0.9291. Even with limited sample size (< 500), which is unfavorable to KDE, the result indicates that DensityPath can still accurately recover the tree trajectory.

Supplementary Note 7: Embedding real datasets in 3-d space of EE

We embed the 3 real datasets of Paul, HSPCs and HPE into 3-d latent space by EE. By adding the third dimension of EE3, EE achieves similar results as in 2-d space and doesn't recover additional information (Fig. S31). For Paul data, EE3 shows a negative correlation with EE2. The five labeled groups can be well separated on the plane of EE1 and EE2, but not on planes that combined with

EE3 (Fig.S31(a)). For HSPCs data, the first two dimension (EE1 and EE2) of EE recovers a clearly branched tree structure with one branching event (Fig. S31(b)). The three clusters are stacked on EE3 against other dimensions. It can also be seen on the 3-d plot (see the movie at https://github.com/ucasdp/DensityPath), where EE3 did not achieve much additional information of the branched structure. For HPE data, EE3 shows highly redundant information as it is strongly correlated with EE1 and no significant structures are newly identified (Fig. S31(c)).

Supplementary Note 8: The comparisons of DensityPath with other methods

Paul data

We compare DensityPath with Monocle2, DPT, Wishbone, and TSCAN on our processed Paul data (Paul *et al.*, 2015). (1) Monocle2 first excludes the missorted cells (Outlier) based on prior information of the cells, identifies 12 states, and reveals a branched structure with two bifurcating events (Fig. S6). Although GMP and Ery cells are correctly separated into 2 branches by Monocle2, a non-negligible number of CMP cells are assigned together along the branch of GMP (Fig. S6(b,c)). (2) The DPT method is applied 3 times iteratively to the Paul data to identify multi-bifurcating events, resulting in a trajectory with 7 branches identified (Fig. S7(a)). The 7 branches cannot fully reveal the CMP, Ery and GMP lineages, and the Outlier are assigned to branch 2, mixed with cells in CMP and Ery (Fig. S7(b,c)). (3) Wishbone reconstructs a bifurcating structure with one branching point on Paul data resulting in the assignment of 560 cells to the trunk, 985 cells to branch 1, and 1185 cells to branch 2 (Fig. S8). About two-thirds of CMP cells are located on the trunk part, while the GMP and Ery cells are located on branch 2 and branch 1, respectively, achieving a high quality of assignment (Fig. S8(b)). However, the Outlier cannot be identified and are clustered on the trunk together with DC cells, which should be independent from the progression of CMP development (Fig. S8(b,c)). (4) TSCAN generates 8 clusters with 2 clusters mostly in GMP, 1 cluster completely in Ery, 2 clusters mostly in Ery, 2 clusters mostly in CMP, and 1 cluster mainly in GMP and CMP (Fig. S9). When considering the detailed cell-type clustering result, TSCAN tends to combine the lineages in GMP closely, while the missorted cells are also distributed along the main trajectory, failing to distinguish the Outlier from the progression of non-lymphoid cells (Fig. S9(c)).

HPE data

We apply Monocle2, DPT, and Wishbone to the HPE data (Petropoulos *et al.*, 2016) to compare pseudotime calculation with the known experimental embryonic time of cells. We do not include TSCAN in this comparison since it cannot provide the pseudotime or pseudo-order on total cells. To

calculate pseudotime, the same start cell as that for DensityPath is assigned to all methods. Since the Monocle2 algorithm fails in results for this dataset, we only compare the results obtained from DPT and Wishbone to those of DensityPath. (1) DPT reconstructs a bifurcating structure wherein 1399 cells are allocated to branch 2, 85 and 29 cells are distributed into branch 1 and branch 3, respectively, and 16 cells are undefined (Fig. S10). The cells in E5 and E6 are tightly collapsed by DPT, and the trifurcating event, as clarified in Petropoulos *et al.*, 2016 , cannot be easily observed by DPT. The diffusion pseudotime of DPT has a PCC of 0.7552 with the experimental embryonic time, which is lower than that of DensityPath (0.8286). (2) Wishbone reveals a bifurcating structure with 183 cells on trunk, 1289 cells on branch 1 and 57 cells on branch 2. The branching structure (Fig. S11) assigns the cells from E3 and part of E4 to the trunk part. Other cells from E4 are assigned to branch 1, and almost all cells from E5-7 are assigned to branch 2. With the branching identified between E4 and E5, the main differentiation process at E5, E6 and E7 are not revealed by Wishbone. The pseudotime obtained in Wishbone has a PCC of 0.8418 with the experimental embryonic time, which is marginally higher than that of DensityPath (0.8286) (Table 1 in the main text).

Simulated datasets of PHATE and SLS3279

We also compare DensityPath with Monocle2, DPT and Wishbone on the simulated PHATE data with complex trajectory and both bifurcating and trifurcating events (Moon *et al.*, 2017). When compared to ground truth of the data, Monocle2, DPT and Wishbone cannot fully reconstruct the branching structure (Fig. S12 and Fig. S13) . We compute the ARI values for each branch assignment obtained, and DensityPath outperforms the other methods with ARI value of 0.7317, while none of the other methods obtained ARI greater than 0.5 (Table 1 in the main text). In addition, the PCCs between the pseudotime reported by other methods (using the same start point) and the actual time reported in the simulation data are calculated. DensityPath again performs best among the others with PCC as high as 0.9528 (Table 1 in the main text).

Since DPT and Wishbone are designed and optimized to reconstruct the trajectory with one branch point, we compare DensityPath with these methods together with Monocle2 on simulated data SLS3279 in Zwiessele and Lawrence (2017). All methods recover the known branched structure (Fig. S14, Fig. S15 and Fig. S16). Monocle2 also reveals a short tiny branch in one of the ends of main branches (Fig. S14). When comparing the PCC of pseudotime and simulated time, DensityPath has the highest PCC of 0.9291, while DPT has the second highest PCC of 0.9270 (Table 1 in the main text). Since no branch assignment information is available for the simulated SLS3279 data, we cannot compare the ARI values.

Supplementary Note 9: Robustness and Sensitivity of DensityPath

Parameter λ in EE of DensityPath

A user-defined parameter in the DensityPath algorithm, λ, serves as a regularization term when calculating the embedded algorithm EE (see Equation (1) in the main text). DensityPath can generate robust results with λ loosely choosing from among a wide range from 5 to 30 around the default value of 10 on the 5 datasets used in this study (Fig. S17). Although the number of RCSs may vary according to different λ, the embedded trajectories are robustly recovered in most cases. In addition, to test the robustness of pseudotime and branch assignment by adjusting λ, the values of PCC and ARI are calculated and robust to the choices of λ around 10 (Fig. S18).

Parameter k in LSC of DensityPath

To get the connected components of $\hat{L}_t$, the LSC of DensityPath constructs a k-NN graph of $\{x_1, \ldots, x_n\}$ on the reduced-dimension space and then finds the connected components of the subgraph with the nodes restricted to the index set $I_t = \left\{ i : \hat{f}_n(x_i) > t \right\}$. The optimal choice of k is still an open question. In general, small k will produce more connected components, resulting in more RCSs detected by DensityPath, while large k will produce fewer connected components, resulting in fewer RCSs detected by DensityPath. When $k = 0$, each cell will be considered as connected component. In this study, we empirically choose the default of k as round$(n/100)$ in the DensityPath algorithm, where n is the number of samples/cells.

We test the robustness of DensityPath on choices of k on the 5 datasets of Paul, HSPCs, HPE, PHATE and SLS3279 (Fig. S19, Fig. S20,Fig. S21, Fig. S22, and Fig. S23). As expected, the number of RCSs on the trajectories by DensityPath will remain the same or will be sightly decreased when choosing k from 60%$*$round$(n/100)$ to 140%$*$round$(n/100)$. However, the major structure of the embedded trajectory remains unchanged, especially in the Paul, HSPCs, SLS3279 and PHATE data. In the HPE data, when k gradually decreases, some new RCSs and bifurcations will appear and contain RCSs with a small number of cells. We regard these newly appearing small RCSs as intermediate cell states or rare cell states, which are not stable and hard to detect (MacLean *et al.*, 2018). More RCSs on the trajectory is representative of a more complex transition process.

In addition, to test the robustness of pseudotime and branch assignment by adjusting k, the values of PCC and ARI are calculated, and the robustness of the choices of k is validated (Fig. S24).

Number of input informative genes

In the analysis of the 3 real datasets, we adopt the same informative genes as the previous study (Haghverdi *et al.*, 2016; Setty *et al.*, 2016; Rizvi *et al.*, 2017). We further test the robustness and sensitivity of DensityPath on choices of informative genes on the HSPCs and HPE datasets.

For the HSPCs data, we subsample subsets of a total 2312 informative genes downloaded, with proportions ranging from 55% to 95%, to reconstruct cell development trajectory. We resample without replacement 10 times for each proportion and calculate the PCC between the pseudotime calculated by DensityPath and Wishbone algorithm for each resampling. The median of PCC values are very stable across proportions, while the variances of the PCCs decrease rapidly, becoming close to 0 when the proportions reach above 80% (Fig. S25(a)).

For HPE data, we select different numbers of genes with top variances as input informative genes, as in Rizvi *et al.*, 2017. The numbers of genes having top variances range from 920 to 8280 with equal increments, covering the number of 4600 used in the main text, as well as in Rizvi *et al.*, 2017. The 9 reconstructed trajectories at different number of informative genes are almost the same, only except that the E5R lineage, are unidentified by the 3 consecutive gene numbers of 1840, 2769 and 3680 (Fig. S26). Meanwhile, the PCC between the pseudotime calculated by DensityPath algorithm and the experimental time of embryonic days demonstrates the robustness under various number of informative input genes (Fig. S25(b)).

In the absence of *a priori* information about informative genes, the Z-score of transformed gene expression can be used to select informative genes with large variance. Since DensityPath is not sensitive to the number of input informative genes, genes with Z-score above 0.5 can be applied in practice.

Subsampling cells

We conduct the robustness analysis of DensityPath by subsampling (without replacement) the cells on the PHATE and HSPCs data. For each of the two datasets, when fixing a proportion of $m\%$ (ranging from 10% to 90%), we subsample $m\%$ of the total cells without replacement as a new dataset, and then apply DensityPath with the complete procedures from A1 to A6 on the new dataset to reconstruct cell state-transition path, map cells onto the path, and cacualate the pseudotime. For a fixed proportion $m\%$, we repeat the subsampling 10 times to obtain 10 new datasets. We find that with the increase of $m\%$, the PCCs on pseudotime and ARIs on branch assignment obtained by subsampling cells are quite stable in median values with the decrease of variances (Fig. S27).

Dropout events

Dropout events exist in the current scRNA-seq studies, meaning that only a small fraction of the transcripts present in each cell are sequenced (Huang *et al.*, 2018). We evaluate whether the dropout events have significant impact on the performance of DensityPath. Since the mechanism of dropout events in scRNA-seq is still not clear, we take an indirect approach to evaluate the dropout events. We adopt the powerful software Saver (Huang *et al.*, 2018), a recently published method to impute dropout events in the scRNA-seq data, to recover gene expression. We take the example of the HSPCs data, which contain 4423 single cells with 2312 informative genes. Although the 2312 informative genes with large variances have been selected out of the total genes, 69.66% of the elements in the gene expression matrix with a dimension of 4423×2312 are zero. We impute the gene expression of HSPCs data with Saver and compare the results before and after the recovery. The trajectory reconstructed by DensityPath after gene expression recovery (Fig. S28) is very consistent with that before the recovery (Fig. S3). In addition, the PCC between the pseudotime calculated by DensityPath after recovery and the time calculated by Wishbone is 0.7585, while the PCC between the pseudotime calculated by DensityPath before and after recovery is 0.9647, showing stable performance in the presence of dropout events.

Supplementary Figures

Figure S1: Overview of Level-set clustering.

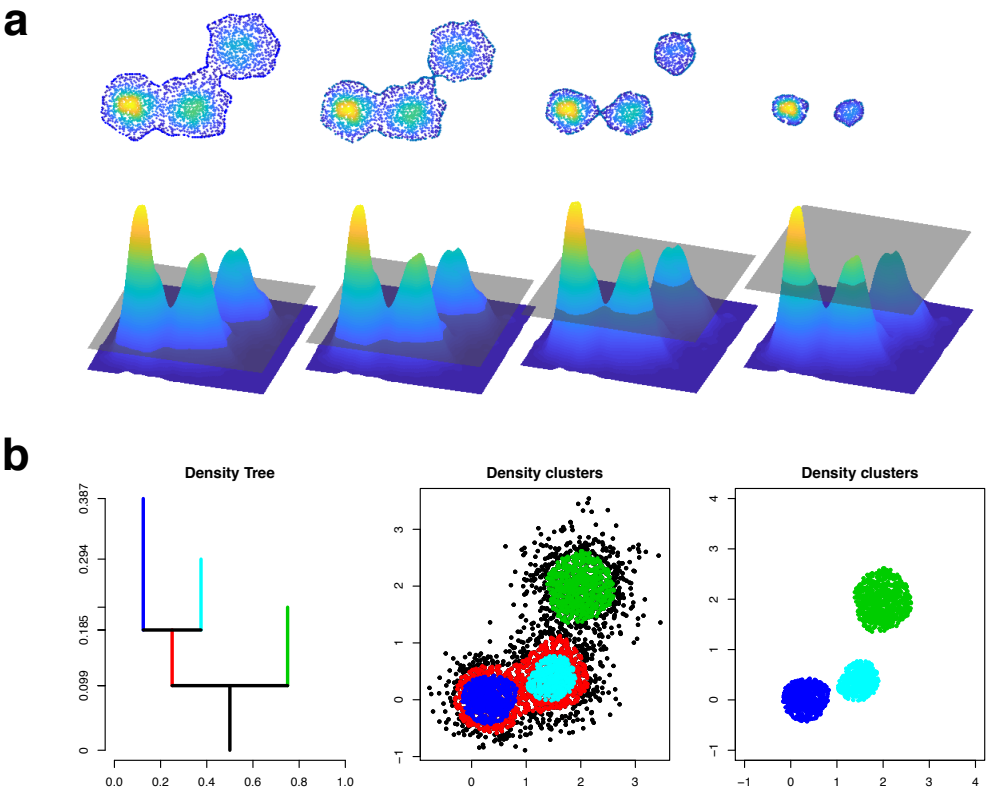

Fig. S1. Overview of level-set clustering (LSC). **(a)** DensityPath estimates the single-cell density landscape (function) of the reduced-dimension space of gene expression (herein $d = 2$); as threshold t increases, LSC of DenistyPath analyzes the structure of the density landscape by calculating the t-upper level-sets $\hat{L}_t$ and the connected component(s) $\hat{\mathcal{C}}_t$. **(b)** DensityPath applies LSC to the level-set results to construct the density tree (left), clusters the single cells according to the density tree (middle), and selects the separate high-density clusters of representative cell states (right), which are represented by the external branches of the constructed density tree.

Figure S2: DensityPath reveals the developmental structure of bone marrow cells.

