



Article

<https://doi.org/10.1038/s41587-023-01728-5>

A relay velocity model infers cell-dependent RNA velocity

Received: 1 August 2022

Accepted: 28 February 2023

Published online: 03 April 2023

Check for updates

Shengyu Li^{1,2,3,4,7}, Pengzhi Zhang^{1,2,3,4,7}, Weiqing Chen^{①,5}, Lingqun Ye^{1,2,3}, Kristopher W. Brannan^{2,3,4}, Nhat-Tu Le^{2,4}, Jun-ichi Abe^{②,6}, John P. Cooke² & Guangyu Wang^{①,2,3,4}✉

RNA velocity provides an approach for inferring cellular state transitions from single-cell RNA sequencing (scRNA-seq) data. Conventional RNA velocity models infer universal kinetics from all cells in an scRNA-seq experiment, resulting in unpredictable performance in experiments with multi-stage and/or multi-lineage transition of cell states where the assumption of the same kinetic rates for all cells no longer holds. Here we present cellDancer, a scalable deep neural network that locally infers velocity for each cell from its neighbors and then relays a series of local velocities to provide single-cell resolution inference of velocity kinetics. In the simulation benchmark, cellDancer shows robust performance in multiple kinetic regimes, high dropout ratio datasets and sparse datasets. We show that cellDancer overcomes the limitations of existing RNA velocity models in modeling erythroid maturation and hippocampus development. Moreover, cellDancer provides cell-specific predictions of transcription, splicing and degradation rates, which we identify as potential indicators of cell fate in the mouse pancreas.

A cell may transition to a new fate during or after development in response to transcriptional factors and epigenetic modifiers that are modulated by intracellular or external signaling^{1–5}. The advent of single-cell RNA sequencing (scRNA-seq) generated insights into cell subpopulations, detecting biological factors that influence cellular state shifts and deciphering cellular response to environmental and immune stimuli in health and disease at single-cell resolution^{6,7}. High-throughput scRNA-seq data provide an unbiased and high-resolution transcriptomic landscape of cellular states⁸. However, scRNA-seq captures only snapshots of a set of cells and does not explicitly demonstrate dynamical transitions between cellular states. Thus, trajectory inference algorithms were developed by constructing a potential branching trajectory based on the similarity in the transcriptomic profiles^{9–11}. A major challenge of trajectory inference is to

determine the direction of the trajectories or the root and terminal cellular states. One way of inferring such directed dynamics of cellular states is to incorporate ‘RNA velocity’¹². RNA velocity correlates the abundance of the nascent, unspliced mRNAs with that of the mature, spliced mRNAs using a simple first-order kinetics model. The progression of the current cellular state shifting toward a future state is extrapolated using the RNA velocities across genes. RNA velocity has brought biological insights to cell differentiation and disease progression^{13–16}.

RNA velocity was proposed to model the dynamic process of transcription, splicing and degradation of mRNA in a single cell. This model was initially applied to circadian-associated genes to extrapolate the progression of the circadian cycle (24 hours) on the bulk RNA-seq data of the mouse liver¹². Later, it was applied to infer the cell fates from scRNA-seq data, assuming that all cells in an scRNA-seq experiment

¹Center for Bioinformatics and Computational Biology, Houston Methodist Research Institute, Houston, TX, USA. ²Center for Cardiovascular Regeneration, Houston Methodist Research Institute, Houston, TX, USA. ³Center for RNA Therapeutics, Houston Methodist Research Institute, Houston, TX, USA. ⁴Department of Cardiothoracic Surgery, Weill Cornell Medicine, Cornell University, New York, NY, USA. ⁵Department of Physiology, Biophysics & Systems Biology, Weill Cornell Graduate School of Medical Science, Weill Cornell Medicine, Cornell University, Ithaca, NY, USA. ⁶Department of Cardiology, The University of Texas MD Anderson Cancer Center, Houston, TX, USA. ⁷These authors contributed equally: Shengyu Li, Pengzhi Zhang.

✉e-mail: gwang2@houstonmethodist.org

share similar kinetics^{12,17}. However, cellular state transitions often involve multiple stages and/or lineages, each of which may have dissimilar kinetics. The existing velocity models assume uniform kinetics of all cells in an scRNA-seq experiment, which may result in poor predictive performance when cell subpopulations have dissimilar RNA velocity kinetics. For example, a number of genes (for example, *Hba-x*) exhibit a boost in their transcription rates during mouse erythrocyte maturation, which have opposite predictions by scVelo¹⁸. It was also reported that there are five major branching lineages during the development of the mouse hippocampus¹³. The expression of some genes (for example, *Ntrk2*), termed branching genes, increase rapidly in several lineages and slowly in the other lineage. RNA velocities inferred by the existing models^{12,17} were inverted, in whole or in part, for the branching genes¹⁸. Thus, the estimation of RNA velocity kinetics is sensitive to heterogeneity in terms of biological conditions and cell populations.

Here we propose a ‘relay velocity model’ that uses the relay of a series of local velocities to provide single-cell resolution inference of velocity kinetics (Fig. 1a). Compared to other kinetic models, in the relay velocity model the cell-specific velocity of each cell is informed by its neighbor cells and then relays cell-specific velocities. To implement the relay velocity model, we developed cellDancer, which is a model-based deep neural network (DNN) framework. The cellDancer algorithm separately trains a DNN for each gene. For a gene, cellDancer assesses the spliced and unspliced mRNA velocities of each cell in a DNN to calculate the cell-specific transcription, splicing and degradation rates (α , β and γ) and to predict the future spliced and unspliced mRNA by the outputted α , β and γ using an RNA velocity model. The key step of cellDancer DNN is to define a loss function to train the DNN based on the similarity between the predicted future spliced and unspliced mRNA of each cell and the observation of its neighbor cells. After optimizing the global similarity between prediction and observation, cellDancer infers α , β and γ at a single-cell resolution rather than bulk rates used in existing methods^{12,17}.

We demonstrate that cellDancer extends the velocity estimation with cell-specific kinetics on heterogeneous cell populations, including those involved in erythroid maturation during gastrulation and those of the hippocampal dentate gyrus during neurogenesis. The cellDancer algorithm outperforms steady and early switching models on multi-stage and multi-lineage cell subpopulations. We show that cell-specific α , β and γ could be indicators of fate for cell identity in the mouse pancreas. cellDancer is available as a highly modularized, parallelized and scalable implementation.

Results

Learning cell-specific RNA kinetics by a relay velocity model

The cellDancer algorithm is a deep learning framework to generalize the estimation of RNA velocity in both homogeneous and heterogeneous cell populations from scRNA-seq data by estimating cell-dependent transcription (α), splicing (β) and degradation (γ) rates. Cell-specific α , β and γ were predicted by an RNA velocity model that incorporated the neighbor cells (see details regarding the selection of the neighbor cells in the Methods). Specifically, we resolved the RNA velocity kinetics by estimating the reaction rates from the weights and biases of the nodes in a DNN, which is a generalized framework of velocity estimation (see a demonstration in Supplementary Note 1). To train the cellDancer DNN, we first discretized the original reaction kinetics as follows:

$$\frac{u(t + \Delta t) - u(t)}{\Delta t} = \alpha(t) - \beta(t)u(t),$$

$$\frac{s(t + \Delta t) - s(t)}{\Delta t} = \beta(t)u(t) - \gamma(t)s(t),$$

where time t is discretized and Δt is a small time slot. In our model, α , β and γ are cell specific. For an individual gene in cell i , cellDancer used a DNN to predict cell-specific rates $\alpha(t_i)$, $\beta(t_i)$ and $\gamma(t_i)$ from the spliced and unspliced mRNA abundances $u(t_i)$ and $s(t_i)$ of genes at time t and

neighboring cells of i (Fig. 1b). Second, we extrapolated $s(t_i + \Delta t)$ and $u(t_i + \Delta t)$ of cell i at time $t + \Delta t$ to infer a velocity vector that points from the current state to the future in the gene phase portrait. We defined a loss function by summing every cell’s maximum cosine similarity for the predicted and observed velocity vectors (Methods). Finally, optimized rates of each cell were obtained by minimizing the loss function (Fig. 1b).

We initially evaluated the training progress of cellDancer on several well-studied genes in pancreatic endocrinogenesis and mouse hippocampus development¹⁷. We observed that cellDancer captured the transcriptional dynamics of these genes (Fig. 1c and Supplementary Fig. 1). Then, we scaled up the performance evaluation of cellDancer on 1,000 simulated mono-kinetic genes with the shared β , γ and two-step α values. The predicted parameters are highly correlated with the ground truth ($R^2 = 0.98$ for α/β and 0.93 for γ/β ; Extended Data Fig. 1a). Remarkably, cellDancer can identify two clusters of α values representing active (positive) and repressive expression phases (centered -0) on a benchmark dataset, without a prior constraint of a two-step transcription rate (Extended Data Fig. 1b).

Inferring RNA velocity in multi-rate kinetics

As cellDancer provides the single-cell resolution of α , β and γ , we next examined whether cellDancer could resolve the multi-rate kinetic regimes. We simulated three multiple kinetic regimes, including transcriptional boost, multi-lineage forward and multi-lineage backward genes (Extended Data Fig. 1c–e, right panels, and Methods). Transcriptional boost refers to a boost in the expression induced by a change in the transcription rate; multi-lineage forward and multi-lineage backward refer to induction and repression in separate lineages, respectively. We generated 2,000 cells and 1,000 genes for each regime. We compared cellDancer with scVelo (dynamic) and velocyto (static) algorithms and two deep learning algorithms, DeepVelo¹⁹ and VeloVAE²⁰. The error rates in cellDancer were significantly lower than those in scVelo, velocyto, DeepVelo and VeloVAE in all three simulated regimes (Extended Data Fig. 1c–e; $P < 0.001$, one-sided Wilcoxon test). Specifically, cellDancer exhibited the lowest error rate for simulated transcriptional boost, multi-forward branching and multi-backward branching kinetics with 13%, 3% and 9% compared to velocyto, scVelo, DeepVelo and VeloVAE, respectively (Supplementary Table 1). To test the effect of imbalanced cell numbers in different lineages or stages, we downsampled the cells at the stage after transcriptional boosting (Extended Data Fig. 1c) and the cells in lineage 1 (Extended Data Fig. 1d,e). Results showed that cellDancer is not affected by the bias of cell distribution. Next, we estimated the required number of epochs to optimize cellDancer DNN. cellDancer converged at 25 epochs for mono-kinetic, multi-forward and multi-backward branching genes and 100 epochs for transcriptional boost genes (Extended Data Fig. 1f–i).

Delineating transcriptional boost on single-cell resolution

We compared cellDancer to the dynamical model of scVelo on the scRNA-seq experiment of mouse gastrulation erythropoiesis² (Extended Data Fig. 2a and Fig. 2a), in which transcriptional boost genes were reported¹³. The vector flow in a uniform manifold approximation and projection (UMAP) embedding of the transcriptome clearly suggests that cellDancer recaptures the progression of erythroid differentiation (Fig. 2a, top), whereas scVelo’s prediction was reversed¹⁸ (Fig. 2a, bottom).

Barile et al.¹⁸ identified 89 multiple rate kinetics (MURK) genes, such as *Smim1* and *Hba-x*, of which transcription rates boost in the middle of erythroid differentiation, and showed that the prediction of scVelo was severely affected by the boost of transcription, resulting in incorrect predicted directions. cellDancer predicted the correct changes of well-known MURK genes, such as *Smim1* and *Hba-x*, on the phase portraits (Fig. 2b), whereas scVelo, DeepVelo and VeloVAE had incorrect predictions. Moreover, cellDancer revealed the transcriptional boost by the cell-specific α (Fig. 2b). We next tested the overall

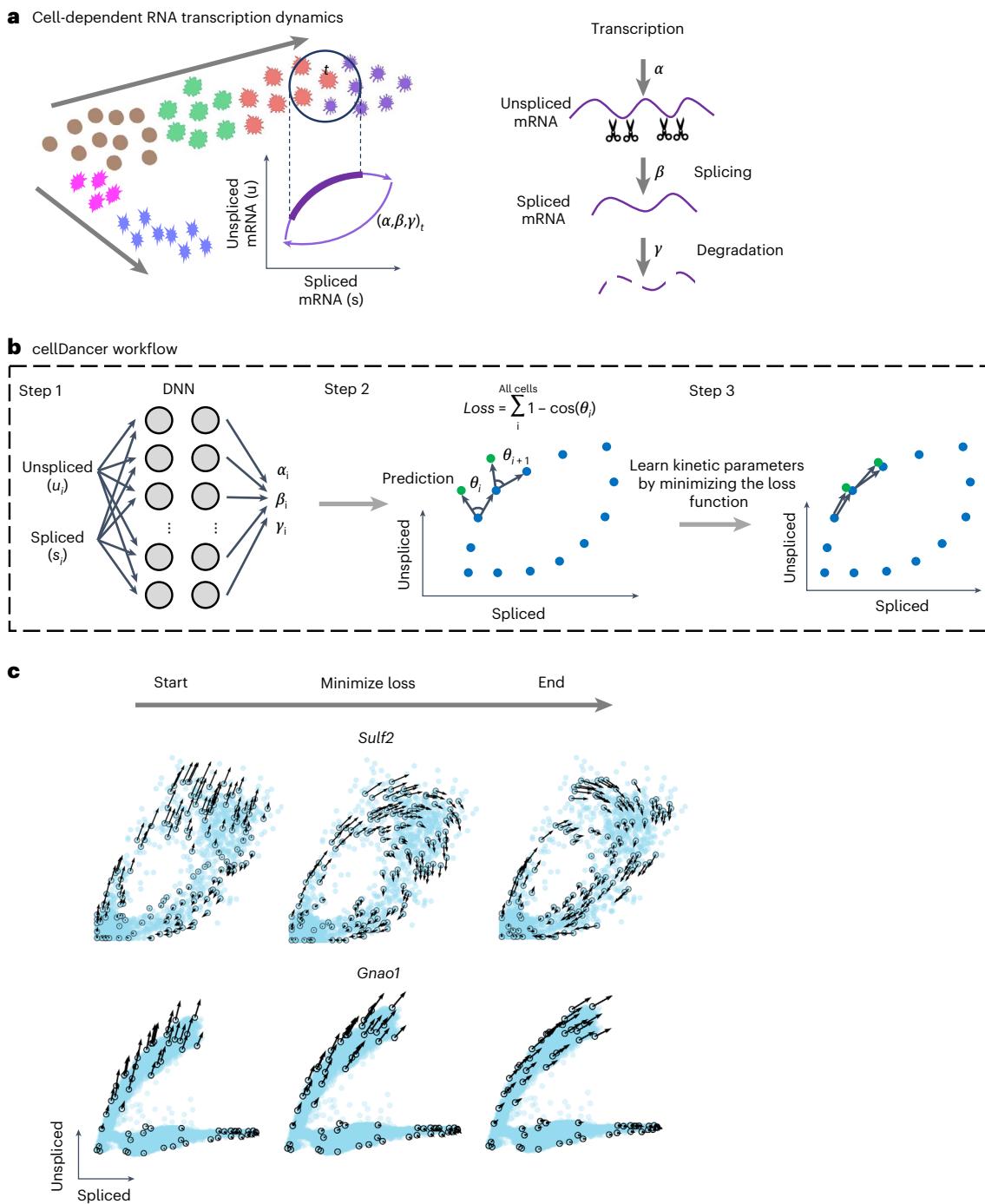


Fig. 1 | Predicting RNA velocity in localized cell populations via DNNs.

a, Transcription dynamics of the premature (unspliced) and mature (spliced) mRNAs are governed by the transcription (α), splicing (β) and degradation (γ) rates. Multi-kinetics genes involve multiple-lineage and/or multi-stage transitions of the cellular states; hence, cell-dependent rates (α, β, γ) are required to accurately capture the transcription dynamics of those genes. In the illustration, the (α, β, γ) for cell t are computed by locating the future state cell in the neighboring cells of t ('local environment'), assuming that the cells in the local environment share the same (α, β, γ) . **b**, cellDancer uses a DNN to predict cell-specific α, β and γ for each gene. The DNN consists of an input layer

with the spliced and unspliced mRNA abundances (u_i, s_i) $i = 1, 2, \dots, n_{\text{cells}}$, two fully connected hidden layers each with 100 nodes and an output layer yielding cell-specific α, β and γ . The loss function is defined as the sum of every cell's cosine similarity of predicted and observed velocity vectors. The DNN is iteratively optimized by minimizing the loss function. **c**, The progress of minimizing the loss function. RNA velocities for the examples of the mono-kinetic gene *Sulf2* in pancreatic endocrinogenesis, and the multi-lineage gene *Gnao1* in mouse hippocampus maturation is projected onto the phase portraits during the training process of their DNNs.

prediction of cellDancer on transcriptional boost genes. We applied cellDancer and scVelo to the 89 MURK genes and projected the velocity inference to the transcriptome UMAP. cellDancer recaptured the correct directional flow of differentiation using only MURK genes (Fig. 2c),

whereas scVelo, DeepVelo and VeloVAE predicted an opposed direction in multiple cell types (Extended Data Fig. 2b).

