DeepVelo: Deep Learning extends RNA velocity to multi-lineage systems with cell-specific kinetics

Haotian Cui[1,2,3]*, Hassaan Maan[2,3,4]*, & Bo Wang[1,2,3,5]†

[1]*Department of Computer Science, University of Toronto, Toronto, ON Canada*
[2]*Vector Institute, Toronto, ON, Canada*
[3]*Peter Munk Cardiac Centre, University Health Network, Toronto, ON, Canada*
[4]*Department of Medical Biophysics, University of Toronto, Toronto, ON, Canada*
[5]*Department of Laboratory Medicine and Pathobiology, University of Toronto, Toronto, ON, Canada*

1 Abstract

The introduction of RNA velocity in single-cell studies has opened new ways of examining cell differentiation and tissue development. Existing RNA velocity estimation methods are based on strong assumptions of either complete observation of cells in steady states or a predefined dynamics pattern parameterized by constant coefficients. These assumptions are violated in complex

*These authors contributed equally.
†Corresponding author. Email: Bo.Wang@uhnresearch.ca

and heterogenous single-cell sequencing datasets and thus limit the application of these techniques. Here we present DeepVelo, a novel method that predicts the **cell-specific** dynamics of splicing kinetics using Graph Convolution Networks (GCNs). DeepVelo generalizes RNA velocity to cell populations containing **time-dependent kinetics and multiple lineages**, which are common in developmental and pathological systems. We applied DeepVelo to disentangle multifaceted kinetics in the processes of dentate gyrus neurogenesis, pancreatic endocrinogenesis, and hindbrain development. DeepVelo infers time-varying cellular rates of transcription, splicing and degradation, recovers each cell's stage in the underlying differentiation process and detects putative driver genes regulating these processes. DeepVelo relaxes the constraints of previous techniques and facilitates the study of more complex differentiation and lineage decision events in heterogeneous single-cell RNA sequencing data.

2 Introduction

The concept of RNA velocity refers to the time derivative of the mRNA abundance in a cell, which reflects the changing rate of RNA processing and degradation. Current velocity estimation methods leverage the observation that the abundance and ratio between unspliced pre-messenger RNAs and spliced mature messenger RNAs can be used to infer changes in gene expression dynamics. Higher abundance and ratio of unspliced mRNAs to spliced mRNAs indicates increasing transcription of a certain gene - in other words, up-regulation/induction and a high velocity estimate. Conversely, a lower abundance and indicated ratio leads to a low velocity estimate and is associated with down-regulation/repression. An equilibrium phase occurs when this dynamical process reaches a stable steady-state, and the rates of spliced and unspliced mRNAs are stable. Since unspliced mRNAs can be distinguished in common single-cell RNA sequencing (scRNA-seq) protocols [16], the idea of estimating dynamic RNA velocity using only static sequencing libraries becomes feasible. The original RNA velocity approach [16] utilized the assumption that the observed transcriptional phases in scRNA-seq last long enough to reach both an apex of induction and a quiescent steady-state equilibrium. This technique infers a per-gene *steady-state ratio* using linear

regression, and then RNA velocities are calculated as the deviation of the observed ratio from the steady-state level. This workflow implies two underlying assumptions, (1)**the assumption of steady-state**: For every gene, the steady states are well captured; (2)**the assumption of cell-agnostic kinetic rates**: The degradation and splicing rate for each gene is shared across all cells. These assumptions are often violated in complex biological systems that single-cell sequencing technologies measure and therefore bring about limitations in downstream applications, particularly when cell states are partially observed or undergo transcription dynamics more complex than the steady-state pattern. Although a later approach, scVelo [4], attempted to generalize the *steady-state* assumption by replacing these states with *four transcriptional states* and modeling them with a dynamical model, the aforementioned second limitation still remains. Further, scVelo assumes a cyclic trajectory within the four transcriptional states for all observed genes, but this assumption also rarely holds in real-world single-cell datasets with complex differentiation trajectories and multifactorial kinetics [9]. Although several related works have been further developed, including MultiVelo [20], Chromatin Velocity [26], protaccel [8] for extending Velocity beyond RNA, VeloAE [24] for denoising velocity with Deep Neural Nets, Dynamo [25] for exploiting the metabolic labeling sequencing data, the core velocity computation follows the original ideas and therefore the aforementioned limitations still hold.

Overall, existing techniques assume each gene follows a pre-defined trajectory depicted by constant cell-agnostic kinetic rates. This workflow implies that each gene goes through the same velocity trajectory across all celltypes, and limits the application in complex real-world systems. To resolve these limitations, we highlight the need for cell-specific kinetics which enables the modeling of multi-lineage systems with heterogeneous cell populations. We propose DeepVelo, a deep neural network based method that models RNA cellular velocities without pre-defined kinetic patterns. Empowered by deep Graph Convolutional Networks (GCN), DeepVelo infers **gene-specific and cell-specific** RNA splicing and degradation rates. Therefore, compared with the cell-agnostic parameters used in existing techniques [16, 4], DeepVelo is able to model RNA velocities for dynamics of high complexity, particularly for cell populations with heterogeneous celltypes and multiple lineages.

We demonstrate the efficacy of DeepVelo on multiple developmental scRNA-

seq datasets including dentate gyrus neurogenesis [11], pancreatic endocrinogenesis [2], and hindbrain development [29]. DeepVelo yields more consistent velocity estimates and accurately identifies transcriptional states compared to the existing models. To highlight the improvement introduced by the cell-specific modeling of DeepVelo, we examine the estimated kinetic rates for individual genes and show that the cell-specific rates accurately recover known differentiation trajectories in challenging scenarios of time-dependent and multi-trajectory gene regulation dynamics. DeepVelo exceeds the capacity of existing models with cell-agnostic rates in realistic single-cell datasets with multiple trajectories/lineages, while also scaling better to larger datasets. For downstream tasks, DeepVelo can identify putative driver genes of these transcriptional changes, which are more likely to characterize and be involved in dictating lineage fate-decisions. The DeepVelo model is available within the DeepVelo package (`https://github.com/bowang-lab/DeepVelo`).

3 Results

3.1 The DeepVelo model

Modeling the transcriptional dynamics in single cells provides the theoretical basis of RNA velocity. For each cell, the dynamics of transcription, splicing, and degradation (Fig.1a) can be approximated as the following differential processes

$$\begin{aligned} \frac{du(t)}{dt} &= \alpha_{i,g}(t) - \beta_{i,g}(t)\, u(t), \\ \frac{ds(t)}{dt} &= \beta_{i,g}(t)\, u(t) - \gamma_{i,g}(t)\, s(t). \end{aligned} \tag{1}$$

where $\alpha_{i,g}, \beta_{i,g}, \gamma_{i,g}$ are the kinetic rates for cell i and gene g. t denotes a time coordinate in cell development. Unspliced immature mRNA is first generated by transcription of DNA and then post-transcriptionally modified and spliced into mature RNA. The dynamics of unspliced RNA abundance, $\frac{du(t)}{dt}$, is modeled by the first equation where $\alpha_{i,g}$ and $\beta_{i,g}$ denote the rates of transcription and splicing, respectively. Similarly, the second equation models the dynamics of spliced RNA abundance, $\frac{ds(t)}{dt}$, and $\gamma_{i,g}$ denotes the rate for RNA degradation. Note that all of the kinetic rates are intrinsically cell-specific since there is a high degree of variability in transcriptional dynamics

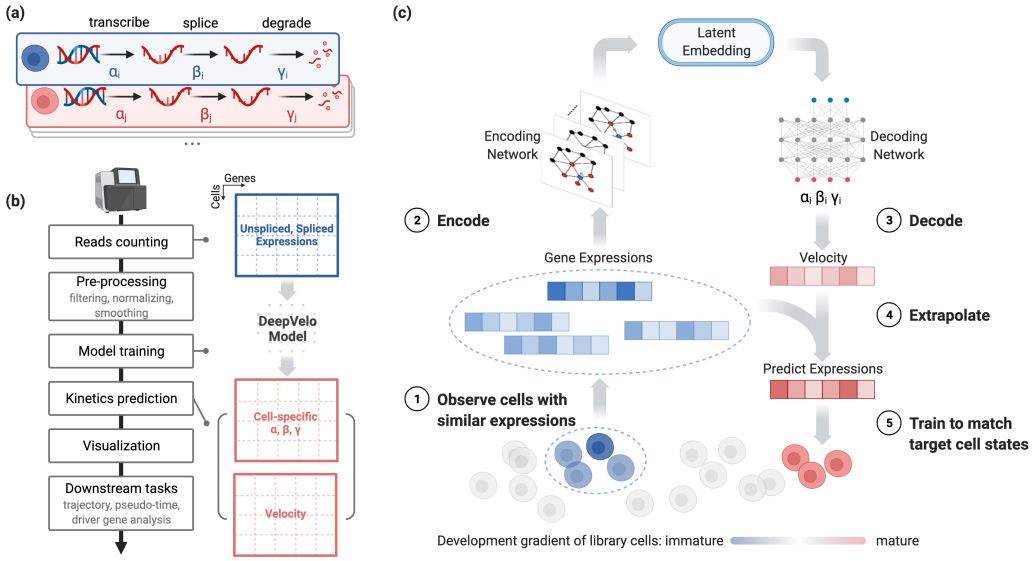

Figure 1: **Overview of the DeepVelo pipeline and velocity prediction method.** (a) DeepVelo estimates cell-specific transcription (α_i), RNA splicing (β_i) and RNA degradation rates (γ_i). (b) Overview of the velocity analysis pipeline using DeepVelo. After read counting of unspliced and spliced mRNA, preprocessing is done to ensure the stability of model training (Online methods), followed by training and prediction of cell-specific kinetic parameters. These are used to estimate the RNA velocity and perform downstream analyses, such as visualization of velocity fields and pseudo-time inference. (c) Overview of the DeepVelo neural network model. Query cells (dark blue) and similar cells (light blue) within a k-nearest neighborhood are input into the model. The Graph Convolutional Network (GCN) [15] encoder module encodes their spliced/unspliced gene expressions into latent space representations. The decoder module then predicts the kinetic rates for RNA velocity and extrapolates expressions to future cell states. The model is optimized to match the extrapolation to observed cell states at later developmental stages. After training and optimization, these rates can be used to determine the RNA velocity vector for each cell through cell-specific rates of transcription, splicing and degradation.

between cells due to the stochastic "bursting" nature of gene expression [12]. Further, these intrinsic cell-specific transcriptional dynamics are likely to be similar for similar celltypes [19], necessitating celltype-specific parameters. **However, previous techniques did not have independent parameters for each cell i, leading to limitations in inferring multi-lineage dynamics**.