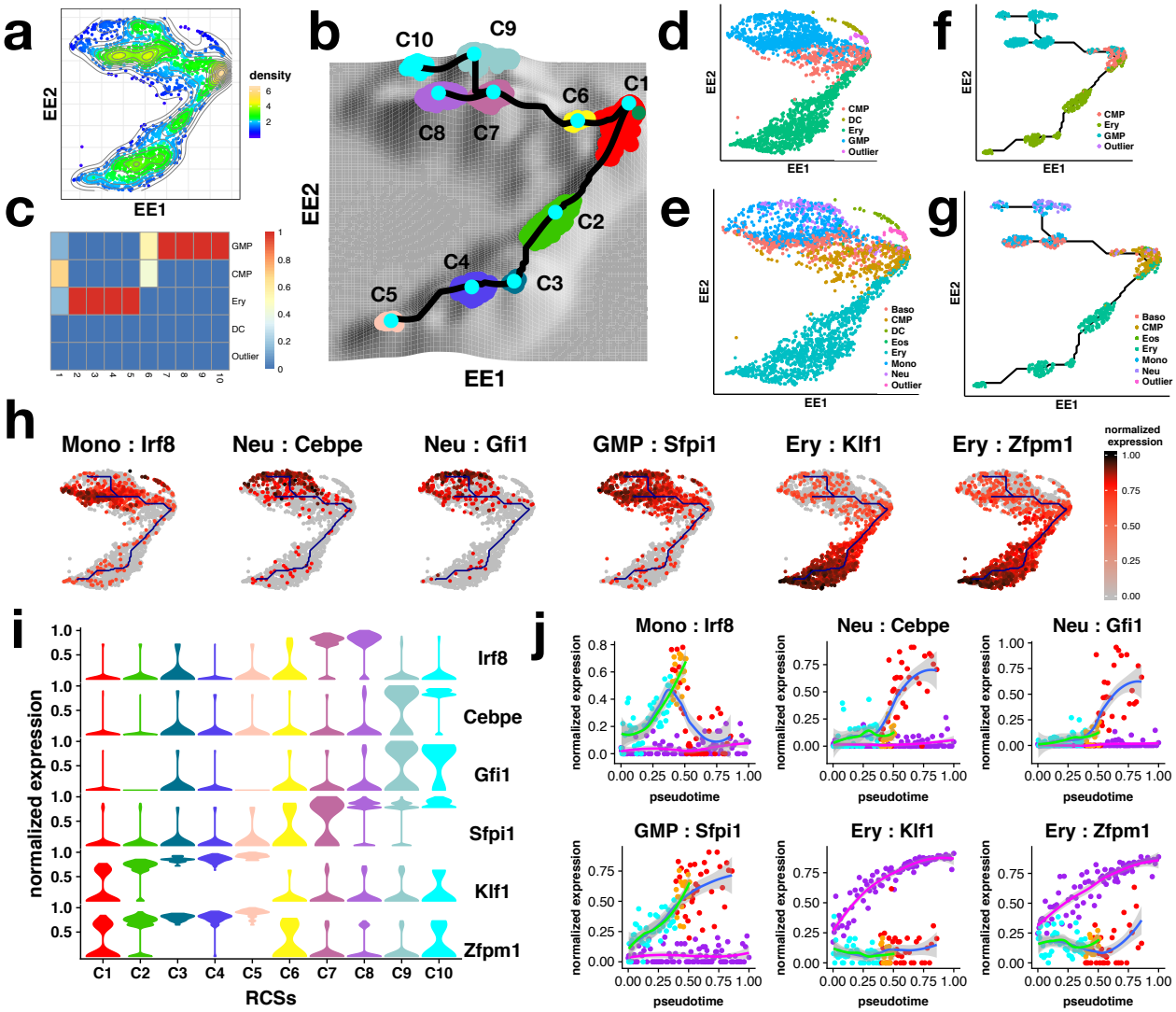

Fig. S2. DensityPath reveals the developmental structure of bone marrow cells. (Continued in the next page)

(a) Density landscape of cells on the 2-d space of EE1 and E2 coordinates. The points (single cells) are colored according their density at their EE coordinates, from blue to yellow, indicating the density level from low to high, respectively. **(b)** The 10 RCSs are extracted by LSC with indexes labeled (C1-C10) in the figure. A branching trajectory is obtained by constructing the MST of density peaks in RCSs using the geodesic distance of density surface. Given the start cell marked in the plot with dark green, a progression with two bifurcating events can be seen. **(c)** The heatmap shows the percentage of cells in each of 10 RCSs (x-axis) that are distributed into clusters CMP, Ery, DC, GMP and Outlier by Paul *et al.*, 2015 (y-axis), respectively. **(d,f)** All cells (d) and the cells in the 10 RCSs (f) are plotted on the EE1 and EE2 coordinates, showing a separation of clusters CMP, Ery, DC, GMP and Outlier by Paul *et al.*, 2015 with progression trends. **(e,g)** are the same as (d,f), except using detailed cluster information as CMP, Ery, DC, Mono, Neu, Eos, Baso, and Outlier by Paul *et al.*, 2015. **(h)** The scatter plots of gene expression levels of known marker genes (*Irf8*, *Cebpe*, *Gfi*, *Sfpi1*, *Klf1* and *Zfpm1*) of clusters Mono, Neu, GMP, Ery on EE1 and E2 coordinates. **(i)** The violin plot of gene expression levels of the known marker genes on the cells from different RCSs annotated in **(b)**. **(j)** The expression of known marker genes on the pseudo-trajectory. The x axis represents the pseudotime of waypoints on the trajectory. The y axis represents the gene expression of each waypoint, which is the mean gene expression of all samples mapped to the waypoint. The scatter points colored in cyan, orange, red and purple represent the waypoints on branches C1-C6-C7, C7-C8, C7-C9-C10 and C1-C2-C3-C4-C5, respectively. The lines colored in green, blue and magenta represent the smooth gene expression of waypoints on branches C1-C6-C7-C8, C1-C6-C7-C9-C10 and C1-C2-C3-C4-C5, respectively.

Figure S3: DensityPath recovers the bifurcating structure of HSPCs data.

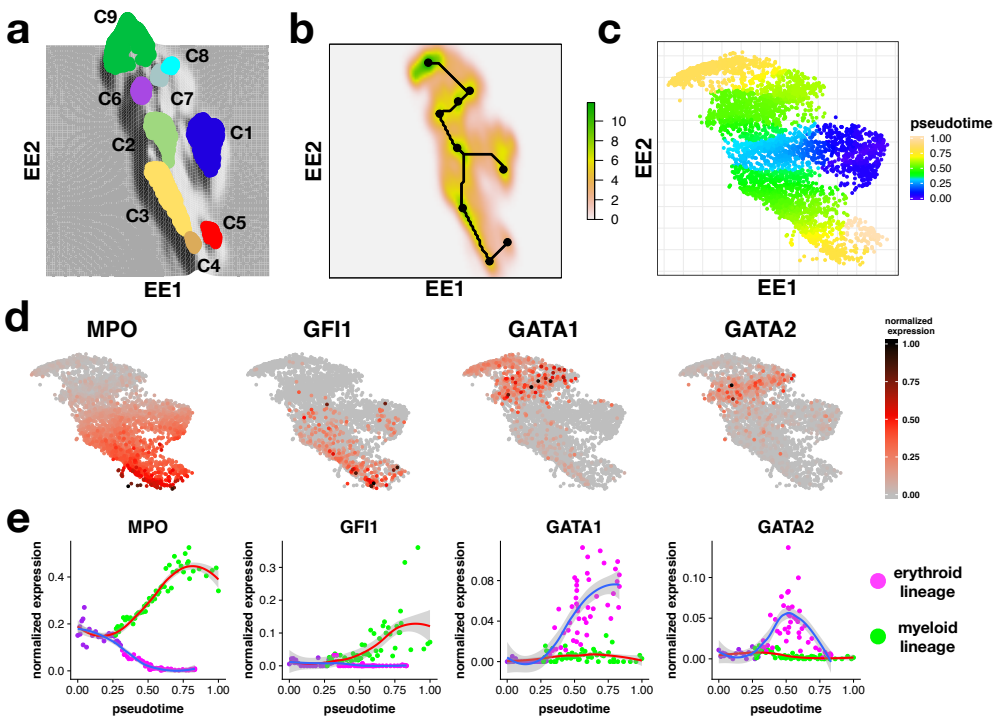

Fig. S3. DensityPath recovers the bifurcating structure of HSPCs data. **(a)** DensityPath extracts 9 separate RCSs, and cells from different RCSs are color coded. **(b)** The bifurcating trajectory constructed by connecting the peaks of the RCSs. **(c)** Given the same start cell (cell no. 1433) as that of Wishbone, DensityPath calculates the pseudotime of each cell, setting the time of the start cell as 0. **(d)** The scatter plots of gene expression levels of known marker genes (*Mpo*, *Gfi1*, *Gata1* and *Gata2*) of clusters Mono, Neu, GMP, Ery on the EE1 and EE2 coordinates. **(e)** The expression levels of the known marker genes on the pseudo-trajectory. The x axis represents the pseudotime of waypoints on the trajectory. The y axis represents the gene expression of each waypoint, which is the mean gene expression of all samples mapped to the waypoint. The scatter points colored in purple, green and magenta represent the waypoints on branches C1-C2, C2-C3-C4-C5 and C2-C6-C7-C8-C9, respectively. The lines colored in red and blue represent the smooth gene expression of waypoints on branches C1-C2-C3-C4-C5 and C1-C2-C6-C7-C8-C9, respectively.

Figure S4: Comparison of different dimension reduction methods, including PCA, tSNE, Diffusion Map, PHATE, and EE, on HSPCs data.

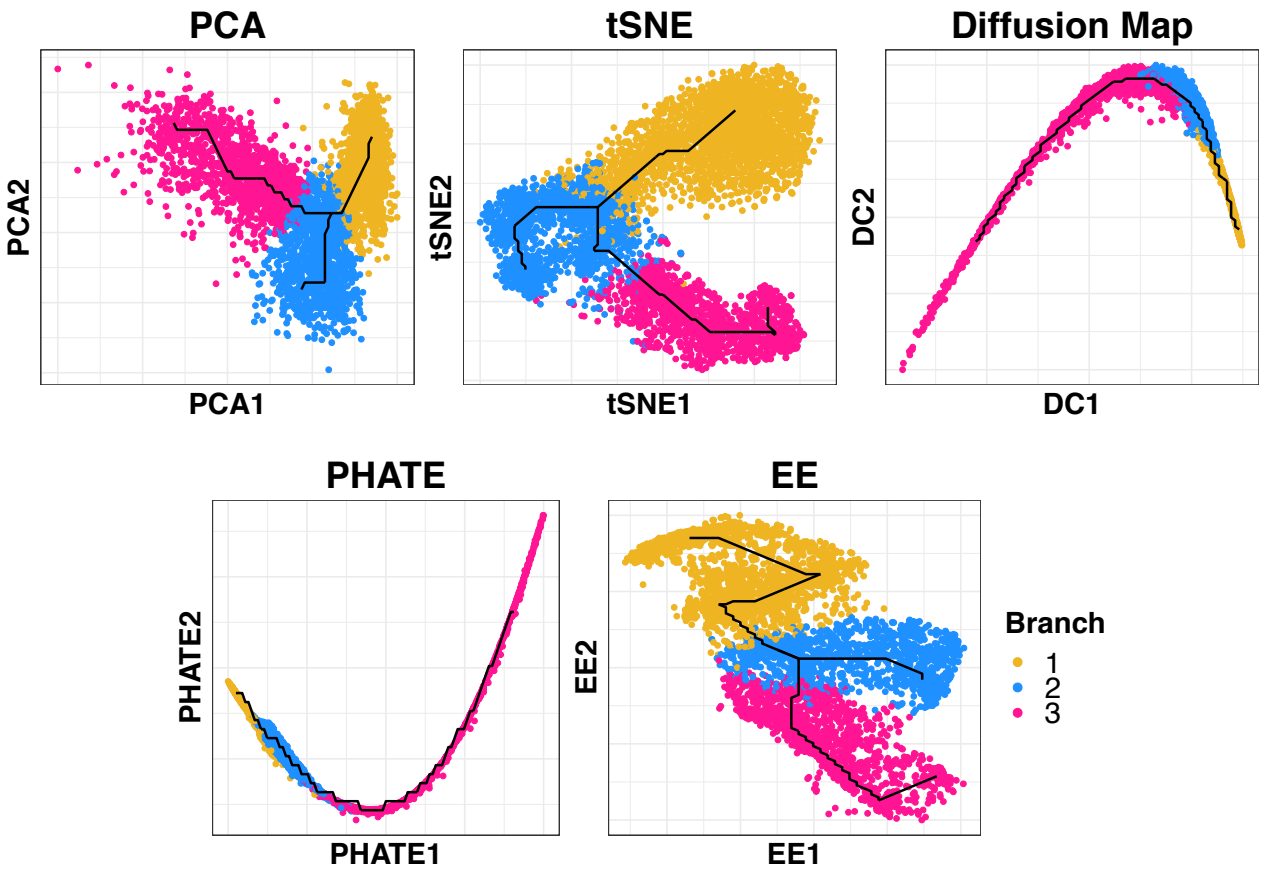

Fig. S4. Comparison of different dimension reduction methods, including PCA, tSNE, Diffusion Map, PHATE algorithm, and EE on HSPCs data. Different colors reflect the assignment of cells in Wishbone, while the black curves correspond to their trajectories reconstructed according to the procedures described in steps A2-A4 of DensityPath in Methods.

Figure S5: DensityPath real the bifurcating trajectory on the simulated SLS3279 data.

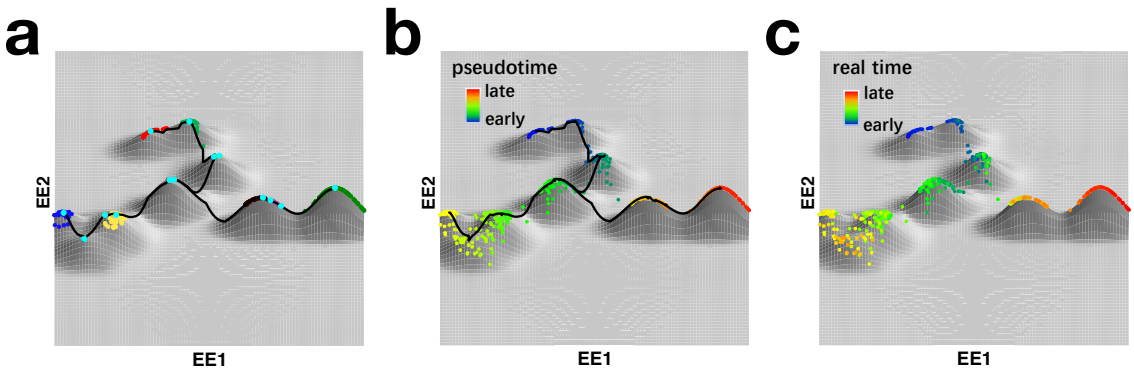

Fig. S5. DensityPath reveals the bifurcating trajectory on the simulated SLS3279 data. **(a)** DensityPath constructs the density landscape of SLS3279 data on the 2-d space of EE1 and EE2 coordinates, extracts 14 RCSs and reconstructs the cell development bifurcating trajectory on the density landscape. **(b)** DensityPath calculates the pseudotime of SLS3279 data by fixing the initial cell in simulation (the first cell of the data) as start cell. **(c)** The real time of cells in the SLS3279 dataset distributed in the density landscape.

Figure S6: Monocle2 analysis on Paul data.

