NeuroVelo: interpretable learning of temporal cellular dynamics from single-cell data

Idris Kouadri Boudjelthia[1,2], Salvatore Milite[3], Nour El Kazwini[1], Yuanhua Huang[4,5], Andrea Sottoriva[3], Guido Sanguinetti[1,6*]

[1*]Data Science, SISSA, Trieste, Italy.
[2]Abdus Salam International Centre for Theoretical Physics, Trieste, Italy.
[3]Computational Biology Research Centre, Human Technopole, Milan, Italy.
[4]School of Biomedical Sciences, University of Hong Kong, Hong Kong, China.
[5]Department of Statistics and Actuarial Science, University of Hong Kong, Hong Kong, China.
[6]School of Informatics, University of Edinburgh, Edinburgh, UK.

*Corresponding author(s). E-mail(s): gsanguin@sissa.it;
Contributing authors: ikouadri@sissa.it; salvatore.milite@fht.org; nelkazwi@sissa.it; yuanhua@hku.hk; andrea.sottoriva@fht.org;

Abstract

Reconstructing temporal cellular dynamics from static single-cell transcriptomics remains a major challenge. Methods based on RNA velocity are useful, but interpreting their results to learn new biology remains difficult, and their predictive power is limited. Here we propose NeuroVelo, a method that couples learning of an optimal linear projection with non-linear Neural Ordinary Differential Equations. Unlike current methods, it uses dynamical systems theory to model biological processes over time, hence NeuroVelo can identify gene interactions that drive the observed temporal dynamics of gene expression. We benchmark NeuroVelo against several state-of-the-art methods using single-cell datasets, demonstrating that NeuroVelo simultaneously reconstructs correct cell-type transitions and identifies gene regulatory networks that drive cell fate directly from the data.

Keywords: single-cell transcriptomics, RNA velocity, neural networks, dynamical systems

1 Introduction

Single-cell transcriptomic (scRNA-seq) technologies have transformed our understanding of cellular diversity and heterogeneity[1], yet the destructive nature of the measurement process poses fundamental limits to their ability to capture temporal biological dynamics. Inferring dynamic information from static snapshots of scRNA-seq has been a major focus of computational research in the last decade. While most early efforts used advanced machine learning techniques to order cells along a pseudo-time trajectory [2–5], more recently the concept of RNA-velocity [6] offered a more mechanistic, biophysically grounded approach to solve the problem.

RNA-velocity leverages the surprisingly abundant pre-mRNA reads present in many scRNA-seq data sets to parametrise a simple first order kinetic model of RNA production and splicing. The relative abundance of spliced and unspliced reads in a given cell can then be used to deduce whether a gene is being transcriptionally activated or repressed. This idea enables researchers to capture a shadow of cellular dynamics even in static data sets. Recent years have witnessed a flourishing of RNA velocity approaches [7–12], permitting a simultaneous inference of pseudo-time and RNA velocity, and coupling the velocity equations with non-linear transformations that can better model the complexity of the data. Nevertheless, both the biochemical foundations and the biological interpretation of RNA-velocity have been questioned [13, 14]. In particular, while RNA velocity enables an intuitive grasp of cellular dynamics at the level of individual cells, much less can be said at the level of the gene regulatory networks (GRNs) which underpin such dynamics, which in general can only be obtained by combining RNA-velocity analyses with post-processing with bespoke GRN inference methods.[15].

Here we propose NeuroVelo, a new approach which bridges the cellular and gene scales, simultaneously inferring temporal cellular dynamics and major gene interactions that drive such dynamics from scRNA-seq data. NeuroVelo is an approach grounded in dynamic system theory that combines ideas from Neural Ordinary Differential Equations (ODE) [16, 17] and RNA velocity in a physics-informed neural network architecture. We test extensively NeuroVelo on five scRNA-seq data sets and compare its performance against six recently published methods. On all data sets NeuroVelo achieved state of the art performance under a variety of metrics. Uniquely to our approach, we demonstrate how the interpretation of NeuroVelo results recovers highly accurate regulatory networks and biological processes underpinning the dynamics in a natural and mathematically consistent way.