Given the unspliced gene counts $u(t)$ and spliced gene counts $s(t)$ for individual cells, DeepVelo estimates the derivatives of $s(t)$ by modeling cell and gene-specific coefficients $\alpha_{i,g}, \beta_{i,g}, \gamma_{i,g}$ using a deep neural network model (Fig.1b,c). As opposed to previous techniques [16, 4], DeepVelo models the coefficients per cell and per gene (Fig.1c), providing sufficient expressive power for more faithful velocity estimates for individual cells.

Specifically, we predict a cell's velocity vector and extrapolate the cell state to match the future states extracted from the sequencing data (Fig.1c). For each cell i in the population, we extract a group of neighbor cells $\mathcal{N}_i$ that have similar expression profiles. Then we compute an initial direction using the velocity heuristic from the steady-state model and extract a group of downstream neighbors along the computed direction. These downstream cells depict the possible future states of cell i. We take the profiles of cell i and neighbor set $\mathcal{N}_i$ as the input to DeepVelo model. The model consists of stacked layers of GCNs and outputs the coefficients $\alpha_{i,g}$, $\beta_{i,g}$, and $\gamma_{i,g}$ in the final layer. Using these coefficients, DeepVelo computes the velocity $v_{i,g} = \frac{ds(t)}{dt}$ for each cell accordingly as in Eq.1.

To train the DeepVelo model, i.e. to update the parameters for accurate velocity prediction, we first extrapolate the cell state by adding the velocity derivative $\frac{ds(t)}{dt}$ onto the original profile $s\,(t)$. Then, DeepVelo computes the difference between the extrapolated state $s(t+1)$ and the real profiles of downstream cells. The DeepVelo model parameters are optimized to minimize the difference between the predicted future state and the actual profile of cells downstream, which works to tune the velocity estimates to be optimal for a given dataset (Online methods - 5.4). After sufficient training iterations, the model is finalized to provide accurate velocity estimates that take into account the transcriptional dynamics unique to individual cells.

We tested DeepVelo on a number of developmental datasets in the con-

text of determining RNA velocity, estimating cell-specific RNA kinetics, inferring developmental pseudotime, and prioritizing genes for their potential role in differentiation through driver gene estimation. We also compared the efficiency between DeepVelo and other velocity estimation methods. We also compared the efficiency between DeepVelo and other velocity estimation methods. In a scRNA-seq dataset of 3,696 cells, DeepVelo is able to finish the velocity and pseudo time analysis in 140 seconds, which is four times faster than the dynamical model which took over 9 minutes. A more extensive comparison over multiple datasets was also performed, which extends these findings (Supplementary Fig.S4).

3.2 Recovering complex transcription dynamics for individual cells using DeepVelo

To test the ability to identify complex kinetics, we applied DeepVelo on a neurogenesis scRNA-seq dataset of the developing mouse dentate gyrus [11]. The data consists of tissue samples from two time points, P12 and P35 (postnatal day 12 and 35), which are collected by a droplet-based single-cell RNA sequencing protocol (10x Genomics Chromium Single-Cell Kit V1). After pre-processing (Online Methods - 5.1), we calculated the RNA velocities using the proposed DeepVelo model and the dynamical model from scVelo [4]. The velocity plots are made by projecting the velocity vectors onto the UMAP [21]-based embedding of the data. In the velocity estimates (Fig.2a), the granule cell lineage dominates the main structure, where the neuroblast cells develop into immature and mature granule cells. The directions of these velocity estimates between celltypes have been validated by existing literature, based on the temporal differentiation trajectories [11].

When examining the main lineage toward the terminal celltype of granule cells, although all models capture the principle direction, DeepVelo has the advantage of showing a more consistent flow from the neurogenic intermediate progenitor cells (nIPC) to neuroblasts, and finally to granule cells. DeepVelo particularly indicates that immature granule cells differentiate into mature granule cells in a manner more faithful to the true trajectory compared with the dynamical model (Fig.2a - zoom-in panel).

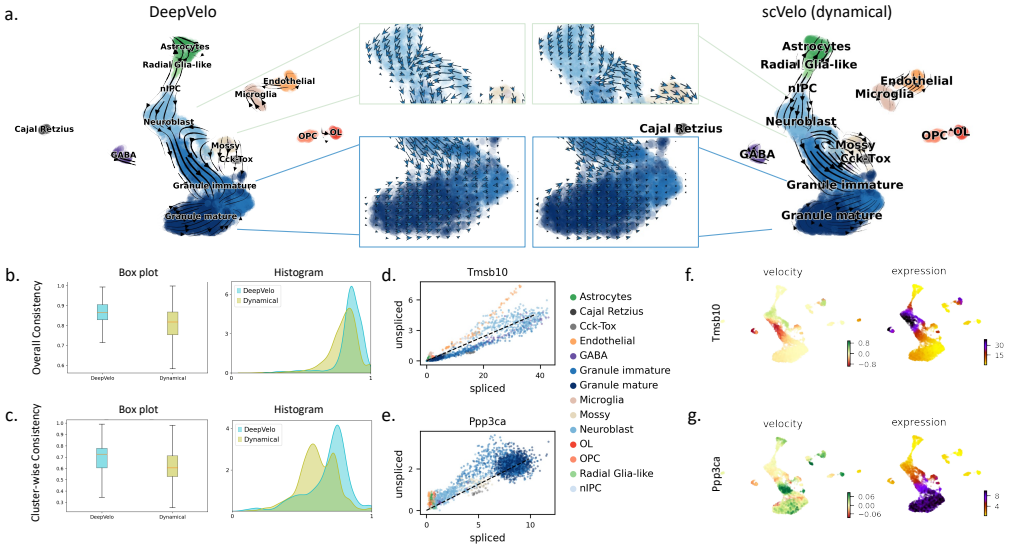

Figure 2: **Fine-grained temporal patterns in neurogenesis predicted by DeepVelo.** (a) Comparison of DeepVelo with the dynamical model from scVelo [4]. The direction and magnitude of velocities are projected as arrows onto the Uniform Manifold Approximation and Projection (UMAP) plot of gene expression values across cells. DeepVelo provides more consistent velocity estimates with respect to the developmental process from immature granule cells to mature granule cells. (b) The box plot and histogram of the overall consistency scores, which indicate the consistency of velocity estimates in a local neighborhood of the data. (c) The box plot and histogram of the cluster/celltype-specific consistency scores, which utilize the neighborhood consistency metric on a per cluster/celltype basis. DeepVelo outperforms the scVelo dynamical methods in both metrics. (d)(e) The spliced/unspliced phase portrait for *Tmsb10* and *Ppp3ca*, respectively. Celltypes are shown in the same color as in panel (b). (f)(g) Velocity and gene expression values projected onto UMAP plots for *Tmsb10* and *Ppp3ca*, respectively. Velocity and gene expression values show consistent patterns across celltypes: high velocity values (green in velocity plot) are correctly shown in the subset of cells with high gene expression values (purple in expression plot), for the same given gene(s).

The estimated velocities by DeepVelo show higher consistency in quantitative analysis. The consistency score is computed as follows - we first compute the average cosine similarity of the velocity vector of each cell to its neighbors, which is defined as the overall consistency. A similar neighbor-wise consistency was also proposed in scVelo [4]. However, the overall consistency could be biased toward over-smoothed estimations which don't account for branching and/or multiple lineages. Therefore, we propose the cluster/celltype-wise consistency as a complement to the overall score, which computes the average cosine similarity of each cell's velocity to all velocity vectors of the same celltype. For both metrics, DeepVelo shows significant improvements over the scVelo dynamical method with higher average scores and unimodal distributions (Fig.2b,c).

Examined at the individual gene level, DeepVelo shows biologically meaningful velocity patterns. For example, *Tmsb10* is one of the major regulators to the inferred dynamics of granule lineage and it plays an important role in the development of hippocampal CA1 region [1]. In Fig.2f, velocities derived from the DeepVelo are consistent across velocities of neighboring cells. The region of cells showing high velocities of *Tmsb10* aligns well with the region of high *Tmsb10* expression. The same alignment is also observed for another regulatory gene, *Ppp3ca* (Fig.2g). In further analysis (Fig.3a), we also observed that DeepVelo clearly disentangles the velocity vectors between the granule (blue) and endothelial lineages (orange), whereas, in the steady-state and dynamical models, both lineages have intertwined velocities. We discuss this limitation of previous techniques in further detail in Section.3.3.