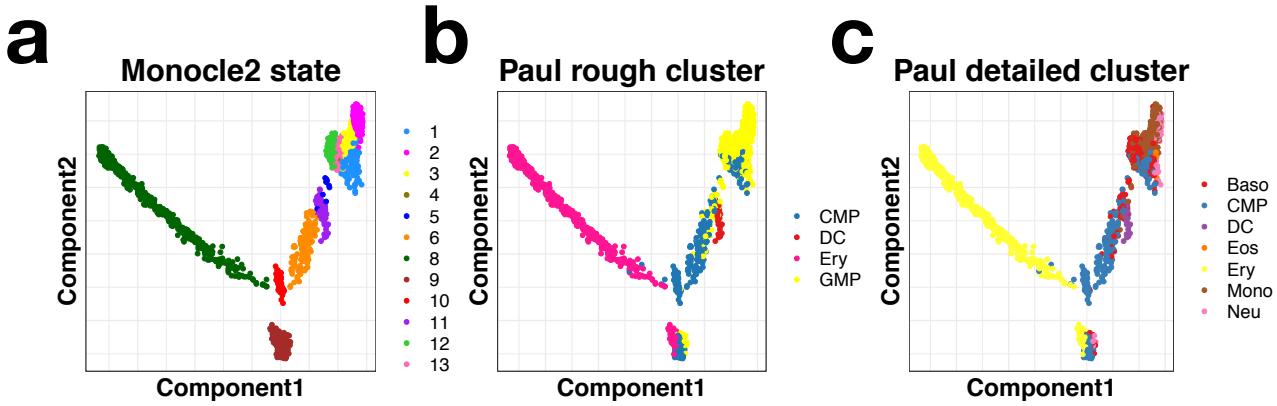

Fig. S6. Monocle2 analysis on Paul data. Monocle2's code, "Paul_dataset_analysis_final.html" (from the online supplementary script of Qiu *et al.*, 2017), is performed on our processed Paul data. **(a)** Monocle2 removes Outlier based on prior cell information, identifies 13 states, and reconstructs a multi-bifurcating trajectory. **(b)** The distribution of the main four cell types. **(c)** The distribution of cells based on detailed classification information on the 2-d space.

Figure S7: DPT analysis on Paul dataset.

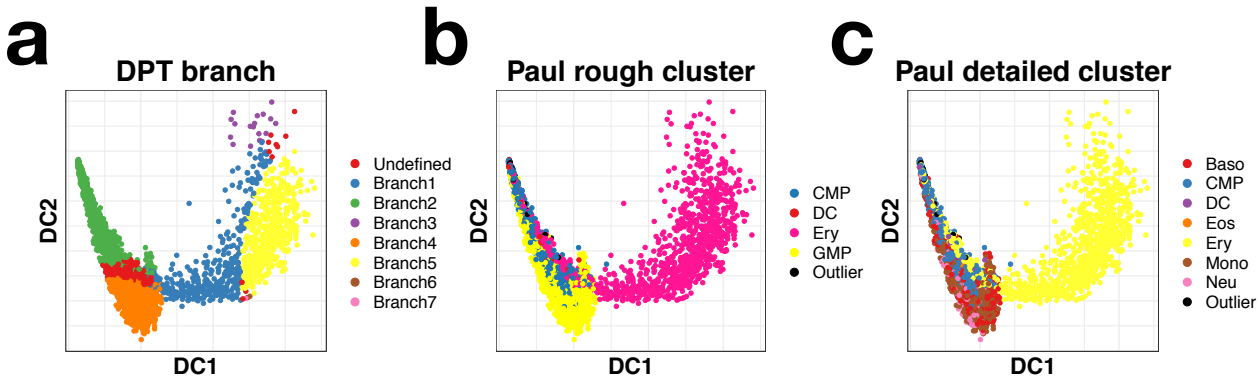

Fig. S7. DPT analysis on Paul dataset. The DPT code, "run_a_general_example.m", downloaded from supplementary material of Haghverdi *et al.*, 2016, is performed by setting the parameter sigma as 20. We carry out DPT analysis three times. We first apply DPT to the whole cells and obtained 3 branches for 2182, 522 and 18 cells, respectively; then, we perform DPT analysis on branch 1 (2182 cells) and branch 2 (522 cells) separately, resulting a total of 7 branches. The results of DPT visualization are demonstrated through Diffusion Map. **(a)** DPT reconstructs a bifurcating trajectory. **(b)** The distribution of the five main cell types. **(c)** The distribution of cells based on detailed classification information on the 2-d space.

Figure S8: Wishbone analysis on Paul data.

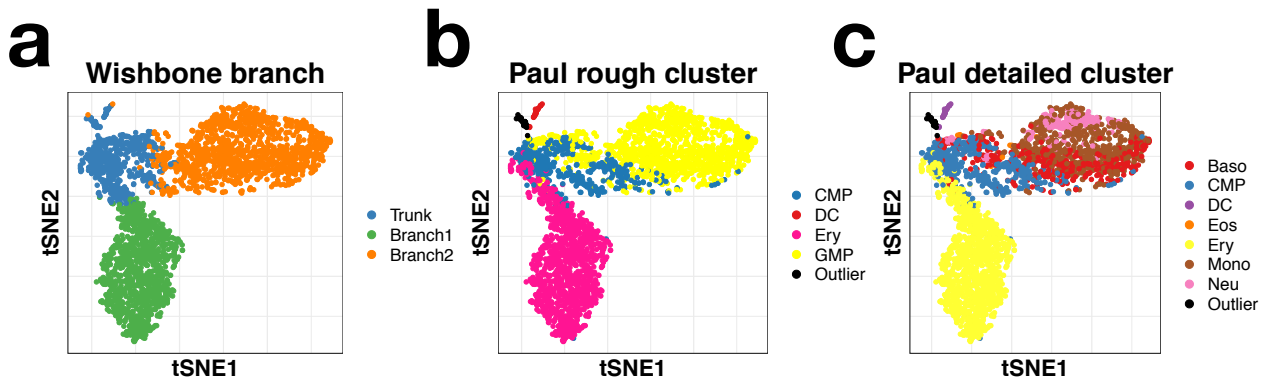

Fig. S8. Wishbone analysis on Paul data. Wishbone is performed by setting the parameters pca and k as 15 and 15, respectively. The visualization of results by Wishbone is demonstrated through tSNE. **(a)** The bifurcating trajectory reconstructed by Wishbone. **(b)** The distribution of the five main cell types in 2-d plane. **(c)** The distribution of cells with detailed classification information on the 2-d space.

Figure S9: TSCAN analysis on Paul data.

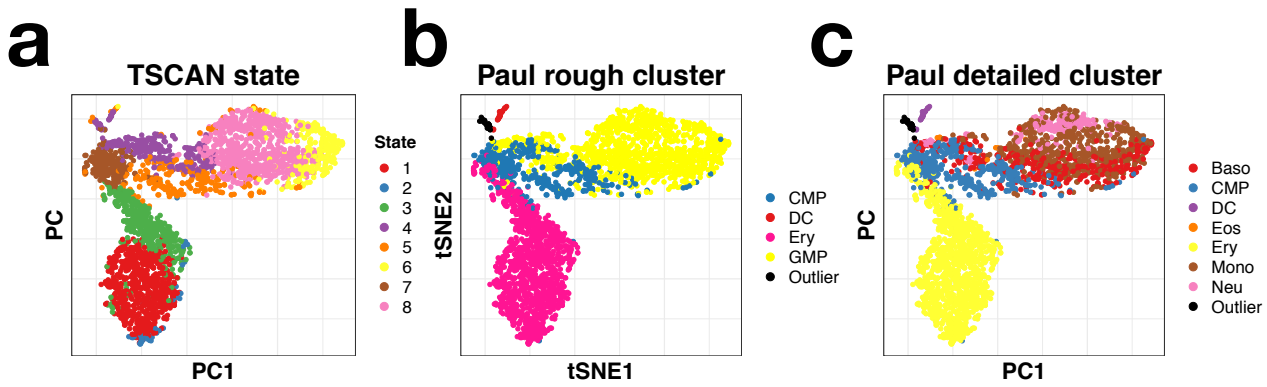

Fig. S9. TSCAN analysis on Paul data. The visualization of results by TSCAN is demonstrated through PCA. **(a)** The bifurcating trajectory reconstructed by TSCAN. **(b)** The distribution of the five main cell types in 2-d plane by TSCAN. **(c)** The distribution of cells with detailed classification information on the 2-d space.

Figure S10: DPT analysis on HPE data.

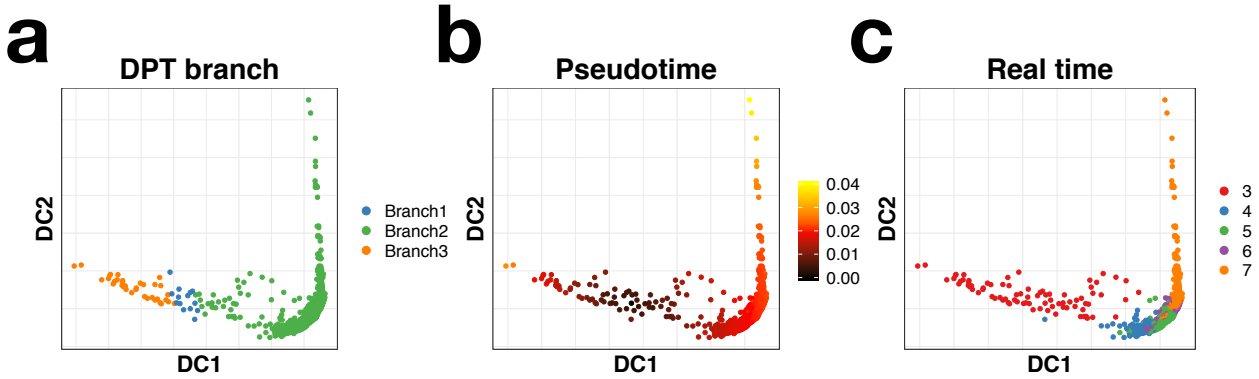

Fig. S10. DPT analysis on HPE data. The DPT code "run_a_general_example.m" downloaded from supplementary material of Haghverdi *et al.*, 2016 is performed by setting the parameter sigma as 200. **(a)** DPT constructs a bifurcating trajectory. **(b)** The pseudotime of each cell calculated by DPT. **(c)** The real time of each cell displayed on the trajectory reconstructed by DPT.

Figure S11: Wishbone analysis on HPE data.

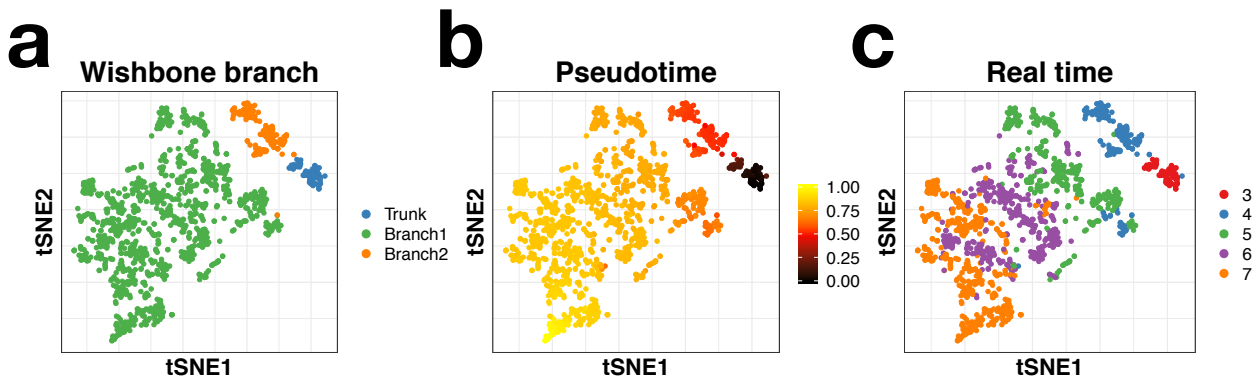

Fig. S11. Wishbone analysis on HPE data. Wishbone is performed by setting the parameters pca and k as 100 and 30, respectively. The visualization of results in Wishbone is demonstrated through tSNE. **(a)** Wishbone reconstructs a bifurcating trajectory. **(b)** The pseudotime of each cell calculated by Wishbone. **(c)** The real time of each cell is displayed on the trajectory reconstructed by Wishbone.

Figure S12: Comparison of pseudotime reconstructed by different methods on PHATE simulation data.

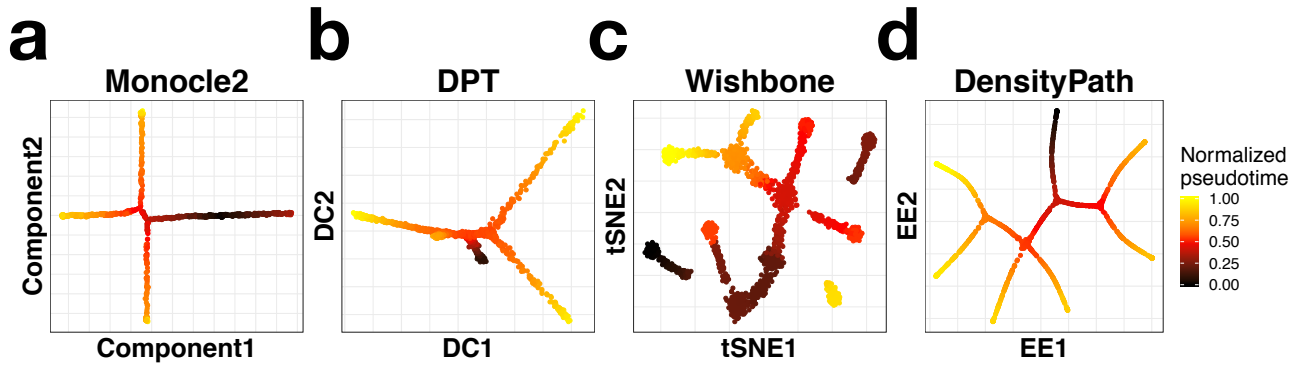

Fig. S12. Comparison of pseudotime reconstructed by different methods on PHATE simulation data. **(a)** The pseudotime reconstructed by Monocle2. The Monocle2 code "analysis_complex_tree_structure.R" is performed by setting the parameters λ and s as 1000 and 0.5, respectively, using the DDRTree method. **(b)** The pseudotime reconstructed by DPT. The DPT code "run_a_general_example.m" downloaded from supplementary material of Haghverdi *et al.*, 2016 is performed by setting the parameter sigma as 150. We carry out DPT analysis four times. We first apply DPT to the whole cells and obtain 3 branches with cell numbers 419, 632 and 337, respectively. We then perform DPT analysis three times on 3 branches, separately, resulting in a total of 9 branches. **(c)** The pseudotime reconstructed by Wishbone. Wishbone is performed by setting the parameters pca and k as 100 and 30, respectively. **(d)** The pseudotime reconstructed by DensityPath.

Figure S13: Comparison of the branches assigned by different methods to the real assignment of branches on PHATE simulation data.

