TSvelo: Comprehensive RNA velocity inference by jointly modeling Transcription and Splicing

Jiachen Li[1], Zhe Wang[1], Hong-Bin Shen[2], Ye Yuan[1,2]*
[1]Key Laboratory of Biopharmaceutical Preparation and Delivery, State Key Laboratory of Biochemical Engineering, Institute of Process Engineering, Chinese Academy of Sciences, Beijing 100190, China
[2]Institute of Image Processing and Pattern Recognition, Shanghai Jiao Tong University, and Key Laboratory of System Control and Information Processing, Ministry of Education of China, Shanghai, 200240, China
*Address correspondence to Y. Y. at: yyuan@ipe.ac.cn or yuanye_auto@sjtu.edu.cn

Abstract

RNA velocity approaches fit gene dynamics and infer cell fate by modeling the splicing process using single-cell RNA sequencing (scRNA-seq) data. However, due to the high noise in the unspliced/spliced data for individual genes, existing RNA velocity models often fail to accurately capture the dynamics between unspliced and spliced abundance in the phase portraits of many genes. In addition, optimizing the high dimensional dynamics of all genes within a global framework and effectively handling multi-lineage data still require further exploration in RNA velocity modeling. Inspired by the role of transcriptional regulation in linking genes into complex networks and providing additional context for gene dynamics modeling, we propose TSvelo. TSvelo integrates cross-gene transcriptional regulation and gene-specific splicing processes of all genes into one neural Ordinary Differential Equation (ODE)-based framework. By incorporating cell-specific transcriptional rates with the unspliced-spliced data, TSvelo precisely models the dynamics of velocity genes, learns global pseudotime, and can be extended to multi-branch datasets. Experiments on six scRNA-seq datasets, including two multi-branch datasets, demonstrate that TSvelo outperforms existing methods in modeling gene dynamics, inferring cell pseudotime and trajectories, and analyzing gene expression patterns across branches.

Introduction

Single-cell RNA sequencing (scRNA-seq) enables the detailed exploration of gene expression at the individual cell level. To move beyond static snapshots and capture the dynamic behavior of cells over time, several trajectory inference (TI) methods have been developed, such as PAGA[1], Monocle[2], Slingshot[3] and Palantir[4]. These methods typically estimate pseudotime by relying on diffusion processes, which require prior annotation of initial cells[5] and may not adequately capture gene-specific dynamics. In contrast, RNA velocity[6] offers a more mechanistic approach by modeling the time derivative of gene expression, linking unspliced (immature) and spliced (mature)

mRNA levels through Ordinary Differential Equations (ODEs). By leveraging velocities from multiple genes, RNA velocity enables the estimation of pseudotime and cell fate without the need for initial cell annotations.

Several RNA velocity methods have been proposed to capture splicing dynamics. The first approach, Velocyto[6], uses least squares solutions to estimate parameters under the assumption of steady-state kinetics. ScVelo[7] improves upon this by employing an Expectation-Maximization (EM) approach for better fitting splicing kinetics. UniTVelo[8] propose a top-down idea that directly models the spliced RNA levels with a time-dependent function. More recently, generative models such as VeloVI[9], veloVAE[10] and Pyrovelocity[11] have been introduced, utilizing Bayesian frameworks to estimate RNA velocity. To handle multi-branch datasets, methods like CellDancer[12], DeepVelo[13] and LatentVelo[14] extend RNA velocity by modeling local dynamics with neural networks, rather than assuming a globally consistent transcriptional rate, which allows for more accurate representation of gene expression variations across different regions. Apart from using unspliced/spliced data, Dynamo[15] enhances RNA velocity further by incorporating labeled RNA-seq data. The advent of single-cell multi-omics[16,17] technologies has allowed RNA velocity analysis to extend to protein abundance (e.g., protaccel[18]) and single-cell ATAC-seq datasets (e.g., MultiVelo[19]). Additionally, methods like DeepCycle[20] and VeloCycle[21] have been developed to focus on cell cycle processes, with specialized modules for capturing periodic signals. STT[22] and SIRV[23] proposed the idea on extending RNA velocity analysis to spatial transcriptomics.

Although RNA velocity theory has significantly advanced the inference of single-cell trajectories, pseudotime, and gene regulation[24], several challenges persist for current RNA velocity models. Firstly, RNA velocity models primarily infer cell fate based on phase portrait fitting of unspliced and spliced abundance for each gene. However, they often fail to accurately capture the splicing dynamics in phase portrait for many genes. This may result from insufficient signal from mRNA splicing dynamics at single-cell resolution[25]. The sparsity and noise in unspliced and spliced mRNA abundance for individual genes, combined with the short time scale of the splicing process, often lead to the mixing of cells from different types on the phase portrait. This complicates the identification of consistent patterns between unspliced and spliced RNA, hindering accurate inference of gene dynamics[26-28]. Secondly, existing RNA velocity models mainly treat each gene independently and fail to incorporate the underlying regulatory interactions[25]. TFvelo[28] attempts to address these issues by modeling gene dynamics based on transcriptional regulation rather than splicing information. Additionally, several computational methods, such as PHOENIX[29] and scKINETICS[30], construct ODE models to capture gene dynamics by incorporating gene-gene transcriptional regulation. However, these methods overlook the valuable splicing data, which can be crucial for determining the directionality of the trajectory. Thirdly, multi-branch tasks remain a significant challenge for current RNA velocity models, despite the development of some approaches to focus on this issue. Classical RNA velocity

approaches, such as scVelo, use interpretable parameters in constructing dynamic models, which helps mitigate overfitting. By comparison, some recent methods that model local transcriptional rates often employ latent space embeddings or neural network-based encoders[12]. While these advanced techniques offer flexibility and improved fitting performance, it makes the model parameters less interpretable in gene-level, which is crucial for understanding the underlying biological mechanisms.

To address the challenges outlined above, we propose TSvelo, a method that integrates the transcription and splicing dynamics of all genes into a single ODE model, whose parameters are highly interpretable. TSvelo directly learns the global pseudotime without the need to separately model the dynamics of each gene or learn gene-specific latent times. By leveraging both unspliced and spliced scRNA-seq data, along with gene regulatory knowledge from TF-target databases, TSvelo iteratively optimizes the ODE model and the global pseudotime using the Expectation-Maximization (EM) algorithm. Experiments on six scRNA-seq datasets, including two multi-branch datasets, demonstrate that TSvelo outperforms existing methods in modeling gene dynamics, inferring cell pseudotime and trajectories and be effective in analyzing gene expression patterns across branches.

Results

Estimate RNA velocity with TSVelo

We present TSvelo, a method for estimating RNA velocity of a target gene by incorporating both cross-gene transcriptional relationships and gene-specific splicing processes.

Initially, the unspliced and spliced RNA abundances are preprocessed with multiple steps, which includes highly variable genes (HVGs) selection, normalization, log transformation, k-nearest neighbor (KNN) smoothing, and clustering. TSvelo aims to enhance the robustness of RNA velocity inference by introducing several strategies to ensure the reliability of intermediate results at each step, including velocity gene filtering and pseudotime initialization. Previous studies[7] have demonstrated that only a subset of genes, termed "velocity genes," can be accurately fitted in the unspliced-spliced phase portrait by RNA velocity models. Inspired by the notion that velocity genes provide high-quality data for dynamic modeling in the phase portrait, we propose to select velocity genes during the preprocessing step. The selection is based on the similarity between cell-cell neighborhood relationships in UMAP space and those observed in gene-specific unspliced-spliced phase portraits. Subsequently, leveraging the fact that unspliced RNA precedes spliced RNA in the expression pattern[31], we identify the initial cluster and apply a trajectory inference tool for pseudotime initialization (**Fig. 1a**). The velocity genes selection and pseudotime initialization could improve both the accuracy and efficiency of the downstream model optimization, which is based on the expectation-maximization (EM) and neural ODE model.