Next, we demonstrated cellDancer's capabilities of deciphering transcriptional changes along the differentiation pseudotime. We

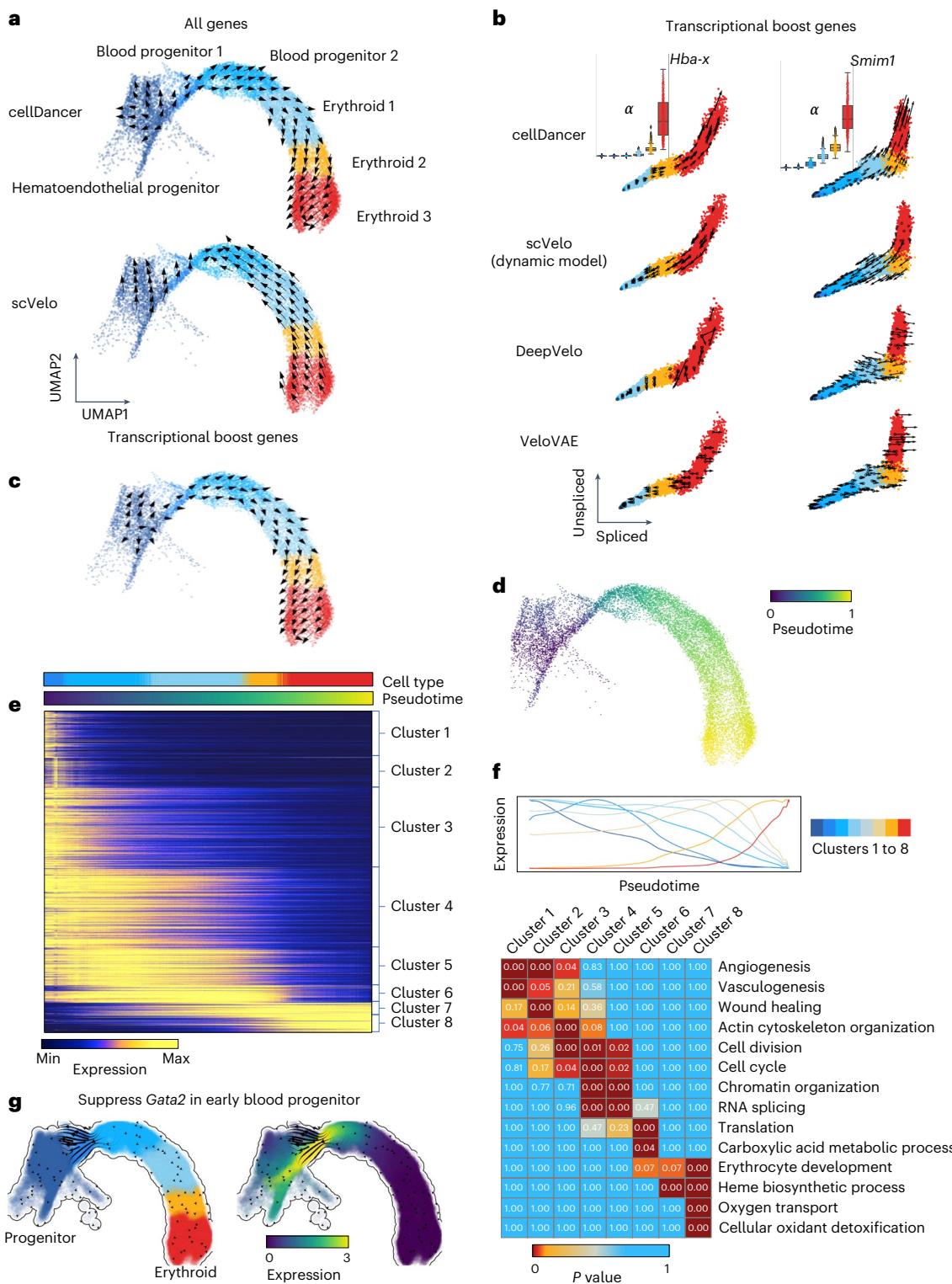


Fig. 2 | Delineating gastrulation erythroid maturation and resolving transcriptional boost. **a**, The velocities derived from cellDancer (top) are consistent with the erythroid differentiation but opposite in scVelo dynamic model (bottom) by using all genes. **b**, The velocities derived from cellDancer, scVelo dynamic model, DeepVelo and VeloVAE for the transcriptional boost genes (*Hba-x* and *Smim1*) are illustrated on the phase portraits. The cells are colored according to the cell types. The box plots of α for each cell type predicted by cellDancer are included to show the boost in the α rates in the course of erythroid maturation, especially in erythroid 3. **c**, The velocities derived from cellDancer for gastrulation erythroid maturation using transcriptional boost genes are projected on the UMAP of the original work, demonstrating that cellDancer can infer the correct cell differentiation direction by using only the

transcriptional boost genes. **d**, Gene-shared pseudotime on UMAP is consistent with the progression of gastrulation erythroid maturation. **e**, Genes that show high similarity in transcriptional changes along time are classified into eight clusters according to their transcriptional changes. The heat map describes the expression of the genes along time (rows: genes; columns: cells ordered according to the pseudotime). Genes were selected by Pearson correlation coefficient (R^2) > 0.8. **f**, Average expression of each cluster along the pseudotime (top) and the enriched pathways for each cluster of genes (bottom) (Benjamini-Hochberg procedure, one-sided, $P < 0.05$). P value indicates the significance of enrichment of a pathway in Fisher's exact test. **g**, In silico perturbation analysis by dynamo shows a critical role of *Gata2* in hematopoiesis.

first inferred major trajectories during cell differentiation from the transition matrix based on the correlation of velocities among neighbor cells (Methods). Then, we estimated a universal pseudotime from trajectories to capture the cell's position along with the erythroid maturation. The pseudotime of cellDancer accurately illustrated the transcriptional changes of genes (Extended Data Fig. 2c) and the terminal of erythroid maturation (Fig. 2d). To delineate the dynamics of transcriptional activity, we grouped genes into eight clusters based on the similarity in the transcriptional changes along pseudotime (Fig. 2e). The expression of genes in the first three clusters was high at the early stage in the hematoendothelial progenitor cells and diminished during differentiation. Gene expression in clusters 4–6 decreased slower than the gene expression in the first three clusters and decreased close to zero in the erythroid 3 subpopulation. Gene expression in clusters 7 and 8 increased during erythroid maturation. We next investigated the biological function of each gene cluster during erythroid cell differentiation. Gene Ontology (GO) analysis through DAVID²¹ showed that these genes are highly enriched in the angiogenesis and wound healing pathways. Genes in clusters 4–6 were enriched in basic cellular functions, including cell cycle, cell division, chromatin organization, RNA splicing and translation pathways. It is not surprising that these genes are enriched in erythrocyte development, heme biosynthetic process, oxygen transport and cellular oxidant detoxification pathways (Fig. 2f). Finally, we applied dynamo²² to *in silico* suppress the expression of *Gata2*, a critical regulator in hematopoiesis, in blood progenitor 1. We observed the diversions of hematopoietic fate after the perturbation (Fig. 2g), which is consistent with the experimental study²³.

Inferring RNA velocities on each branch for branching genes

We evaluated cellDancer using data from the branching lineages in mouse hippocampus development. There are five major branching lineages in the mouse hippocampus, corresponding to dentate gyrus granule neurons, pyramidal neurons in subiculum and CA1, pyramidal neurons in CA2/3/4, oligodendrocyte precursors (OPCs) and astrocytes¹². The cell velocity graph shows that cellDancer accurately inferred five major branching lineages in hippocampus development (Fig. 3a), confirming the reliable performance of cellDancer on multi-lineage populations.

We further studied the velocity inference of individual branching genes. As branching genes have different reaction rates among lineages, they have lineage-specific regulation of transcription, splicing and degradation and often play an important role in hippocampus development. For example, branching genes are vital to neurogenesis (*Diaph3*, *Klf7* and *Ncald*; Extended Data Fig. 3)^{24–26} and are involved in the differentiation of the neural system (*Cadm1* and *Gpm6b*)^{27,28}. Branching genes are also related to neurological or neuropsychiatric disorders. For instance, mutations of *Gnao1* may contribute to epilepsy, developmental delay and movement disorders in the neural system²⁹. Aberrant *Psd3* proteins are related to autism spectrum disorder and schizophrenia³⁰. We applied cellDancer to the branching genes. Phase portraits show that cellDancer can accurately infer the velocities of branching genes on each lineage (Fig. 3b and Extended Data Fig. 3), whereas scVelo, velocyto, DeepVelo and VeloVAE predicted the correct velocities on a limited number of cells (Fig. 3b and Supplementary Fig. 2). Moreover, cell-specific α , β and γ were inferred on each branch. For instance, neurotrophic tyrosine kinase receptor type 2 (*Ntrk2*)³¹ has two major branches: the upper branch corresponds to astrocytes and OPCs, and the lower branch corresponds to dentate gyrus granule neurons and pyramidal neurons (Fig. 3b). Astrocytes and OPCs have high α and low β , resulting in high expression of unspliced *Ntrk2* on the upper branch. Dentate gyrus granule neurons and pyramidal neurons have high β and low γ , resulting in high expression of spliced *Ntrk2* on the lower branch (Extended Data Fig. 3).

cellDancer calculates a minimized loss function after optimizing a DNN for each gene. A small loss score indicates a good fit with the

RNA velocity model. We ranked genes based on their loss function score. Top-ranking genes include both mono-kinetic and branching genes (Fig. 3c). Next, we performed GO pathway enrichment analysis through DAVID²¹ for the top 500 genes. The enriched pathways are associated with neurogenesis, nervous system development, neuron differentiation, synaptic signaling, chemical synaptic transmission and brain development (Fig. 3d).

We applied pseudotime analysis to infer the differentiation order of cells in hippocampus development. cellDancer automatically identified radial glia cells as a shared root state of hippocampus development (Fig. 3e), which is in good agreement with the previous study³². We also identified five terminal states without prior knowledge of the number of branches in the development process and applied dynamo to predict the most probable path of each terminal state (Fig. 3e). The pseudotime analysis of cellDancer suggests that astrocytes and OPCs are produced earlier than granule neurons and pyramidal neurons. Together, cellDancer has the capability to infer the global differentiation pseudotime of branching cell lineages.

We investigated the temporal progression of transcription during hippocampus development. We observed multiple expression patterns of individual genes on different branches. For instance, *Dcx* transiently upregulates in neuroblasts with consistently low expression in astrocytes (Fig. 3f), which is supported by previous studies that *Dcx* transiently expresses in the early neurogenesis stage and is a widely used marker for neurogenesis^{33,34}. By contrast, genes associated with neurogenesis, such as *Slc4a10* (ref. 35), *Ncald*²⁶ and *Ntrk2* (ref. 31), show increasing expression in all branches at different rates (Extended Data Fig. 4).

Vector fields analysis using cell-specific RNA velocity

cellDancer extends the bulk reaction rates (α , β and γ) to single-cell resolution in an scRNA-seq experiment. As gene expression is regulated by transcription, splicing and degradation, the reaction rates tend to be more stable than expression in a cell type during cell differentiation (Fig. 4a). Thus, we asked if the cell-dependent reaction rates in cellDancer provide biological insights into cell identity. We applied cellDancer to infer cell-dependent α , β and γ in the endocrine development of the mouse pancreas profiled from embryonic day 15.5 (E15.5)³⁶. Previous works reported four terminal cell types in endocrinogenesis, including glucagon-producing alpha-cells, insulin-producing beta-cells, somatostatin-producing delta-cells and ghrelin-producing epsilon-cells³⁷. UMAP of transcriptome shows that alpha-, beta-, delta- and epsilon-cells are distributed closely (Fig. 4b). Reaction parameters are always more consistent than transcriptomes in a cell type. For instance, expression of *Sulf2* increases in Ngn3-low endocrine progenitors and decreases in pre-endocrine (Fig. 4c), whereas α is a similar positive value in Ngn3-low endocrine progenitors and -0 in pre-endocrine. Next, we investigated the overall similarity of α , β and γ in each cell type. We applied UMAP to embed α , β and γ into two dimensions. Alpha-, beta-, delta- and epsilon-cells separate into distinct groups on UMAP of α , β and γ (Fig. 4d and Supplementary Fig. 3), suggesting that cell-specific α , β and γ are available as an indicator of cell identity. Notably, the cycling subpopulation of ductal cells and endocrine progenitors was separated from those without cycling (Fig. 4e).

Furthermore, we inputted the cell velocity to the established framework dynamo, which provides rich downstream analyses by learning differentiable velocity vector fields and inferring gene regulation networks. Noticeably, absorbing fixed points are identified in the alpha-, beta- and epsilon-cells, and an emitting fixed point is identified in the pancreas progenitor cells (Fig. 4f). To investigate the alpha-cell and beta-cell fate determination, we inspected the expression of *Arx* and *Pax4*, two well-known transcription factors that determine the endocrine cell fates (the alpha and beta lineages)³⁸. Consistent with the previous study³⁸, we observed exclusively high expression of *Arx* and *Pax4* in the alpha-cells and beta-cells, respectively (Fig. 4g).