3.3 DeepVelo's cell-specific kinetic rate estimates enable accurate quantification of time-dependent and branching gene dynamics

Due to the cell-specific estimation of ($\alpha_{i,g}, \beta_{i,g}, \gamma_{i,g}$ in Eq.1 in Eq.1), DeepVelo for the first time provides a profile of individual kinetic rates for each cell. This enables new approaches for cell-specific trajectory analysis, visualization, and characterization. We show the UMAP projection of all cell-specific kinetic rates of 2930 cells (Fig.3a). Although DeepVelo is **unaware of the celltypes during training**, the learned kinetic rates are naturally clustered

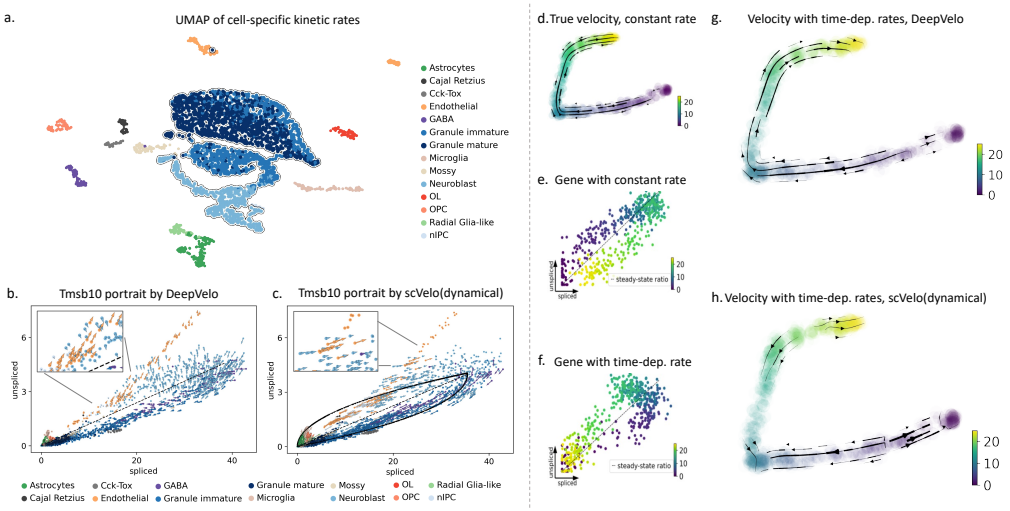

Figure 3: **Velocity estimation for branching kinetics and time-dependent kinetic rates.** (a) The UMAP projection of the estimated kinetic rates of 2930 cells in the dentate gyrus developmental data. Cells of the same celltypes are clustered together by kinetic rates, reflecting similarity in predictions by DeepVelo. Further, clusters of cells from the same lineage (e.g the outlined Granule lineage) are positioned closely compared with other cells. In general, the similarity of learned kinetic rates reflects the biological similarity of cells, although the DeepVelo model is unaware of cell-type labels. (b) Projection of estimated velocity (arrows) onto the unspliced and spliced phase portrait of *Tmsb10* by DeepVelo. The endothelial cells undergo a separate trajectory on the phase portrait, aside from the main trajectory containing neuroblast cells, granule immature and granule mature cells. DeepVelo successfully captures both trajectories. In the zoomed view, cells within the same region comprising of different celltypes are correctly predicted to have distinct velocity directions. (c) Phase portrait of *Tmsb10* with RNA velocity predicted by the dynamical model of scVelo. Only the main trajectory of granule lineage is well captured, but other celltypes including endothelial cells are predicted with incorrect direction. (d) Reference velocity on simulated data with constant kinetic rates. (e)(f) Constant and time-dependent degradation rates as shown on phase portraits. The gene with the time-dependent rate (f) undergoes a reversed trajectory. (g)(h) Estimated velocities by DeepVelo and scVelo, respectively, for a simulated population of 500 cells for the subset with time-dependent kinetic rates. Colors indicate pseudo-time in the simulated data. DeepVelo correctly recovers the directions from regions of earlier to later time values.

into groups aligned with celltypes. Further, clusters of cells from the same lineage (e.g. the outlined granule lineage) are positioned closely compared to other cells. Overall, the similarity of learned kinetic rates reflects the biological similarity of cells at both the celltype and lineage levels. This indicates that DeepVelo can estimate kinetics that reflect the dynamics of individual cell populations as opposed to the entire dataset.

Velocity-associated kinetic rates across cells may vary for genes undergoing dynamic regulation involving multiple processes. For example Battich, Stoeger, and Pelkmans [3] observed varying kinetic rates in the differentiation of intestinal stem cells. These varying kinetics are often misinterpreted in existing velocity methods [5]. This stems from the fact that the kinetic rates in previous methods are modeled as constant cell-agnostic coefficients in first-order equations (Eq.2), which lack the ability to model complex dynamical variation. In contrast, DeepVelo provides estimates for different celltypes and states by introducing cell-specific kinetic rates, and the relaxation of the cell-agnostic constraint leads to better velocity estimation in time-dependent and complex systems. Here, we show this improvement using two challenging scenarios:

(1) Estimate velocity for genes that are separately regulated in two lineages. We used the previously analyzed dentate gyrus cell population and determined genes with complex kinetics [11]. *Tmsb10* shows multiple kinetic regimes and undergoes multiple trajectories in the phase portrait of spliced and unspliced reads in this dataset. The cells in the granule lineage (including neuroblast, granule immature and granule mature celltypes) form a cyclic trajectory. Meanwhile, the endothelial cells are located outside of the granule lineage (Fig.2d). These two regimes are possibly regulated by different kinetic rates. We applied DeepVelo and the dynamical model from scVelo and computed the velocity estimates (Fig.3b,c).

DeepVelo correctly predicted the patterns for both regimes (Fig.3b). For the granule lineage, DeepVelo captures the direction of velocity from neuroblast cells to granule immature cells and then to granule mature cells. For the endothelial cells, the predicted velocity direction correctly points to the position of the same celltype with amplified spliced reads. Additionally, we found that DeepVelo learns to assign similar velocity directions for cells of the same type, even though the celltype labels are not available during training.

This is because cells of the same celltype have similar gene expression profiles, and are therefore embedded closer together in representations learned by the neural network model. In contrast to DeepVelo, scVelo forces the velocities to follow the assumed cyclic trajectory by the model (Fig.3c). As a result, although scVelo successfully captures the trajectory for the granule lineage, it incorrectly points the velocities of endothelial cells to the position of neuroblasts, (Fig.3c - Zoom-in panel).

However, DeepVelo is capable of predicting different velocity directions for cells within the same region (Fig.3b). The cells in the zoomed view, including both the endothelial and neuroblast cells, employ similar RNA dynamics (through the levels of spliced and unspliced reads) of *Tmsb10*. However, the distinct directions for each celltype are correctly predicted by DeepVelo. This is due to the ability of DeepVelo to estimate distinct sets of kinetic rates per celltype, as shown in Fig.3a. In contrast, scVelo uses constant kinetic rates per gene and predicts a uniform direction for the same region of cells. Overall, a cell-specific model such as DeepVelo broadens the application of RNA velocity for genes with multifaceted kinetics, such as *Tmsb10* in the dentate gyrus developmental data.

(2) Estimate velocity for genes with time-dependent kinetic rates. We simulated the population of 500 cells and 30 genes using the simulator provided by the scVelo package [4]. We first show the reference velocity in the setting of constant kinetic rates (Fig.3d). The color of the cells indicates their associated pseudo-time in the simulation process. Then, the degradation rates *gamma* of 3 out of 30 genes were set to increase over time. As a result, the genes undergo a reversed trajectory as shown in the phase portrait (Fig.3f) as opposed to the original phase portrait of constant degradation rates (Fig.3e). This sets up a challenging scenario for the estimation of velocity. The resulting velocity plots of DeepVelo and the dynamical model of scVelo are shown in Fig.3g,h, and scVelo struggles to predict constant velocities across cells while DeepVelo is able to recover the correct velocity directions from regions of earlier to later timepoints. The DeepVelo model learns to predict velocity directions toward the possible future cell states (Eq.3). By correctly predicting the future cell states using all genes, DeepVelo was able to correct for the directions of small subset of time-dependent reversed trajectories in the data.

3.4 Tracking the ordering of development using DeepVelo-estimated diffusion pseudo-time

The velocities estimated by DeepVelo can also be used to improve the prediction of pseudo-time for cell states across a developmental trajectory. We first compute the velocity connectivity graph to represent cell-cell relationships and use this graph as the basis to compute a diffusion pseudo-time [10] mapping (Online Methods - 5.5). We compared the pseudo-time estimates (Supplementary Fig.S3a) using DeepVelo with the latent time (Supplementary Fig.S3c) estimates by the dynamical model from scVelo on a scRNA-seq dataset of pancreatic endocrinogenesis with ground-truth temporal measurements. DeepVelo pseudo-time provided a more accurate reconstruction of the pseudo-time trajectory compared to the scVelo estimate (Supplementary Fig.S3(a,b)). Particularly, DeepVelo more reliably predicts the continuous ordering of pseudotime values for the terminal states of Alpha and Beta cells (Supplementary Fig.S3(a,c)). Similar to scVelo, DeepVelo successfully demonstrates that the main stream of EP cells develop into terminal cell-types - alpha, beta and delta. However, the earliest-developed cell cluster at the upper left of the velocity plot, indicating the terminal state for the alpha cells, is much better captured and assigned with a clearly more discrete and smaller pseudotime value than the terminal state for beta cells. (Supplementary Fig.S3a). This indicates that the DeepVelo inferred pseudo-time is better aligned with the ground-truth, as the difference between the terminal celltypes is emphasized where the alpha-cells are developed earlier at E12.5 and the beta-cells appear later at E15.5. Although the dynamical model from scVelo also captures the overall temporal order, this result has more ambiguity compared to the pseudo-time result of DeepVelo.