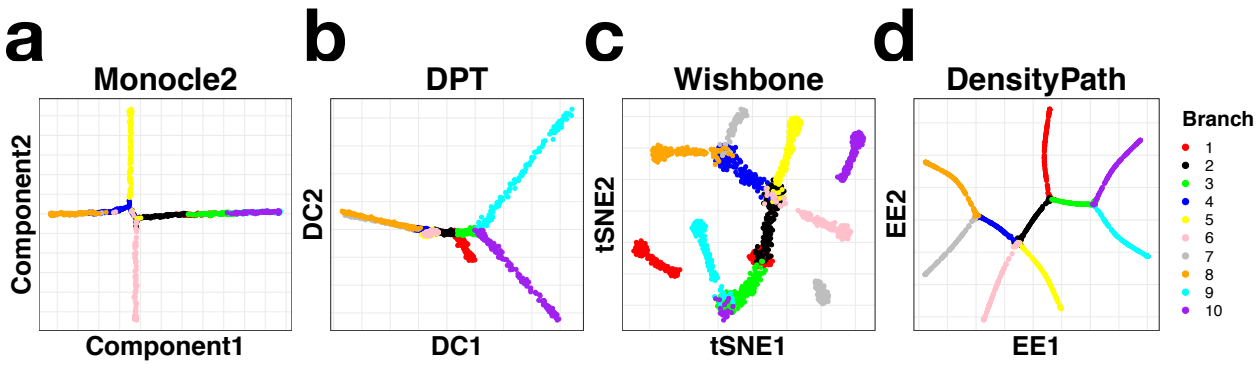

Fig. S13. Comparison of the branches assigned by different methods to the real assignment of branches on PHATE simulation data. Here, the 10 colors correspond to the 10 segments of path according to the known structure. The procedures and parameter settings for these methods are taken as Fig. S12. **(a)** The branches assigned by Monocle2. **(b)** The branches assigned by DPT. **(c)** The branches assigned by Wishbone. **(d)** The branches assigned by DensityPath.

Figure S14: Monocle2 analysis on simulation data SLS3279.

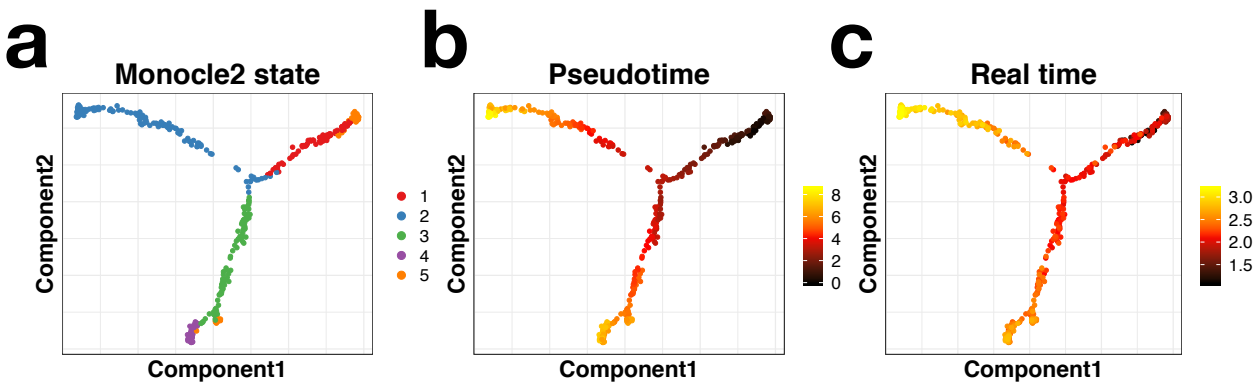

Fig. S14. Monocle2 analysis on simulation data SLS3279. The Monocle2 code "analysis_complex_tree_structure.R" is performed by setting the parameters λ and s as 0.0464 and 0.05, respectively, using the DDRTree method. **(a)** Monocle2 identifies 5 states in SLS3279 data. **(b)** The pseudotime of each cell reconstructed by Monocle2 distributed in 2-d plane. **(c)** The real time of each cell obtained by Monocle2 distributed in 2-d plane.

Figure S15: DPT analysis on simulation data SLS3279.

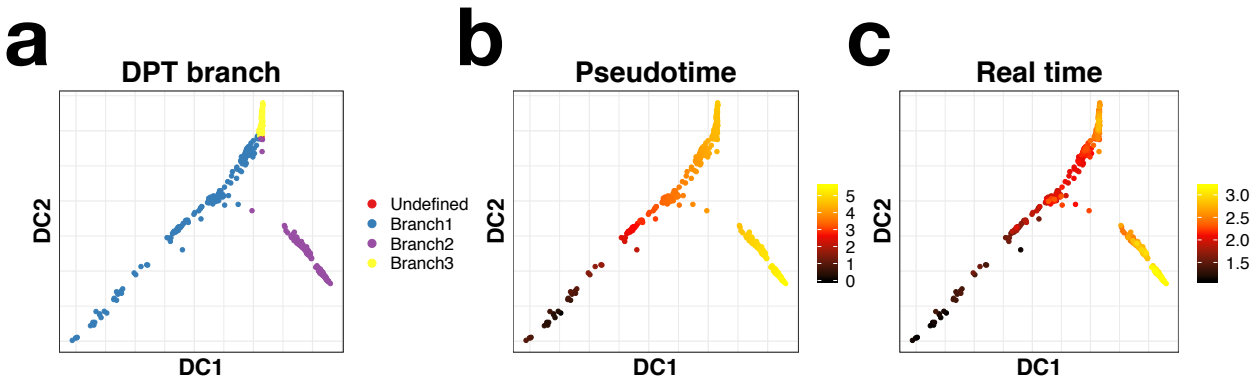

Fig. S15. DPT analysis on simulation data SLS3279. The DPT code "run_a_general_example.m" downloaded from supplementary material of Haghverdi *et al.*, 2016 is performed by setting the parameter sigma as 2.5. **(a)** The bifurcating trajectory reconstructed by DPT. **(b)** The pseudotime of each cell calculated by DPT. **(c)** The real time of each cell displayed on the trajectory reconstructed by DPT.

Figure S16: Wishbone analysis on simulation data SLS3279.

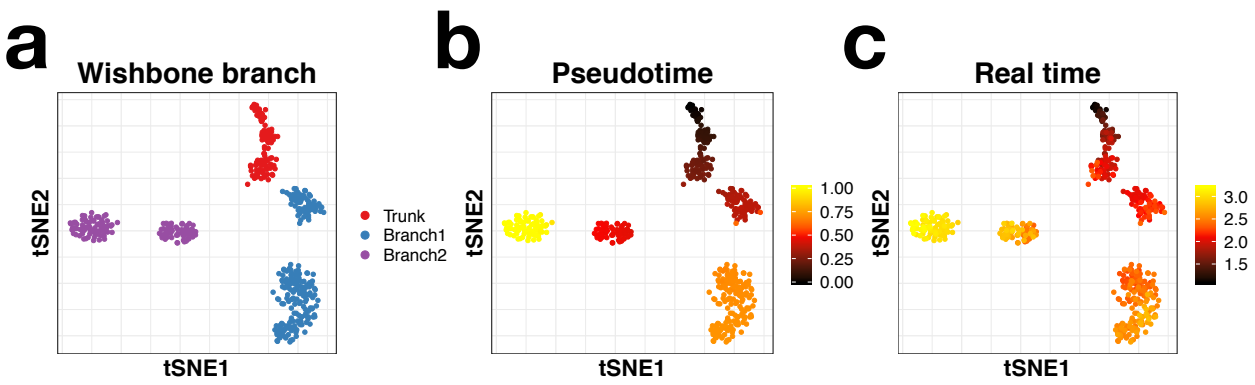

Fig. S16. Wishbone analysis on simulation data SLS3279. Wishbone is performed by setting the parameters pca and k as 20 and 70, respectively. **(a)** The bifurcating trajectory reconstructed by Wishbone. **(b)** The pseudotime of each cell calculated by Wishbone. **(c)** The real time of each cell displayed on the trajectory reconstructed by Wishbone.

Figure S17: DensityPath trajectory under different λ on Paul, HSPCs, HPE, PHATE and SLS3279 data.

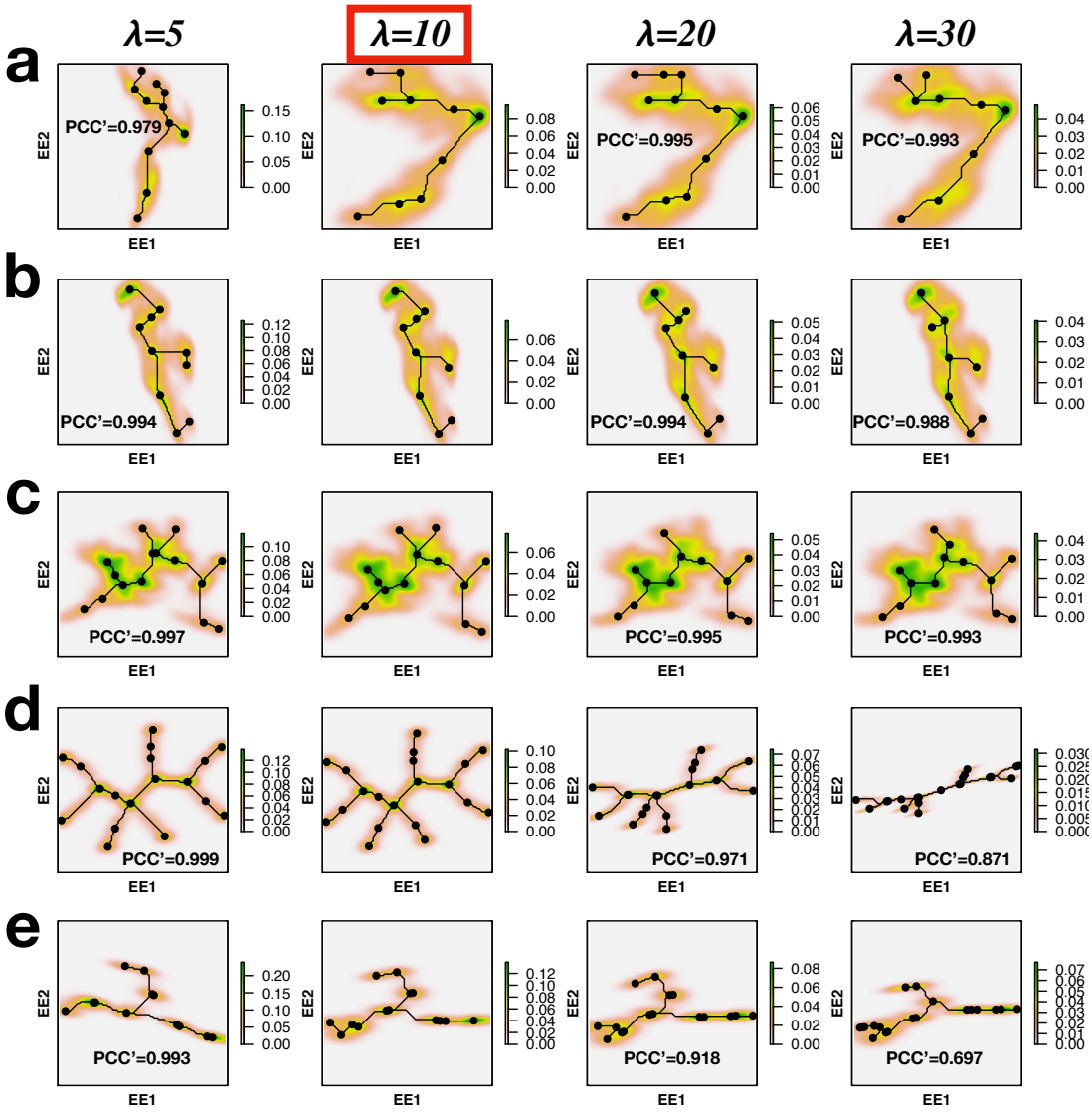

Fig. S17. DensityPath trajectory results under different λ on Paul, HSPCs, HPE, PHATE and SLS3279 data. The λ with red box is the default value of DensityPath (second column). The PCC' in each plot refers to PCC of pseudotimes calculated by DensityPath between different λ and the default value. **(a)** Trajectories obtained by DensityPath under different λ on Paul data. **(b)** Trajectories obtained by DensityPath under different λ on HSPCs data. **(c)** Trajectories obtained by DensityPath under different λ on HPE data. **(d)** Trajectories obtained by DensityPath under different λ on PHATE data. **(e)** Trajectories obtained by DensityPath under different λ on SLS3279 data.

Figure S18: The robustness of DensityPath on the parameter λ.

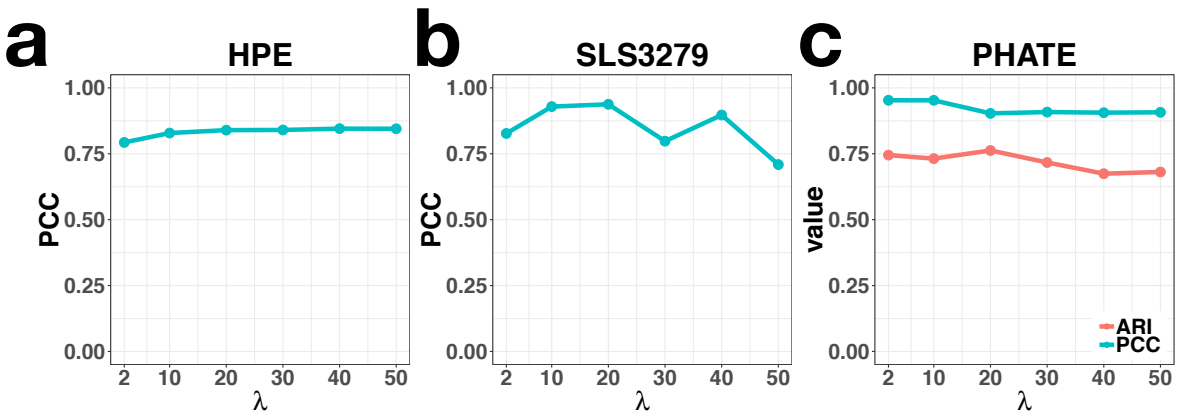

Fig. S18. The robustness of DensityPath on the parameter λ. The robustness analysis for parameter λ in EE where **(a)** is the values of the PCC in HPE data, **(b)** is the PCC values in SLS3279 data and **(c)** is the results of PCC and ARI of PHATE data.

Figure S19: The trajectories under different k on Paul data.