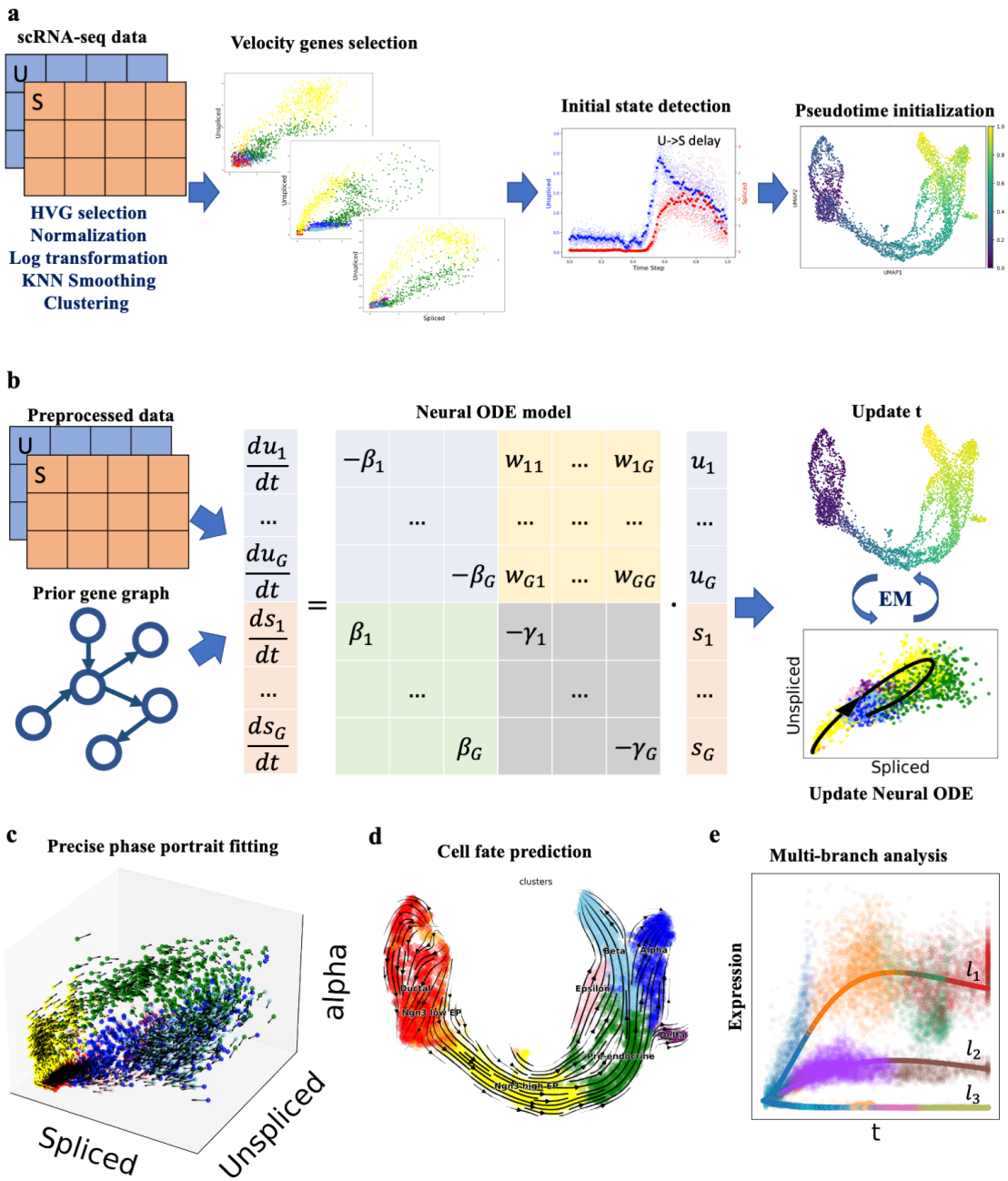

Fig. 1. **The framework of TSvelo.** (**a**) The preprocessing strategy in TSvelo, including velocity genes selection, initial state detection and pseudotime initialization. (**b**) The Neural ODE model and its optimization in TSvelo. (**c-e**) The downstream application tasks of TSvelo, including precise phase portrait fitting (c), cell fate prediction (d) and gene expression pattern analysis for multi-branch dataset (e).

For modeling the gene expression dynamics, suppose u_g and s_g are the abundance of unspliced and spliced RNA, and $\alpha_g(t), \beta_g, \gamma_g$ are the transcription rate, splicing rate and degradation rate. The dynamics of gene g is modeled as:

$$\frac{du_g(t)}{dt} = \alpha_g(t) - \beta_g u_g(t) \tag{1}$$

$$\frac{ds_g(t)}{dt} = \beta_g u_g(t) - \gamma_g s_g(t) \tag{2}$$

We assume that the gene and cell-specific transcriptional rate $\alpha_g(t)$ is influence by the expression of Transcriptional Factors (TFs). Considering the wide usage of linear models for gene-relations in previous studies[28,32-35], we model $\alpha_g(t)$ as

$$\alpha_g(t) = ReLU(\sum_{i \in TFs(g)} w_{gi} * s_i(t)) \tag{3}$$

$TFs(g)$ refers to the TFs that could regulated the target gene g, which is selected according to the ChEA and ENCODE TF-target database. Then, we can combine these dynamics function on each gene into a global Ordinary Differentia Equation (ODE) model (**Fig. 1b**). In order to include more TFs in the model, we reserve those TFs that are not selected as velocity genes, and directly model the dynamics between transcription to mature RNA on them. More details are provided in **Methods**.

TSvelo employs an EM framework to iteratively optimize both pseudotime t and the parameters in ODE. The global pseudotime t can be directly optimized through grid search, eliminating the need for model optimization for each individual gene or the learning of gene-specific time points. Because the analytical solutions of $u_g(t)$ or $s_g(t)$ can not be obtained, TSvelo adopted a Neural ODE for optimizing B, Γ and W. Details are provided in **Methods**. Finally, TSvelo can accurately model the gene dynamics of both transcription and splicing processes (**Fig. 1c**). TSvelo predicts cell states using both pseudotime and velocity stream analysis (**Fig. 1d**), and is also applicable to multi-branch tasks for analyzing expression patterns across different lineages (**Fig. 1e**).

TSvelo can accurately model gene dynamics and cell fate on pancreas dataset

To validate the TSvelo model, we applied it to the pancreas scRNA-seq dataset[6], which is widely used in RNA velocity studies to model cell differentiation from ductal to endocrine cells. Here we provide details about the pseudotime initialization using the time delay between unspliced to spliced. During preprocessing, cell clustering is performed using the Leiden algorithm. Diffusion pseudotime (dpt) is then calculated through PAGA trajectory inference and defining a Leiden cluster as the initial state. The unspliced-to-spliced delay for each velocity gene is assessed by computing the correlation between the dpt-ordered unspliced and spliced abundances under varying time delays. By comparing the mean unspliced-to-spliced delay across genes for each Leiden cluster initialization, TSvelo can accurately identify the initial Leiden cluster and use the corresponding dpt for pseudotime initialization (**Fig. 2a**).