2 Results

2.1 The NeuroVelo model

We first present a high-level overview of our method, focussing on its main features and innovative aspects. A detailed mathematical description of the model is provided in the Methods section. Figure 1(a) illustrates the basic concept of NeuroVelo. scRNA-seq data consisting of both spliced and unspliced reads are encoded to a low dimensional space through two dimensionality reduction channels: one is a non-linear 1D encoder

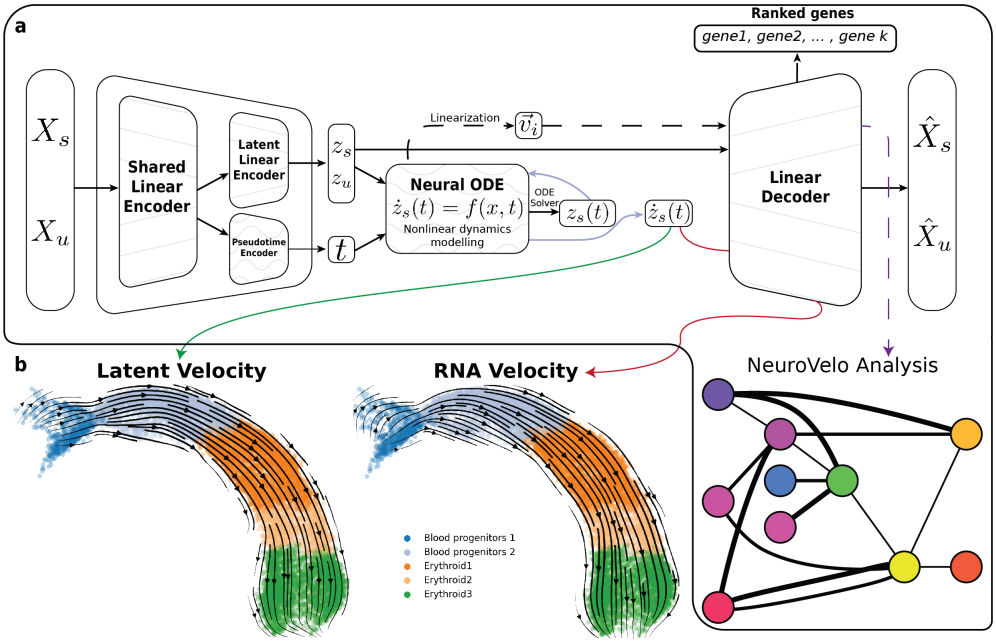

Fig. 1 (a) Schematic representation of the neuroVelo model: two autoencoders, a linear one to define the phase space, and a non-linear 1-D one to define pseudotime, are coupled in latent space through a non-linear neural ODE. RNA-velocity constraints are added as an additional penalty in the loss function. (b) Example output of velocity field in the latent space and in the gene space on the mouse gastrulation dataset projected to the UMAP-based embedding

learning a pseudo-time coordinate associated with each cell, while the second is a linear projection to an *effective phase space* for the system. The observed single-cell transcriptomes are modelled as discrete-time snapshots of a continuous time trajectory happening on this latent phase space.

Cellular dynamics are defined through an autonomous system of differential equations where the rate of change of the state variables (right hand side of the ODE) is parametrised by a neural network (a neural ODE). The resulting dynamics can be extremely rich: since neural networks are universal function approximators, a neural ODE with a sufficiently large network may give rise to any first order dynamics. To ground biophysically these dynamics and avoid overfitting, we impose the RNA velocity principle as a further penalty in the loss function in the spirit of physics-informed neural networks. This introduces a trade-off in the fitting procedure: the parameters of the neural ODE and of the projections must strike a balance between providing an accurate interpolation of the observed transcriptomes, and maintain a balance of predicted spliced/ unspliced reads consistent with the RNA velocity principle. Notice that this constraint is applied locally to each cell, thus removing limiting assumptions on global transcriptional dynamics which are often a major problem for RNA velocity approaches. Because the effective phase space is defined via a linear projection, the RNA velocity constraint can be applied *exactly* in the reduced dimensional space.

The choice of defining the effective phase space as a linear subspace of gene space, while it may limit the expressivity of the model, brings additional significant advantages: it avoids confounding the non-linear dynamics in the effective phase space with the effects of a non-linear projection, leading to more robust results (see Supp. Fig. S1). Most importantly, a major benefit of the linearity of the phase space is that standard techniques for the analysis of low-dimensional non-linear dynamical systems can be applied to the trained model. The resulting insights, thanks to the linearity of the encoder/ decoder model, can readily be translated in terms of genes and regulatory interactions.