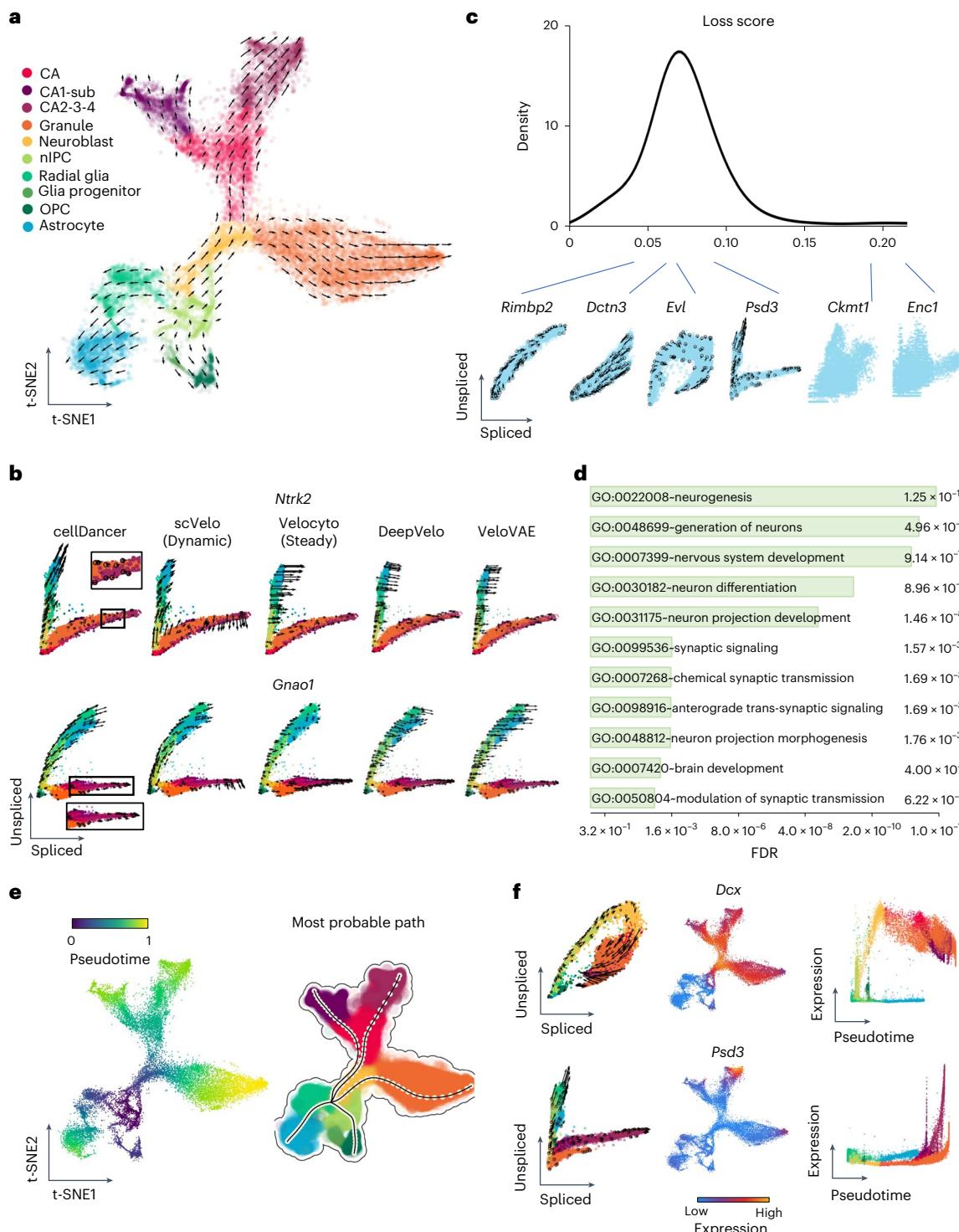


Fig. 3 | Identifying the branching lineage in the hippocampus development.

a, The velocities derived from cellDancer for the mouse hippocampus development dataset are visualized on the pre-defined t-SNE embedding. Directions of the projected cell velocities on t-SNE are in good agreement with the reported directions. **b**, The phase portraits of two branching genes (*Ntrk2* and *Gnao1*) predicted by cellDancer, scVelo dynamic mode, velocyo, DeepVelo and VeloVAE demonstrate the advantage of cellDancer in predicting the velocities of the branching genes. The RNA velocities of *Ntrk2* and *Gnao1* predicted by cellDancer are consistent with the expectation of hippocampus developmental progress, whereas the directions predicted by others are inconsistent in part. The cells are colored according to the cell types. **c**, Distribution of the minimized loss for all the genes. Those genes with low loss scores show mono-kinetic or divergent dynamics, whereas genes with high loss scores show pattern-less phase portraits.

d, The GO pathway enrichment analysis using adjusted *P* values of Fisher's exact test (Benjamini–Hochberg procedure, one-sided, $P < 0.05$) of DAVID for the 500 genes with the lowest training loss score shows that these genes are highly involved in pathways associated with nervous and brain development. **e**, Gene-shared pseudotime is projected on t-SNE by cellDancer, and the most probable paths are inferred by dynamo, showing the order of cell differentiation during hippocampus development. **f**, The phase portraits (left, cells colored according to **a**), the expression on t-SNE embedding (middle) and the expression pseudotime profiles (right) for the genes *Dcx* and *Psd3*. *Dcx* (top) and *Psd3* (bottom) have distinct dynamic behaviors. *Dcx* is a mono-kinetic gene (left), and its expression gradually increases in neuroblasts (right). *Psd3* is a branching gene (left), and its expression increases in each branching lineage at different speeds (right). FDR, false discovery rate; nIPC, neural intermediate progenitor cell.

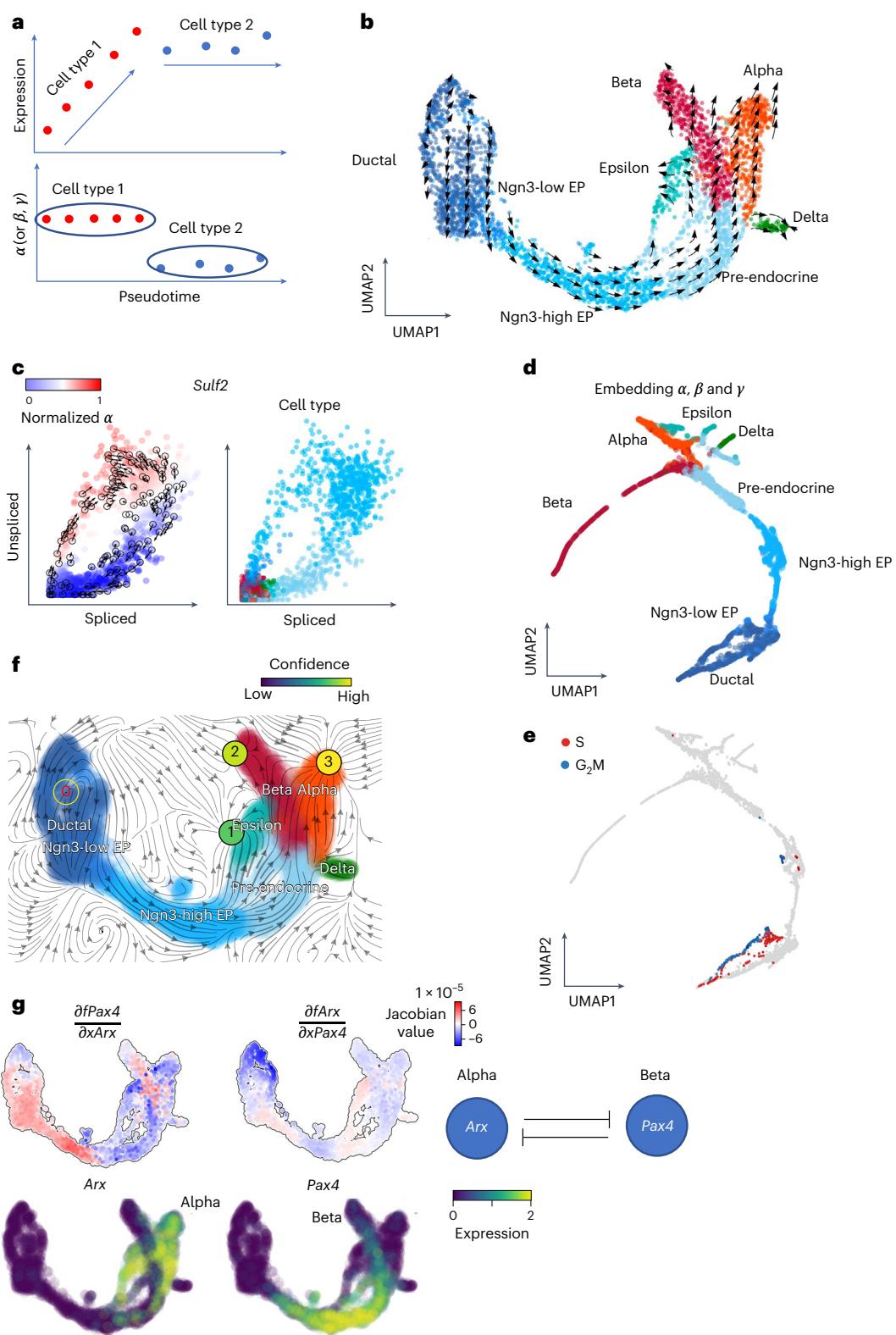


Fig. 4 | Deciphering cell identity with cell-specific reaction rates and analyzing gene regulation through vector fields. **a**, Schematic illustration shows that the α , β or γ rates of the genes may be a good indicator of the cell types rather than the expressions of the genes. **b**, The velocities derived from cellDancer for the pancreatic endocrinogenesis cells are visualized on the pre-defined UMAP embedding. **c**, Phase portraits of the gene *Sulf2*. The α rates of the *Sulf2* gene for each cell calculated by cellDancer clearly illustrate the gene's induction and regression phases (left). *Sulf2* is in induction in the Ngn3-high embryonic progenitor (EP) cell type and in regression in the pre-endocrine

cell type, whereas it is barely transcribed in other cell types (right). **d,e**, UMAP embedding using the cell-specific α , β and γ rates calculated by cellDancer indicates that our computed kinetics rates might be useful in assigning cell subpopulations (**d**) and cell identity (**e**). **f**, The velocity vector fields were learned by dynamo. The red digit 0 reflects the identified emitting fixed point. The black digits 1, 2 and 3 reflect the absorbing fixed points. **g**, Jacobian analysis and the gene expression of *Arx* and *Pax4* on the UMAP space. It shows that *Pax4* is downregulated by *Arx* in alpha-cells. *Arx* is downregulated by *Pax4* in beta-cells.

Then, we used dynamo to perform Jacobian analyses and detected mutual inhibition between *Arx* and *Pax4* in the alpha-cells and beta-cells. These analyses are in line with the experimental findings³⁹ and provide mechanistic insight from gene regulation at single-cell resolution, showing that cellDancer can be seamlessly integrated with downstream analysis, such as dynamo vector field analysis.

Revealing the turnover strategies of mRNA during cell cycle

A previous study showed that metabolic labeling technology, such as sequencing mRNA labeled with 5-ethynyl-uridine (EU) in single cells (scEU-seq), can measure the synthesis and degradation of mRNA using the sequencing method⁴⁰. Furthermore, Qiu et al.²² showed that scEU-seq can be used to predict the dynamics of the cell cycle. To investigate whether the predicted kinetic parameters are consistent with the experimental measurements, we used metabolic labeling data (that is, scEU-seq) of RPE1-FUCCI cells at specific points during cell cycle progression as a benchmark⁴⁰. We first clustered RPE1-FUCCI cells into eight groups based on cell cycle stages and calculated the average spliced and unspliced expression of cell-cycle-associated genes, which also have synthesis and degradation rates in scEU-seq (Extended Data Fig. 5a). We applied cellDancer to predict the velocities and kinetic parameters of cell cycle genes and compared the predicted α and γ to the experimentally derived synthesis and degradation rates measured by scEU-seq⁴⁰ (Extended Data Fig. 5b). Overall, the predicted α and γ are associated with the experimental measurements of mRNA synthesis and degradation (Extended Data Fig. 5b,c), especially in the highly expressed genes (Extended Data Fig. 5a). We also observed a difference between the predicted α and scEU-seq synthesis rates in the G1 state for the low-expression genes, of which expression starts to increase at the G1 state (Extended Data Fig. 5a). Our prediction captures this increase by a relatively large α in the G1 state, whereas scEU-seq shows a low synthesis rate, which may be due to the potential limitation of scEU-seq in the low-expression genes. Next, we predicted the velocity flow and pseudotime of the cell cycle procession using cell cycle genes. cellDancer predicts the direction of transcriptome shifting and the pseudotime during the cell cycle (Extended Data Fig. 5d). Together, the cellDancer-predicted kinetic parameters reflect the reality of mRNA turnover rates in cell cycle.

We further investigated the functions of genes with different kinetic patterns. We grouped genes into seven clusters according to dynamic patterns of α and γ (Extended Data Fig. 6a). We calculated the correlation of α and γ and the average expression in each cluster (Extended Data Fig. 6b). We identified three positively correlated groups and four negatively correlated groups, indicating different turnover strategies in the clusters. Next, we investigated the functions of genes in each cluster through DAVID²¹ (Extended Data Fig. 6c). Overall, all clusters are associated with cell cycle pathways, including cell division, proliferation, chromatin remodeling, DNA replication and cell cycle checkpoints. We noticed that the genes in cluster F have large transcription and degradation rates in the mitosis stage, indicating a fast turnover of mRNAs. The genes in cluster F are enriched in pathways related to cell communication, including signal transduction, enzyme-linked receptor protein signaling, TGF- β receptor signaling and stress-activated protein kinase signaling, suggesting a quick communication of cells during mitosis.

To investigate the capacity of cell-specific rates in identifying cell subpopulations, we recaptured that pseudotime is continuous in the gene expression space during the cell cycle. Specifically, the G2 phase (pseudotime 0.8–1) is in proximity to the M phase (pseudotime 0–0.2) (Extended Data Fig. 6d). Then, we clustered the cells into 17 subpopulations according to the cell-specific rates (Extended Data Fig. 6d) using SCANPY⁴¹ and used the hierarchical method to further cluster each subpopulation (Extended Data Fig. 6e). We found that these subpopulations were globally clustered together in good agreement with cell cycle pseudotime except clusters 3 and 4 (a cell subpopulation at the

M phase). The reaction rates of this cell subpopulation are more in line with clusters 1 and 2, which are at the G1 and S stages (Extended Data Fig. 6e). Next, we compared the gene expression and reaction rates of this intricate cell subpopulation with the other cells. We identified 116 differentially expressed genes and 181 genes having differential transcriptional rates by comparing this subpopulation to the rest and found that only 10% of genes having differential transcriptional rates were captured by the raw expression (Extended Data Fig. 6f). We further investigated the enriched pathways of these 163 genes that are uniquely identified by the rates through DAVID²¹. Those genes are enriched with cell division pathways, such as cytokinesis, cell division and mitotic metaphase congression (Extended Data Fig. 6g), suggesting that transcriptional regulation plays an important role in cell division at the M stage.

Decoding human embryonic glutamatergic neurogenesis

We further investigated RNA velocity on an scRNA-seq dataset of the developing human forebrain at 10 weeks after conception, which was used as a benchmark in previous studies^{12,42}. We used cellDancer to predict RNA velocity on human embryonic glutamatergic neurogenesis. The velocity on the embedding space and the derived pseudotime show that cellDancer accurately recaptures the cell fate of human embryonic glutamatergic neurogenesis (Extended Data Fig. 7a,b). The velocities of genes that are vital to neural development and neurogenesis, such as *ELAVL4* (ref. 43) and *DCX*^{33,34}, were also correctly predicted (Extended Data Fig. 7c).

To test whether cellDancer is sensitive to the methods of neighbor cell detection, we applied cellDancer to predict velocity vector flow based on the nearest neighbors defined by the spliced RNAs or by the spliced and unspliced RNAs. Results suggest that the prediction of velocities using spliced RNAs is consistent with the prediction using spliced and unspliced RNAs (Extended Data Fig. 7a).

cellDancer has a robust and efficient performance

The high proportion of zero reads is a key feature in scRNA-seq data, one cause of which is technical dropout. We tested whether cellDancer is robust with technical dropout (Extended Data Fig. 8a). cellDancer was able to correctly predict the gene dynamics even with high dropout ratios and learned RNA velocities in noisy scRNA-seq data (Extended Data Fig. 8b).

Next, we tested the robustness of our algorithm among different cell numbers. We gradually reduced the number of cells from 10,000 to 1,000 in the simulation dataset to predict RNA velocity and compared the prediction of α/β and α/γ . Results show that our model is robust in data with sparsity (Extended Data Fig. 8c).

We tested the sensitivity of the stopping criteria for the training of cellDancer DNN. Two key parameters, ‘checkpoint’ and ‘patience’, are associated with the stopping criteria. We performed the full cellDancer analysis in the mouse hippocampus development experiment using a different number of checkpoints and patience for training. cellDancer shows low sensitivity to the stopping criteria of training (Extended Data Fig. 9). Furthermore, cellDancer independently predicted an individual DNN for each gene, which allows us to apply the multi-processing approach to speed up the efficiency. Overall, cellDancer has an optimized runtime (Extended Data Fig. 10).

Discussion

In this study, we first showed that RNA velocity was automatically inferred from a neural network by optimizing a simple loss function based on local cosine similarity and implemented this deep learning algorithm to cellDancer, which is a flexible, robust and extensible framework for velocity inference. Our algorithm delivers four innovations. First, cellDancer overcomes the barriers for inferring RNA velocity with multiple kinetics, such as branching genes and transcriptional boost genes by local but not global velocity estimation. cellDancer also

largely improves the reaction rates inference from bulk to single-cell resolution and illuminates the regulation of transcription, splicing and degradation at a single-cell resolution.