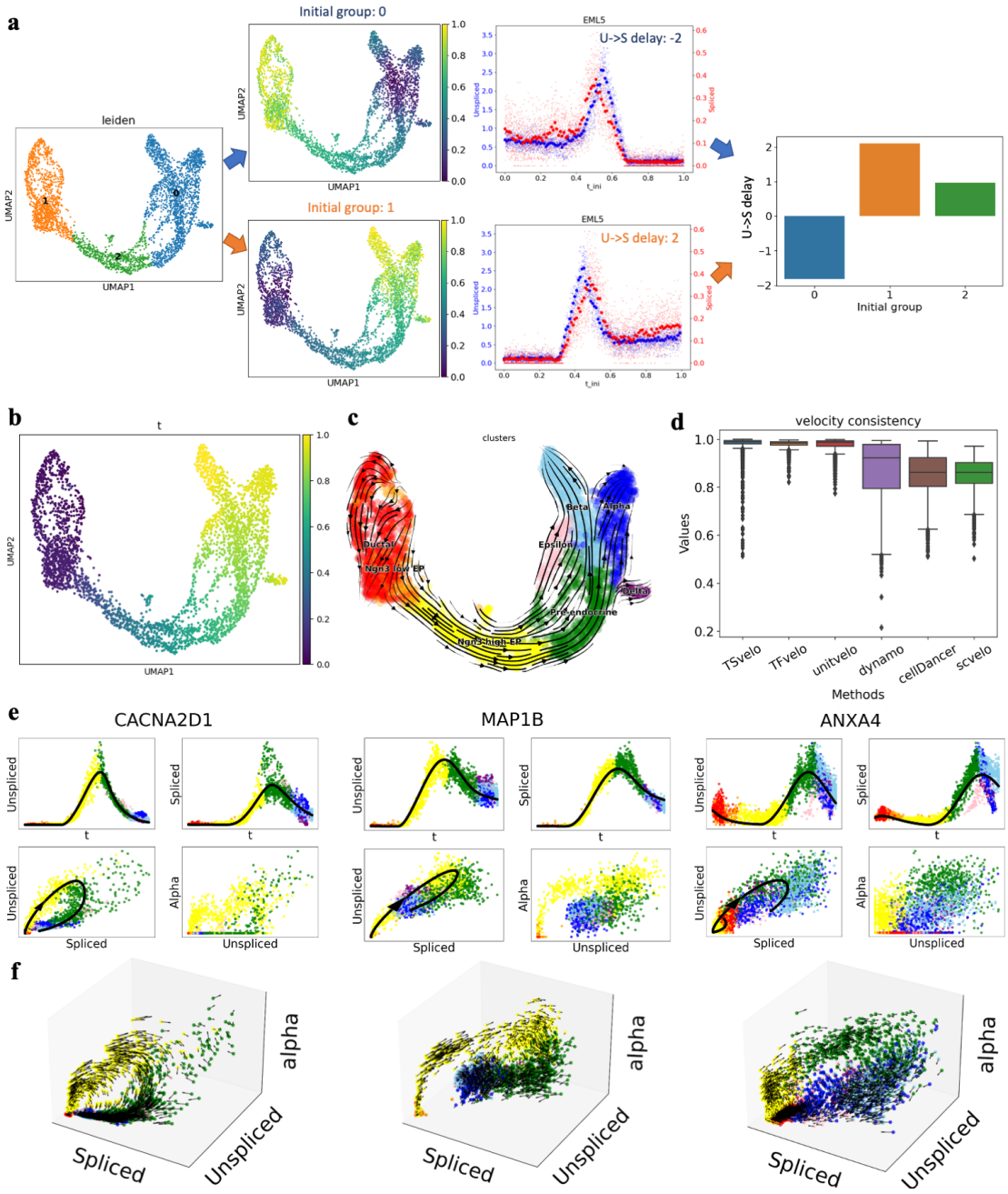

Fig 2. Results on pancreas dataset. (**a**) The initial Leiden group detection in preprocessing, which is based on the time delay between unspliced to spliced mRNA. (**b**) The pseudotime learned with TSvelo. (**c**) The Stream-plot for visualization the RNA velocity inferred by TSvelo. (**d**) The quantitative comparison between TSvelo and multiple baseline approaches. The down, central and up hinges correspond to the first quartile, median value and third quartile, respectively. The whiskers extend to 1.5× the interquartile range of the distribution from the hinge. 3696 samples are included in the boxplot for each method. (**e**) The dynamics fitting on CACNA2D1, MAP1B and ANXA4. For each gene, four plots are displayed in a 2x2 layout: the u-t, s-t, u-s, and alpha-u plots. (**f**) The dynamics fitting in alpha-u-s 3D phase portrait for CACNA2D1, MAP1B and ANXA4.

Using the initialized pseudotime, the TSvelo optimizes both the gene dynamics ODE model and the pseudotime t. Using the predicted RNA velocity, TSvelo can also generate velocity stream plot using the functions provided in scVelo. Both the final pseudotime t (**Fig. 2b**) and the velocity stream plot (**Fig. 2c**) effectively capture the cell differentiation process. We quantitively compare TSvelo and baseline approaches, including scVelo, dynamo, UniTVelo, cellDancer and TFvelo, using velocity consistency. TSvelo achieves the highest mean velocity consistency (**Fig. 2d**), which demonstrates that the high-dimensional velocity vectors learned by TSvelo are mostly coherent within neighbor cells.

Next, we analyze the dynamics fitting for individual genes and demonstrate how the TSvelo model aids gene dynamics fitting by incorporating both transcriptional and splicing information. Using CACNA2D1, MAP1B, and ANXA4 as examples, **Fig. 2e** and **Fig. 2f** present the results of the TSvelo model on these genes. For CACNA2D1, the unspliced-spliced phase portrait reveals an almond-shaped distribution, a characteristic assumption in many RNA velocity models. TSvelo accurately fits the data within this phase portrait, and also detects the time-delay pattern between transcription and unspliced abundance by showing another almond-shaped distribution in the alpha-unspliced portrait. For MAP1B, while the unspliced-spliced distribution predominantly follows the almond-shaped pattern, some cells, specifically Alpha cells (blue) and Beta cells (light blue), overlap with Ngn3 high EP cells (red), indicating that these cell types cannot be distinctly separated using only the 2D phase portrait. However, TSvelo can infer transcription representation from the expression of multiple transcription factors, enabling the model to distinguish these cell types within the 3D dynamics space of α-u-s. For ANXA4, the expression pattern exhibits an initial decrease followed by an increase, a dynamic that is not easily captured in the U-S phase portrait. Nevertheless, TSvelo can still accurately predicts the abundance of unspliced spliced RNA. The visualization of dynamics fitting on more genes are provided at **Fig. S2** in **Supplementary Information**

TSVelo can better predict RNA velocity on gastrulation erythroid dataset.

We applied TSvelo to the gastrulation erythroid dataset, which is derived from the transcriptional profile of mouse embryos [36] and has been used in previous RNA velocity studies. This dataset primarily describes the differentiation process from blood progenitors to erythroid cells. First, using Gene Ontology (GO) term enrichment analysis (**Fig. 3a**), we find that the velocity genes selected through TSvelo's preprocessing strategies are mostly enriched in heme biosynthetic process (GO:0006783) and porphyrin-containing compound biosynthetic process (GO:0006779), both of which are related to the erythropoiesis process.

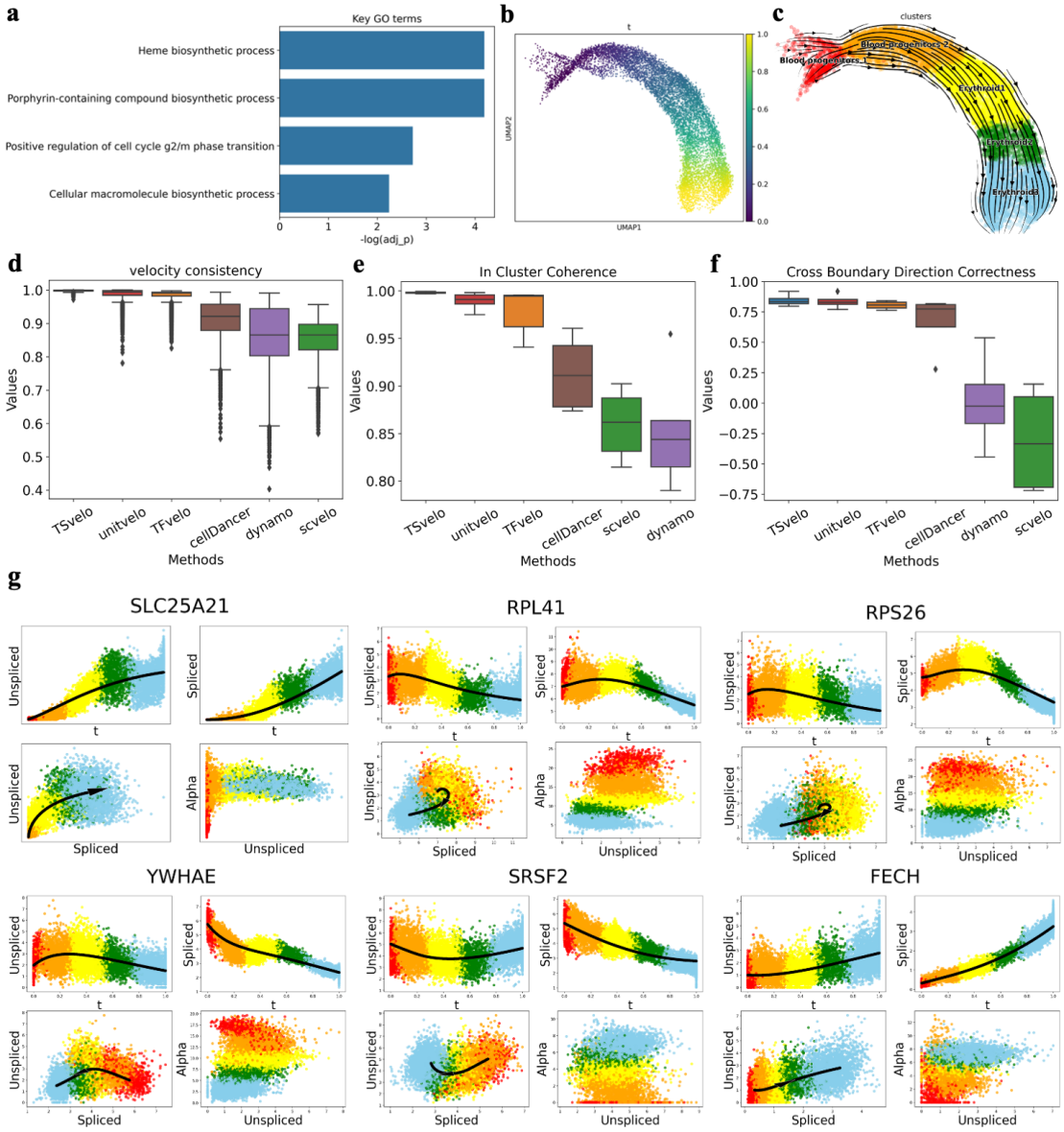

Fig 3. Results on gastrulation erythroid dataset. (**a**) The GO terms which are mostly enriched in the selected velocity genes of TSvelo. (**b**) The pseudotime learned with TSvelo. (**c**) The Stream-plot for visualization the RNA velocity inferred by TSvelo. (**d-f**) The quantitative comparison between TSvelo and multiple baseline approaches in terms of velocity consistency (d), in-cluster coherence (e), and cross boundary direction correctness (f). The down, central and up hinges correspond to the first quartile, median value and third quartile, respectively. The whiskers extend to 1.5× the interquartile range of the distribution from the hinge. 3696 samples are included in the boxplot for each method. (**g**) The dynamics fitting on multiple genes, including SLC25A21, RPL41, RPS26, YWHAE, SRSF2 and FECH. For each gene, four plots are displayed in a 2x2 layout: the u-t, s-t, u-s, and alpha-u plots, where u, a and alpha mean the abundance of unspliced mRNA, the abundance of spliced mRNA, and the learned transcriptional representation, respectively.