3.5 DeepVelo infers functionally relevant lineage-specification genes and processes in hindbrain development

To test velocity methods in a complex setting with multiple lineages, we applied all methods on a mouse hindbrain development dataset [29] (Fig.4(a)). Specifically, we filtered the data corresponding to the junction and differentiation between the GABAergic and gliogenic lineages (Methods.5.1). In a

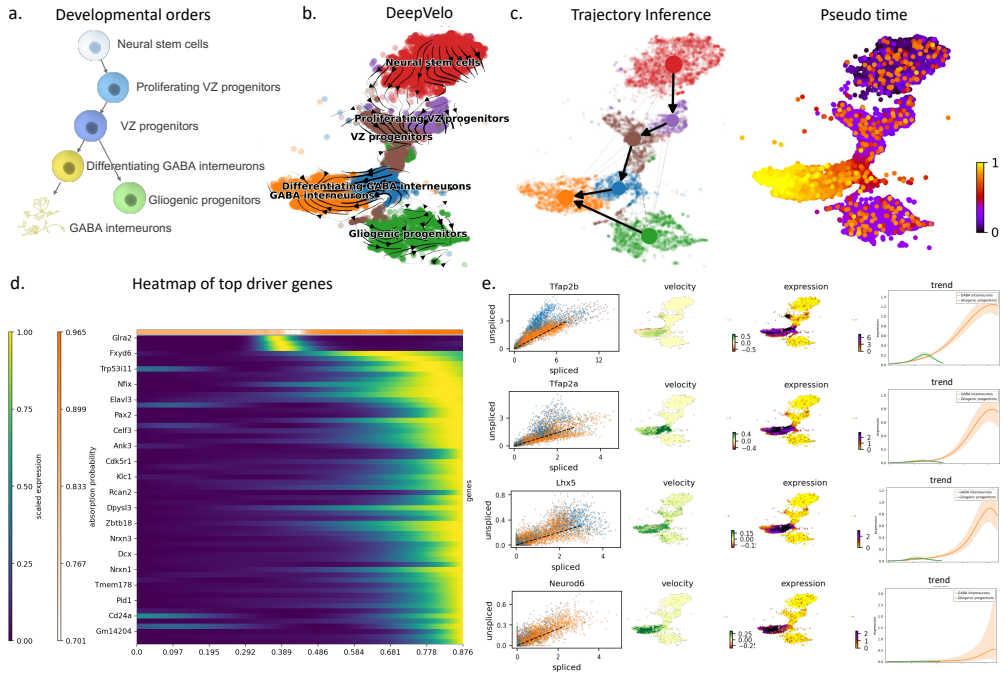

Figure 4: **Velocity, trajectory, and driver gene estimation of developing mouse hindbrain cells.** (a) The putative development order for 6 celltypes in early mouse hindbrain, reported by Vladoiu et al. [29]. (b) The velocity projected onto the tsne plot of gene expression. DeepVelo reveals the temporal order among stem cells and early progenitors in the developing mouse hindbrain, including cells from early progenitors, GABAergic, and gliogenic lineages. (c) Trajectory inference using velocity-based PAGA [31] and velocity-based pseudo-time analysis. The predicted trajectory correctly reflects the relations shown in (a). Only the direction to gliogenic progenitors is projected incorrectly, this is due to the sequencing noise within the two clusters, gliogenic progenitors and VZ progenitors. It can be seen in the tsne plot the two clusters (green and brown) have a portion of cells mixed up. (d) The top 60 driver genes predicted through DeepVelo's velocity estimation, sorted by the difference of velocity scales across celltypes. (e) Gene phase portrait, velocity, gene expression plots and expression trends of selected driver genes. The expression trend shows the abundance of expression for specific genes in cells at corresponding pseudo-time estimated timepoints. Tfap2b, Tfap2a, Lhx5, and Neurod6 are computed among the top driver genes for the GABAergic lineage.

complex multi-faceted system such as this, which is typical of developmental scRNA-seq datasets, considering cell-agnostic kinetic rates is haphazard because of the presence of multiple lineages which can contain different RNA velocity dynamics. DeepVelo's ability to learn cell-specific kinetic rates alleviates this assumption and accounts for the multi-faceted differentiation of GABAergic and gliogenic lineages and their respective celltypes. The result of DeepVelo (Fig.4(b)) shows the RNA velocity over the developmental process from Neural stem cells to the differentiating GABA interneurons and gliogenic progenitors. We performed trajectory inference using directional PAGA [31] over the velocity graph of DeepVelo. We found that DeepVelo was able to recapitulate ground-truth differentiation patterns - specifically the branching between VZ progenitors and differentiation VZ progenitors and gliogenic progenitors (Fig.4 (c)). The cluster of neural stem cells is well recognized as the origin celltype with outward velocity arrows and a low pseudo-time index, while the GABA interneurons are confirmed as a terminal celltype with incoming velocity arrows and a high pseudo-time index. Overall velocity visualizations were similar for the dynamical and steady-state methods (Supplementary material).

Using the velocity vector for each cell, we built a connectivity graph (Methods Section.5.5) of the scRNA-seq data. CellRank [18] is a recent visualization and analysis toolbox for scRNA-seq data that utilizes the connectivity graph to predict cell's fate mapping, which corresponds to the probability of the cell differentiating to a terminal state in the lineage(s). After determining cell fate, gene importance for differentiation can be calculated based on the correlation of gene expression with transition and differentiation probabilities towards all terminal states. The genes that display dynamical behavior across a lineage are termed putative "driver genes", as these are the genes most likely to be involved in the regulation of the differentiation process itself. Driver genes are systematically detected via their gene expression correlation with lineage specification (Online methods). This procedure presents a dynamics-based alternative to the standard differential expression paradigm and is thus more likely to capture regulatory genes involved in the differentiation process. CellRank has been reported to work well with other velocity methods, such as scVelo, to infer lineage-specific drivers. We incorporated this toolbox with the predicted velocity connectivity graph from DeepVelo and determined driver genes in the variable gene subset of the data for both the GABAergic and gliogenic lineages.

Within the top 100 driver genes across both lineages of interest, we observed groups of genes showing particular abundance in specific celltypes in a temporal manner (Fig.4(d)). For example, *Tfap2a*, *Tfap2b*, and *Lhx5*, which are two known differentiation genes involved in the specification of GABAergic interneurons during hindbrain development, are listed in the top 100 driver genes from DeepVelo for the GABAergic lineage (Fig.4(e)) [33, 23]. Similar results were found for the gliogenic lineage from DeepVelo, with detection of known glial cell differentiation regulators in *Hes1* and *Sox9* (Supplementary Table 1) [32, 30]. Within the top 100 driver genes, DeepVelo also picked up hits that were novel and not detected by scVelo, such as *Neurod6* in the GABAergic developmental lineage (Fig.4(e)). Although the role of *Neurod6* in the differentiating GABAergic interneurons and their development is unclear, previous literature has indicated the gene's involvement in regulating specification of inhibitory GABAergic interneuron subpopulations in the hindbrain and spinal cord [27]. This indicates a testable link and hypothesis for the differentiation of these cells in the junction within the GABAergic and gliogenic lineages, highlighting the ability of DeepVelo to guide searches of functional genes in scRNA-seq data and potential drivers of the differentiation process.

To compare the results of driver analysis when employing CellRank with different velocity outputs, we determined driver genes for the gliogenic and GABAergic lineages using both scVelo and DeepVelo. As the complete set of genes driving differentiation in the complex hindbrain developmental system is not known, we sought to infer the relevance of inferred driver genes in two ways: 1) By considering their overlap with predicted marker genes from the original analysis, as these genes are characteristic of celltype identity and should be correlated with lineage specification, and 2) By considering their overlap with transcription factors (TFs), as TFs are the main elements responsible for differentiation and establishing transcriptional and cellular identity. We analyzed and compared the top 100 driver genes for both the GABAergic and gliogenic lineages predicted by the scVelo and DeepVelo methods (Supplementary Table 1). DeepVelo predicted driver genes that overlapped with more of the original markers from Vladoiu et al. [29], for both the GABAergic and gliogenic lineages (Fig.5(a)) (Supplementary Table 2). Although the GABAergic driver genes overlapped with much fewer marker genes, when examining the density of correlations between the marker genes that were found for this lineage, DeepVelo had a stronger signal for