Second, cellDancer can be adapted to other velocity ordinary differential equations (ODEs) using the same framework. cellDancer does not require an analytic solution for ODEs. Therefore, cellDancer can be conveniently extended from original velocity ODEs to other extended ODEs. For example, scVelo and another recent study, UniTVelo⁴⁴, proposed two stochastic models that considered the second-order moments of dynamics of the transcriptome to resolve cell-specific dynamics. To adapt to those velocity models, we could modify step 2 (computing predicted spliced/unspliced mRNA abundance) in the cellDancer workflow by using the velocity ODEs without changing other steps.

Third, cellDancer is highly modularized and extensible to multi-omics velocity models. As explained in the Methods, cellDancer is applicable to dynamics governed by first-order rate equations. More generally, in principle, cellDancer fits any dynamics following these rate kinetics:

$$\frac{d\mathbf{T}(t)}{dt} = f(\mathbf{T}(t), \mathbf{R}(t))$$

where $\mathbf{T}(t)$ is the abundance vector of mRNAs, proteins, etc.; $\mathbf{R}(t)$ is the reaction rates vector; and f is a function of $\mathbf{T}(t)$ and $\mathbf{R}(t)$ and does not explicitly contain time t . For instance, Gorin et al.⁴⁵ developed a protein velocity model by extending the RNA velocity model to cell surface protein translation. The protein velocity model has one more equation than the RNA velocity model to delineate the translation process. cellDancer can adapt to protein velocity by adding protein abundance into the input matrix and updating the module of loss function from RNA velocity to protein velocity. Moreover, chromatin accessibility measured by single-cell assay for transposase-accessible chromatin with sequencing (scATAC-seq)⁴⁶ can be likewise included in cellDancer to reinforce the estimation of the transcription rates.

Finally, cellDancer DNN is scalable. A small, fully connected DNN was used in cellDancer to boost the running speed. If the relationship between kinetic parameters and spliced/unspliced mRNA abundance is complex, or multi-omics data are included in the velocity model, the fully connected DNN can be replaced or extended by other DNNs, such as a long short-term memory (LSTM) network⁴⁷ or a convolutional neural network (CNN)⁴⁸. This feature allows us to customize an optimal network structure based on the complexity of the velocity model and experimental data. Furthermore, due to the limitation that scRNA-seq captures only spliced and unspliced mRNA abundances, it is unfeasible to infer the absolute magnitude of the RNA velocity and the underlying (α, β, γ) values using only scRNA-seq data. Additional time information introduced by experimental techniques, such as metabolic labeling or different timepoint datasets, could be incorporated to obtain such absolute kinetic rates. This functionality would be included in a future version of cellDancer.

Together, cellDancer represents a notable advance to quantitatively predict the time evolution of cellular transcriptomics, applicable to numerous biological models and disease processes at a genome-wide scale.

Online content

Any methods, additional references, Nature Portfolio 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/s41587-023-01728-5>.

References

- Qiu, C. et al. Systematic reconstruction of cellular trajectories across mouse embryogenesis. *Nat. Genet.* **54**, 328–341 (2022).
- Pijuan-Sala, B. et al. A single-cell molecular map of mouse gastrulation and early organogenesis. *Nature* **566**, 490–495 (2019).
- Cao, J. et al. The single-cell transcriptional landscape of mammalian organogenesis. *Nature* **566**, 496–502 (2019).
- Cheng, S. et al. Single-cell RNA-seq reveals cellular heterogeneity of pluripotency transition and X chromosome dynamics during early mouse development. *Cell Rep.* **26**, 2593–2607 (2019).
- Mohammed, H. et al. Single-cell landscape of transcriptional heterogeneity and cell fate decisions during mouse early gastrulation. *Cell Rep.* **20**, 1215–1228 (2017).
- Kharchenko, P. V. The triumphs and limitations of computational methods for scRNA-seq. *Nat. Methods* **18**, 723–732 (2021).
- Stephenson, E. et al. Single-cell multi-omics analysis of the immune response in COVID-19. *Nat. Med.* **27**, 904–916 (2021).
- Kolodziejczyk, A. A., Kim, J. K., Svensson, V., Marioni, J. C. & Teichmann, S. A. The technology and biology of single-cell RNA sequencing. *Mol. Cell* **58**, 610–620 (2015).
- Setty, M. et al. Characterization of cell fate probabilities in single-cell data with Palantir. *Nat. Biotechnol.* **37**, 451–460 (2019).
- Bendall, S. C. et al. Single-cell trajectory detection uncovers progression and regulatory coordination in human B cell development. *Cell* **157**, 714–725 (2014).
- Haghverdi, L., Buttner, M., Wolf, F. A., Buettner, F. & Theis, F. J. Diffusion pseudotime robustly reconstructs lineage branching. *Nat. Methods* **13**, 845–848 (2016).
- La Manno, G. et al. RNA velocity of single cells. *Nature* **560**, 494–498 (2018).
- Bergen, V., Soldatov, R. A., Kharchenko, P. V. & Theis, F. J. RNA velocity-current challenges and future perspectives. *Mol. Syst. Biol.* **17**, e10282 (2021).
- Couturier, C. P. et al. Single-cell RNA-seq reveals that glioblastoma recapitulates a normal neurodevelopmental hierarchy. *Nat. Commun.* **11**, 3406 (2020).
- Guerrero-Juarez, C. F. et al. Single-cell analysis of human basal cell carcinoma reveals novel regulators of tumor growth and the tumor microenvironment. *Sci. Adv.* **8**, 7981 (2022).
- Lederer, A. R. & La Manno, G. The emergence and promise of single-cell temporal-omics approaches. *Curr. Opin. Biotechnol.* **63**, 70–78 (2020).
- Bergen, V., Lange, M., Peidli, S., Wolf, F. A. & Theis, F. J. Generalizing RNA velocity to transient cell states through dynamical modeling. *Nat. Biotechnol.* **38**, 1408–1414 (2020).
- Barile, M. et al. Coordinated changes in gene expression kinetics underlie both mouse and human erythroid maturation. *Genome Biol.* **22**, 197 (2021).
- Cui, H., Maan, H. & Wang, B. DeepVelo: deep learning extends RNA velocity to multi-lineage systems with cell-specific kinetics. Preprint at <https://www.biorxiv.org/content/10.1101/2022.04.03.486877v2> (2022).
- Gu, Y., Blaauw, D. & Welch, J. D. Bayesian inference of RNA velocity from multi-lineage single-cell data. Preprint at <https://www.biorxiv.org/content/10.1101/2022.07.08.499381v1> (2022).
- Huang, D. W. et al. DAVID Bioinformatics Resources: expanded annotation database and novel algorithms to better extract biology from large gene lists. *Nucleic Acids Res.* **35**, W169–W175 (2007).
- Qiu, X. et al. Mapping transcriptomic vector fields of single cells. *Cell* **185**, 690–711 (2022).
- Eich, C. et al. In vivo single cell analysis reveals Gata2 dynamics in cells transitioning to hematopoietic fate. *J. Exp. Med.* **215**, 233–248 (2018).
- Lau, E. O.-C. et al. DIAPH3 deficiency links microtubules to mitotic errors, defective neurogenesis, and brain dysfunction. *eLife* **10**, e61974 (2021).

25. Laub, F. et al. Transcription factor KLF7 is important for neuronal morphogenesis in selected regions of the nervous system. *Mol. Cell. Biol.* **25**, 5699–5711 (2005).
26. Upadhyay, A. et al. Neurocalcin delta knockout impairs adult neurogenesis whereas half reduction is not pathological. *Front. Mol. Neurosci.* **12**, 19 (2019).
27. Yamagata, M., Duan, X. & Sanes, J. R. Cadherins interact with synaptic organizers to promote synaptic differentiation. *Front. Mol. Neurosci.* **11**, 142 (2018).
28. Michibata, H. et al. Inhibition of mouse GPM6A expression leads to decreased differentiation of neurons derived from mouse embryonic stem cells. *Stem Cells Dev.* **17**, 641–651 (2008).
29. Feng, H., Khalil, S., Neubig, R. R. & Sidiropoulos, C. A mechanistic review on GNAO1-associated movement disorder. *Neurobiol. Dis.* **116**, 131–141 (2018).
30. Grant, S. G. Synaptopathies: diseases of the synaptome. *Curr. Opin. Neurobiol.* **22**, 522–529 (2012).
31. Bartkowska, K., Paquin, A., Gauthier, A. S., Kaplan, D. R. & Miller, F. D. Trk signaling regulates neural precursor cell proliferation and differentiation during cortical development. *Development* **134**, 4369–4380 (2007).
32. Malatesta, P. et al. Neuronal or glial progeny: regional differences in radial glia fate. *Neuron* **37**, 751–764 (2003).
33. Brown, J. P. et al. Transient expression of doublecortin during adult neurogenesis. *J. Comp. Neurol.* **467**, 1–10 (2003).
34. Couillard-Despres, S. et al. Doublecortin expression levels in adult brain reflect neurogenesis. *Eur. J. Neurosci.* **21**, 1–14 (2005).
35. Jacobs, S. et al. Mice with targeted *Slc4a10* gene disruption have small brain ventricles and show reduced neuronal excitability. *Proc. Natl Acad. Sci. USA* **105**, 311–316 (2008).
36. Bastidas-Ponce, A. et al. Comprehensive single cell mRNA profiling reveals a detailed roadmap for pancreatic endocrinogenesis. *Development* **146**, dev173849 (2019).
37. Byrnes, L. E. et al. Lineage dynamics of murine pancreatic development at single-cell resolution. *Nat. Commun.* **9**, 3922 (2018).
38. Li, X. Y., Zhai, W. J. & Teng, C. B. Notch signaling in pancreatic development. *Int. J. Mol. Sci.* **17**, 48 (2015).
39. Hoffman, B. G., Zavaglia, B., Beach, M. & Helgason, C. D. Expression of Groucho/TLE proteins during pancreas development. *BMC Dev. Biol.* **8**, 81 (2008).
40. Battich, N. et al. Sequencing metabolically labeled transcripts in single cells reveals mRNA turnover strategies. *Science* **367**, 1151–1156 (2020).
41. Wolf, F. A., Angerer, P. & Theis, F. J. SCANPY: large-scale single-cell gene expression data analysis. *Genome Biol.* **19**, 15 (2018).
42. Gorin, G., Fang, M., Chari, T. & Pachter, L. RNA velocity unraveled. *PLoS Comput. Biol.* **18**, e1010492 (2022).
43. Jung, M. & Lee, E. K. RNA-binding protein HuD as a versatile factor in neuronal and non-neuronal systems. *Biolog (Basel)* **10**, 361 (2021).
44. Gao, M., Qiao, C. & Huang, Y. UniTVelo: temporally unified RNA velocity reinforces single-cell trajectory inference. *Nat. Commun.* **13**, 6586 (2022).
45. Gorin, G., Svensson, V. & Pachter, L. Protein velocity and acceleration from single-cell multiomics experiments. *Genome Biol.* **21**, 39 (2020).
46. Buenrostro, J. D. et al. Single-cell chromatin accessibility reveals principles of regulatory variation. *Nature* **523**, 486–490 (2015).
47. Hochreiter, S. & Schmidhuber, J. Long short-term memory. *Neural Comput.* **9**, 1735–1780 (1997).
48. Lecun, Y., Bottou, L., Bengio, Y. & Haffner, P. Gradient-based learning applied to document recognition. *Proc. IEEE* **86**, 2278–2324 (1998).

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this license, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2023

Methods

Modeling RNA transcriptional dynamics

The reaction kinetics of a single gene is described by two ordinary differential equations:

$$\frac{du(t)}{dt} = \alpha(t) - \beta(t)u(t) \quad (1)$$

$$\frac{ds(t)}{dt} = \beta(t)u(t) - \gamma(t)s(t) \quad (2)$$

where $u(t)$ and $s(t)$ are time-dependent concentrations of the premature and mature mRNAs, and α, β, γ indicate the transcription, splicing and degradation rates, respectively. For simplicity, one of the key assumptions in previous models for estimation of RNA velocity is that α is either a constant (velocity model) or a binary (scVelo model) value, and β and γ are shared by all the genes and cells. However, the assumption fails in evaluation of a heterogeneous cell subpopulation. In this study, we developed cellDancer, a deep learning framework, to generalize estimation of RNA velocity in both homogeneous and heterogeneous cell populations by predicting cell-specific time-dependent α, β and γ from premature and mature reads. A unique feature of the cellDancer framework is its capability to determine gene-specific kinetics that can be described by the rate equations (Eqs. 1 and 2).

In cellDancer, we use a DNN with a set of network parameters (θ) to learn the unknown functions that map the predictive features to the rates. Specifically, for gene i in the scRNA-seq dataset, there are n captured cell snapshots ($t = t_1, t_2, \dots, t_n$) at different stages of the cell development (for simplicity, we also refer to time t_j as 'cell j ' throughout the paper), and we formulated the reaction rates as functions of the abundances of the unspliced and spliced mRNAs in Eq. 3:

$$(\alpha^i(t), \beta^i(t), \gamma^i(t))^T = \Phi_{\theta^i}(u^i(t), s^i(t)) \quad (3)$$

where the DNN is described as a mapping Φ with gene-specific network parameters (θ^i). To train the DNN, we send one gene to the DNN at a time. We randomly sample a subset of cells (details in the 'Model parameters' subsection) as the input in each epoch of training. We leave out the superscript notation i in the following detailed steps for prediction.

First, the reaction kinetics ODEs in Eqs. 1 and 2 are discretized:

$$\frac{u(t + \Delta t) - u(t)}{\Delta t} = \alpha(u(t), s(t)) - \beta(u(t), s(t))u(t), \quad (4)$$

$$\frac{s(t + \Delta t) - s(t)}{\Delta t} = \beta(u(t), s(t))u(t) - \gamma(u(t), s(t))s(t), \quad (5)$$

where pseudotime t is discretized and Δt is an infinitesimal time increment. We use cellDancer to jointly predict cell-specific $\alpha(u(t_j), s(t_j))$, $\beta(u(t_j), s(t_j))$ and $\gamma(u(t_j), s(t_j))$ given spliced and unspliced mRNA abundance $u(t_j)$ and $s(t_j)$ of cell j . Second, we use the predicted rates to calculate the extrapolated mRNA abundance $s(t_j + \Delta t)$ and $u(t_j + \Delta t)$ by the discretized reaction kinetics. To measure the difference between predicted and true velocity vectors, we define a loss function \mathcal{L} based on every cell's cosine similarity between the predicted and observed velocity vectors:

$$\mathcal{L} = \sum_{j=1}^n \mathcal{L}_j, \quad (6)$$

$$\mathcal{L}_j = 1 - \max_{\{j'\}} \frac{v_j \cdot v'_j}{|v_j| * |v'_j|}, \quad (7)$$

$$v_j = (u(t_j + \Delta t) - u(t_j), s(t_j + \Delta t) - s(t_j)), \quad (8)$$

$$v'_j = (u(t_{j'}) - u(t_j), s(t_{j'}) - s(t_j)), \quad (9)$$

$\mathcal{L}(\mathcal{L}_j)$ is the overall (cell j) loss function; $v_j (v'_j)$ is the predicted (observed) RNA velocity vector, where $\{j'\}$ is a collection of cells in the neighborhood of cell j ; and t_j is the observed cell in the neighboring cells $\{j'\}$ that minimizes the loss function for cell j . Note that the neighboring cells are controlled by the number of $n_{neighbors}$ and can be either gene-specific (calculated in the phase space of each gene) or gene-shared (calculated in the embedding space using the abundances of the spliced mRNA or the abundances of both the spliced and the unspliced mRNA).

Finally, we obtain θ^i by minimizing the overall loss function \mathcal{L} for gene i by applying the Adam optimization algorithm in a DNN. The configuration of the DNN is as follows: an input layer with $2n$ nodes; two fully connected hidden layers each with 100 nodes and the leaky ReLU activation function; and an output layer with $3n$ nodes. The sigmoid activation function $\sigma(x) = \frac{1}{1+e^{-x}}$ is applied as a regularization to constrain the outputs (α, β and γ) within the range [0, 1]. The learning rate of the Adam optimizer is 0.001. The weight decay is 0.004, which adds L2 penalty to the weights parameters and prevents overfitting. The training of the DNNs is terminated if the loss function does not decrease after three checkpoints. Those training parameters are fully controllable by the user in the cellDancer command line interface. The DNN in cellDancer is implemented using PyTorch Lightning⁴⁹, a widely used Python library.