After applying TSvelo, both the inferred pseudotime (**Fig. 3b**) and the velocity stream plot (**Fig. 3c**) accurately reflect the cell differentiation process. We also compared TSvelo with previous approaches using quantitative metrics. Since the edges between cell types in this dataset are well-defined, we evaluated the model using not only velocity consistency, but also in-cluster coherence and cross-boundary correctness, which are based on the ground truth annotations of cell types and their orientation. The results, shown in **Fig. 3d-f**, demonstrate that TSvelo achieves the highest velocity consistency, the highest in-cluster coherence, and the highest cross-boundary correctness. These findings suggest that the high-dimensional dynamics predicted by TSvelo most accurately capture the underlying biological processes.

We next analyze TSvelo modeling at the gene level and provided some examples in **Fig. 3g**. For the gene SLC25A21, TSvelo accurately captures the dynamics, which exhibit a simple and clear pattern on the phase portrait. We further observe that the learned transcriptional representation of SLC25A21 remains stable throughout the process, consistent with the assumptions of several previous approaches, such as scVelo. Actually, scVelo predicts a similar dynamic for SLC25A21. However, for genes exhibiting higher noise or more complex patterns, TSvelo demonstrates its advantage in dynamic fitting by integrating both transcriptional and splicing signals. For genes such as RPL41, RPS26, YWHAE, SRSF2, and FECH, the unspliced-spliced data is so noise that overlap cell types in phase portrait. TSvelo can still accurately captures the gene dynamics for both unspliced and spliced abundances and reveals differences in transcription rates across cell types. In contrast, methods that rely solely on unspliced-spliced data from individual genes, such as scVelo, fail to accurately fit these dynamics and do not identify them as velocity genes. This highlights TSvelo's robustness in handling more complex scenarios.

TSvelo can accurately fit cell fate and gene dynamics on mouse brain data.

We apply TSvelo to a 10x multi-omics dataset from the embryonic mouse brain, which includes both Assay for Transposase-Accessible Chromatin with sequencing (ATAC-Seq) [37] and scRNA-seq data. This dataset was previously used in the Multivelo study, which introduced an RNA velocity model designed for multi-omics datasets to capture the dynamics between chromatin accessibility and unspliced mRNA[20]. In this dataset, Radial Glia (RG) cells located in the outer subventricular zone give rise to neurons, astrocytes, and oligodendrocytes. During cortical development, neurons migrate in an inside-out manner, with younger cells moving toward the upper layers of the cortex, while older cells settle in the deeper layers. Additionally, RG cells are capable of producing intermediate progenitor cells (IPCs), which serve as neural stem cells and generate a variety of mature excitatory neurons within the cortical layers.

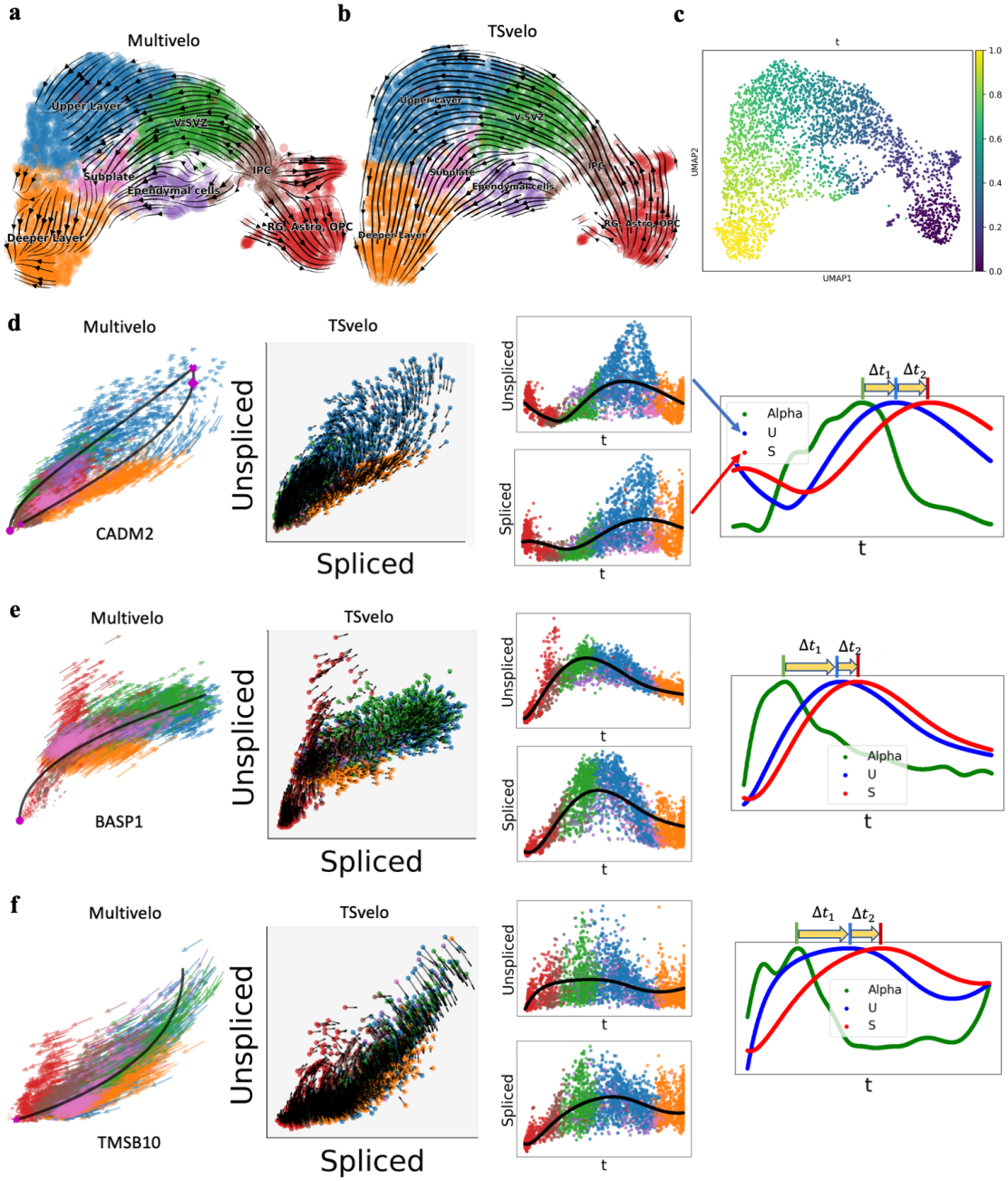

Fig 4. Results on mouse brain dataset. (**a**) The velocity stream inferred by Multivelo. (**b**) The Stream-plot for visualization the RNA velocity inferred by TSvelo. (**c**) The pseudotime learned with TSvelo. (**d-f**) The dynamics fitting on gene CADM2 (d), BASP1 (e) and YMSB10 (f) by both Multivelo and TSvelo. The far-right plot in each panel shows the dynamics of the learned transcriptional representation (in green), predicted unspliced abundance (in blue), and predicted spliced abundance (in red) along the pseudotime. Since the transcriptional representation is calculate for each individual cell, we apply a Generalized Additive Model (GAM) to transcriptional representation across cells along the pseudotime and present GAM-fitting results to better visualize its trends in these plots.

Multivelo is chosen as the baseline on this study, which also aimed at connecting the regulation signal into RNA velocity modeling by using 3D phase portrait fitting in ATAC-unspliced-spliced space. For a fair comparison, TSvelo uses the same data processed be Multivelo. As shown in **Fig. 4a** and **Fig. 4b**, the velocity stream obtained by TSvelo is smoother than Multivelo and can correctly reflect the differentiation from RG (in red) to IPCs (in brown). The pseudotime inferred by TSvelo is also coherent with the whole cell development process (**Fig. 4c**).

We next analyze the phase portrait fitting for further understanding the connection and difference between Multivelo and TSvelo. For genes such as CADM2 (**Fig. 4d**), both methods have similar results and can accurately model the gene dynamics. However, due to the high noise and sparsity of ATAC-seq data, Multivelo may struggle with genes where different cell types overlap significantly in the unspliced-spliced phase portrait. For example, for BASP1 (**Fig. 4e**) and TMSB10 (**Fig. 4f**), Multivelo incorrectly models their expression patterns as monotonically increasing and monotonically decreasing, respectively, failing to capture the true dynamics, which involve initial upregulation followed by downregulation. In contrast, TSvelo successfully captures the correct gene dynamics and identifies the delay from transcriptional signals to gene expression. We also show the learned transcriptional rate α as well as the predicted dynamics of unspliced and spliced RNA along pseudotime t, which clearly illustrate the time delays between trancription to unspliced and unspliced to spliced RNAs. The visualization of dynamics fitting on more genes are provided at **Fig. S3** in **Supplementary Information**. These results suggest that the transcription should not only be modeled with ATAC, but also be modeled with the TFs abundance.

TSvelo can accurately infer cell fate and model lineage-specific gene dynamics for multi-branch task.