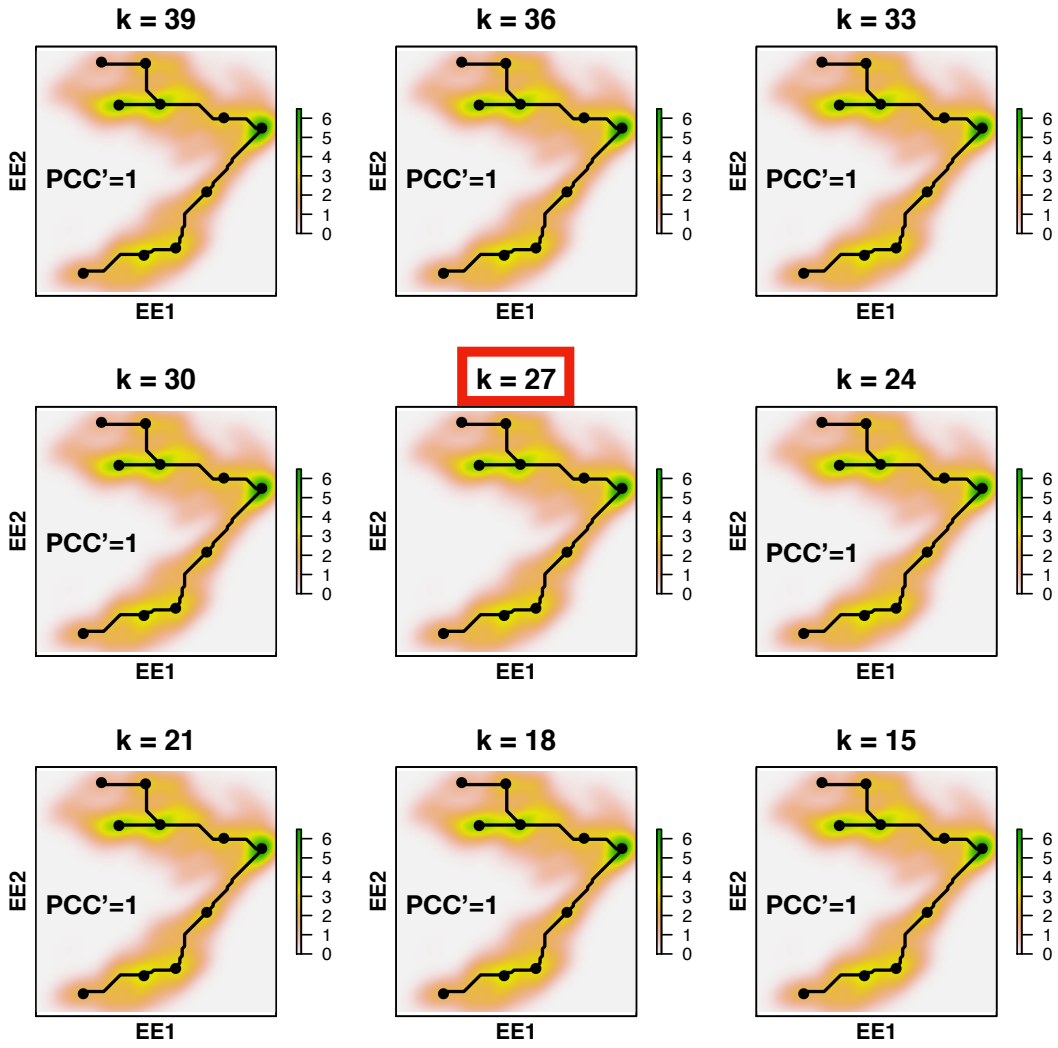

Fig. S19. The trajectories under different k on Paul data. The value of k with red box in the center is the default choice of DensityPath. The PCC' in each plot refers to PCC of pseudotime calculated by DensityPath between different k values and the default k value of DensityPath.

Figure S20: The trajectories under different k on HSPCs data.

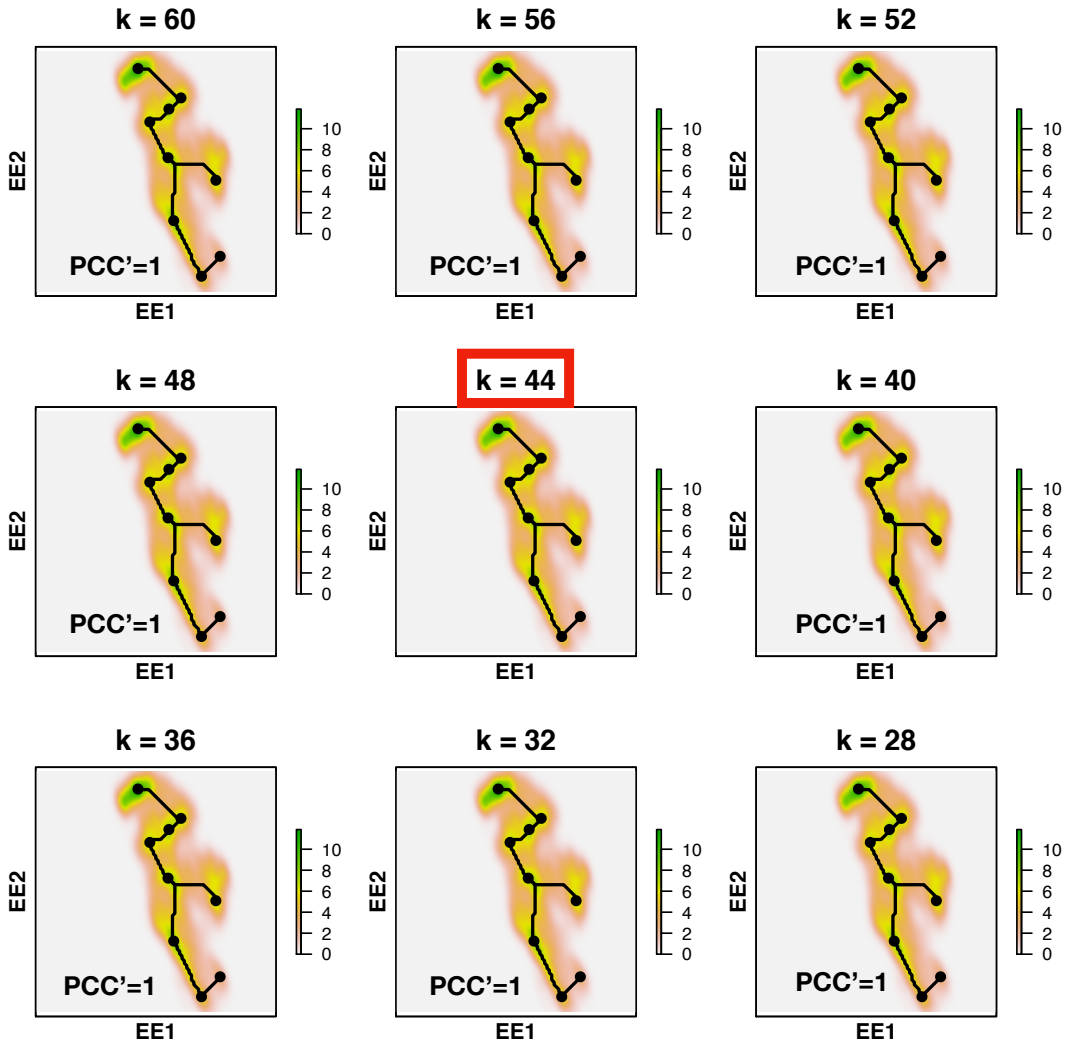

Fig. S20. The trajectories under different k on HSPCs data. The value of k with red box in the center is the default choice of DensityPath. The PCC' in each plot refers to PCC of pseudotime calculated by DensityPath between different k values and the default k value of DensityPath.

Figure S21: The trajectories under different k on HPE data.

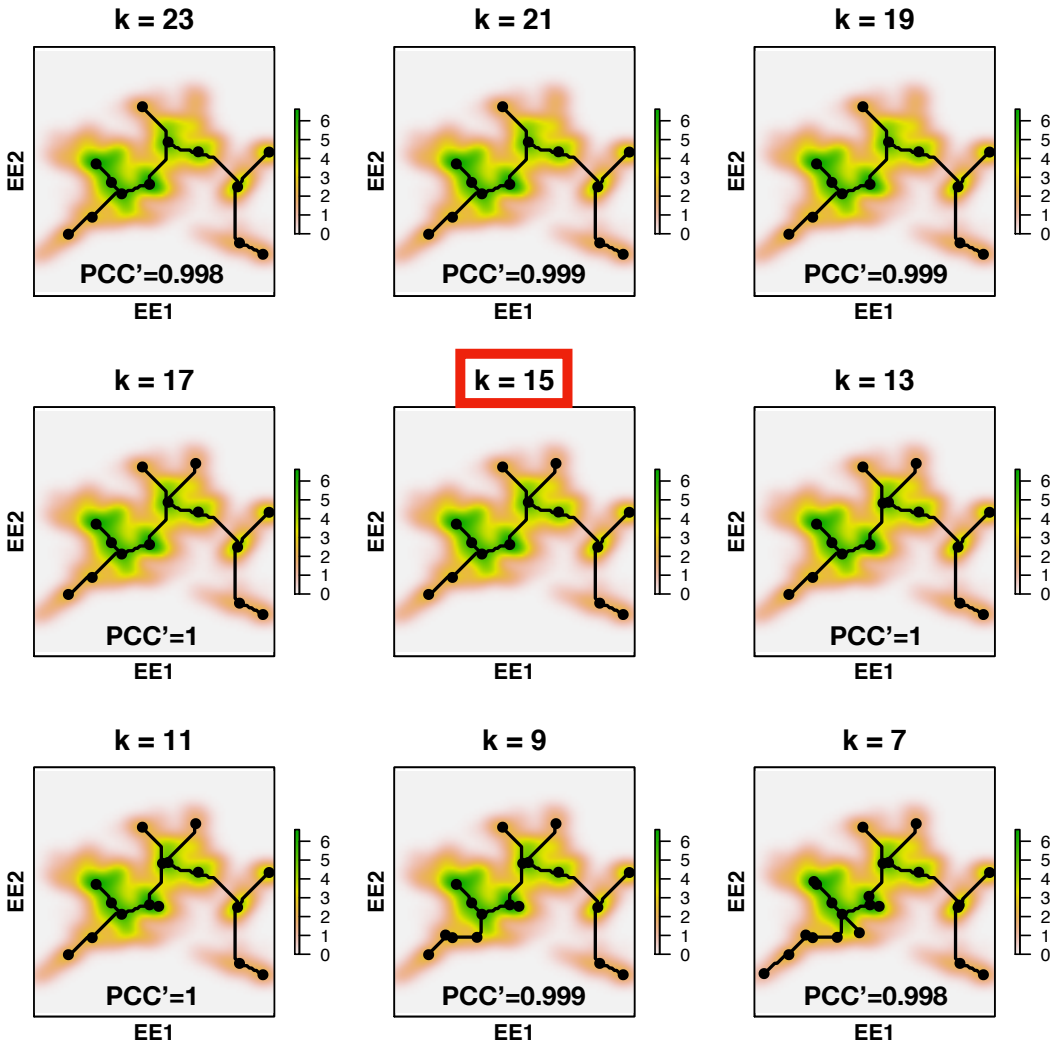

Fig. S21. The trajectories under different k on HPE data. The value of k with red box in the center is the default choice of DensityPath. The PCC' in each plot refers to PCC of pseudotime calculated by DensityPath between different k values and the default k value of DensityPath.

Figure S22: The trajectories under different k on PHATE data.

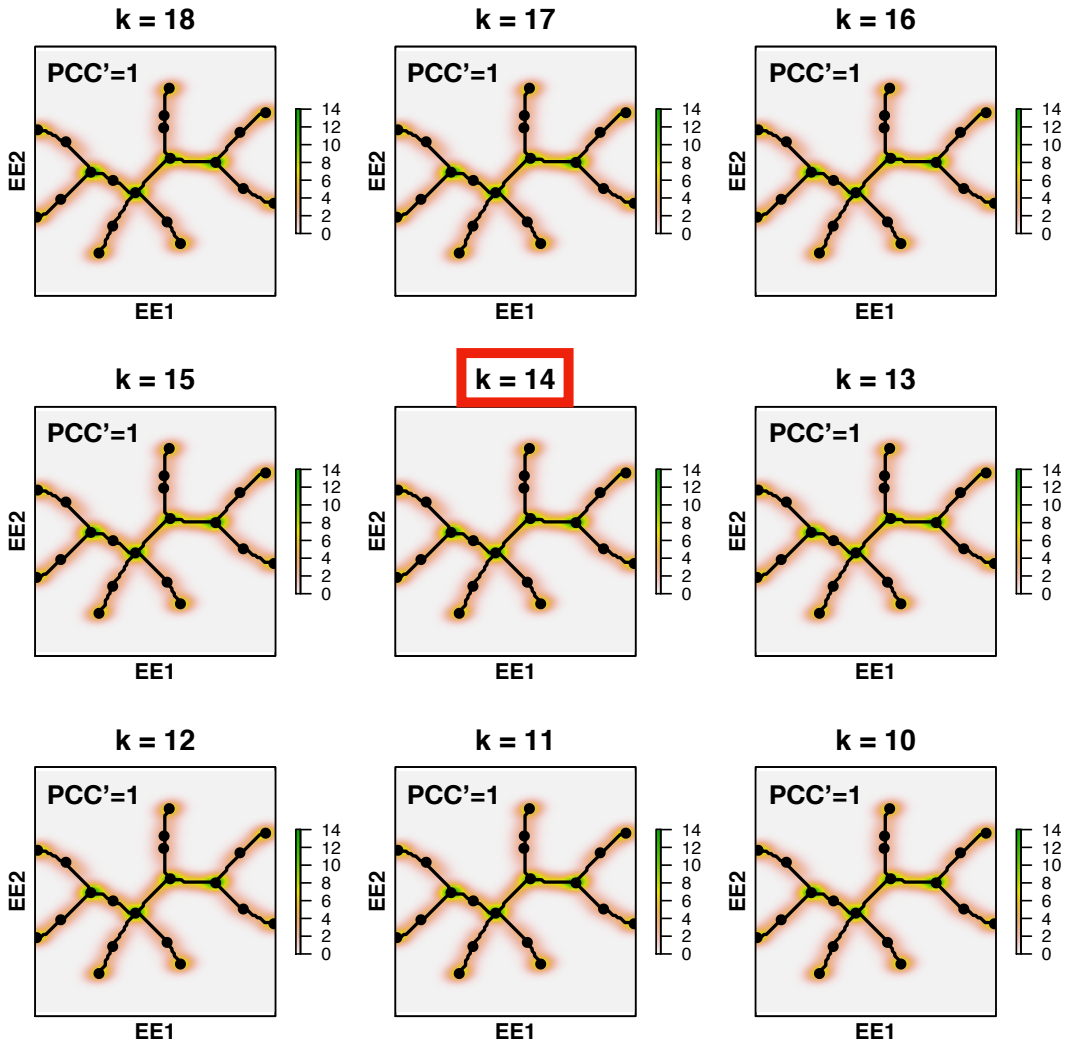

Fig. S22. The trajectories under different k on PHATE data. The value of k with red box in the center is the default choice of DensityPath. The PCC' in each plot refers to PCC of pseudotime calculated by DensityPath between different k values and the default k value of DensityPath.

Figure S23: The trajectories under different k on SLS3279 data.