Simulation details

To assess the accuracy and limitation of cellDancer, we generate various kinetic regimes of the expression profiles using time-dependent rates of transcription, splicing and degradation (α, β, γ). Specifically, for one gene, a set of differential equations is solved by numerical integration using the function `integrate.solve_ivp` under the SciPy package⁵⁰ with the Runge–Kutta method^{51,52}. The unspliced and spliced reads are initialized to 0. Gaussian noises are added to the generated gene expression level for each cell.

We simulate the spliced and unspliced expression of 2,000 cells and 1,000 genes for transcriptional boost, multi-forward branching and multi-backward branching regimes. For transcriptional boost genes, α is sampled from a uniform distribution of $U(1.6, 2.4)$ before boosting and $U(4, 6)$ for cells after boosting where the lower and upper limits are set by varying 20% from the mean values of 2 (before boosting) and 5 (after boosting). β is sampled from a uniform distribution of $U(1.8, 2.2)$ for all cells where the lower and upper limits are set by varying 10% from the mean value of 2. γ is sampled from a uniform distribution of $U(0.9, 1.1)$ where the lower and upper limits are set by varying 10% from the mean value of 1 for all cells. For multi-forward branching genes, α is sampled from a uniform distribution of $U(0.8, 1.2)$ for cells in the first lineage and $U(4, 6)$ for cells in the second lineage where the lower and upper limits are set by varying 20% from the mean values of 1 (first lineage) and 5 (second lineage). β is sampled from a uniform distribution of $U(0.4, 0.6)$ for cells in the first lineage and $U(0.8, 1.2)$ for cells in the second lineage where the lower and upper limits are set by varying 20% from the mean value of 0.5 (first lineage) and 1 (second lineage). γ is sampled from a uniform distribution of $U(0.2, 0.3)$ for cells in the first lineage and $U(4, 6)$ for cells in the second lineage where the lower and upper limits are set by varying 20% from the mean values of 0.25 (first lineage) and 5 (second lineage). For multi-backward branching genes, α is set to 0 in all cells. β and γ are sampled from a uniform distribution of $U(0.9, 1.1)$ where the lower and upper limits are set by varying 10% from the mean value of 1 for all cells. In the first lineage, cells start from a region around a point of ($s = 1.3, u = 0.2$) to decrease. In the second lineage, cells start from a region around a point of ($s = 1, u = 1$) to decrease. The data are used as input of a standard cellDancer analysis pipeline. After velocity estimation, we calculate an error rate to

evaluate the accuracy of cellDancer against the ground truth velocity. The error rate is calculated as the percentage of cells having a low correlation coefficient (lower than 0.7 as a cutoff) between the estimated velocity and the ground truth velocity.

To investigate the robustness of cellDancer in data with high technical dropout, we simulate dropout in the expression of unspliced and spliced mRNAs. According to the experimental datasets in this study, the average dropout ratios for the unspliced and spliced mRNA reads are in the range of 50% to 70% for the top 2,000 highly variable genes. Therefore, for dropout ratios of 50%, 60% and 70%, we simulate 1,000 genes each. To achieve this, we first generate the spliced and unspliced abundances (u_j^i and s_j^i for gene i of cell j), which follow the transcriptional dynamics equations (Eqs. 1 and 2). We assume that those abundances are averaged over the raw counts ($U_{\{j'\}}$ and $S_{\{j'\}}$) of the neighboring cells, as in real scRNA-seq data those raw counts are zero-inflated. Based on this assumption, for a gene i in any given cell j , we randomly generate spliced and unspliced raw counts that follow the Poisson law ($U_{\{j'\}} \sim \text{Poisson}(u_j^i)$ and $S_{\{j'\}} \sim \text{Poisson}(s_j^i)$) for 200 neighboring cells $\{j'\}$. We perform a grid search for the kinetic rate parameters (α, β, γ) in the range [0.1, 1.0] at a step of 0.1. We use kinetic parameters that lead to dropout ratios ($50\% \pm 3\%$, $60\% \pm 3\%$ and $70\% \pm 3\%$) in our RNA velocity estimation, where the averaged raw counts (sample average) are used for the unspliced and spliced abundances.

Pseudotime estimation

The RNA velocity vector for a cell j is represented by a high-dimensional vector $v_j = (v_j^1, v_j^2, \dots, v_j^g)$, where g is the total number of genes and v_j^i is the velocity for gene i in cell j . Following the method of velocyto and scVelo, we project the velocity vectors of the cells into the low-dimensional embedding space $\{\xi\}_{\text{dim}}$ using embedding algorithms such as PCA, t-distributed stochastic neighbor embedding (t-SNE) or UMAP for visualization and gene-shared pseudotime estimation. Under the assumption that the more correlated the change in the gene expression $\delta_{jj'} = s_j - s_{j'}$ from cells j and j' with the direction of the velocity v_j , the higher chance that cell j could transition to cell j' , we construct the transition probability matrix by applying an exponential kernel to the correlation between $\delta_{jj'}$ and v_j :

$$P_{jj'} \propto e^{\frac{\text{corr}(v_j, \delta_{jj'})}{\sigma}}, \quad (10)$$

where $\sigma = 0.05$. A normalization factor is applied to ensure the sum of transition probabilities for cell j to its neighboring cells (N , which is determined by k -nearest neighbors in the high-dimensional space or optionally the low-dimensional embedding space) is 1:

$$\sum_{j' \in N} P_{jj'} = 1, \quad (11)$$

The velocity of cell j on the low-dimensional embedding space $\{\xi\}$ is estimated as

$$\bar{v}_j = \sum_{j' \in N} (P_{jj'} - 1) \hat{\theta}_{jj'}, \quad (12)$$

where $\hat{\theta}_{jj'}$ is the unitary vector of the displacement between cell j and j' in the embedding space.

To detect the cell state transition paths and track the continuous changes in transcriptome along those paths, we sort the cells in temporal order by carrying out cell (gene-shared) pseudotime analysis based on the RNA velocities. First, we divide the low-dimensional embedding space $\{\xi\}$ to a customized grid to smooth the abrupt velocity vector flows, and the velocity of a cell j in a grid l (or ‘meta cell’) is estimated as the mean velocity \bar{v}_l of the enclosed cells. We then generate a pool of trajectories $\{\xi_j^{(t_0)}, \xi_j^{(t_1)}, \xi_j^{(t_2)}, \dots\}_{j=1, \dots, n_{\text{cells}}}^{r=1, \dots, n_{\text{repeats}}}$ tracing the

velocity streamlines starting from any cell j using the following equation of motion:

$$\xi_j(t + \Delta t) = \xi_j(t) + \bar{v}_l \Delta t. \quad (13)$$

A Gaussian-distributed swaying angle $\theta \in N(0, \pi/6)$ is applied at every timestep to allow a slight deviation from the direction of the velocity flow. Second, from the trajectory pool, we select m trajectories $\{L_k(t)\}_{k=1, \dots, m}$ whose traverse length is a local maximum (or long trajectories, as shown in Extended Data Fig. 2d for the erythroid maturation dataset). The traverse length is computed as the accumulated distance of a trajectory $\sum_t \| \xi(t + \Delta t) - \xi(t) \|$. The long trajectories are determined by iteratively selecting the longest trajectory and eliminating its similar trajectories within a cutoff until no trajectory is left in the pool. The fate of a neighboring cell j is decided by whether most of the trajectories originated from the position of cell j , $\xi_j(t_0)$, terminate on/around a long trajectory $L_l(t)$. The pseudotime t_l of cell j is then assigned as the time on $L_l(t)$, where $L_l(t_l)$ is closest to $\xi_j(t_0)$ (Extended Data Fig. 2d). Finally, at this moment, all the cells are assigned a relative time according to the respective paths, or ‘time zones’, and we need to adjust the relative time of the cells by finding the time shift between those ‘time zones’. This is done based on an assumption that ‘overlapping’ cells (in practice, we consider cells in close proximity) in the embedding space (or optionally in the high-dimensional expression space) also share the same time. The assumption is consistent with the assumption on which the transition probability matrix is based. The time for the cells in each ‘time zone’ (or cluster) is adjusted using a graph-based approach. The time adjustment algorithm is outlined below.

- 1) Construct the graph. Every cluster forms a node, and an edge is formed between nodes l and m if there is a time shift $\Delta t_{lm} = t_l - t_m$ between the ‘overlapping’ cells going for path L_l and path L_m . Therefore, each cell abiding by the L_m ‘time zone’ needs addition of Δt_{lm} to the original cell time to consolidate all the cell time in the two clusters.
- 2) Divide the graph into individual trees. If the graph is a forest, divide it into trees. If a cycle exists, the time adjustment algorithm fails. In the latter scenario, we suggest reducing the n_{path} parameter to reduce the number of long trajectories.
- 3) Compute the accumulative time shift τ_k^{abs} needed for each node $k \in \{1, 2, \dots, n_{\text{nodes}}\}$ in each tree T in a few steps.
- 4) Initiate $\{\tau_k^{\text{abs}}\}$ with 0 for each node $k \in \{1, 2, \dots, n_{\text{nodes}}\}$. Initiate a marker for each node $\{\text{flag}_k\}$ with 0.
- 5) Start from a node o and set the marker to 1. Traverse all the connections. For a connection between node l and m : add τ_m^{abs} by Δt_{lm} if l equals o and set the marker for node m to 1; subtract τ_l^{abs} by Δt_{lm} if m equals o and set the marker for node l to $\text{flag}_l = 1$. Repeat the process until all the nodes are marked as 1.

scRNA-seq data and pre-processing

All scRNA-seq data in this study were downloaded publicly (see details in the ‘Data availability’ section).

- 1) For the pancreatic endocrinogenesis data, we followed the method by Bergen et al. in the scVelo study¹⁷ and filtered 3,696 cells with 2,000 genes for further analysis.
- 2) For the mouse hippocampal dentate gyrus neurogenesis data, we followed the gene and cell filtering methods by La Manno et al.¹² and selected 18,140 cells with 2,159 genes.
- 3) For the erythroid lineage of the mouse gastrulation data, we selected 12,329 cells from cell types, including hematoendothelial progenitors, blood progenitors 1/2 and erythroid 1/2/3 in stages of E7.0, E7.25, E7.5, E7.75, E8.0, E8.25 and E8.5. We followed the standard data pre-processing procedures in scVelo with default parameters except that we used 100 nearest neighbors

- for the calculation of the first moment to reduce the noise in transcripts.
- (4) For the human embryo glutamatergic neurogenesis dataset, we kept cells with at least 200 genes expressed and kept genes that were captured in at least three cells. We identified all the high variable genes with the default settings of `scipy.pp.highly_variable_genes()` by using SCANPY⁴¹. In total, 1,054 cells with 1,720 genes were selected. We used 200 nearest neighbors for the calculation of the first moment to reduce the noise in transcripts.
- (5) For the cell cycle progression in the REP1-FUCCI cells, we used the scEU-seq data, in which we took 3,058 cells with the top 2,000 high variable genes from the pulse experiment. The unspliced mRNA reads were calculated as the addition of the unspliced labeled and unspliced unlabeled reads, likewise for the spliced mRNA reads. We used 300 nearest neighbors for the calculation of the first moment to reduce the noise in transcripts. The synthesis and degradation rates (molecules per hour) measured by scEU-seq data were obtained from the study of the original paper⁴⁰.

Model parameters

In DNN training, the learning rate and patience are associated with the total number of training epochs. In all case studies, the learning rate was set to 0.001, which is widely used in Adam optimizer. The patience was set to 3 in all case studies. The time increment Δt in Eqs. 4 and 5 was set to 0.5. The permutation ratio determines how many cells were sent to train the model in each epoch. We recommend using a large permutation ratio for datasets with a small number of cells or datasets presenting a clear pattern in gene phase portraits. Specifically, for gastrulation erythroid maturation (12,329 cells) and the cell cycle progression in REP1-FUCCI data (3,058 cells), we used the default permutation ratio of 0.125; for the mouse hippocampus development dataset (18,140 cells), we set the permutation ratio to 0.1; for the pancreatic endocrinogenesis data (3,696 cells), we set the permutation ratio to 0.5; and for the human embryo glutamatergic neurogenesis data (1,720 cells), we set the permutation ratio to 0.3. For all genes within the same dataset, the DNN parameters were kept the same.

Reporting summary

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

Data availability

All the scRNA-seq raw data are publicly accessible. The pancreatic endocrinogenesis data can be extracted using scVelo's CLI: `scvelo.datasets.pancreas()` or accessed from the original work³⁶ under accession number GSE132188 of the Gene Expression Omnibus (GEO). The hippocampal dentate gyrus neurogenesis data can be accessed at <http://pklab.med.harvard.edu/velocyto/DentateGyrus/DentateGyrus.loom> or the original paper⁵³ under GEO accession number GSE95753. The erythroid lineage of mouse gastrulation data can be extracted using scVelo's CLI: `scvelo.datasets.gastrulation()` or from the original work² under accession number E-MTAB-6967 of ArrayExpress. Human embryo glutamatergic neurogenesis can be accessed at https://github.com/pachterlab/GFCP_2022/blob/main/notebooks/data/hgForebrainGlut.loom or the original work¹² under Sequence Read Archive accession code SRP129388. Cell cycle progression in

REP1-FUCCI cells can be extracted using dynamo's CLI: `dyn.sample_data.scEU_seq_rpe1()` or from the original work²² under GEO accession number GSE128365.

Code availability

cellDancer is implemented in Python and is available at <https://github.com/GuangyuWangLab2021/cellDancer>.

References

49. Paszke, A. et al. PyTorch: an imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems* (Eds. Wallach, H. et al.) (NeurIPS, 2019).
50. Virtanen, P. et al. SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nat. Methods* **17**, 261–272 (2020).
51. Dormand, J. R. & Prince, P. J. A family of embedded Runge–Kutta formulae. *J. Comput. Appl. Math.* **6**, 19–26 (1980).
52. Shampine, L. F. Some practical Runge–Kutta formulas. *Math. Comput.* **46**, 135–150 (1986).
53. Hochgerber, H., Zeisel, A., Lonnerberg, P. & Linnarsson, S. Conserved properties of dentate gyrus neurogenesis across postnatal development revealed by single-cell RNA sequencing. *Nat. Neurosci.* **21**, 290–299 (2018).

Acknowledgements

This work was supported by a Houston Methodist internal grant (G.W.). The cartoon in Fig. 1a was created with BioRender (<https://biorender.com>).

Author contributions

G.W. supervised the study. S.L. and G.W. designed and developed the relay velocity model. S.L., P.Z. and W.C. analyzed the data. P.Z. and G.W. developed the pseudotime algorithm. L.Y., P.Z. and G.W. developed the simulation method. S.L. and W.C. examined the robustness. G.W., S.L. and P.Z. wrote the manuscript. K.W.B., N.-T.L., J.A. and J.P.C. were involved in the discussion and helped to improve the manuscript. All authors gave approval to the final version of the manuscript.

Competing interests

The authors declare no competing interests.