Given that multi-branch differentiation is a common phenomenon in many scRNA-seq datasets, developing the RNA velocity model that can handle such complexity is essential. In TSvelo, the lineages are effectively separated during the preprocessing steps. To demonstrate its application, we apply TSvelo to a multi-branch dentate gyrus scRNA-seq dataset, which captures the differentiation process from neural blast cells to various cell types[38] and initially be introduced to RNA velocity studies by cellDancer. During preprocessing, the velocity genes selected by TSvelo are enriched in GO terms related to neural development, such as axonogenesis (GO:0007409), axon guidance (GO:0007411) and axon development (GO:0061564) (**Fig. 5a**), which aligns with the biological processes described by the dataset.

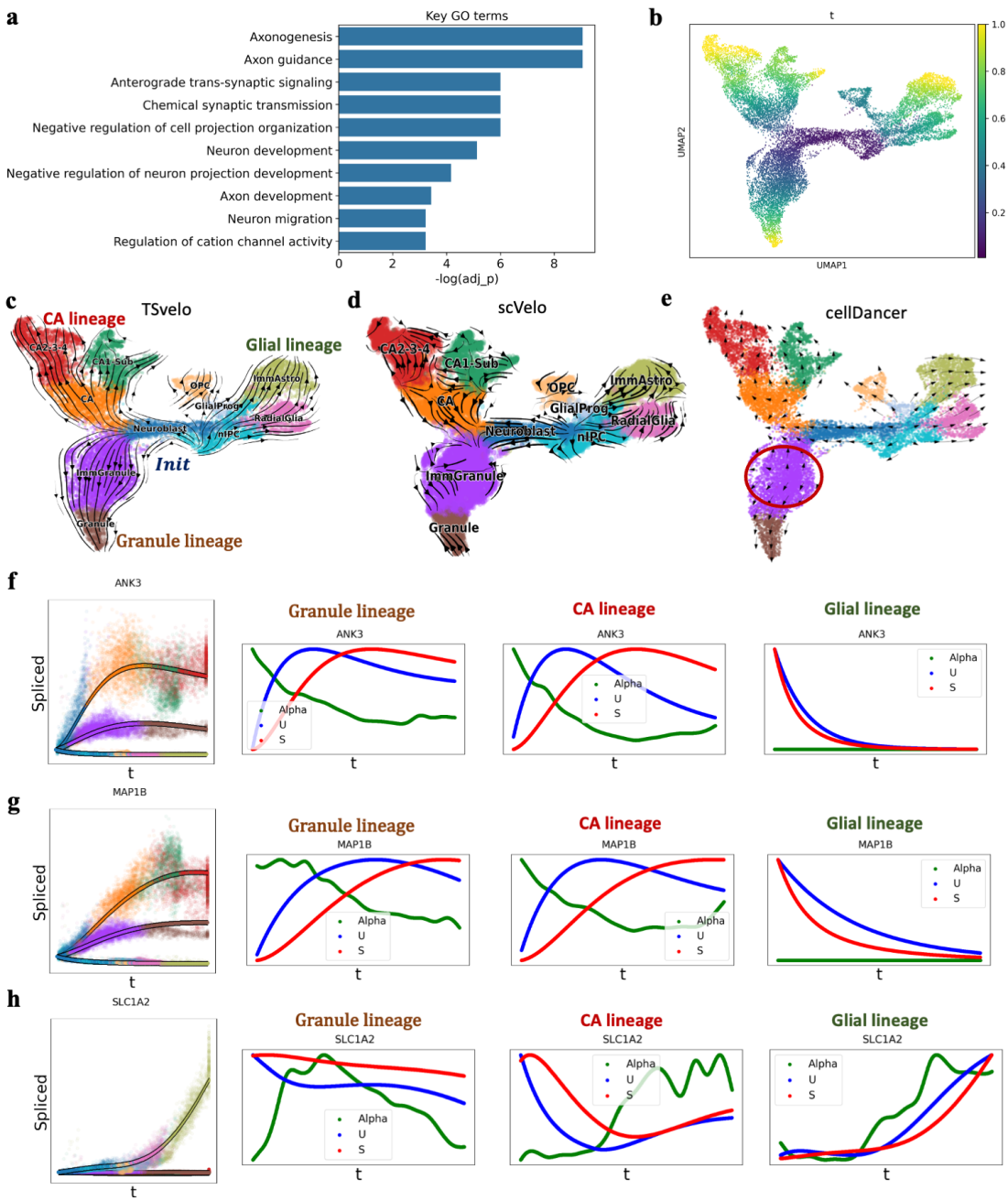

Fig 5. Results on the multi-branch dentate gyrus dataset. (**a**) The GO terms enriched in the selected velocity genes of TSvelo. (**b**) The pseudotime learned with TSvelo. (**c**) The Stream-plot for visualization the RNA velocity inferred by TSvelo. Three lineages are detected, which are Granule lineage, CA lineage and glial lineage. (**d**) The velocity stream inferred by scVelo. (**e**) The velocity stream inferred by cellDancer. (**f-h**) The dynamics modeling of TSvelo on three axonogenesis-related genes, ANK3 (f), MAP1B (g) and SLC1A2 (h). In the far-left plot of each panel, the lines represent the predicted spliced abundance across all lineages, with the color indicating the cell types most strongly associated with each pseudotime point along the corresponding lineage. Additionally, expression data for each branch are shown as translucent points. The remaining plots in each panel display the dynamics of the learned transcriptional representation (in green), predicted unspliced abundance (in blue), and predicted

spliced abundance (in red) along pseudotime for each branch. The transcriptional representation in these plots is also processed using GAM fitting.

TSvelo could correctly identify the initial cell group and detect three lineages from this data. Details about the lineage segmentation are provided at **Fig. S4** in **Supplementary Information**. After applying the TSvelo model to each lineage and combine the results, the inferred pseudotime and velocity stream plots (**Fig. 5b** and **Fig. 5c**) align with the ground truth differentiation trajectory, which begins with neural blast cells. We also compare TSvelo to baseline methods, including scVelo and cellDancer. Notably, cellDancer is designed to handle such multi-branch data by modeling cell- and gene-specific transcriptional rates, splicing rates, and degradation rates using neural networks. As shown in **Fig. 5d** and **Fig. 5e**, scVelo struggles with this multi-branch dataset, and cellDancer fails to correctly capture the trajectory of immature granule cells (colored in purple).

Next, we analyze gene expression patterns across different lineages. TSvelo provides inferred dynamics along each lineage, revealing that the expression of ANK3 in the granule and Cornu Ammonis (CA) lineages follows an increasing and then decreasing pattern. In contrast, the expression of ANK3 in the glial lineage decreases to zero (**Fig. 5f**). The similar pattern is also observed for other genes, such as MAP1B (**Fig. 5g**). Both ANK3 and MAP1B are associated with the GO terms axonogenesis and axon guidance. This observation is consistent with the fact that the granule and CA lineages predominantly consist of neurons, where axons are a critical component. In contrast, glial cells, such as astrocytes, typically lack axons, providing a potential explanation for these expression patterns. SLC1A2 exhibits a distinct expression pattern, showing a significant increase in the glial lineage and a decrease in the granule and CA lineages. This pattern aligns with the observation that astrocytes are the primary cell type expressing SLC1A2[39]. Details about dynamics fitting for genes on each lineage are provided at **Fig. S5** in **Supplementary Information**.

TSvelo can accurately infer cell fate and model lineage-specific gene dynamics on Larry dataset

The Lineage and RNA Recovery (LARRY) method utilizes barcoded hematopoietic cells to trace both cell lineage and gene expression over time. LARRY has been successfully employed to track the in vitro differentiation of human blood cells, accurately capturing lineage trajectories and cell fates[40].