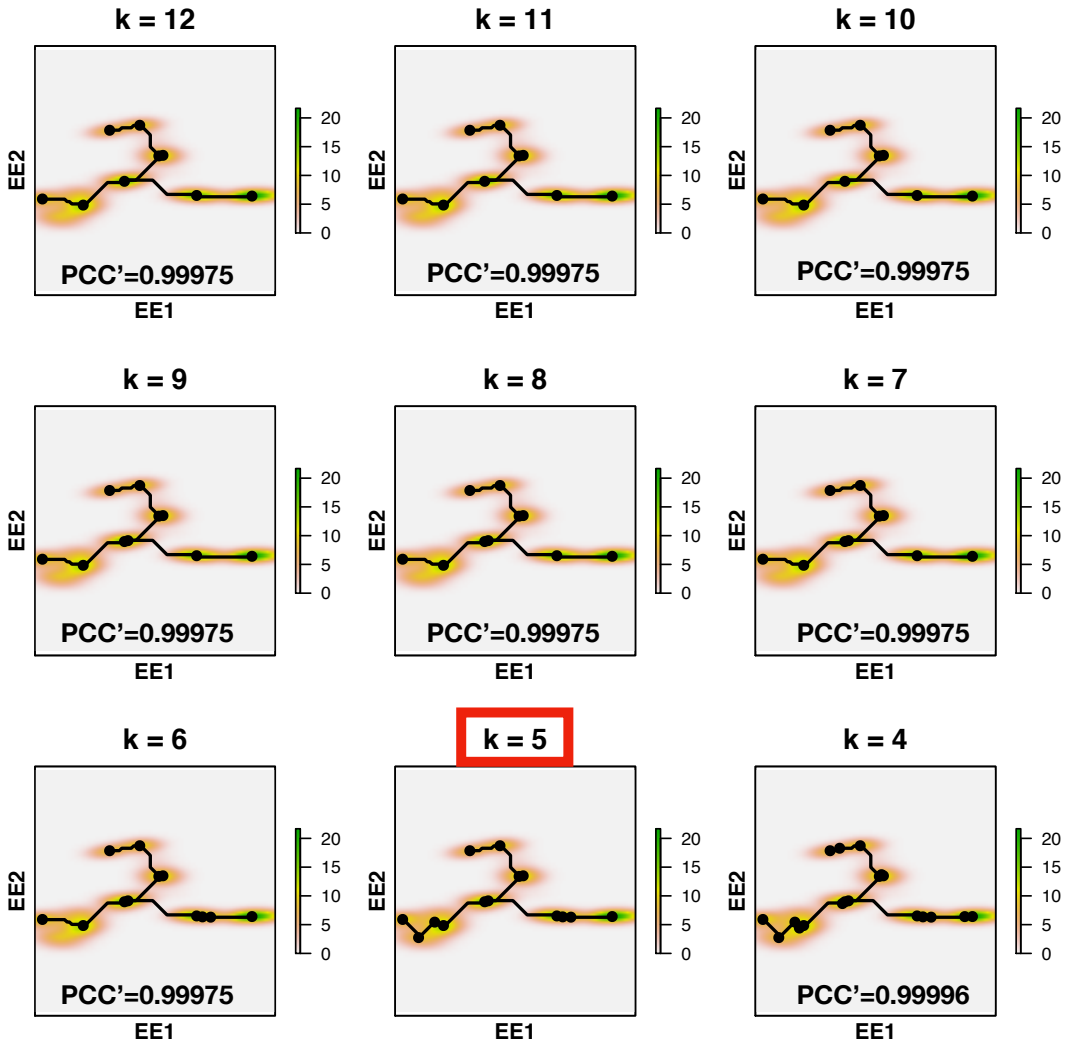

Fig. S23. The trajectories under different k on SLS3279 data. The value of k with red box is the default choice of DensityPath. The PCC' in each plot refers to PCC of pseudotime calculated by DensityPath between different k values and the default k value of DensityPath.

Figure S24: The robustness of DensityPath on the parameter k.

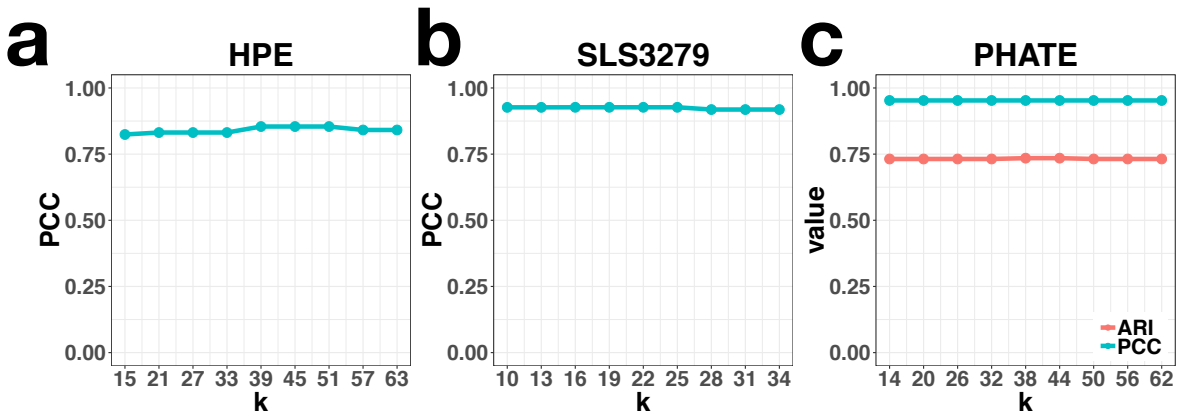

Fig. S24. The robustness of DensityPath on the parameter k. The robustness analysis on parameter k of LSC where **(a)** is the values of the PCC in HPE data, **(b)** is the PCC values in SLS3279 data and **(c)** is the results of PCC and ARI of PHATE data.

Figure S25: The robustness of PCC with different numbers of input informative genes on HSPCs and HPE data.

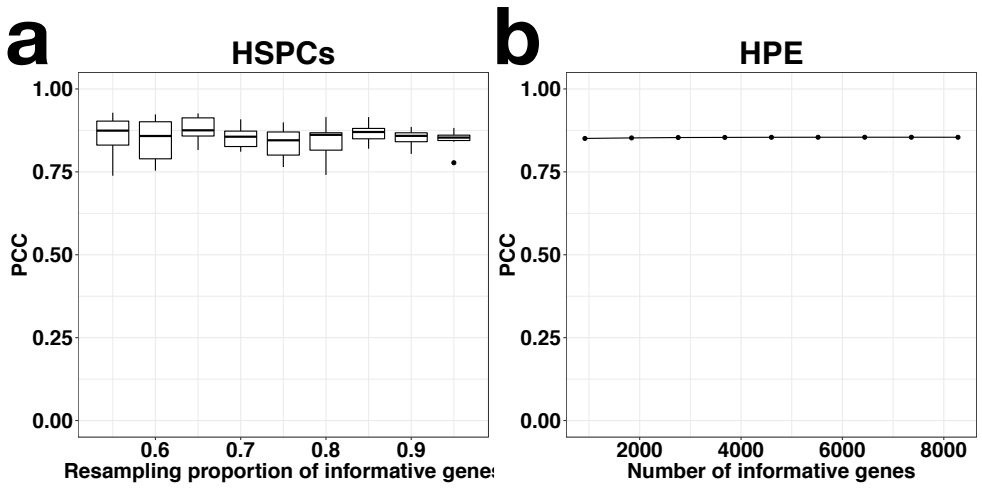

Fig. S25. The robustness of PCC with different numbers of input informative genes on HSPCs and HPE data. **(a)** Since the HSPCs data have been processed by Wishbone, selecting genes based on raw data is inappropriate. Hence, we sample genes from the processed data with the proportion varying from 55% to 95% and calculate the results of PCC on pseudotime on HSPCs data. **(b)** The results of PCC on pseudotime with varying number of informative genes from 920 to 8280 selected according to their variance across all cells, around the default number of genes 4600.

Figure S26: The trajectories under different numbers of input informative genes on HPE data.

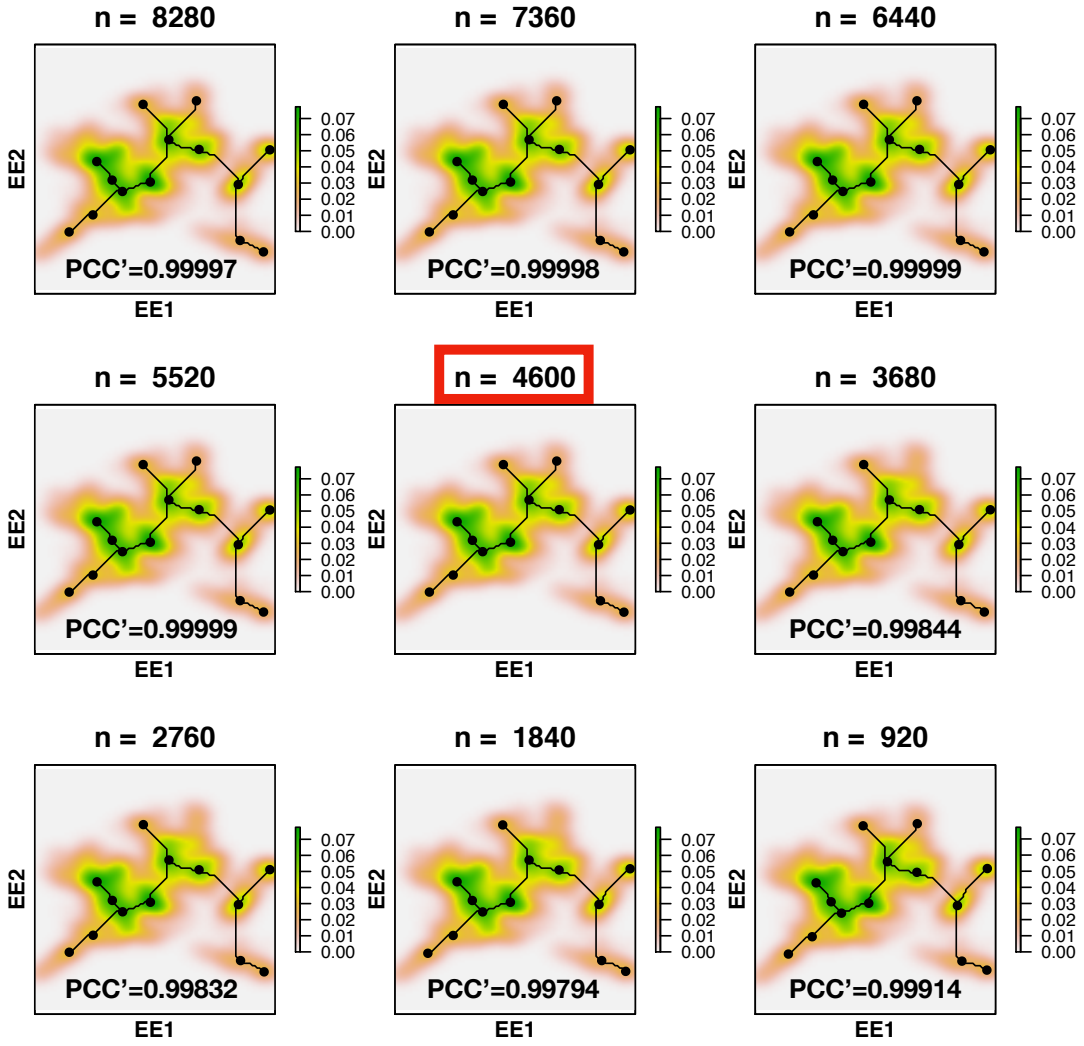

Fig. S26. The trajectories under different numbers of input informative genes on HPE data. The n represents the number of input informative genes. The value of n with red box in the center is the default result matching Fig. 2 in the manuscript. The PCC' in each plot refers to PCC of pseudotime calculated by DensityPath between different input informative genes and the default input informative genes.

Figure S27: The robustness analysis of DensityPath by subsampling cells.

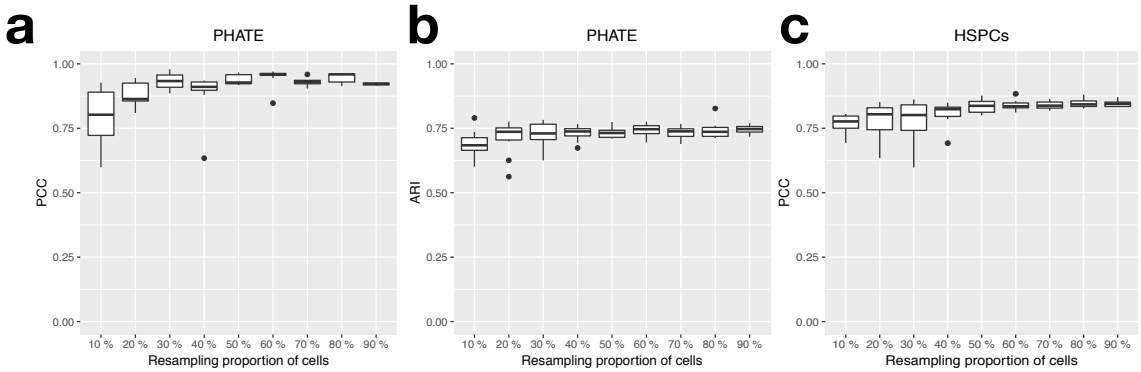

Fig. S27. The robustness analysis of DensityPath by subsampling cells. For each of the PHATE and HSPCs datasets, when fixing a proportion of $m\%$ (ranging from 10% to 90%), we subsample $m\%$ of the total cells without replacement as a new dataset, and then apply DensityPath with the complete procedures from A1 to A6 on the new dataset to reconstruct cell state-transition path, map cells onto the path, and cacualate the pseudotime. For a fixed proportion $m\%$, we repeat the subsampling 10 times to obtain 10 new datasets. **(a)** Boxplot of PCCs by subsampling the PHATE cells at different $m\%$. **(b)** Boxplot of ARIs by subsampling the PHATE cells at different $m\%$. **(c)** Boxplot of PCCs by subsampling the HSPCs cells at different $m\%$.

Figure S28: DensityPath result on HSPCs data after recovery by Saver.

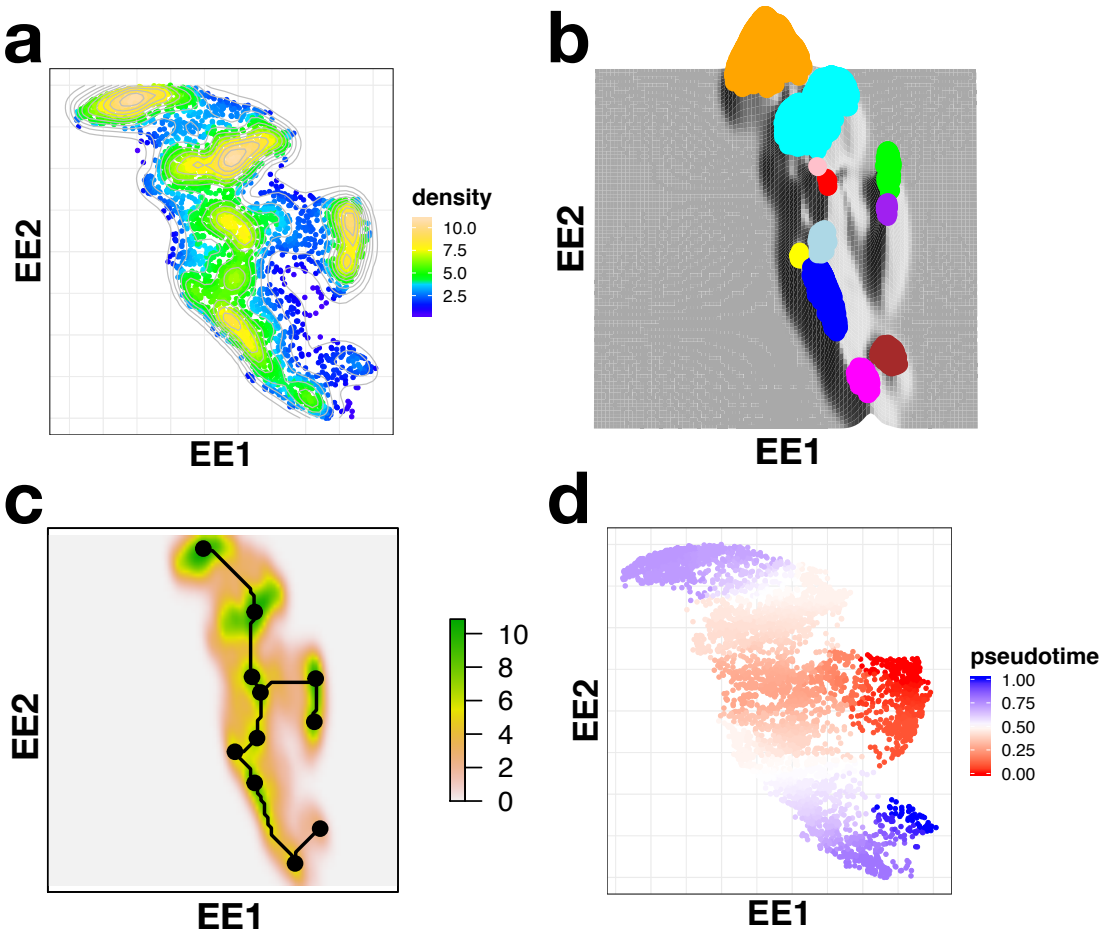

Fig. S28. DensityPath result on HSPCs data after recovery by Saver. **(a)** Density landscape. **(b)** RCSs on the density landscape. **(c)** Cell state-transition path on the density landscape. **(d)** Pseudotime of the cells.