Additional information

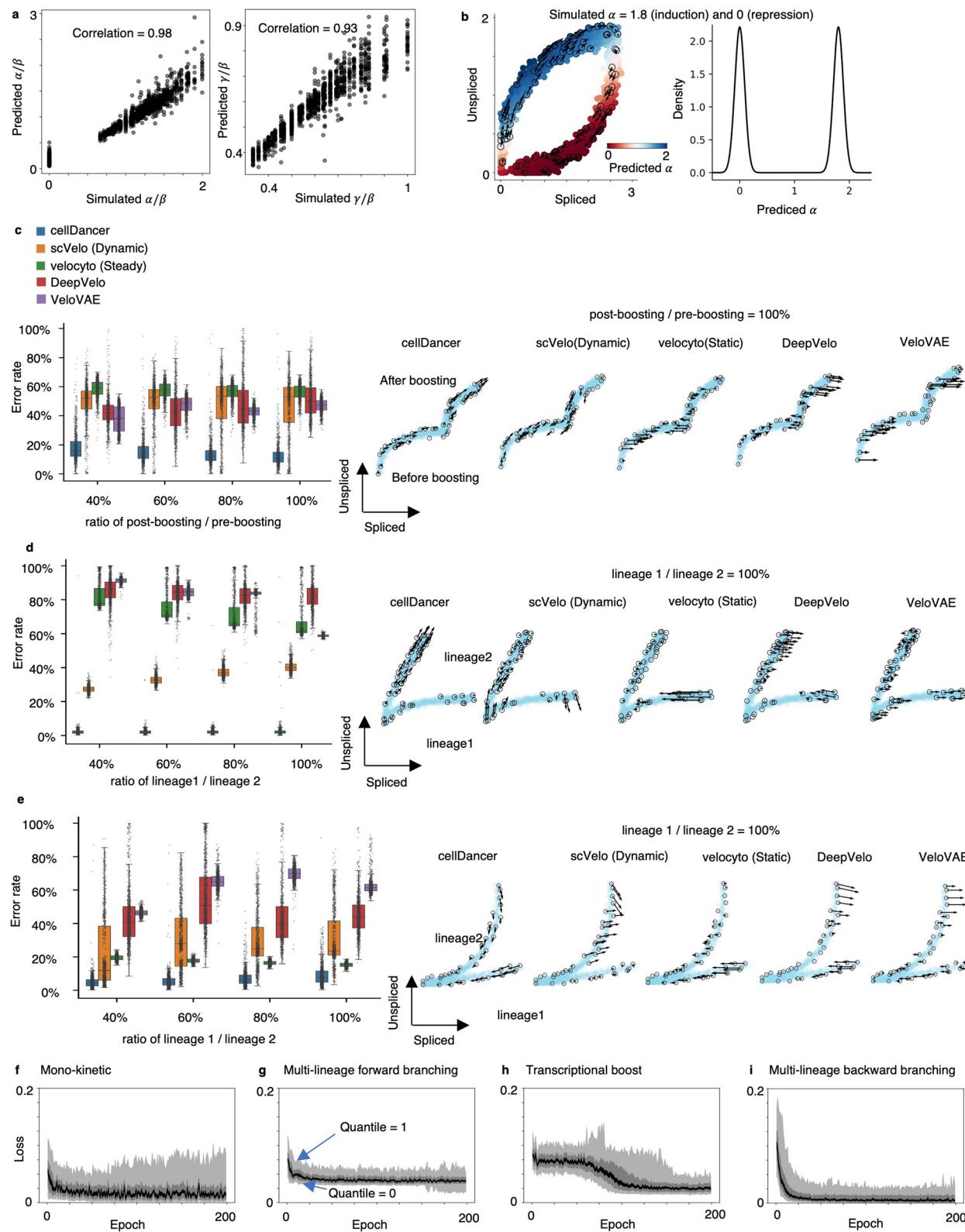
Extended data is available for this paper at <https://doi.org/10.1038/s41587-023-01728-5>.

Supplementary information The online version contains supplementary material available at <https://doi.org/10.1038/s41587-023-01728-5>.

Correspondence and requests for materials should be addressed to Guangyu Wang.

Peer review information *Nature Biotechnology* thanks Xiaojie Qiu and the other, anonymous, reviewer(s) for their contribution to the peer review of this work.

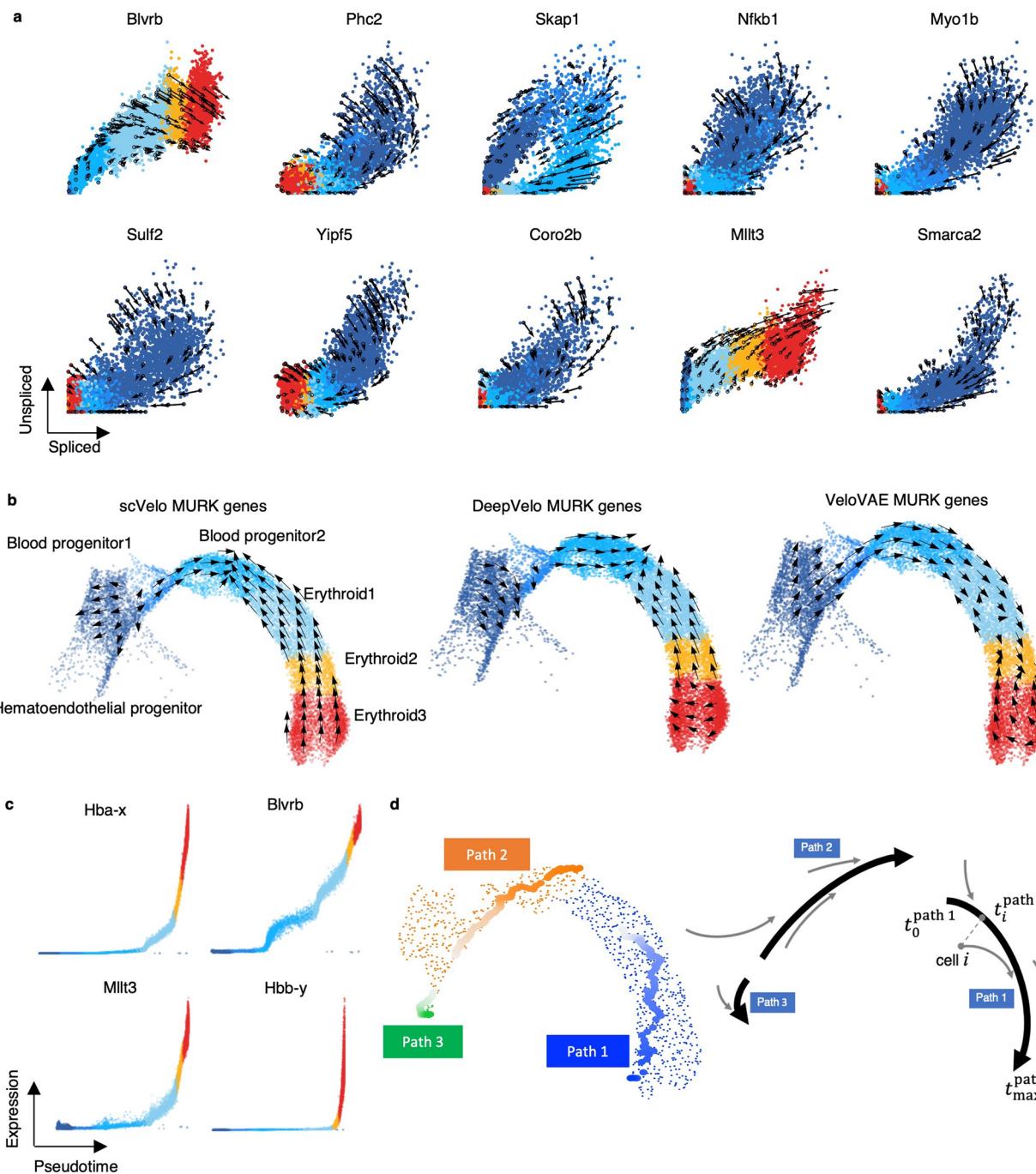
Reprints and permissions information is available at www.nature.com/reprints.



Extended Data Fig. 1 | See next page for caption.

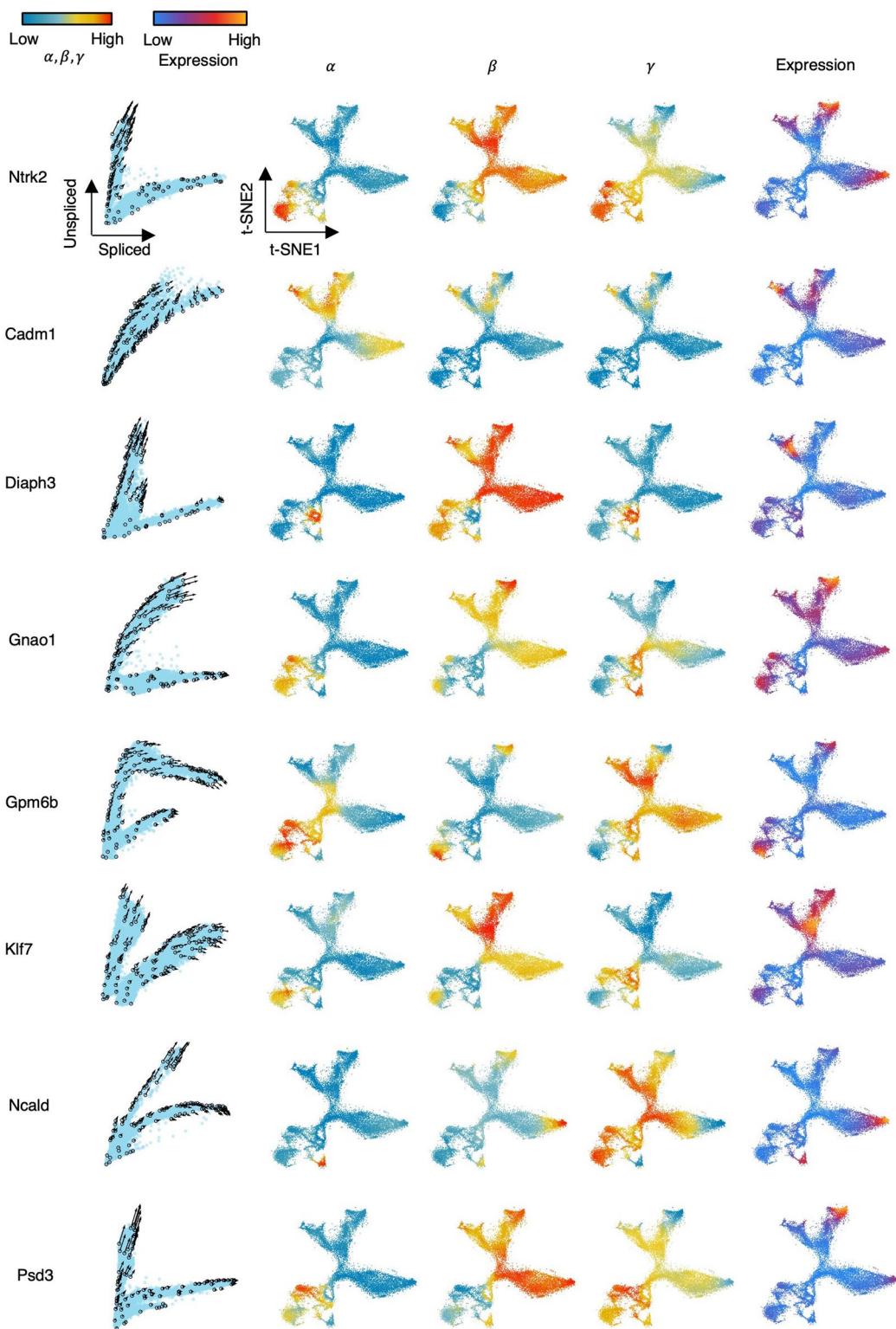
Extended Data Fig. 1 | Resolving RNA velocity of simulated multiple rate kinetics genes. (a) High correlation between the simulated (background truth) and the predicted α/β (left) and γ/β (right). The Pearson correlation coefficients (R^2) between the prediction and the simulation are computed. (b) RNA velocity predicted by cellDancer is projected onto the phase portraits of a simulated gene with α equals 1.8 (induction) and 0 (repression) (left) and the density plot of the predicted α (right). (c-e) We measured the accuracy by computing the error rate as the percentage of cells whose predicted RNA velocity is poorly correlated with the ground truth velocity (cosine similarity < 0.7). The box plots of the error rates show that cellDancer outperforms scVelo, velocyto, DeepVelo, and VeloVAE in the estimation of RNA velocities for the simulated transcriptional boost (c), multi-lineage forward branching (d), and multi-lineage backward branching (e) genes. Middle line in box plot, median; box boundary, interquartile range; whiskers, 10–90 percentile; minimum and maximum, not indicated in the box

plot; gray dots, individual datapoints. The error rate is defined as the percentage of falsely predicted directions. Different sampling ratios were investigated at 40%, 60%, 80%, and 100% ($n = 1,000$ genes), representing the ratio of the number of post-boosting cells to the number of pre-boosting cells in (c) and the ratio of the number of cells in lineage 1 to the number of cells in lineage 2 (d-e). Example phase portraits for sampling ratio (1:1) are provided in each case. (f-i) The loss scores are plotted against epochs of training on the simulated mono-kinetic (top left), multi-lineage forward branching (top right), transcriptional boost (bottom left), and multi-lineage backward branching (bottom right) genes at quantiles 0, 0.1, 0.4, 0.6, 0.9, and 1. In all cases, the loss scores converge in about 25 epochs, except for the transcriptional boost genes, for which the convergence emerges in about 100 epochs.



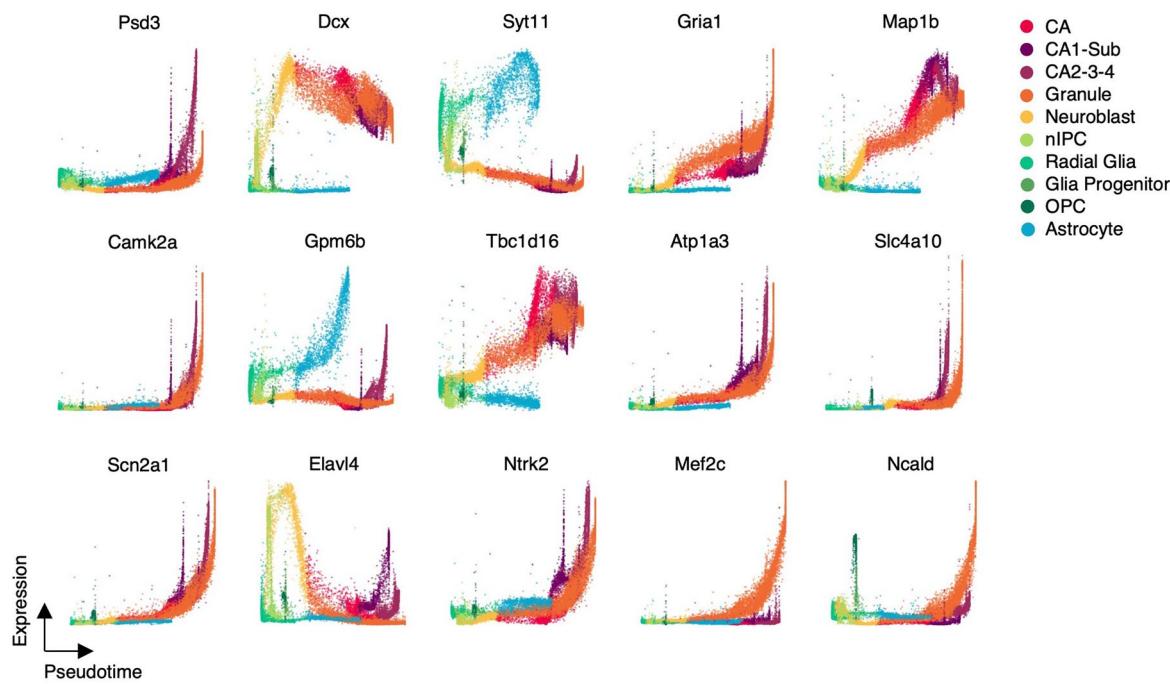
Extended Data Fig. 2 | RNA velocity estimation for the multiple rate kinetics (MURK) genes in the gastrulation erythroid maturation dataset. (a) RNA velocities predicted by cellDancer are projected onto the spliced-unspliced phase portraits for a set of selected genes. (b) The velocities derived from scVelo dynamic model, DeepVelo, and VeloVAE for gastrulation erythroid maturation cells using MURK genes are visualized on the pre-defined UMAP embedding. Inverted flows from the erythroid 3 to the blood progenitors 2 cell type are observed for the scVelo and DeepVelo predictions; inverted flows from the erythroid 3 to the erythroid 1 cell type are observed for the VeloVAE prediction. (c) Expression pseudotime profiles for four MURK genes *Hba-x*, *Blvrb*, *Mllt3*,

and *Hbb*-y show the expression patterns of transcriptional boost in gastrulation erythroid maturation. **(d)** Long trajectories used for pseudotime estimation in gastrulation erythroid maturation are visualized on the UMAP embedding. The long trajectories are local maxima of traverse length and are colored from light to dark according to their unadjusted pseudotime. The schematic diagram demonstrates how the unadjusted pseudotime of cells is determined according to the time in the long trajectories. The black bold lines stand for the long trajectories and the pseudotime for the originating cells of the short trajectories (gray lines) is obtained as the time of the closest cell in the corresponding long trajectory.