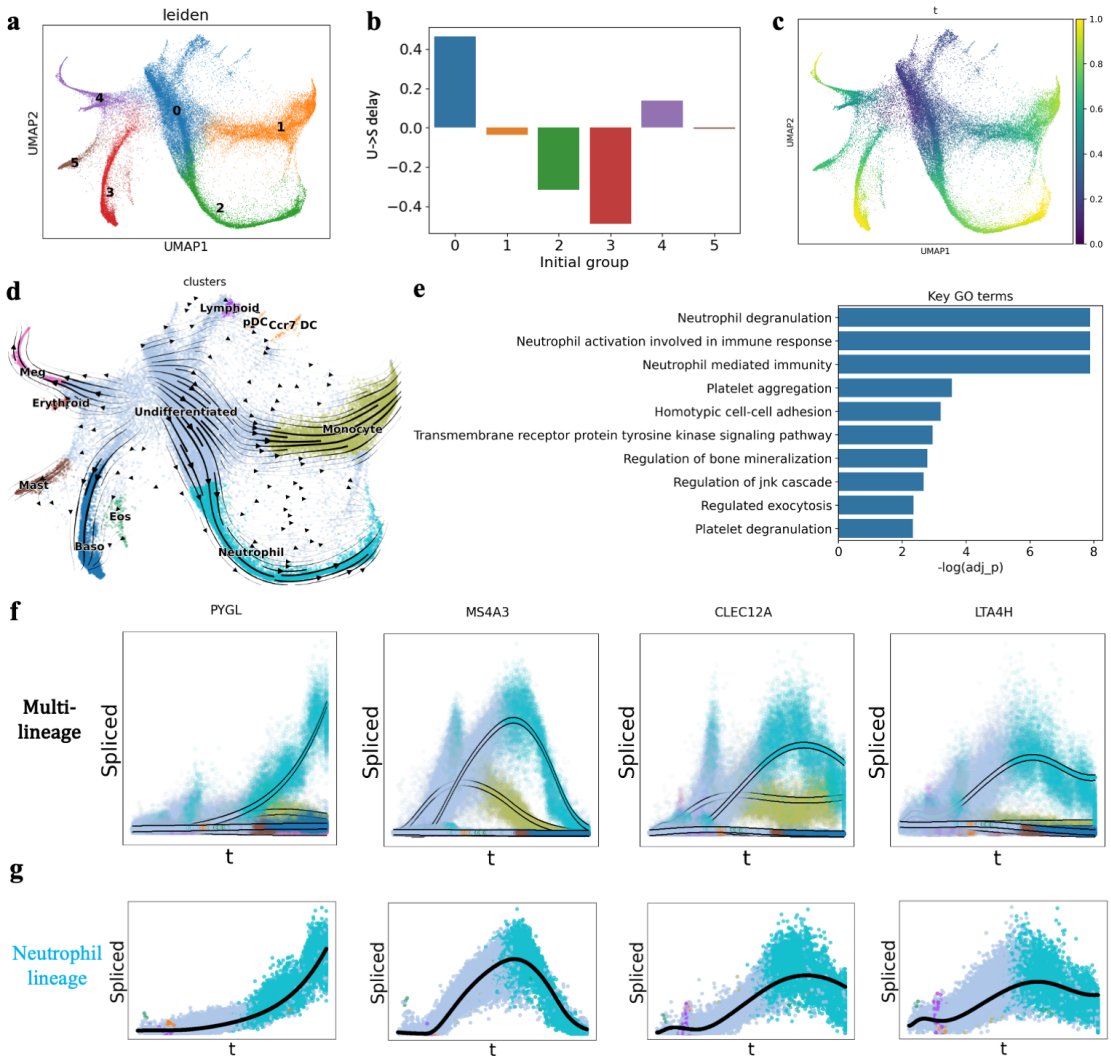

Fig 6. Results on the LARRY dataset. (**a**) The Leiden clustering on LARRY. (**b**) The initial Leiden cluster detection in preprocessing, which is based on the time delay between unspliced to spliced mRNA. (**c**) The pseudotime learned with TSvelo. (**d**) The Stream-plot for visualization the RNA velocity inferred by TSvelo. (**e**) The GO terms enriched in the selected velocity genes of TSvelo. (**f**) The dynamics modeling of TSvelo on four genes related to neutrophil development, which are PYGL, MS4A3, CLEC12A and LTA4H. The lines represent the predicted spliced abundance across all lineages, with the color indicating the cell types most strongly associated with each pseudotime point along the corresponding lineage. On the second row, we also add the expression data on each branch as translucent points into the plots, for visualizing more details about model fitting. **(g)** The dynamics of the learned transcriptional representation (in green), predicted unspliced abundance (in blue), and predicted spliced abundance (in red) along pseudotime for each branch. The transcriptional representation in these plots is also processed using GAM fitting.

On this LARRY dataset which encompass a total of 49,302 cells on multiple lineages, TSvelo effectively detects the initial Leiden clusters based on the unspliced-to-spliced

delay (**Fig. 6a** and **Fig. 6b**) and separate its lineages. Furthermore, **Fig. 6c** and **Fig. 6d** show that the pseudotime and velocity stream inferred by TSvelo accurately capture the differentiation process, progressing from undifferentiated cells to distinct cell fates. The velocity genes selected by TSvelo are significantly enriched in processes related to neutrophil biology, such as neutrophil degranulation (GO:0043312), neutrophil activation in immune response (GO:0002283), and neutrophil-mediated immunity (GO:0002446).

We further analyzed the expression patterns of genes associated with neutrophil degranulation, focusing specifically on the neutrophil lineage. Using four representative genes as examples (**Fig. 6f,g**), TSvelo can help to observe the significant variation in the expression patterns across these genes. PYGL, which has been widely reported as a key gene in neutrophil[41], is almost exclusively expressed in neutrophil cells and exhibits a strictly increasing expression pattern during neutrophil differentiation. In contrast, MS4A3 was examined across all branches, and using TSvelo, we observed that its expression initially increases and then decreases in both the neutrophil and monocyte branches. This pattern is consistent with prior studies identifying MS4A3 as a gene specifically expressed by granulocyte-monocyte progenitors[42]. Similarly, CLEC12A, another gene associated with neutrophil degranulation, shows a comparable expression trajectory during neutrophil differentiation. Previous research has indicated that CLEC12A expression is highest in granulocyte-macrophage progenitors[43]. LTA4H exhibits a pattern similar to that of CLEC12A, suggesting it may play a significant role in the early stages of neutrophil differentiation. These findings further confirm the utility of TSvelo for gene-level analysis in multi-branch differentiation datasets.

Discussion

RNA velocity, utilizing unspliced/spliced data, has become a widely adopted concept for predicting cell fate and modeling gene dynamics. In our view, the key advantages of RNA velocity models include: the ability to infer cell fate without requiring initial state annotations, and the provision of a physically-inspired gene-level model with high interpretability. While many RNA velocity models have been proposed in recent years, they are fundamentally based on phase portrait fitting in the unspliced-spliced space. However, due to the limited information and high noise inherent in the splicing data of individual genes, many genes cannot be well-fitted by these models.

TSvelo addresses these limitations by introducing a strategy that models both transcription and splicing dynamics using an ODE framework, allowing for precise fitting of gene dynamics. Additionally, TSvelo is designed to handle multi-branch datasets, extending its applicability to more complex tasks. Results on six scRNA-seq datasets demonstrate that TSvelo is a valuable approach for RNA velocity modeling (results on pons dataset are provided in **Fig. S1** in **Supplementary Information**).

Looking ahead, TSvelo could be further improved by more accurately modeling the splicing process. Currently, TSvelo assumes a constant gene-specific splicing rate. Since splicing factors[44] may play a significant role and also regulate other genes expression, relaxing this assumption could enhance the model's flexibility in the future.

Methods

Preprocessing for scRNA-seq data

Following the data preprocessing procedures outlined in scVelo[7], we first select the top 2,000 highly variable genes (HVGs) and normalize their expression profiles by dividing by the total counts in each cell. A nearest-neighbor graph (with 30 neighbors by default) was constructed based on Euclidean distances in principal component analysis space (with 30 principal components by default) on log-transformed gene expression data. Subsequently, we compute the first- and second-order moments (mean and uncentered variance) for each cell across its nearest neighbors. These steps are performed by using scvelo.pp.filter_and_normalize() and scvelo.pp.moments().

Acquiring Prior Knowledge of Gene Regulatory Relations

To construct a prior gene network for the selected HVGs, we utilize TF-target annotations from the ENCODE[45] and ChEA[46] databases. If a regulatory interaction between a transcription factor (TF) and its target gene is identified in either of these two databases, the TF is included in the set of regulators for modeling the dynamic behavior of the target gene.

Velocity genes selection

The velocity genes are selected based on the premise that they exhibit clear dynamics in the unspliced-spliced phase portrait, which is necessary for precise modeling splicing dynamics. To identify these genes, we first compute the cell-cell neighborhood graph, $Graph$, using scvelo.pp.neighbors() on all highly variable genes (HVGs). Next, we construct an anndata object $adata_g$ for each gene, which contains only the unspliced and spliced expression data of that gene. We then apply scvelo.pp.neighbors() to each $adata_g$ to calculate the gene-specific neighborhood graph, denoted as $Graph_g$. Subsequently, we compute the similarity between $Graph$ and each $Graph_g$. The top 100 genes with the highest similarity are selected as velocity genes. These genes are characterized by phase portraits that exhibit a structure similar to that of the UMAP space, thereby enhancing the separation of cells from different types. As a result, the splicing dynamics of these velocity genes are more likely to be captured by RNA velocity models.

lineages segmentation and pseudotime initialization

Based on the normalized gene expression data, we perform Leiden clustering using scanpy.tl.leiden() with a low resolution (default resolution = 0.1). These Leiden clusters are utilized to identify lineages and determine the differentiation direction for each lineage. Subsequently, we apply PAGA (scanpy.tl.paga()) to assess the connectivity between Leiden groups.

By setting one Leiden group as the initial state, we infer pseudotime using diffusion pseudotime (DPT) with scanpy.tl.dpt(). To detect lineages, we start from the selected initial Leiden cluster and filter the PAGA graph by applying a threshold (default = 0.02). Edges with weights below this threshold are removed, leaving only the shortest path from the initial state to each group. The remaining paths are considered as the detected lineages. Given a lineage, for each gene we compute the Spearman correlation between unspliced and spliced abundance across different moving steps along the DPT. The time number of moving steps along unspliced to spliced expression, which maximizes the Spearman correlation, is considered the U-to-S delay. **Fig. 2a** illustrates how U-to-S delay can be used to validate the accuracy of pseudotime, with EML5 serving as an example. Then the U-to-S delay for that lineage is calculated as the mean U-to-S delay for all genes on it. And the average U-to-S delay across all lineages provides the overall U-to-S delay for the initial Leiden cluster. Finally, the initial Leiden cluster with the highest U-to-S delay is considered the correct initiation. The DPT under this condition is used to initialize pseudotime for TSvelo. If multiple lineages are detected, TSvelo will model each lineage independently and subsequently merge them in the final step. A detailed explanation with example illustrating how lineages are segmented is provided in the **Fig. S2** in **Supplementary Information**.