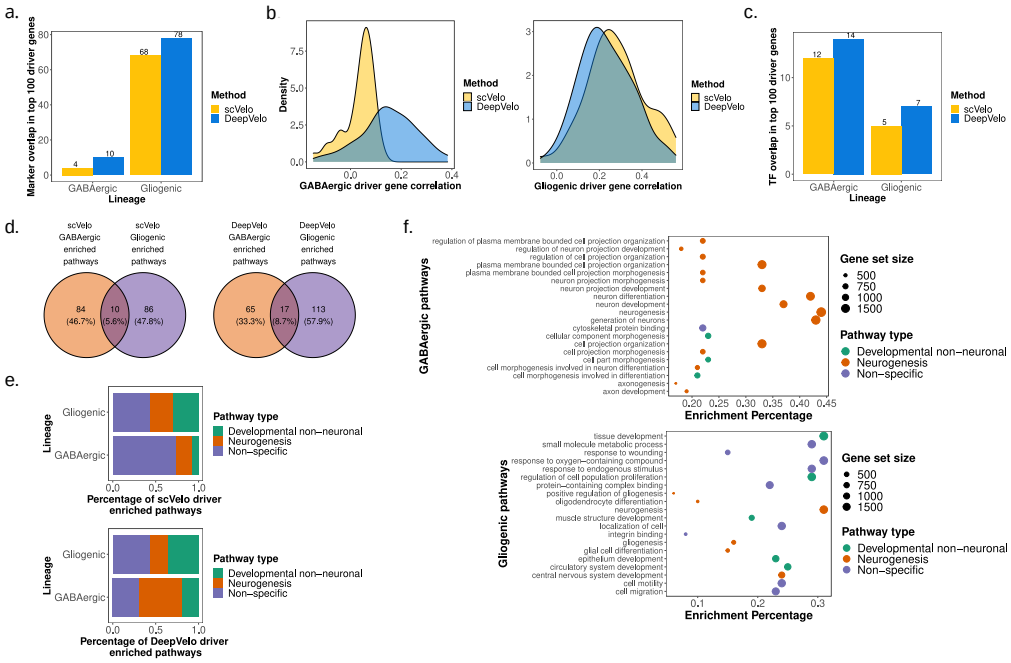

Figure 5: **Functional enrichment of DeepVelo predicted driver genes.** (a) Overlap of the top 100 driver genes from scVelo and DeepVelo for GABAergic and gliogenic lineages with annotated lineage marker genes. (b) Marker-overlapping driver gene correlation values for scVelo and DeepVelo, separated by the GABAergic and gliogenic lineages, respectively. (c) Overlap of top 100 driver genes from DeepVelo and scVelo for both lineages with annotated transcription factors. (d) Pathway enrichment analysis results for the top 100 scVelo and DeepVelo driver genes, respectively, in the GABAergic and gliogenic lineages. (e) Functional signal in the enriched pathways for scVelo and DeepVelo, based on the presence of pathways involved directly in neurogenesis ("Neurogenesis"), not specific to neurogenesis but involved in development ("Developmental non-neuronal"), and not specific to either development or neurogenesis ("Non-specific"). (f) The top 20 DeepVelo pathway enrichment analysis results, based on FDR corrected p-values, for the GABAergic and gliogenic lineages, respectively.

the marker genes (Kolmogorov-Smirnov Test $p = 2.423 \times 10^{-09}$) (Fig.5(b)). When examining the transcription factor overlap in the top 100 driver genes, DeepVelo also had more hits than scVelo for both the GABAergic and gliogenic lineages (Fig.5(c)).

For further examination of the results of driver analysis, we took the top 100 driver genes for the GABAergic and gliogenic lineages from DeepVelo and sought to determine their functional signal as gene-sets through pathway enrichment analysis. Overall, 82 and 130 pathways were found to be significantly enriched for the GABAergic and gliogenic lineages, respectively for DeepVelo (Fig.5(d)) (Supplementary Table 3). These pathways were analyzed for the presence of neurogenesis and developmental results, for which we did see a functional enrichment in both lineages (Fig.5(e)). More specifically, the top 20 pathways, ranked in terms of FDR-corrected p values, for each lineage revealed enrichment of pathways relevant to neuronal differentiation processes (Fig.5(f)). In the GABAergic lineage, enriched pathways included: *regulation of neuron projection development*, *neuron differentiation*, and *neurogenesis* (Fig.5(f)). The results from the gliogenic lineage had an even more relevant terms, including *positive regulation of gliogenesis*, *oligodendryocyte differentiation*, and *glial cell differentiation* (Fig.5(f)). When comparing these results with pathway analysis performed on the scVelo top 100 driver genes, we observed a much lower percentage of functional enrichment for neurogenesis and developmental pathways compared to DeepVelo for the GABAergic lineage (Fisher's Exact Test $p = 4.578 \times 10^{-08}$) (Fig.5(e)), while the difference between the gliogenic results was non-significant. These functional pathway enrichment results highlight the relevance of the driver genes predicted by the DeepVelo method and increased functional relevance compared to those predicted by scVelo.

4 Discussion

DeepVelo offers a novel and more robust velocity estimation framework that goes beyond assumptions of constant RNA splicing and degradation rates, and instead estimates these rates at a cell-specific level. By analyzing the performance of DeepVelo and existing velocity estimation techniques, we have demonstrated that DeepVelo's cell-specific estimation through a novel deep

learning method allows for the detection and specification of multiple lineages in calculating RNA velocity. Realistic single-cell RNA sequencing settings will likely have more than one lineage/trajectory in a given sample, and thus it is imperative to develop methods that can account for these multifaceted dynamical systems. DeepVelo's ability to model these multifaceted dynamics was demonstrated through analysis of complex differentiation systems, such as the development of the dentate gyrus, pancreatic endocrinogenesis, and the hindbrain development. Lastly, we demonstrated that DeepVelo can be utilized to identify functionally relevant genes that are enriched along the differentiation trajectory of given lineages - potentially having a regulatory role that can be further validated. We envision that DeepVelo will be more readily applicable to these realistic developmental settings as compared to previous techniques.

The kinetic rates of transcription, splicing and degradation of mRNA determine the dynamics of RNA processing in cell differentiation. DeepVelo highlights this importance in cell-specific modeling of RNA velocities. Apart from this, we provide a systematic way to estimate the kinetic rates based on the fact that kinetics are correlated with the complete set of processes affecting the transcriptomic state of a cell. So, by using the entirety of the transcriptome-wide information from scRNA-seq data, one can approximately represent the cell status and estimate the kinetic rates accordingly. DeepVelo, as one of the first attempts, shows positive results of such an estimation, and we believe the cell-specific modelling of the transcriptional processes is a promising direction for inference of dynamical information.

DeepVelo internally predicts the first-order derivative of expression per gene based on the transcriptome-wide reads of all selected genes. The ability to learn the interaction/regulation between genes could be further explored, for example by replacing the GCN model with recent transformer networks [28] which could explicitly model the interaction of internal gene representations. This could allow for more interpretable velocity and driver-gene estimates, by considering correlations of kinetics and expression patterns between genes and cells. Recent work shows promising research directions by extending the velocity of cellular dynamics from RNA to proteins [8], epigenomics [26], and multi-omics velocities [20]. DeepVelo could be naturally updated and well fitted into these settings by enriching input and output space with additional -omics information. Ultimately, the estimation of cell-

specific kinetics across multiple steps in the central dogma may increase the signal-to-noise ratio [5] and further accurately capture information related to cellular development.

A major limitation of driver analysis through RNA velocity estimation is potentially spurious driver genes being picked up due to correlation of gene expression during differentiation. Although key regulators will display dynamical expression behavior during lineage specification, the same is likely to be true of their downstream targets and other "passenger" genes, resulting in high likelihoods towards being a driver/regulator. This is likely the reason why a significant transcription factor enrichment was not observed in the top 100 driver genes in the hindbrain developmental data for either scVelo or DeepVelo. We envision a more comprehensive driver analysis technique would take into account multi-lineage probabilities (preventing negative correlation between top drivers of two lineages) and would factor in correlations between driver analysis results, thereby removing spurious hits. Apart from the driver gene analysis, building up a theorem to verify the confidence of velocity estimation is another challenge. Empirical metrics, such as the consistency of velocity directions among neighbor cells, have been used in existing techniques [4, 24]. However, there is a lack of probabilistic tools to test the kinetics estimated by either previous methods or DeepVelo. We leave this to future works.

RNA velocity techniques have allowed for insights into biological differentiation from single-cell RNA sequencing data that go beyond the oversimplified trajectory inference models, and instead infer dynamic processes that indicate the direction and magnitude of differentiation potential. Although many major limitations and assumptions for RNA velocity methods still exist, we anticipate that continued methodological development in this relatively novel field will lead to better tools to study differentiation and development in a single-cell setting. DeepVelo overcomes limitations of previous techniques in a major aspect with regards to cell-specific model estimates, and can be used for more robust velocity estimation in multi-lineage systems, yielding better biological insights into real and complex developmental systems.

5 Online methods

5.1 Preprocessing the scRNA-seq data for DeepVelo

The dentate gyrus neurogenesis [11] and pancreatic endocrinogenesis [2] data are available at the National Center for Biotechnology Information's Gene Expression Omnibus repository. The accession number is GSE95753 and GSE132188. In this work, we use the zipped data of these two sequencing datasets provided by the scVelo packageBergen et al. [4](`https://scVelo.org`). The data is in h5py file format and contains spliced and unspliced gene readout.