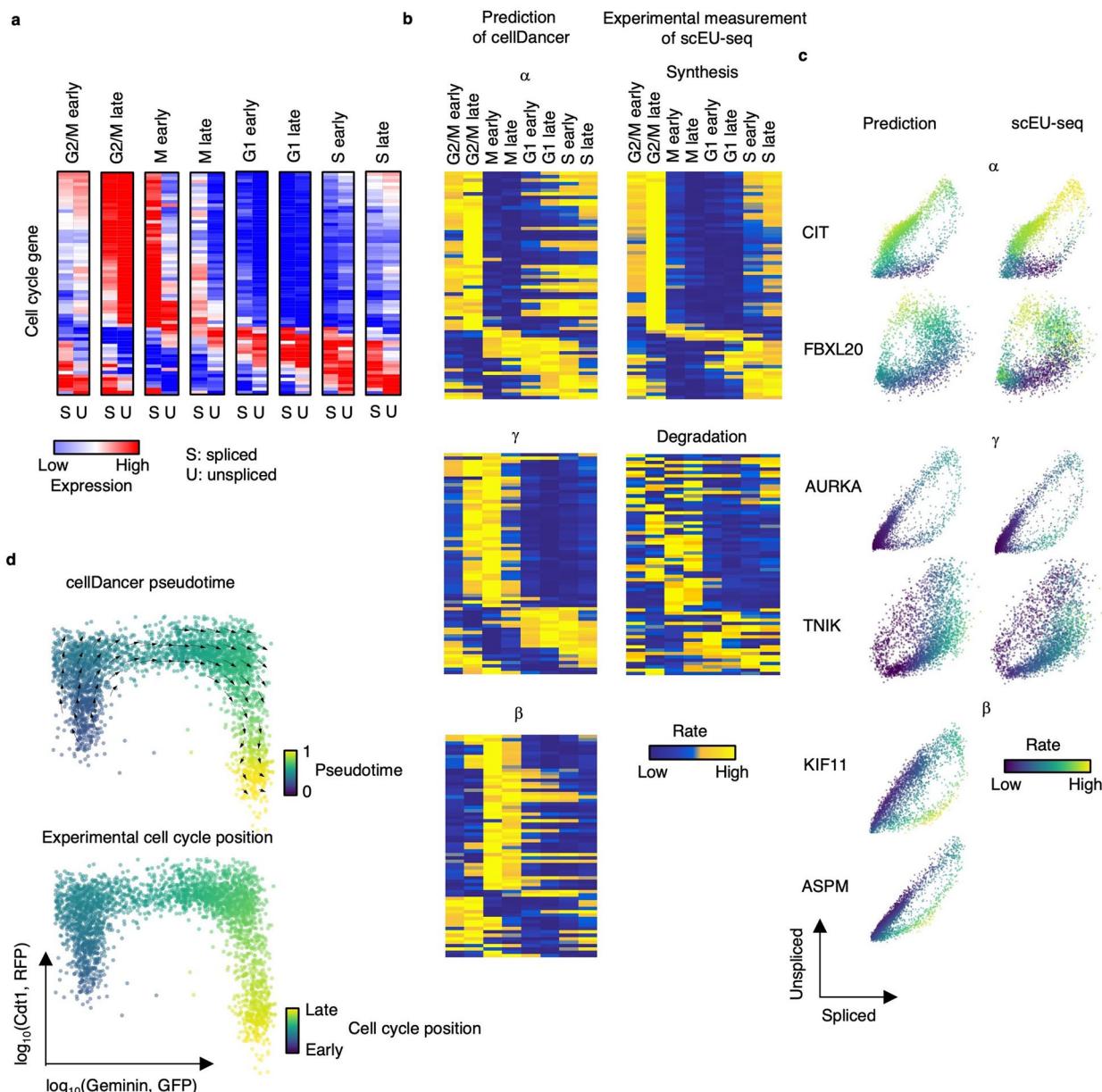


Extended Data Fig. 3 | Cell-specific kinetic rate parameters improve RNA velocity inference in hippocampus development. Panel (1) Velocities of selected genes inferred by cellDancer are projected onto the phase portraits;

Panels (2–4) Cells are colored according to the cell-specific α , β , and γ rates for the referenced gene in the t-SNE embedding for the hippocampus development; Panel (5) Cells are colored according to the gene expression.

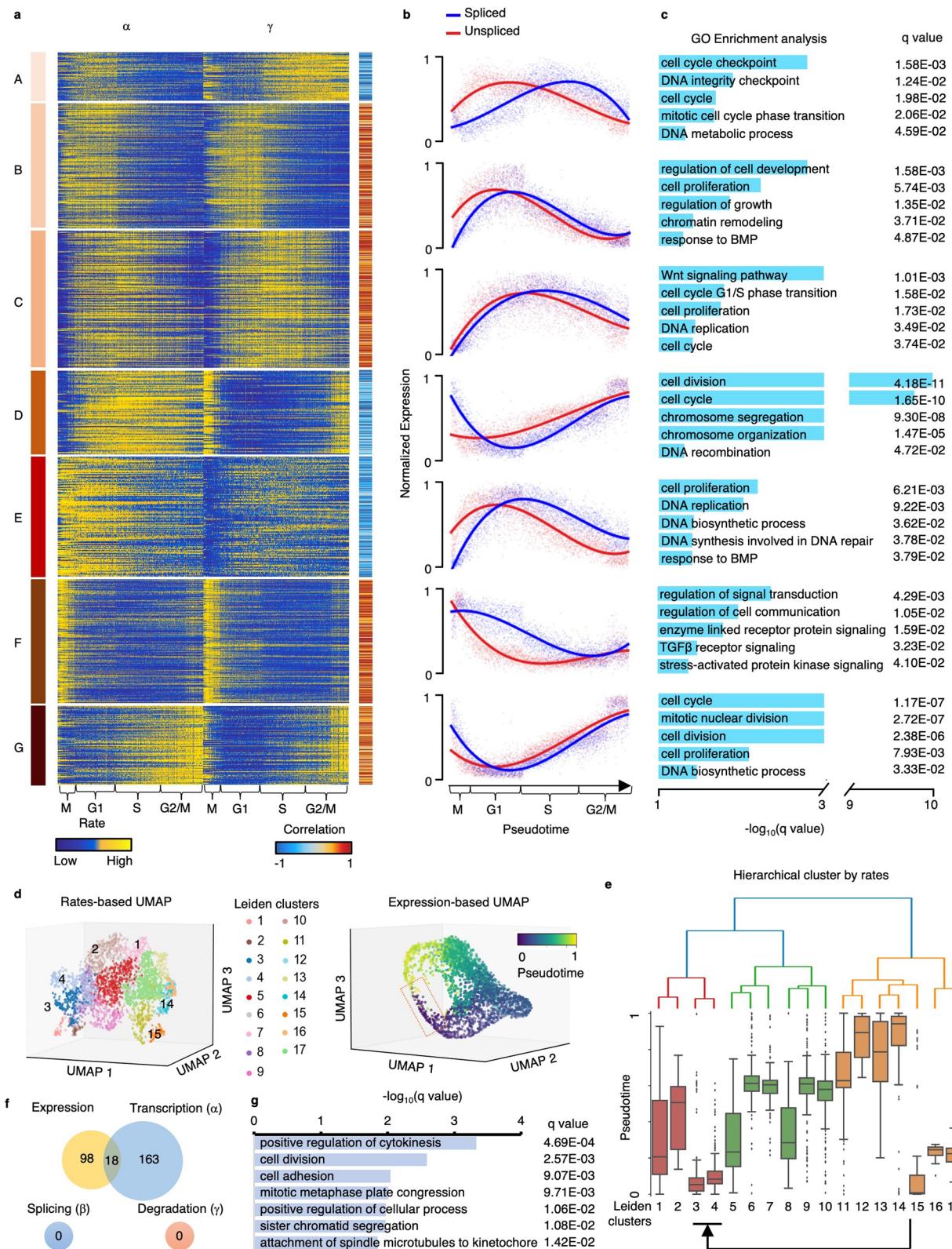


Extended Data Fig. 4 | The expressions along pseudotime of genes in hippocampal neurogenesis data. The expression pseudotime profiles for a selected set of genes in hippocampal neurogenesis.



Extended Data Fig. 5 | Revealing the turnover strategies of mRNA in the cell cycle process. (a) The spliced and unspliced reads of cell cycle genes in the cell cycle progression. The averaged spliced and unspliced reads were calculated for each cell cycle group. (b) Heatmaps show α , β , and γ estimated by cellDancer (first column) is associated with the experimentally derived synthesis and degradation status in scEU-seq (second column) in the cell cycle progress. (c) The phase

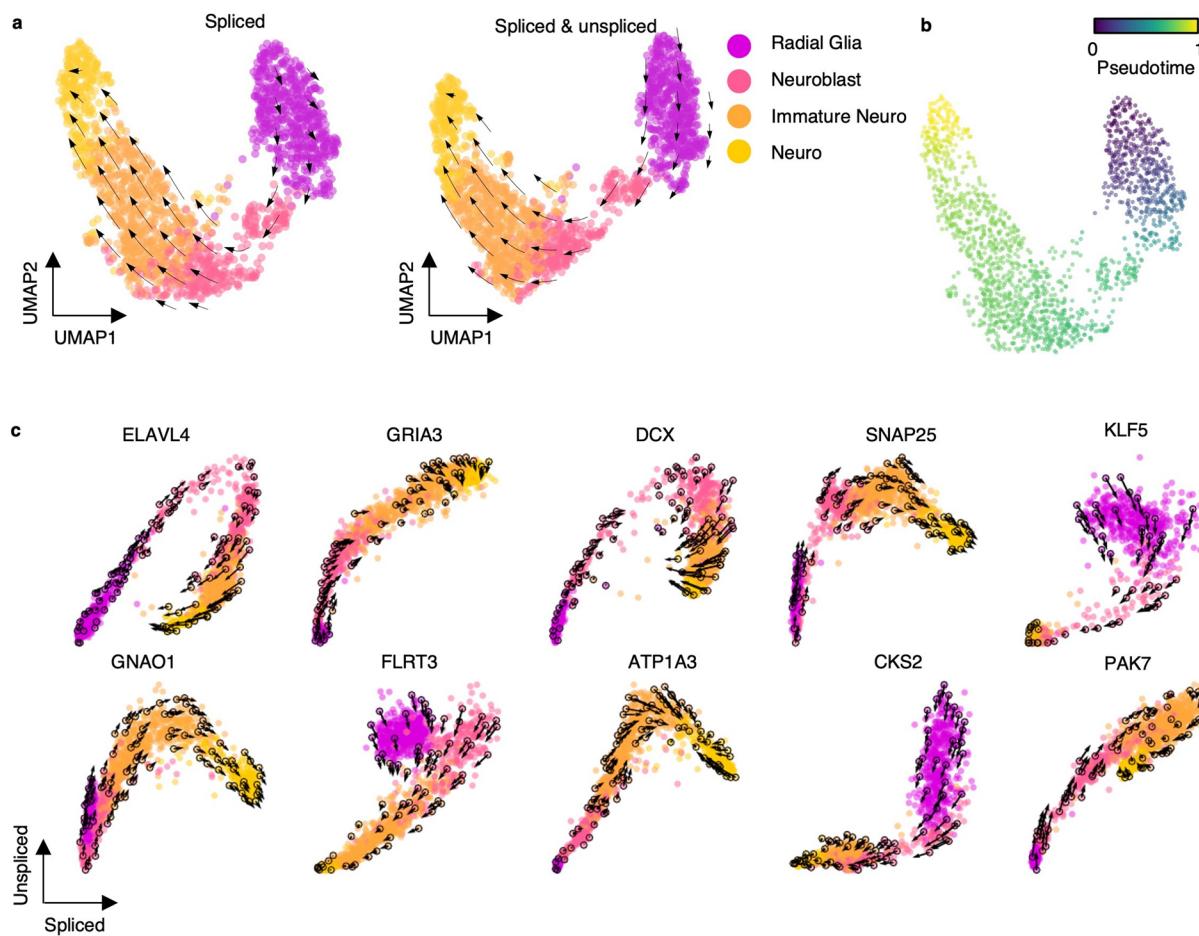
portraits of cell cycle genes show the predicted kinetic parameters are related to experimental measurements in scEU-seq. **(d)** The velocities derived from cellDancer for metabolic labeling dataset are visualized on the relative position along the cell cycle using the Geminin-GFP and the Cdt1-RFP signals from the FUCCI system. Gene-shared pseudotime on the relative position is consistent with the experimental cell cycle position.



Extended Data Fig. 6 | See next page for caption.

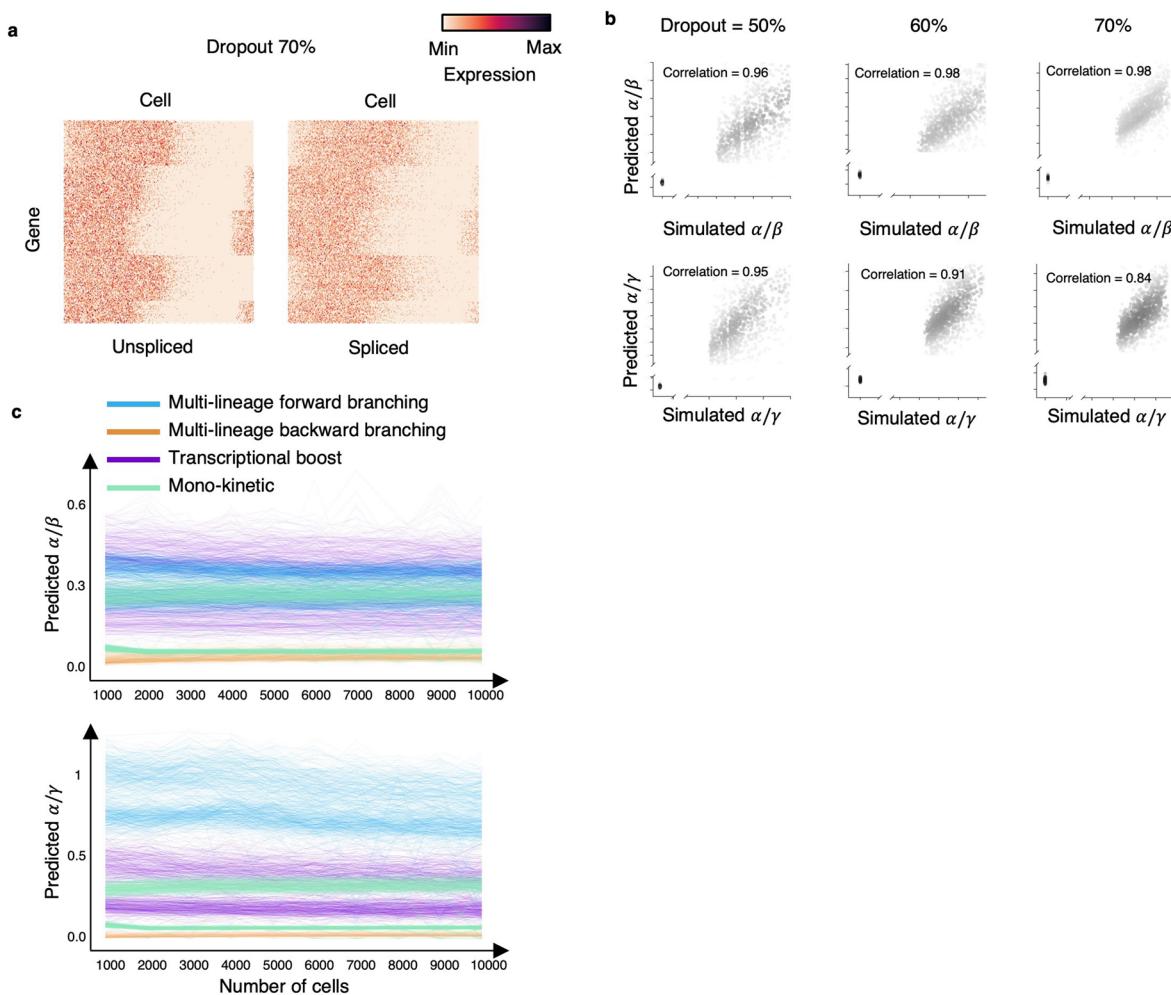
Extended Data Fig. 6 | The dynamic pattern of rates identifies the different turnover strategies of genes, and cell-specific reaction rates reveal cell subpopulation uncaptured by expression. (a) α and γ of genes along pseudotime. Genes are clustered into seven groups according to their dynamic patterns of α and γ . The Pearson correlation coefficients between α and γ are calculated. (b) The normalized spliced and unspliced reads of genes along pseudotime in each clustered group. (c) The GO pathway enrichment analysis using adjusted p -values of Fisher's Exact test (Benjamini–Hochberg procedure, one-sided, $p < 0.05$) for genes in each group. (d) The 3D UMAP based on α , β , and γ colored by Leiden clusters (top) and the 3D UMAP based on expression

colored by cell cycle pseudotime (bottom). (e) The hierarchical tree of the Leiden clusters. The box plot ($n = 3,058$ cells) shows the pseudotime of each cluster. Middle line in box plot, median; box boundary, interquartile range; whiskers, 10–90 percentile; minimum and maximum, not indicated in the box plot; gray dots, individual datapoints. (f) Venn diagram of genes with significant difference ($p < 0.05$, $FC > 1.5$ or $FC < 1/1.5$) on expression, α , β , and γ between the clusters 3 & 4 and other clusters. (g) The GO pathway enrichment analysis using adjusted p -values of Fisher's Exact test (Benjamini–Hochberg procedure, one-sided, $p < 0.05$) of DAVID for the 163 genes that only differential in α .