Neural ODE-based TSvelo model

Suppose u_g and s_g are the abundance of unspliced and spliced RNA, and $\alpha_g(t), \beta_g, \gamma_g$ are the transcription rate, splicing rate and degradation rate, respectively. The dynamics of a velocity gene g is modeled as:

$$\frac{du_g(t)}{dt} = \alpha_g(t) - \beta_g u_g(t) \tag{4}$$

$$\frac{ds_g(t)}{dt} = \beta_g u_g(t) - \gamma_g s_g(t) \tag{5}$$

We assume that the gene and cell-specific transcriptional rate $\alpha_g(t)$ is influence by the expression of Transcriptional Factors (TFs), while β_g and γ_g are gene-specific constant parameters. Considering the wide usage of linear models for gene-relations in previous studies[28,32-35], we model $\alpha_g(t)$ as

$$\alpha_g(t) = ReLU\left(\sum_{i \in TFs(g)} w_{gi} * s_i(t)\right) \tag{6}$$

In order to include more TFs in the modeling, we reserve those TFs that are not selected as velocity genes. These TFs are excluded from the velocity genes selection because

their unspliced-spliced expression does not provide sufficient information, primarily due to high noise in the unspliced RNA. Consequently, we incorporate these TFs into the TSvelo by directly modeling the process between transcriptional signal to mature RNA, $\frac{ds_g(t)}{dt} = \alpha_g(t) - \gamma_g s_g(t)$.

As a result, we can arrive the ODE model with parameters matrix A:

$$\begin{bmatrix} dU/dt \\ dS/dt \\ dS'/dt \end{bmatrix} = A \begin{bmatrix} U \\ S \\ S' \end{bmatrix} = \begin{bmatrix} -B & W & W' \\ B & -\Gamma & 0 \\ 0 & W & W' - \Gamma' \end{bmatrix} \begin{bmatrix} U \\ S \\ S' \end{bmatrix} \tag{7}$$

$U = [u_1, u_2 \dots, u_G]^T \in R^G, S = [s_1, s_2 \dots, s_G]^T \in R^G$, $S' = [s_{G+1}, s_{G+2} \dots, s_{G+K}]^T \in R^K$, G is the number of velocity genes and K is the number of additional TFs which are not selected as velocity genes. $\mathrm{B} \in R^{G\times G}$ and $\Gamma \in R^{G\times G}$ are the diagonal matrices consist of splicing rates $[\beta_1, \beta_2 \dots, \beta_G]$ and degradation rates $[\gamma_1, \gamma_2 \dots, \gamma_G]$ for velocity genes, respectively. $\Gamma' \in R^{K\times K}$ is the diagonal matrices consist of degradation rates $[\gamma_{G+1}, \gamma_{G+2} \dots, \gamma_{G+K}]$ for these additional TFs. $W \in R^{G\times G}$ and $W' \in R^{K\times K}$ denote the matrices representing the TF-target relationships for the TFs selected as velocity genes and those not selected as velocity genes, respectively. Using these gene-gene weight matrices, TSvelo can model the gene- and cell-specific transcriptional rate α_g. The whole parameters matrix $A \in R^{(2G+K)\times(2G+K)}$

Optimizing global time and Neural ODE in EM framework

In TSvelo, the dynamics described in Equation 7 are implemented using a neural ordinary differential equation (ODE) model. Given an initial state and the parameters in matrix A, the neural ODE model can compute the values of U and S at any time step. This approach eliminates the need for an analytical solution for unspliced and spliced expression, enhancing the flexibility of TSvelo. By default, the number of time steps in the ODE is set to 1,000. The ReLU activation function is applied to the parameters α_g, β_g and γ_g, in order to prevent them from taking negative values. Notably, TSvelo does not incorporate deep neural networks or encoders. Since TSvelo models all genes simultaneously, we normalize the unspliced and spliced abundances of each gene by its standard deviation before inputting the data into the Neural ODE model to ensure that the influence of each gene is balanced.

TSvelo optimize the pseudotime t and parameters matrix A in ODE iteratively using an Expectation-Maximization (EM) approach. The maximum number of iterations is set to 30 by default, with an early stopping criterion applied. Given the parameters in ODE system, we can compute the predicted values of unspliced and spliced at all time steps, denoted as $\hat{U}(t_step)$ and $\hat{S}(t_step)$, $t_{step} \in \{0,1,2 \dots, 999\}$. Given the time steps assignment $t_c \in t_{step}$ for each cell, we can get the expected values for all cells as $\hat{U}(t_c), \hat{S}(t_c)$. The loss function used in EM algorithm is the mean squared error

between the expected value with the observed data. Taking those TFs which are not velocity genes into consideration, whose expression is denoted as S', we have:

$$Loss = \frac{1}{c}\sum_{c}[(\hat{U}(t_c) - U_c)^2 + (\hat{S}(t_c) - S_c)^2 + (\hat{S'}(t_c) - S'_c)^2] \quad (8)$$

The EM algorithm will iteratively update the time assignment and parameters in neural ODE by minimizing the above loss function.

In the E-step, TSvelo updates the time step assigned to each cell using grid search, which is achieved by finding the minimum distance between the observed unspliced-spliced expression and the model prediction ($\hat{U}$ and $\hat{S}$) at all time steps. In the M-step, the parameter matrix A is updated using gradient descent within the neural ODE model to minimize the loss function. Here we use the prior gene regulation knowledge to constraint the weight matrix W and W' in A. Only If gene i is annotated as a TF for gene j in the ENCODE or ChEA databases, the corresponding w_{ij} could be optimized. Otherwise w_{ij} is fixed at zero. This could keep the matrix sparse and avoid overfitting. During training the neural ODE, he Adam optimizer is used, with a learning rate of 0.05, and the maximum number of epochs is set to 500, with an early stopping criterion to prevent overfitting.

By performing the EM optimization, TSvelo could fit the high-dimensional unspliced-spliced data across multiple genes well, and learn the global pseudotime for each cell. After the training procedure, the RNA velocity could be calculated using those parameters in matrix A, which could be further used for cell fate prediction.

Metrics for evaluating

(1) Velocity consistency. We used the scvelo.velocity_confidence() function from scVelo to evaluate velocity consistency, interpreting the results as a measure of how consistent velocities are within neighboring cells.

(2) Cross-boundary direction correctness. Cross-boundary direction correctness assesses the accuracy of transitions from a source cluster to a target cluster by examining the boundary cells, and requires ground truth annotations. We directly run the function unitvelo.evaluate() provided in UniTVelo[8] to obtain the Cross-boundary direction correctness.

(3) Within-cluster velocity coherence. Within-cluster velocity coherence measures the coherence of velocities within a single cluster using a cosine similarity score between cell velocities. We applied the function unitvelo.evaluate() provided by UniTVelo[8] to directly compute the within-cluster velocity coherence.

GO term enrichment analysis

To perform the GO term enrichment analysis, we utilized the "gseapy.enrichr()" function in Python with using Fisher's exact test by default.

Data Availability Statement

The pancreatic endocrinogenesis dataset[47] comprises the single-cell RNA-seq (10X) data of pancreatic epithelial and Ngn3-Venus fusion cells sampled from mouse embryonic day 15.5, which could be loaded using scVelo's package scvelo.datasets.pancreas().
The gastrulation erythroid dataset, which is selected from the transcriptional profiles of mouse embryos[36], which could be loaded using scVelo's package scvelo.datasets.pancreas().
10x embryonic mouse brain dataset is provided at the 10x website at https://www.10xgenomics.com/resources/datasets/fresh-embryonic-e-18-mouse-brain-5-k-1-standard-1-0-0. The data preprocessed by Multivelo is utilized in this study, (https://multivelo.readthedocs.io/en/latest/MultiVelo_Fig2.html).
The dentate gyrus neurogenesis data is available at http://pklab.med.harvard.edu/velocyto/DentateGyrus/DentateGyrus.loom
The LARRY dataset has been shared by pyrovelocity, which could be accessed at https://figshare.com/articles/dataset/larry_invitro_adata_sub_raw_h5ad/20780344.
The raw data of Hindbrain (pons) of adolescent mice is from https://pklab.med.harvard.edu/ruslan/velocity/oligos/.
The ENCODE TF-target database[45] website: https://maayanlab.cloud/Harmonizome/dataset/ENCODE+Transcription+Factor+Targets.
The ChEA TF-target database[46] website: https://maayanlab.cloud/Harmonizome/dataset/CHEA+Transcription+Factor+Targets.

Code Availability Statement

TSvelo is implemented in Python. The source code can be downloaded from the GitHub repository, https://github.com/lijc0804/TSvelo.