Figure S29: The comparisons between level-set clustering (LSC) and mean-shift clustering on Paul, HSPCs, HPE, PHATE and SLS3279 data.

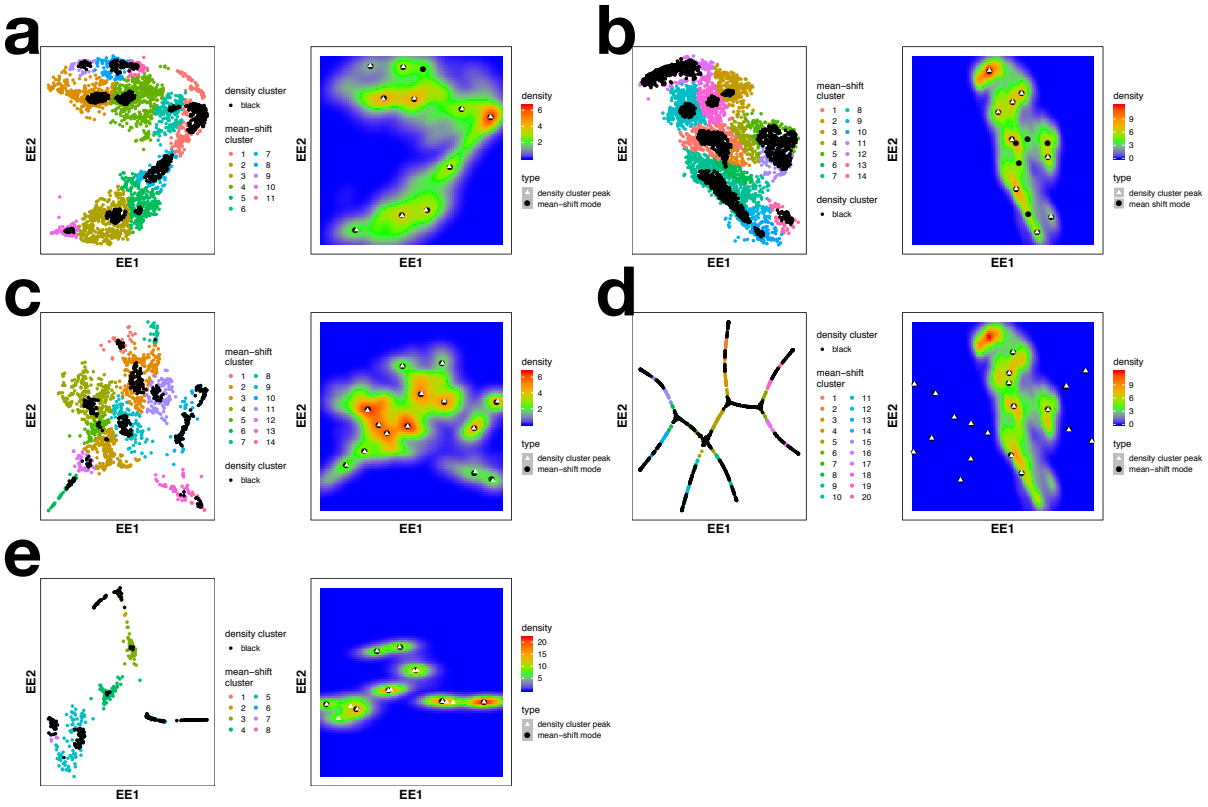

Fig. S29. The comparisons between level-set clustering (LSC) and mean-shift clustering. The parameter of LSC is set as the default of $k = \text{round}(n/100)$. The mean-shift algorithm works on the dimensionality reduced data by the step A1 of DensityPath, and is performed by the "kms" function in the R package "ks", using the same parameter H estimated as in the KDE of DensityPath in step A2. **(a)** Paul data, **(b)** HSPCs data, **(c)** HPE data, **(d)** PHATE data, and **(e)** SLS3279 data. For each dataset, the points with different colors are in different mean-shift clusters, and the black points are in level-set clusters in the left panel. The black points are modes of mean-shift clusters, and the white triangles represent the peak points of level-set clusters on the density heatmap in the right panel.

Figure S30: The validation for the accuracy of cell mapping in step A5 of DensityPath based on the computational cell fate determination results by mean-shift clustering on Paul, HSPCs, HPE, PHATE and SLS3279 data.

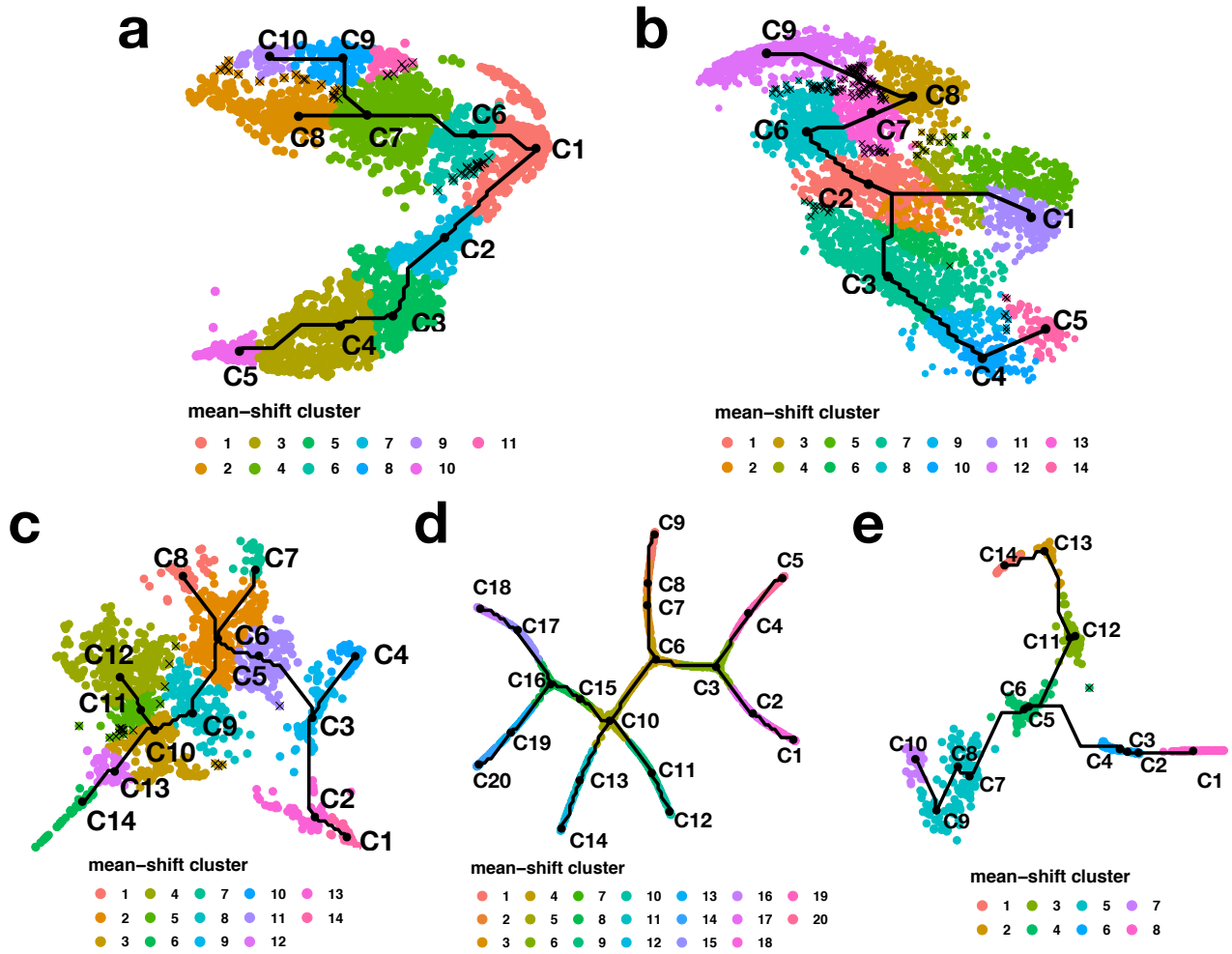

Fig. S30. The validation for the accuracy of cell mapping in step A5 of DensityPath based on the computational cell fate determination results by mean-shift clustering on Paul, HSPCs, HPE, PHATE and SLS3279 data. The mean-shift algorithm works on the dimensionality reduced data by the step A1 of DensityPath, and is performed by the "kms" function in the R package "ks", using the same parameter H estimated as in the KDE of DensityPath in step A2. (Continued in the next page)

The scatter points with different colors represent different mean-shift clusters. The black line represents the trajectory reconstructed by DensityPath, and black points represent the peak points of density clusters. The points with cross dots are mapped wrongly by DensityPath (see the **Discussion** of main text). **(a)** Paul data have 11 mean-shift clusters and 10 level-set clusters. Except for C9, which corresponds to MS 8 and MS 11, each of the other mean-shift clusters contains one level-set cluster. We merge MS 8 and MS 11 as a computational cell fate and then validate the mapping. Paul data have 44 points with a cross. **(b)** HSPCs data have 14 mean-shift clusters and 9 level-set clusters. In detail, C1 corresponding to MS 5 and MS 11, C2 corresponding to MS 1, MS 2 and MS 6, C3 corresponding to MS 7 and MS 9; there is a one-to-one correspondence between the remaining mean-shift clusters and the remaining level-set clusters. In order to match all the level-set clusters and all the mean-shift clusters in one-to-one correspondence, we merge MS 5, 11 and MS 1, 2, 6 and MS 7, 9 as a computational cell fate for validation of mapping, respectively. In addition, MS 4 does not intersect with any level-set cluster, but the local trajectory C1-C2 passed MS 4; therefore, points in MS 4, which are mapped to C1-C2, are considered to be mapped correctly. The result is 142 points with a cross on HSPCs data. **(c)** On HPE data, each mean-shift cluster contains one level-set cluster. After the validating the mapping, 20 mapping errors were found. **(d)** On PHATE data, each mean-shift cluster contains one level-set cluster, and all points are mapped correctly. **(e)** On SLS3279 data, a mean-shift cluster would contain more than one level-set cluster, and each peak point has a unique computational cell fate label. After validating the mapping, the SLS3279 data have one point with a cross.

Figure S31: Pairwise plots of the EE1, EE2, and EE3 coordinates by EE on Paul, HSPCs and HPE data.

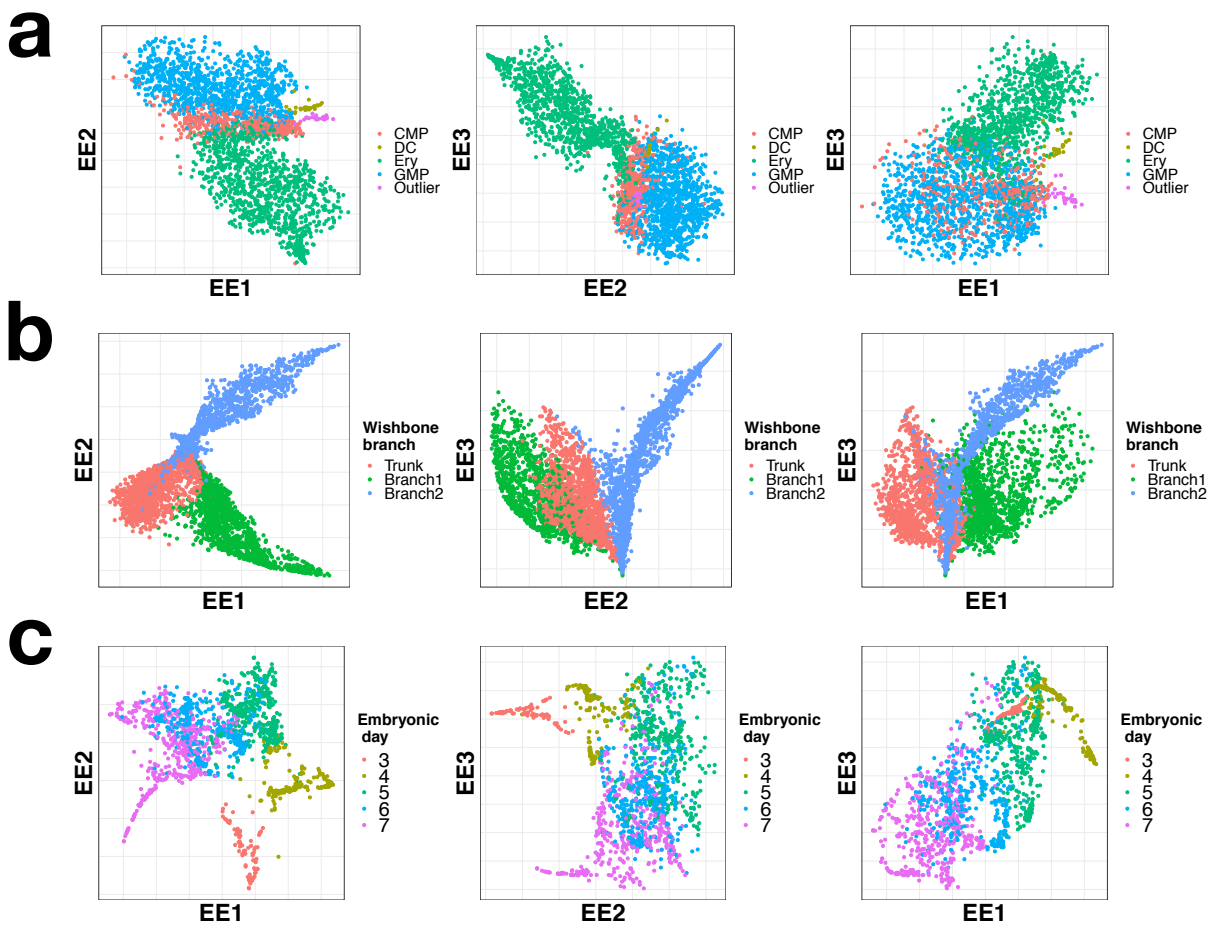

Fig. S31. Pairwise plots of the EE1, EE2, and EE3 coordinates by EE on Paul, HSPCs and HPE data. **(a)** Paul data (cells are annotated by clusters CMP, Ery, DC, GMP and Outlier of Paul *et al.*, 2015). **(b)** HSPCs data (cells are annotated by Wishbone's branch assignments). **(c)** HPE data (cells are annotated by the embryonic data).

Figure S32: DensityPath recovers the refined local structures on subset of cells of Paul data.