Mouse hindbrain developmental data from Vladoiu et al. [29] was used to test velocity techniques for estimation at a lineage junction. As the data was not available in loom format for velocity analysis, fastq files were reprocessed into loom files using kallisto reference-free alignment through the loompy pipeline [6]. This was done individually for each timepoint (E10, E12, E14, E16, E18, P0, P5, P7, P14) and processed loom files were concatenated. For the purposes of the analysis, the junction between the GABAergic and gliogenic lineages was utilized. The following celltypes were subset from timepoints E10, E12, E14, E16, E18, P0, P5, P7, and P14 - Neural stem cells, Proliferating VZ progenitors, VZ progenitors, Differentiating GABA interneurons, gliogenic progenitors, and GABA interneurons. Estimates of spliced and unspliced counts from the kallisto quantification method were used for testing DeepVelo and scVelo.

Processing of unspliced and spliced counts in differing formats was done via three steps and uses the scVelo package. First, the spliced and unspliced gene matrices are normalized across genes. The preprocessing includes expression matrix normalization, nearest-neighbor-based smoothing. We used the `scv.pp.filter_and_normalize` from scVelo for these steps with default parameters. We select the top 2000 genes with the most spliced and unspliced gene counts across cells. The principal components are computed afterward using logarithmized spliced counts, and then we smooth the expression reads using the average of 30-nearest-neighbors for each cell.

5.2 Steady-state estimation of velocity

DeepVelo starts from the heuristics of initial velocity directions and then the deep learning model updates the prediction for future dynamics on the individual cell level. Here we employ the steady-state model [16] to compute the initial velocity directions. The steady-state model computes linear regression over spliced/unspliced expressions per gene and assigns positive velocity to cells with above-average unspliced reads, and vice versa. This computation is simple and fast and in experiments works well as the initial direction heuristics for DeepVelo. In the following, we first briefly review the concepts of RNA velocity and the steady-state model from La Manno et al. [16].

The transcriptional dynamics depicts the process from generation to degradation of mRNA molecules. It captures unspliced premature mRNAs u(t) with transcription rate α, its splicing into mature mRNAs $s(t)$ with rate β and the degradation of spliced mRNA with rate γ. The simplified gene-specific dynamics with constant splicing and degradation rates are

$$\begin{array}{rcl} \frac{du(t)}{dt} & = & \alpha\left(t\right)-\beta u\left(t\right), \\ \frac{ds(t)}{dt} & = & \beta u\left(t\right)-\gamma s\left(t\right). \end{array} \tag{2}$$

The derivative equations depict the changes of mRNA abundances over time. RNA velocity is defined as the time derivative of spliced mRNA, i.e. $\nu\left(t\right):=\frac{ds(t)}{dt}$. In, the steady-state model, it assumes most observed cells are in saturation stages ($\nu(t)=0$) and the splicing rate β constantly equals to 1. Then the average degradation rate $\tilde{\gamma}$ is computed as the ratio of $\tilde{\gamma}=\frac{(u-\bar{u})^{T}(s-\bar{s})}{||s-\bar{s}||^{2}}$, where u and s are the unspliced and spliced reads of all cells. $\bar{u}$ and $\bar{s}$ denote the average. Then the velocity for a cell i is computed as the difference to the average ratio:

$$\tilde{\nu}_{i}=u_{i}-\tilde{\gamma}s_{i}, \tag{3}$$

where u_i and s_i are the unspliced and spliced reads for the cell.

5.3 Modeling individual transcriptional dynamics

Taking into consideration that the steady-state model heavily depends on the often violated steady-state assumption, and it omits the difference of kinetic rates (α, β, γ) across celltypes, we propose a new method using deep learning to capture individual cell kinetics. DeepVelo assumes that the sequencing data captures a continuous spectrum of cells in consecutive development stages, so that the changing rate of mRNA abundance, i.e. the velocity, could be well captured by the difference of mRNA reads within developmentally close-by cells. We term this the **continuity assumption**. In comparison to the previous strict assumptions (i.e. the observation of steady states or the global constant rates) in existing approaches, the continuity is commonly satisfied in sequencing data of large cell populations; hence, it broadens the scope of application for DeepVelo.

Given a spectrum of cell population satisfying the continuity assumption, DeepVelo learns cell-specific kinetic rates and predict velocities at current cellular time t. The velocities extrapolate the gene expressions to the next time-point to match the future cell states at $t+1$. The future cell states are depicted by possible cell expressions in the sequencing data. The details of this computation pipeline are as follows:

First, we build a graph convolutional network model to predict cell-specific kinetic rates. In this work, we build a nearest neighbor graph based on the expressions of all sequenced cells $G = (\mathcal{V}, \mathcal{E})$. The vertex $\mathfrak{v}_i \in \mathcal{V}$ in the graph denotes the expression reads of a cell i, which include its spliced and unspliced gene expressions $\mathfrak{v}_i = [s_i, u_i]$. A cell i is connected to cell j (i.e. $\mathcal{E}_{ij} = 1$) if cell j is the one of top 30 nearest neighbors based on the Euclidean distance of the gene expressions. We input this neighbor graph to DeepVelo. We chose the graph representation because it considers the vicinity of local cell expressions. This has more expressive power than the expressions of individual cells because of the sparse and noisy nature of gene expressions. Taking the neighborhood expressions into smooth the velocity estimation.

Graph convolutional network (GCN) is a type of deep neural networks that learns node embeddings based on message passing along the graph edges [15]. Given a graph with nodes V and adjacency matrix A, a multi-layer

neural network is constructed on the graph with the following layer-wise propagation rule:

$$H^{(l+1)} = \sigma(\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}H^{(l)}W^{(l)}), \tag{4}$$

where $H^{(l)}$ denotes the node feature vectors at the l-th layer, $\tilde{A} = A + I_N$ is the adjacency matrix with self-connections, D is the diagonal degree matrix such that $D_{ii} = \sum_j \tilde{A}_{ij}$, $W^{(l)}$ is the layer-specific trainable parameter matrix, σ is the RELU activation function.

In this work, the input feature $H^{(0)} \in \mathbb{R}^{N\times 2D}$ to GCN is the cellular gene matrix. Each row in H stands for the aforementioned vertex $\mathfrak{v}_i$. H contains the population of N cells and the dimension $2D$ equals the number of selected spliced and unspliced genes combined, $D = 2000$ by default. The adjacency matrix $A \in \mathbb{R}^{N\times N}$ depicts the aforementioned nearest neighbor graph, where the element at position i, j has value 1 if the cell j is one of the nearest neighbors of cell i, otherwise the value is 0. The GCN model consists of stacked graph convolution layers, i.e. Eq. 4. The output of the final layer H^L is processed by a fully connected neural network, which then yields the estimated velocity parameters $\alpha \in \mathbb{R}^{N\times D}$, $\beta \in \mathbb{R}^{N\times D}$ and $\gamma \in \mathbb{R}^{N\times D}$ for all cells and genes.

Finally, the estimated velocity $v_i \in \mathbb{R}^D$ for each cell is computed as

$$v_i = \beta_i u_i - \gamma_i s_i, \tag{5}$$

where β_i and γ_i are the i-th row in β and γ, u_i and s_i are the unspliced and spliced reads of cell i.

DeepVelo also supports estimation of the derivative of unspliced RNA, namely v_i^{uns}, which is an estimation for the $\frac{du(t)}{dt}$ in Eq.1.

$$v_i^{uns} = \alpha_i - \beta_i s_i.$$

5.4 Training the DeepVelo model

To optimize the parameters in the DeepVelo model, we introduce a combined loss function with two terms.

$$\mathscr{L} = \underbrace{\mathscr{L}_r}_{\text{regression loss}} + \lambda \underbrace{\mathscr{L}_c}_{\text{constraint loss}} \tag{6}$$

The first term, named the regression loss, depicts the difference between the estimated future cellular state $s_i + v_i$ and the average cellular state of downstream neighbors. Minimization of this loss promotes the estimated velocities to match the observations of cellular state distribution in the dataset.

$$\mathscr{L}_r(cell_i) = (s_i + v_i - \frac{1}{|\mathscr{N}_i^+|} \sum_{j \in \mathscr{N}_i^+} s_j)^2, \tag{7}$$

where $\mathscr{N}_i^+$ is the set of cells that have the expression profile most comparable to the future cell state of i. This is selected by computing the Euclidean distance between the extrapolation of cell i with initial direction $\tilde{\nu}_i$ (Eq.3) and every cell j in the sequenced population, i.e. $||s_i + \tilde{\nu}_i - s_j||, \forall j \in U$. Then the k cells with nearest distances are selected. We use $k = 30$ by default.

The second term, constraint loss, smooths the velocities among similar cells by minimizing the variance of velocity vectors.

$$\mathscr{L}_c(cell_i) = \sum_{m=1}^{M_{gene}} \mathrm{Var}[v_j^{(m)}], j \in \mathscr{N}_i^{(m)}, \tag{8}$$

where $\mathscr{N}_i^{(m)}$ denotes the set of top k cells that have the most similar expression counts on gene m as cell i (including cell i itself). For each sequencing dataset, the DeepVelo model is trained by gradient back-propagation using the Adam [14] optimizer for 100 to 150 epochs. The updated model at the last epoch is used to compute the estimated velocities.

5.5 Computing cell-to-cell connectivity graph

The similarity of velocity vectors of cells could model cell-to-cell connectivities. We use the connectivity graph for downstream tasks including driver gene analysis and developmental trajectory inference.