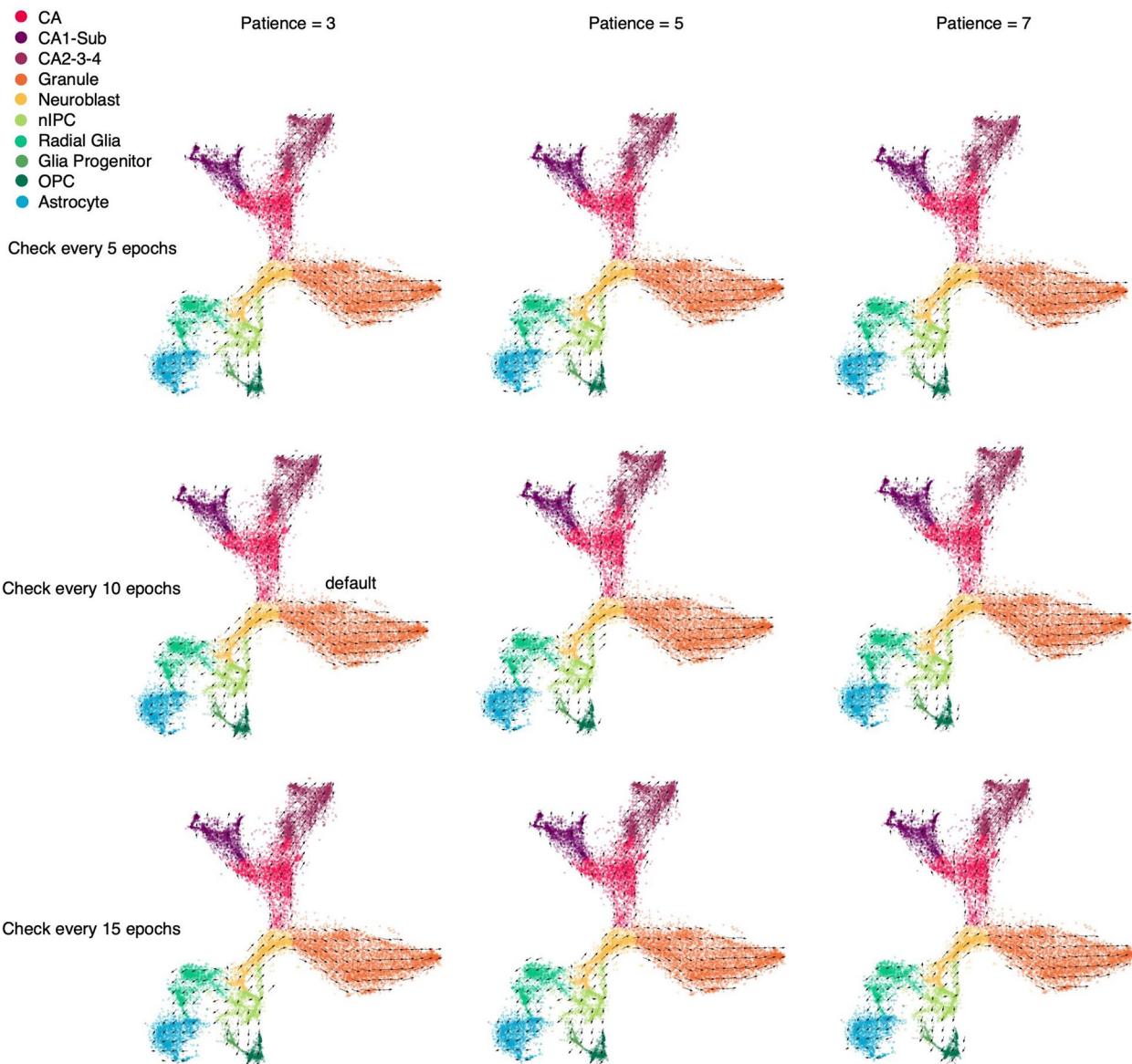
Extended Data Fig. 7 | cellDancer decodes human embryonic glutamatergic neurogenesis. (a) The velocities derived from cellDancer for human embryo glutamatergic neurogenesis are visualized on the UMAP embedding based on spliced reads (left), and on the UMAP embedding based on the spliced and

unspliced reads (right). (b) Gene-shared pseudotime projected on UMAP shows the order of cell development during neurogenesis. (c) RNA velocities predicted by cellDancer are projected onto the phase portraits for a set of selected genes.



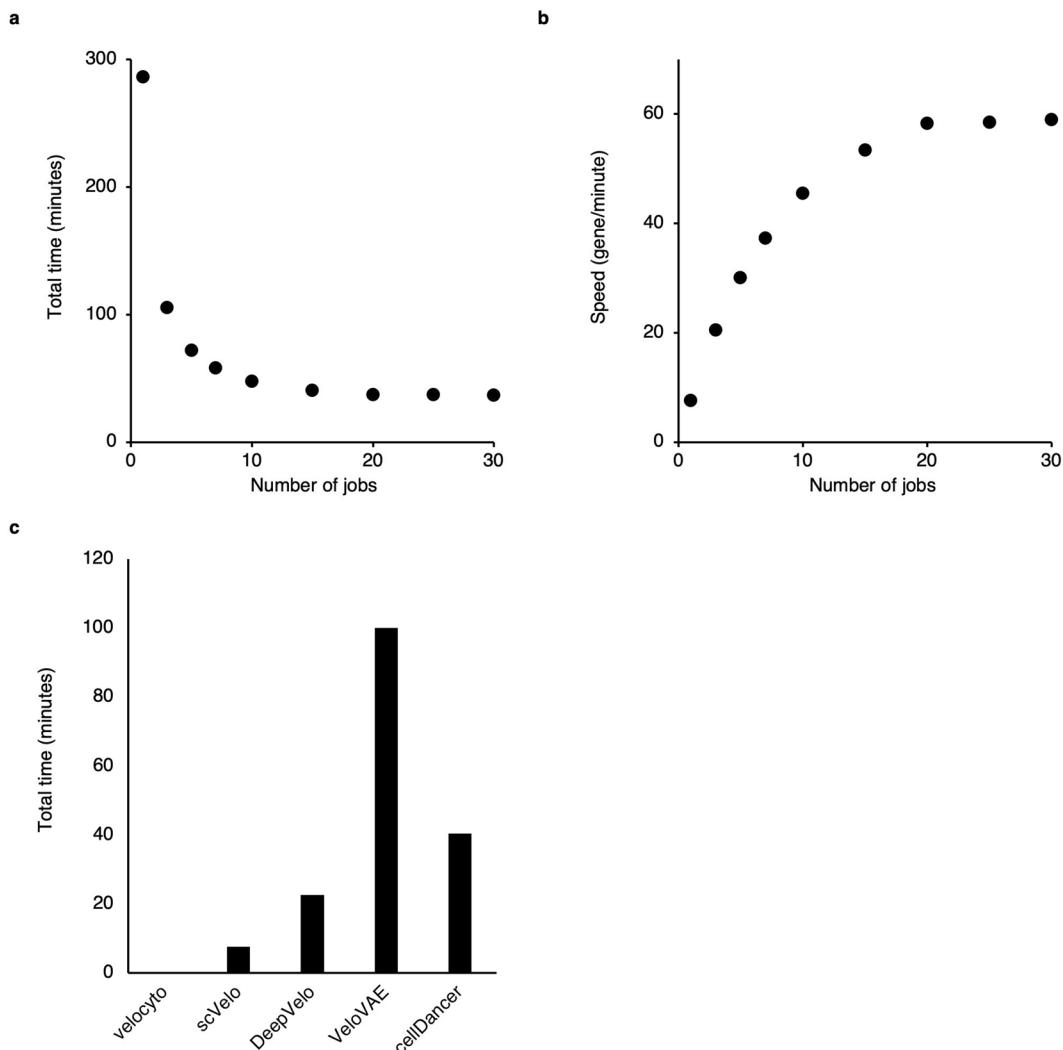
Extended Data Fig. 8 | Robustness under different dropout ratios and number of cells. (a) Heatmaps show the overview of the simulated genes with a dropout of 70% on both unspliced and spliced reads. We simulated raw mRNA counts using a Poisson distribution to obtain the unspliced and spliced abundances with 50%, 60%, and 70% technical zeros. (b) Scatter plot shows a high correlation between the simulated (background truth) and the predicted α/β (top) and α/γ (bottom) under different dropout ratios of the spliced and

unspliced reads. The dropout was applied to both spliced and unspliced reads. The Pearson correlation coefficients between the prediction and the simulation are computed. The Pearson correlation coefficient in data with different dropout rates is larger than 0.96 and 0.84 for α/β and α/γ , respectively. (c) The predicted α/β (top) and α/γ (bottom) are plotted against numbers of cells on the simulated mono-kinetic, multi-lineage forward branching, transcriptional boost, and multi-lineage backward branching genes.



Extended Data Fig. 9 | Sensitivity of the stopping criteria for the DNN training. The velocities derived by cellDancer with different combinations of stopping criteria parameters. The ‘check every n epoch’ means the number of epochs to skip (or a checkpoint) when computing the loss function. cellDancer

calculates the loss of DNN every several epochs, which is specified by the checkpoint. The patience means the number of checkpoints waited before stopping the training when the loss score doesn’t decrease.



Extended Data Fig. 10 | The speedup of cellDancer. (a, b) Scatter plots showing the total time (a) and the training speed (b) of cellDancer when applying multiprocessing. We tested the parallel speedup ratio of cellDancer by increasing job numbers from 1 to 30. We applied full analysis of cellDancer to 2,159 genes in 18,140 cells (the hippocampal dentate gyrus neurogenesis dataset) with the default parameters and calculated the runtime and speed of different job numbers. The evaluation of all the algorithms and the speedup ratio analysis was performed on a 2.7 GHz 24-Core Intel Xeon W processor. Total runtime decreases from 286 to 36 minutes when adding job numbers from 1 to 30 and reaches saturation at 15 jobs with 40 minutes. cellDancer has a feasible runtime

of 53 genes per minute using 15 jobs. The training speed (number of genes per unit time) increases with the number of jobs. (c) Bar plot showing the total time of the comparison between velocyto, scVelo, DeepVelo, VeloVAE, and cellDancer. We compared the runtime of cellDancer with velocyto, scVelo, DeepVelo, and VeloVAE. The benchmark is based on 18,140 cells and 2,159 genes in the hippocampal dentate gyrus neurogenesis dataset with the default parameters. We set the number of jobs (threads) to 15 for scVelo, DeepVelo, VeloVAE, and cellDancer. cellDancer shows a comparable running time with the other two deep learning algorithms.

Reporting Summary

Nature Portfolio 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 Portfolio policies, see our [Editorial Policies](#) 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	No software was used for data collection.
Data analysis	<p>The main functions of our study are open-source and could be obtained through https://github.com/GuangyuWangLab/cellDancer. The comparison between our algorithm and other algorithms was performed by using cellDancer (v1.1.3), scVelo (v0.2.4), DeepVelo (v0.2.5rc1), VeloVAE (v0.1.0; under commit number 018ee60998c9b63f441fdfc4836dbd5baadf1459; https://github.com/welch-lab/VeloVAE). The GO enrichment analysis was performed by DAVID (v2021.Dec).</p> <p>The analysis of Rates-based UMAP and Expression-based UMAP of scEU-seq was based on SCANPY (v1.8.2).</p> <p>The network and Adam optimizer of our software were implemented based on PyTorch Lightning (v1.5.2).</p> <p>The Runge-Kutta method we used to solve differential equations in our simulation was performed by SciPy (v1.7.2).</p>

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 and reviewers. We strongly encourage code deposition in a community repository (e.g. GitHub). See the Nature Portfolio [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 description of any restrictions on data availability
- For clinical datasets or third party data, please ensure that the statement adheres to our [policy](#)

All the scRNA-seq raw data are publicly accessible. The pancreatic endocrinogenesis data can be extracted using scVelo's CLI: scvelo.datasets.pancreas() or accessed from the original work under accession number GSE132188 of Gene Expression Omnibus (GEO) repository. The hippocampal dentate gyrus neurogenesis data can be accessed from the link <http://pklab.med.harvard.edu/velocyto/DentateGyrus/DentateGyrus.loom> or the original paper under accession number GSE95753 of GEO. The erythroid lineage of mouse gastrulation data can be extracted using scVelo's CLI: scvelo.datasets.gastrulation() or from the original work under accession number E-MTAB-6967 of ArrayExpress. Human embryo glutamatergic neurogenesis can be accessed from https://github.com/pachterlab/GFCP_2022/blob/main/notebooks/data/hgForebrainGlut.loom or from the original work under accession code of Sequence Read Archive (SRA) SRP129388. Cell cycle progression in REP1-FUCCI cells can be extracted using dynamo's CLI: dyn.sample_data.scEU_seq_rpe1() or from the original work under accession number GSE128365 of GEO.

Human research participants

Policy information about [studies involving human research participants and Sex and Gender in Research](#).

Reporting on sex and gender No human research participants in this study.

Population characteristics No human research participants in this study.

Recruitment No human research participants in this study.

Ethics oversight No human research participants in this study.

Note that full information on the approval of the study protocol must also be provided in the manuscript.

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 Sample sizes were determined in originally published publications. Mouse gastrulation erythropoiesis data: n=12,329 cells; mouse hippocampus development data: n=18,140 cells; mouse pancreas endocrine development data: n=3,696 cells; scEU-seq data: n=3,058 cells; human embryonic glutamatergic neurogenesis data: n=1,054 cells.

Data exclusions No data were excluded.

Replication No experiments in this study.

Randomization No experiments in this study.

Blinding No experiments in this study.

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
<input checked="" type="checkbox"/>	Antibodies
<input checked="" type="checkbox"/>	Eukaryotic cell lines
<input checked="" type="checkbox"/>	Palaeontology and archaeology
<input checked="" type="checkbox"/>	Animals and other organisms
<input checked="" type="checkbox"/>	Clinical data
<input checked="" type="checkbox"/>	Dual use research of concern

Methods

n/a	Involved in the study
<input checked="" type="checkbox"/>	ChIP-seq
<input checked="" type="checkbox"/>	Flow cytometry
<input checked="" type="checkbox"/>	MRI-based neuroimaging