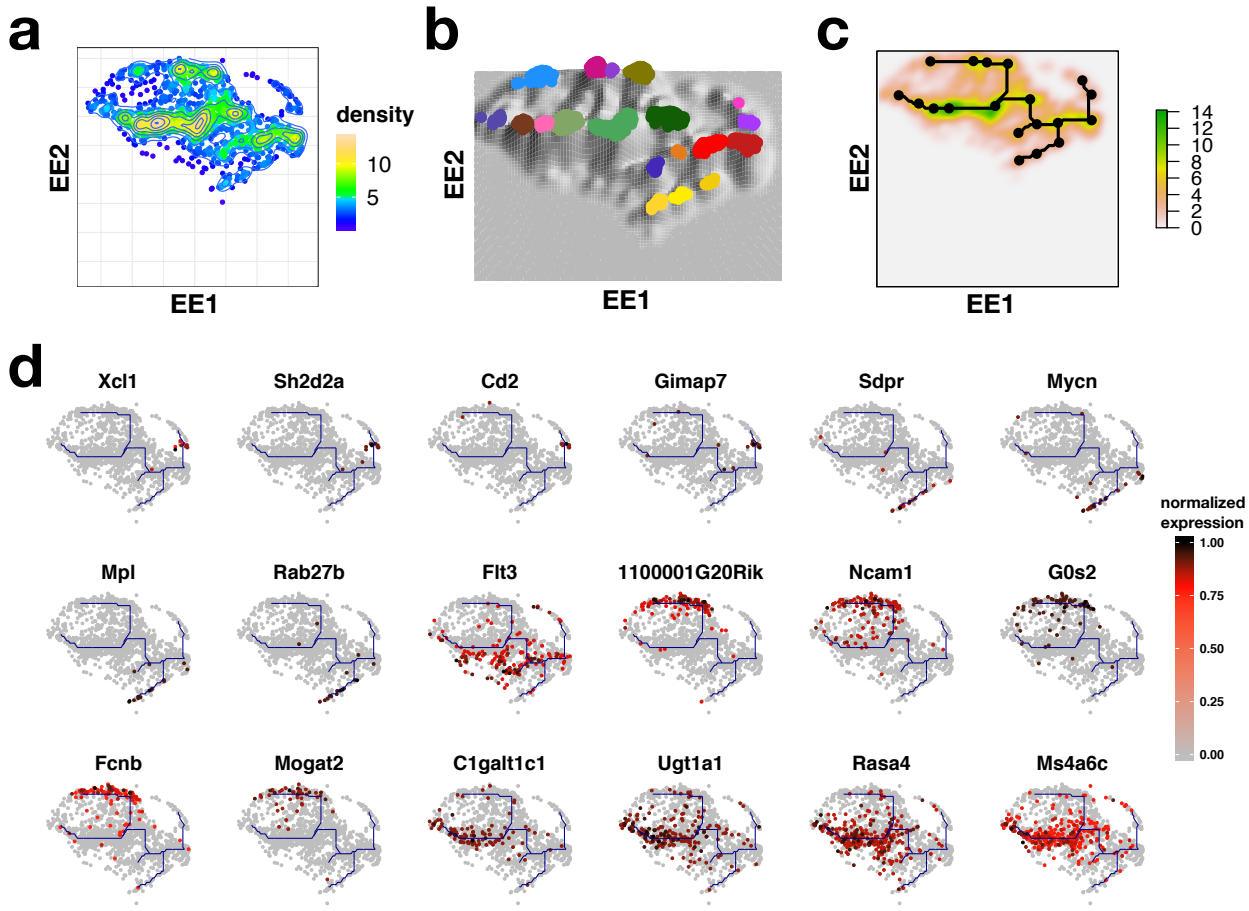

Fig. S32. DensityPath recovers the refined local structures on subset of cells of Paul data. To recover the elaborate local structures, we extract the 1407 cells that are mapped onto the major branch C1-C6-C7-C8(C9-C10) of Fig. S2(b) based on step A5, out of the 2730 total cells. These cells are mainly constituted by GMP and CMP cells. We conduct steps A2-A4 of DensityPath on this subset by setting the bandwidth of KDE as $\frac{1}{5}H$, where H is estimated by the plug-in method based on full samples which is used in Fig. S2. **(a)** DensityPath estimates the refined density landscape of subset of single cells on a 2-d plane of EE. **(b)** DensityPath extracts 19 separate RCSs on the refined density landscape. **(c)** DensityPath reconstructs the cell state-transition subpath on the refined density landscape by connecting the peaks of the RCSs. The right-most cell is selected as start cell. **(d)** The genes show the branch-specific expression patterns on the 2-d space of EE.

Figure S33: Comparison between DensityPath and Monocle2 on recovering the multi-scaled structure of Paul data.

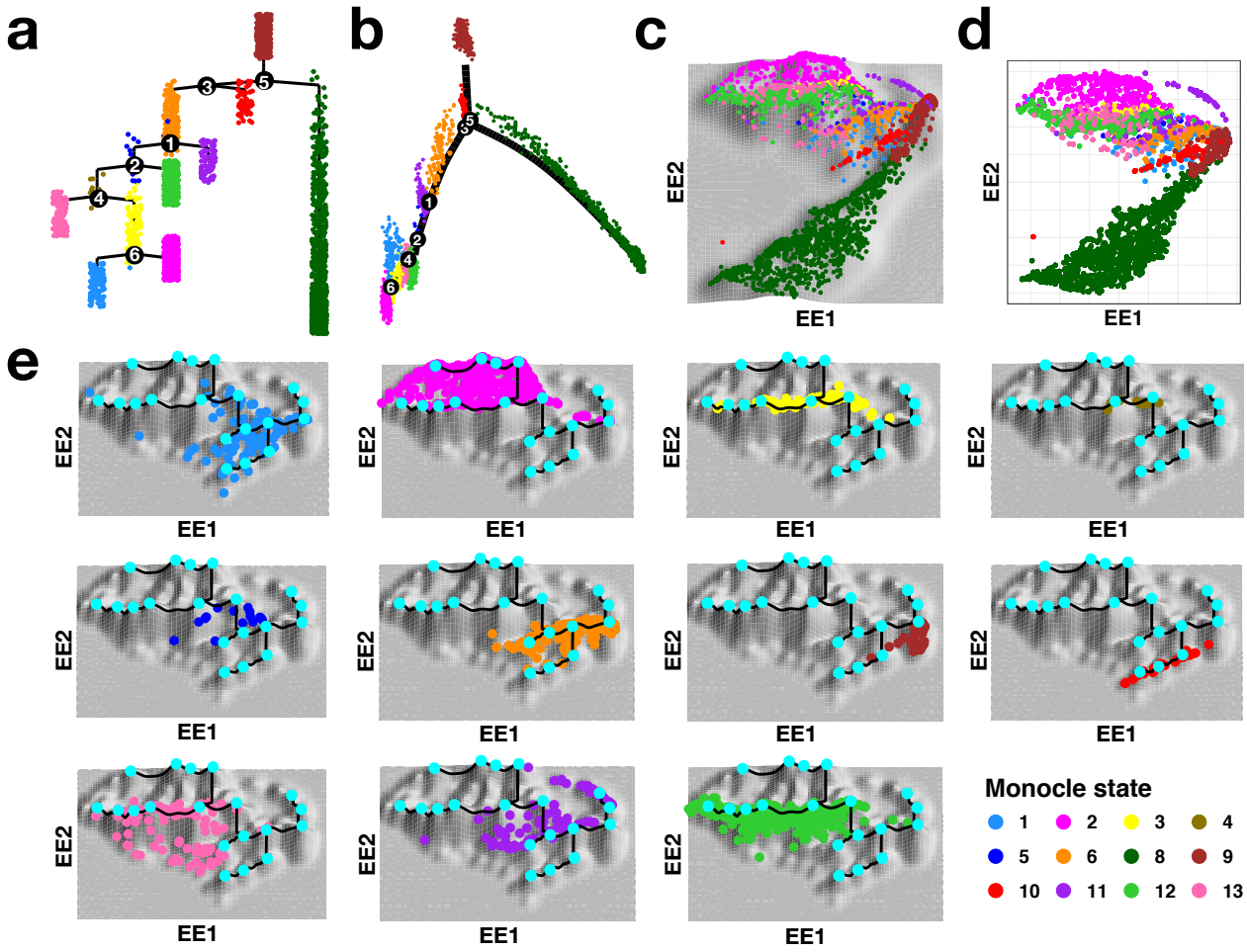

Fig. S33. Comparison between DensityPath and Monocle2 on recovering the multi-scaled structure of Paul data. **(a)** Monocle2 reconstructs the pseudo trajectory with 6 bifurcating events connecting by 12 cell states using full samples of Paul data. The branching points are labeled from 1 to 6. The cell is colored with the different states identified by Monocle2. **(b)** Cells are projected into a 2D space by Monocle2 using DDRTree. Black segments indicate cells connected in a minimum spanning tree reconstructed by Monocle2. The branch points labeled from 1 to 6 are corresponding to **(a)**. **(c)** The distribution of cell states classified by Monocle2 is displayed on density landscape of DensityPath. **(d)** The scatter plot of cell states classified by Monocle2 on the 2-d EE space. **(e)** The states classified by Monocle2 is displayed on the refined density landscape by DensityPath. The cell state-transition subpath is reconstructed by DensityPath, corresponding to Fig. S32(c).

Supplementary Tables

Table. S1: Running times of the DensityPath algorithm on 5 datasets.

Table S1: Running times of DensityPath algorithm on 5 datasets. The total running times of DensityPath, as well as the corresponding running times for the steps of PCA (A1), EE (A1), reconstruction (including density estimation in A2, LSC in A3 and construction of the cell state-transition path in A4), and mapping and pseudotime calculations in A5 and A6 are also shown. The computation is performed on a MacBook Pro laptop with 2.9GHz Processor and 16 GB DDR3 memory.

	Running Times in 5 Datasets (Time Unit: Second)				
DensityPath	Paul	HSPCs	PHATE	HPE	SLS3279
PCA in A1	76.3	32.1	0.1	148.9	0.1
EE in A1	18.6	21.4	5.3	3.2	0.4
A2-A4	11.1	19.6	6.4	6.7	4.4
A5-A6	106.4	101.3	42.8	55.1	20.4
Total Time	212.4	174.4	54.6	213.9	25.3

Table. S2: Pseudo code of DensityPath algorithm.

Table S2: Pseudocode of DensityPath Algorithm.

DensityPath Algorithm

Input: $Y_{n\times D} = \{y_1, \cdots, y_n\}$, single cell gene expression profile with N cells and G genes;

s, the index of the start cell.

Output: **RCSs**, the representative cell states;

pseudotime, the pseudotime of single cell;

pseudo branch, the indexes of the cells in each segment of trajectory, and the segment is defined as the part of trajectory starting from one branching point or the peak point in the RCSs where start cell is mapped, to another directly connected branching point or the point at the end;

MSTtree, the topological structure of RCSs;

allpath, the index of RCSs and the 2-d coordinates of their peak points in each single path starting from rom the RCS where start cell is mapped, to the end of RCS;

pseudo order, the pseudo order of single cell in each single path as defined in allpath.

(1) Normalize gene expression data $Y = (Y - \min Y)/(\max Y - \min Y)$);

(2) Reduce dimension using PCA and EE, obtaining the 2-d coordinates of single cells Y;

(3) Calculate density of cells based on KDE;

(4) Build k-NN network with $k = \text{round}(n/100)$, and extract RCSs by LSC;

(5) Construct grid covering cells on 2-d plane, calculate density on grid points, and construct MST of RCSs as cell state-transition path based on geodesics on density surface. The MST is saved as the $MSTtree$ representing the topological structure of trajectory;

(6) Map each cell to the point, which has the smallest geodesics on path;

(7) Calculate the geodesic distance of each mapping point to the mapping point of start cell, normalize the distance ranging from 0 to 1, and define the distance as pseudotime of each point;

(8) Divide the trajectory into segments and count the points which are mapped onto the segments of trajectory as pseudo branch;

(9) Record the path starting from RCSs where start cell is mapped, to the end RCSs, as well as the 2-d coordinates saved as allpath;

(10) Calculate the pseudo order of cells in each path defined as in allpath, saved as pseudo order.

Supplementary References

Guo, J. and Zheng, J. (2017). HopLand: single-cell pseudotime recovery using continuous Hopfield network-based modeling of Waddington's epigenetic landscape. *Bioinformatics*, **33**(14), I102–I109.

Haghverdi, L., Buettner, M., Wolf, F. A., Buettner, F., and Theis, F. J. (2016). Diffusion pseudotime robustly reconstructs lineage branching. *Nat. Methods*, **13**(10), 845–848.

Huang, M., Wang, J., Torre, E., Dueck, H., Shaffer, S., Bonasio, R., Murray, J. I., Raj, A., Li, M., and Zhang, N. R. (2018). SAVER: gene expression recovery for single-cell RNA sequencing. *Nature Methods*, **15**(7), 539+.

Kaneko, H., Shimizu, R., and Yamamoto, M. (2010). GATA factor switching during erythroid differentiation. *Curr. Opin. Hematol.*, **17**(3), 163–168.

MacLean, A. L., Hong, T., and Nie, Q. (2018). Exploring intermediate cell states through the lens of single cells. *Current Opinion in Systems Biology*, **9**, 32 – 41.

Moon, K. R., van Dijk, D., Wang, Z., Burkhardt, D., Chen, W., van den Elzen, A., Hirn, M. J., Coifman, R. R., Ivanova, N. B., Wolf, G., and Krishnaswamy, S. (2017). Visualizing transitions and structure for high dimensional data exploration. *bioRxiv*.

Paul, F., Arkin, Y., Giladi, A., Jaitin, D. A., Kenigsberg, E., Keren-Shaul, H., Winter, D., Lara-Astiaso, D., Gury, M., Weiner, A., David, E., Cohen, N., Lauridsen, F. K. B., Haas, S., Schlitzer, A., Mildner, A., Ginhoux, F., Jung, S., Trumpp, A., Porse, B. T., Tanay, A., and Amit, I. (2015). Transcriptional Heterogeneity and Lineage Commitment in Myeloid Progenitors. *Cell*, **163**(7), 1663–1677.

Petropoulos, S., Edsgard, D., Reinius, B., Deng, Q., Panula, S. P., Codeluppi, S., Reyes, A. P., Linnarsson, S., Sandberg, R., and Lanner, F. (2016). Single-Cell RNA-Seq Reveals Lineage and X Chromosome Dynamics in Human Preimplantation Embryos. *Cell*, **165**(4), 1012–1026.

Qiu, X., Mao, Q., Tang, Y., Wang, L., Chawla, R., Pliner, H. A., and Trapnell, C. (2017). Reversed graph embedding resolves complex single-cell trajectories. *Nat. Methods*, **14**(10), 979–982.

Rand, W. M. (1971). Objective criteria for the evaluation of clustering methods. *J. Am. Stat. Assoc.*, **66**(336), 846–850.

Rizvi, A. H., Camara, P. G., Kandror, E. K., Roberts, T. J., Schieren, I., Maniatis, T., and Rabadan, R. (2017). Single-cell topological RNA-seq analysis reveals insights into cellular differentiation and development. *Nat. Biotechnol.*, **35**(6), 551–560.

Setty, M., Tadmor, M. D., Reich-Zeliger, S., Ange, O., Salame, T. M., Kathail, P., Choi, K., Bendall, S., Friedman, N., and Pe'er, D. (2016). Wishbone identifies bifurcating developmental trajectories from single-cell data. *Nat. Biotechnol.*, **34**(6), 637–645.

van Etten, J. (2017). R package gdistance: Distances and routes on geographical grids. *J. Stat. Softw.*, **76**(13), 1–21.

Zwiessele, M. and Lawrence, N. D. (2017). Topslam: Waddington landscape recovery for single cell experiments. *bioRxiv*.