The weight in the connectivity graph, w_{ij} denotes the estimated magnitude of connection. Higher w_{ij} means the future state of cell i is close to the current state of cell j. w_{ij} could be computed by possible similarity measures

between velocity v_i and the gene expression difference $s_j - s_i$. Here, we used the cosine similarity, which is also adopted in scVelo [4], therefore,

$$w_{ij} = \frac{v_i^T(s_j - s_i)}{||v_i|| \cdot ||s_j - s_i||}.$$

For the visualization of the velocity plot, we adopted the same projection computation provided by exiting methods [16, 4] to project velocity as arrows onto low-dimensional embeddings, such as tsne and UMAP. To summarize, the transition probability $\pi_{i,j}$ from a cell i to possible target cell j is computed by the Gaussian normalized connectivity weight w_{ij}. Then the velocity vector for v_i in a low-dimensional space is computed by the weighted sum of $\sum_j \pi_{i,j}\delta_{ij}$, where δ_{ij} is the direction vector pointing from cell i to j in the low-dimensional space.

5.6 Driver gene estimation and comparison

To determine functional signals in the driver genes, the top 100 genes based on a correlation with each lineage were determined, in particular for the hindbrain developmental data from [29]. Overlap with marker genes based on the original analysis used to annotate celltypes was performed, as well as overlap with transcription factors. Transcription factors were pulled from the manually annotated Human Transcription Factors list curated by Lambert et al., and were lifted over to mouse data using orthologous gene-matches [17].

Analysis of marker overlap was further extended by determining the correlation values of markers in the top 100 driver gene lists for both scVelo and DeepVelo per lineage in the Vladoiu et al. [29] data. The density of correlation values for marker genes in the top 100 list was analyzed for significant differences using the Kolmogorov-Smirnov (KS) non-parametric test for significant distributional differences. The KS test utilizes the maximum difference between the cumulative distributions of the data as a test statistic. The two-sided version of the test was used in this case, allowing either DeepVelo or scVelo to have a greater or lesser density of correlations.

5.7 Pathway enrichment analysis

To determine functional signal in the driver gene results, pathway enrichment analysis was done using the ActivePathways R package [22]. The top 100 driver genes, based on correlation values for both the GABAergic and gliogenic lineages from the Vladoiu et al. [29] data, were input into the ActivePathways gene-set enrichment analysis model. The latest Gene-Matrix-Transposed (GMT) files containing gene-set information from the Gene Ontology Molecular Function, GO Biological Process, and REACTOME databases were used [7, 13]. Pathways were labelled as being involved in "Neurogenesis", as "Developmental non-neuronal", and "Non-specific" using manual annotation and the presence of known terms (such as "neuron projection" or "proliferation" for "Neurogenesis" and "Developmental non-neuronal", respectively). "Non-specific" pathways indicated those that did not have immediately obvious roles in either neurogenesis or general development. To determine significant differences between pathway labelling and potential enrichment of neurogenic/development specific pathways, a two-sided Fisher's exact test based on the hypergeometric distribution was done for the contingency table comprising of scVelo and DeepVelo pathway results and functional labels ("Neurogenesis", "Developmental non-neuronal", "Non-specific") for the gliogenic and GABAergic lineages independently.

References

[1] Benedetta Artegiani et al. "A single-cell RNA sequencing study reveals cellular and molecular dynamics of the hippocampal neurogenic niche". In: *Cell reports* 21.11 (2017), pp. 3271–3284.

[2] Aimée Bastidas-Ponce et al. "Comprehensive single cell mRNA profiling reveals a detailed roadmap for pancreatic endocrinogenesis". In: *Development* 146.12 (2019).

[3] Nico Battich, Thomas Stoeger, and Lucas Pelkmans. "Control of transcript variability in single mammalian cells". In: *Cell* 163.7 (2015), pp. 1596–1610.

[4] Volker Bergen et al. "Generalizing RNA velocity to transient cell states through dynamical modeling". In: *Nature biotechnology* (2020), pp. 1–7.

[5] Volker Bergen et al. "RNA velocity—current challenges and future perspectives". In: *Molecular systems biology* 17.8 (2021), e10282.

[6] Nicolas L. Bray et al. "Near-optimal probabilistic RNA-seq quantification". In: *Nature Biotechnology* 34.5 (2016), pp. 525–527.

[7] S. Carbon et al. "The Gene Ontology Resource: 20 years and still GOing strong". In: *Nucleic Acids Research* 47 (2019), pp. D330–D338.

[8] Gennady Gorin, Valentine Svensson, and Lior Pachter. "Protein velocity and acceleration from single-cell multiomics experiments". In: *Genome biology* 21.1 (2020), pp. 1–6.

[9] Gennady Gorin et al. "RNA velocity unraveled". In: *bioRxiv* (2022).

[10] Laleh Haghverdi et al. "Diffusion pseudotime robustly reconstructs lineage branching". In: *Nature methods* 13.10 (2016), pp. 845–848.

[11] Hannah Hochgerner et al. "Conserved properties of dentate gyrus neurogenesis across postnatal development revealed by single-cell RNA sequencing". In: *Nature neuroscience* 21.2 (2018), pp. 290–299.

[12] Ian S. Hsu and Alan M. Moses. "Stochastic models for single-cell data: Current challenges and the way forward". In: *The FEBS Journal* 289.3 (), pp. 647–658.

[13] Bijay Jassal et al. "The reactome pathway knowledgebase". In: *Nucleic Acids Research* 48 (2020), pp. D498–D503.

[14] Diederik P. Kingma and Jimmy Ba. "Adam: A Method for Stochastic Optimization". In: *CoRR* abs/1412.6980 (2015).

[15] Thomas N Kipf and Max Welling. "Semi-supervised classification with graph convolutional networks". In: *arXiv preprint arXiv:1609.02907* (2016).

[16] Gioele La Manno et al. "RNA velocity of single cells". In: *Nature* 560.7719 (2018), pp. 494–498.

[17] Samuel A. Lambert et al. "The Human Transcription Factors". In: *Cell* 172.4 (2018), pp. 650–665.

[18] Marius Lange et al. "CellRank for directed single-cell fate mapping". In: *BioRxiv* (2020).

[19] Anton J.M. Larsson et al. "Genomic encoding of transcriptional burst kinetics". In: *Nature* 565.7738 (2019), pp. 251–254.

[20] Chen Li et al. "Single-cell multi-omic velocity infers dynamic and decoupled gene regulation". In: *bioRxiv* (2021).

[21] Leland McInnes, John Healy, and James Melville. "Umap: Uniform manifold approximation and projection for dimension reduction". In: *arXiv preprint arXiv:1802.03426* (2018).

[22] Marta Paczkowska et al. "Integrative pathway enrichment analysis of multivariate omics data". In: *Nature Communications* 11.1 (2020), pp. 1–16.

[23] Andrea Pillai et al. "Lhx1 and Lhx5 maintain the inhibitory-neurotransmitter status of interneurons in the dorsal spinal cord". In: *Development* 134.2 (2007), pp. 357–366.

[24] Chen Qiao and Yuanhua Huang. "Representation learning of RNA velocity reveals robust cell transitions". In: *Proceedings of the National Academy of Sciences* 118.49 (2021).

[25] Xiaojie Qiu et al. "Mapping transcriptomic vector fields of single cells". In: *Cell* 185.4 (2022), pp. 690–711.

[26] Martina Tedesco et al. "Chromatin Velocity reveals epigenetic dynamics by single-cell profiling of heterochromatin and euchromatin". In: *Nature Biotechnology* 40.2 (2022), pp. 235–244.

[27] Svetlana Tutukova, Victor Tarabykin, and Luis R. Hernandez-Miranda. "The Role of Neurod Genes in Brain Development, Function, and Disease". In: *Frontiers in Molecular Neuroscience* 14.June (2021), pp. 1–13.

[28] Ashish Vaswani et al. "Attention is all you need". In: *Advances in neural information processing systems* 30 (2017).

[29] Maria C Vladoiu et al. "Childhood cerebellar tumours mirror conserved fetal transcriptional programs". In: *Nature* 572.7767 (2019), pp. 67–73.

[30] Keng Ioi Vong et al. "Sox9 is critical for suppression of neurogenesis but not initiation of gliogenesis in the cerebellum". In: *Molecular Brain* 8.1 (2015), pp. 0–17.

[31] F Alexander Wolf et al. "PAGA: graph abstraction reconciles clustering with trajectory inference through a topology preserving map of single cells". In: *Genome biology* 20.1 (2019), pp. 1–9.

[32] Yuanyuan Wu et al. "Hes1 but not Hes5 regulates an astrocyte versus oligodendrocyte fate choice in glial restricted precursors". In: *Developmental Dynamics* 226.4 (2003), pp. 675–689.

[33] Norliyana Zainolabidin et al. "Distinct activities of tfap2a and tfap2b in the specification of GABaergic interneurons in the developing cerebellum". In: *Frontiers in Molecular Neuroscience* 10.August (2017), pp. 1–14.

6 Acknowledgements

We thank Lin Zhang, Mehran Karimzadeh, Phil Fradkin, and Shihao (Rex) Ma of the Bo Wang Lab and Jiao Zhang of the Michael Taylor Lab for their insightful comments on the manuscript.