Acknowledgement

This work was supported by National Natural Science Foundation of China (No. 62103262 to Y.Y.), and the National Key R&D Program of China (2023YFF1204500 to Y.Y.)

Competing Interests Statement

The authors declare no competing interests.

1 Wolf, F. A. *et al.* PAGA: graph abstraction reconciles clustering with trajectory inference through a topology preserving map of single cells. *Genome biology* **20**, 1-9 (2019).
2 Qiu, X. *et al.* Reversed graph embedding resolves complex single-cell trajectories. *Nature methods* **14**, 979-982 (2017).
3 Street, K. *et al.* Slingshot: cell lineage and pseudotime inference for single-cell transcriptomics. *BMC genomics* **19**, 1-16 (2018).
4 Setty, M. *et al.* Characterization of cell fate probabilities in single-cell data with Palantir. *Nature biotechnology* **37**, 451-460 (2019).
5 Gayoso, A. *et al.* Deep generative modeling of transcriptional dynamics for RNA velocity analysis in single cells. *Nature Methods*, 1-10 (2023).
6 La Manno, G. *et al.* RNA velocity of single cells. *Nature* **560**, 494-498 (2018).
7 Bergen, V., Lange, M., Peidli, S., Wolf, F. A. & Theis, F. J. Generalizing RNA velocity to transient cell states through dynamical modeling. *Nature biotechnology* **38**, 1408-1414 (2020).
8 Gao, M., Qiao, C. & Huang, Y. UniTVelo: temporally unified RNA velocity reinforces single-cell trajectory inference. *Nature Communications* **13**, 6586 (2022).
9 Gayoso, A. *et al.* Deep generative modeling of transcriptional dynamics for RNA velocity analysis in single cells. *Nature methods* **21**, 50-59 (2024).
10 Gu, Y., Blaauw, D. T. & Welch, J. in *International Conference on Machine Learning.* 7887-7901 (PMLR).
11 Qin, Q., Bingham, E., La Manno, G., Langenau, D. M. & Pinello, L. Pyro-Velocity: Probabilistic RNA Velocity inference from single-cell data. *bioRxiv* (2022).
12 Li, S. *et al.* A relay velocity model infers cell-dependent RNA velocity. *Nature biotechnology*, 1-10 (2023).
13 Cui, H. *et al.* DeepVelo: deep learning extends RNA velocity to multi-lineage systems with cell-specific kinetics. *Genome Biology* **25**, 27 (2024).
14 Farrell, S., Mani, M. & Goyal, S. Inferring single-cell transcriptomic dynamics with structured latent gene expression dynamics. *Cell Reports Methods* **3** (2023).
15 Qiu, X. *et al.* Mapping transcriptomic vector fields of single cells. *Cell* **185**, 690-711. e645 (2022).
16 Ma, A., McDermaid, A., Xu, J., Chang, Y. & Ma, Q. Integrative methods and practical challenges for single-cell multi-omics. *Trends in biotechnology* **38**, 1007-1022 (2020).
17 Subramanian, I., Verma, S., Kumar, S., Jere, A. & Anamika, K. Multi-omics data integration, interpretation, and its application. *Bioinformatics and biology insights* **14**, 1177932219899051 (2020).
18 Gorin, G., Svensson, V. & Pachter, L. Protein velocity and acceleration from single-cell multiomics experiments. *Genome biology* **21**, 1-6 (2020).
19 Li, C., Virgilio, M. C., Collins, K. L. & Welch, J. D. Multi-omic single-cell velocity models epigenome–transcriptome interactions and improves cell fate prediction. *Nature Biotechnology* **41**, 387-398 (2023).
20 Riba, A. *et al.* Cell cycle gene regulation dynamics revealed by RNA velocity and deep-learning. *Nature communications* **13**, 1-13 (2022).
21 Lederer, A. R. *et al.* Statistical inference with a manifold-constrained RNA velocity model

uncovers cell cycle speed modulations. *Nature Methods*, 1-16 (2024).

22 Zhou, P., Bocci, F., Li, T. & Nie, Q. Spatial transition tensor of single cells. *Nature Methods*, 1-10 (2024).

23 Abdelaal, T. *et al.* SIRV: Spatial inference of RNA velocity at the single-cell resolution. *NAR genomics and bioinformatics* **6**, lqae100 (2024).

24 Liu, R., Pisco, A. O., Braun, E., Linnarsson, S. & Zou, J. Dynamical Systems Model of RNA Velocity Improves Inference of Single-cell Trajectory, Pseudo-time and Gene Regulation. *Journal of Molecular Biology*, 167606 (2022).

25 Bergen, V., Soldatov, R. A., Kharchenko, P. V. & Theis, F. J. RNA velocity—current challenges and future perspectives. *Molecular systems biology* **17**, e10282 (2021).

26 Gorin, G., Fang, M., Chari, T. & Pachter, L. RNA velocity unraveled. *PLOS Computational Biology* **18**, e1010492 (2022).

27 Soneson, C., Srivastava, A., Patro, R. & Stadler, M. B. Preprocessing choices affect RNA velocity results for droplet scRNA-seq data. *PLoS computational biology* **17**, e1008585 (2021).

28 Li, J., Pan, X., Yuan, Y. & Shen, H.-B. TFvelo: gene regulation inspired RNA velocity estimation. *Nature Communications* **15**, 1387 (2024).

29 Hossain, I., Fanfani, V., Fischer, J., Quackenbush, J. & Burkholz, R. Biologically informed NeuralODEs for genome-wide regulatory dynamics. *Genome Biology* **25**, 127 (2024).

30 Burdziak, C. *et al.* scKINETICS: inference of regulatory velocity with single-cell transcriptomics data. *Bioinformatics* **39**, i394-i403 (2023).

31 Lin, Z. TIVelo: RNA velocity estimation leveraging cluster-level trajectory inference. (2024).

32 Tran, A., Yang, P., Yang, J. Y. & Ormerod, J. T. scREMOTE: Using multimodal single cell data to predict regulatory gene relationships and to build a computational cell reprogramming model. *NAR genomics and bioinformatics* **4**, lqac023 (2022).

33 Song, Q., Ruffalo, M. & Bar-Joseph, Z. Using single cell atlas data to reconstruct regulatory networks. *Nucleic Acids Research* **51**, e38-e38 (2023).

34 Liu, Y.-Y., Slotine, J.-J. & Barabási, A.-L. Controllability of complex networks. *nature* **473**, 167-173 (2011).

35 Wang, L. *et al.* Dictys: dynamic gene regulatory network dissects developmental continuum with single-cell multiomics. *Nature Methods* **20**, 1368-1378 (2023).

36 Pijuan-Sala, B. *et al.* A single-cell molecular map of mouse gastrulation and early organogenesis. *Nature* **566**, 490-495 (2019).

37 Buenrostro, J. D., Giresi, P. G., Zaba, L. C., Chang, H. Y. & Greenleaf, W. J. Transposition of native chromatin for fast and sensitive epigenomic profiling of open chromatin, DNA-binding proteins and nucleosome position. *Nature methods* **10**, 1213-1218 (2013).

38 Hochgerner, H., Zeisel, A., Lönnerberg, P. & Linnarsson, S. Conserved properties of dentate gyrus neurogenesis across postnatal development revealed by single-cell RNA sequencing. *Nature neuroscience* **21**, 290-299 (2018).

39 Sun, Y. *et al.* Promotion of astrocyte-neuron glutamate-glutamine shuttle by SCFA contributes to the alleviation of Alzheimer's disease. *Redox Biology* **62**, 102690 (2023).

40 Weinreb, C., Rodriguez-Fraticelli, A., Camargo, F. D. & Klein, A. M. Lineage tracing on transcriptional landscapes links state to fate during differentiation. *Science* **367**, eaaw3381

(2020).

41 Borella, R. *et al.* Metabolic reprograming shapes neutrophil functions in severe COVID-19. *European Journal of Immunology* **52**, 484-502 (2022).

42 Liu, Z. *et al.* Fate mapping via Ms4a3-expression history traces monocyte-derived cells. *Cell* **178**, 1509-1525. e1519 (2019).

43 Bill, M. *et al.* Mapping the CLEC 12A expression on myeloid progenitors in normal bone marrow; implications for understanding CLEC 12A-related cancer stem cell biology. *Journal of Cellular and Molecular Medicine* **22**, 2311-2318 (2018).

44 Rogalska, M. E. *et al.* Transcriptome-wide splicing network reveals specialized regulatory functions of the core spliceosome. *Science* **386**, 551-560 (2024).

45 Feingold, E. *et al.* The ENCODE (ENCyclopedia of DNA elements) project. *Science* **306**, 636-640 (2004).

46 Lachmann, A. *et al.* ChEA: transcription factor regulation inferred from integrating genome-wide ChIP-X experiments. *Bioinformatics* **26**, 2438-2444 (2010).

47 Bastidas-Ponce, A. *et al.* Comprehensive single cell mRNA profiling reveals a detailed roadmap for pancreatic endocrinogenesis. *Development* **146**, dev173849 (2019).