I Supplementary

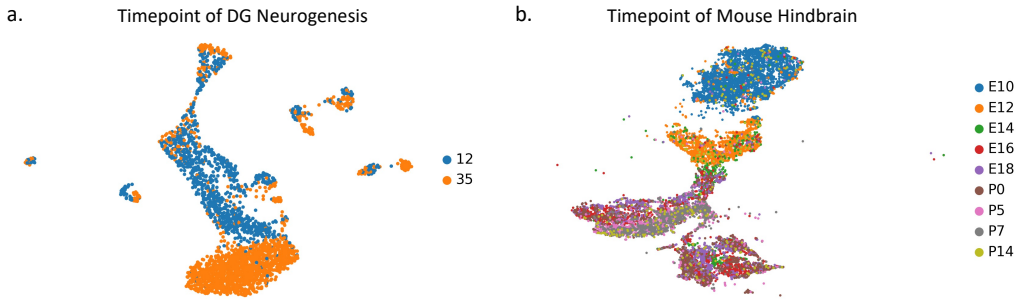

Figure S1: **The time points of sequenced cells in datasets.** (a) The dentate gyrus neurogenesis data. (b) The mouse hindbrain data

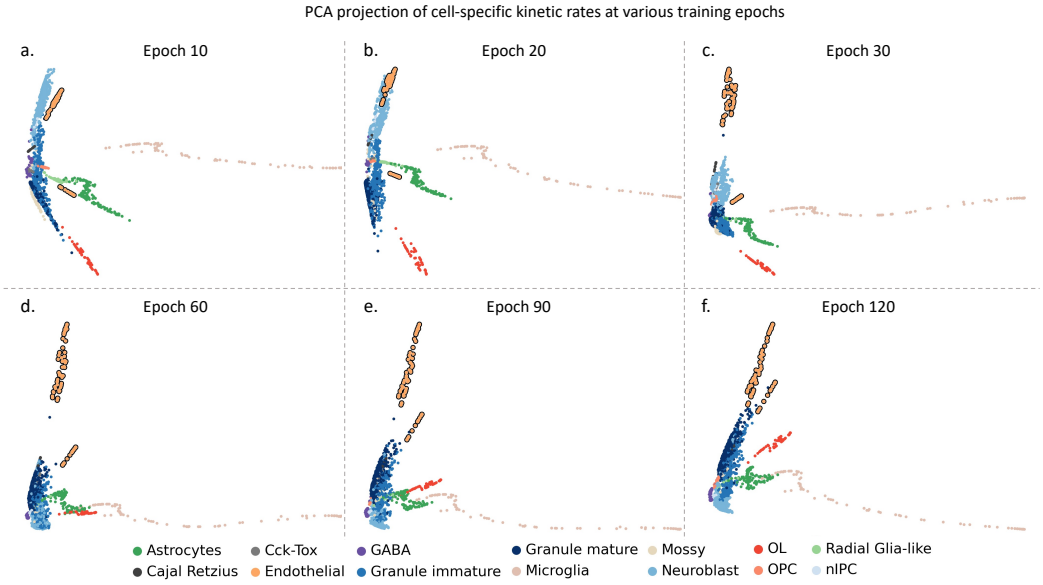

Figure S2: **The PCA projection of cell-specific kinetic rates at various training epochs.** (a-f) Scatter plot of the first two PCA dimensions at training epochs 10, 20, 30, 60, 90, 120. DeepVelo learns to predict similar kinetic rates for cells of same celltype. For example, the kinetic rates of Endothelial cells (outlined) are gradually clustered together and locate away from the less-related granule lineage.

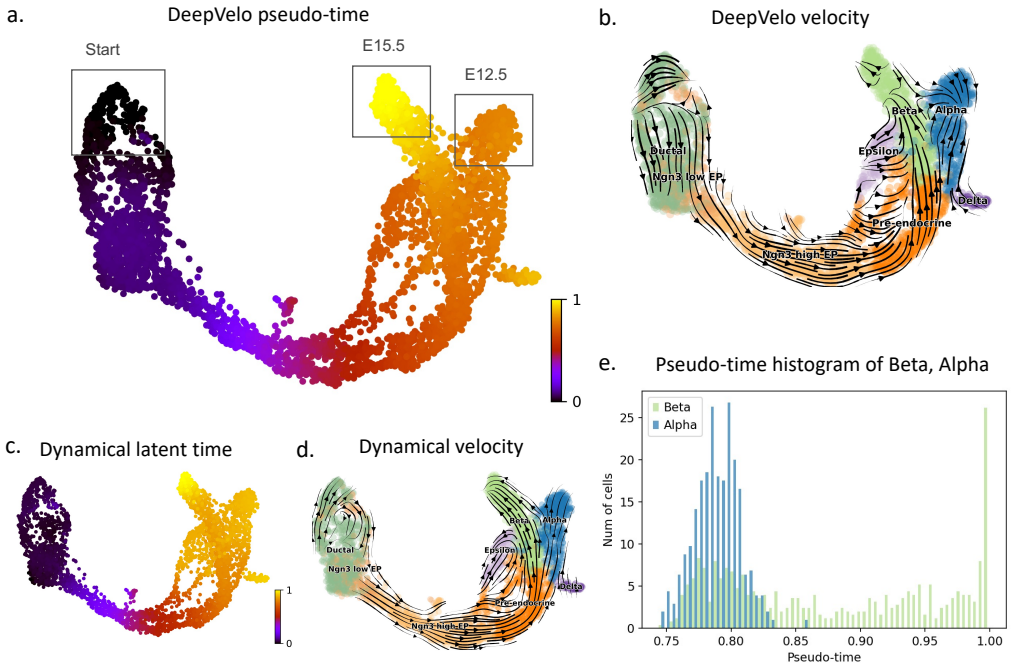

Figure S3: **Velocity and pseudo-time plots for pancreatic endocrinogenesis [2].** (a) The pseudo-time prediction from DeepVelo accurately assignes time indices to different celltypes. Particularly, the starting cell cluster correctly locates at the upper left. Also, the difference between the terminal celltypes are well captured, where the group of Alpha-cells were developed earlier then E12.5 and the Beta-cells appeared later until E15.5. (b) Velocity values derived from DeepVelo are projected onto the UMAP-based embedding and visualized in arrows. DeepVelo successfully captures the main stream of EP cells developing into terminal celltypes of alpha, beta and delta cells. (c),(d) For comparison, the latent time and velocity computed by the dynamical model from scVelo. (e) The histogram of pseudo-time of Beta and Alpha cells, using DeepVelo estimated velocities. Beta cells have a larger percentage of cells with higher pseudo-time values, which correctly reflects the later development of these cells at E15.5

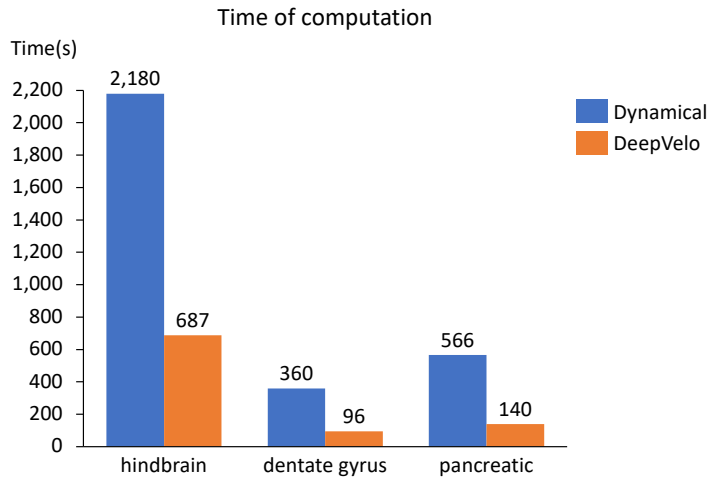

Figure S4: **Time comparison across datasets.** Comparison of computation time between Dynamical and DeepVelo on three datasets. Here we train the DeepVelo model on CPU without GPU accelaration. DeepVelo has a 3-5 folds acceleration compared to the dynamical model from scVelo using the same computation resources.

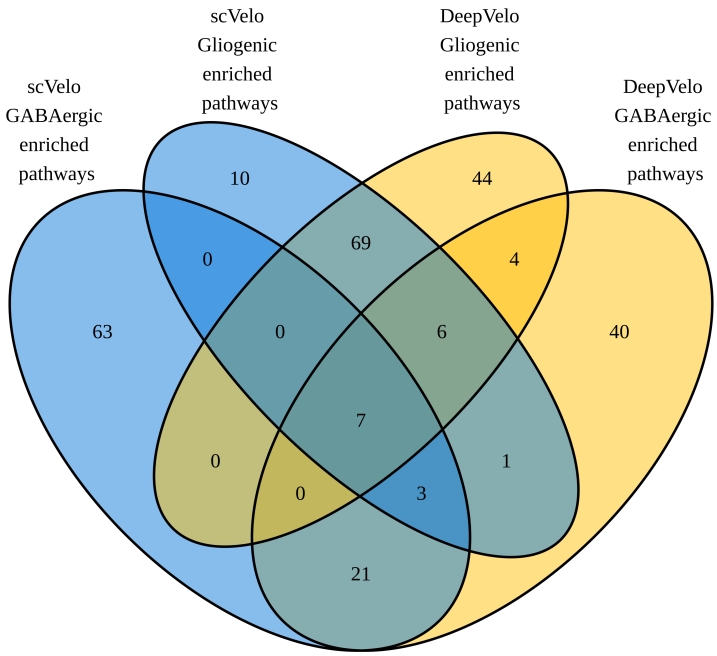

Figure S5: **Full pathway enrichment analysis results overlap.** Overlap of scVelo and DeepVelo pathway enrichment analysis results, between methods, for the top 100 GABAergic and gliogenic driver genes.