The main tool we use to interpret the regulatory interactions underlying the dynamics is the Jacobian matrix of the system. The Jacobian is obtained by taking first derivatives of the neural network with respect to the state variables, and it tells us how small changes in state variables affect their rates of change. The leading eigenvectors -corresponding to the largest eigenvalues in magnitude- highlight the direction in phase space that contributes most to the system's dynamics. In principle, both the Jacobian and its eigenvectors are time dependent. By decoding the Jacobian matrix in gene space, we can obtain estimates of how a gene affects the transcription rate of another gene at a particular time, hence providing a method to infer time-varying gene regulatory networks. Decoding the leading eigenvectors of the Jacobian into gene space provides ranked lists of important genes that can then be used for enrichment analyses and model validation.

2.2 Datasets used, competing methods and evaluation metrics

To test NeuroVelo, we utilize a total of five recent scRNA-seq data sets:

- a data set of erythroid cells differentiation [18] during mouse gastrulation, comprising of 9815 cells quantifying a median of 2791 non-zero expressed genes per cell;
- a human bone marrow hematopoiesis data set [5], widely used in RNA-velocity studies, which quantifies a median of 1190 non-zero expressed genes in 5780 individual cells;
- a pancreatic endocrinogenesis [19] data set measuring a median of 2447 non-zero expressed genes in 3696 cells;
- two smaller CRISPR-based lineage barcoded mouse cancer data sets [20], measuring gene expression in 754 and 1082 cells at a higher depth (3490 and 2352 median non-zero expressed genes respectively.

To contextualise the performance of NeuroVelo, we analysed each data set with a number of recently proposed methods to reconstruct cellular dynamics. Perhaps the most widely used RNA-velocity method is scVelo [7], which remains the main benchmark. Very recently, multiple methods extended the RNA-velocity framework in several directions: we choose here UniTVelo [9], which focusses on improving the pseudotime estimation, and latentVelo [21] and deepVelo [8], which couples RNA-velocity with non-linear dimensionality reduction. We also use scTour [17], which does not use RNA-velocity but directly fits a neural ODE to the data, as a splicing-independent comparison. On the lineage barcoded data, we use PhyloVelo [20], a specialised method

which exploits barcode information to obtain robust estimates of pseudotime. As performance metrics, we use the cross-boundary direction (CBDir) and the in-cluster coherence (ICCoh), two widely used metrics which summarise the consistency of the inferred dynamics with known biological dynamics (in data sets where cell-type annotation is present). For all methods, we report the best performance across several random restarts, when a default initialisation is not provided. Table 1 provides a comprehensive assessment of all methods on all data sets with respect to the inferred RNA-velocity fields.

To benchmark the ability of NeuroVelo to reconstruct GRNs, we compare its performance to GENIE3 [22] and GRNboost [23], two machine learning-based methods which were highlighted as state of the art in a recent review [24]. Assessing the performance of GRN methods on real data is notoriously difficult, due to the lack of gold standards (and indeed the difficulty in associating precise molecular semantics to the interactions summarised by a network edge). To overcome this hurdle, we assembled a data base of ChIP-seq tracks from ChIP-Atlas [25], a number of transcription factors in a number of cell types which are relevant in gastrulation and hematopoiesis, the two most studied processes we consider. A list of all data sets used with accession codes is provided in supplementary table ST 1 and ST 2. This enables us to obtain a ground truth network for each cell type, where the various methods can be assessed using standard metrics such as area under the precision-recall (PR) curve.

2.3 Reconstructing dynamical mechanisms from scRNA-seq data

We start by assessing NeuroVelo performance on the three standard (non-barcoded) scRNA-seq, focussing on the ability to reconstruct lineage differentiation and GRNs.

An analysis of the erythroid lineage in mouse gastrulation is shown in Figure 2. NeuroVelo effectively captures the single-branched nature of the process, providing the right cell-type transitions of the dynamics both in latent space and in gene space (Figure 2a), and a very good sequential ordering of the different cell types (Figure 2c). The quality of the inferred dynamics is further supported by Figure 2(d-e), which plots histograms of the evaluation metrics for each cell type, clearly showing NeuroVelo to provide state-of-the art performance on every cell type.

Figure 2f-g instead concern the interpretation of the decoded dynamics. The barplots in Figure 2f show the normalised area under PR curve for NeuroVelo, Genie3 and GRNboost for eight cell types which are associated with gastrulation (the normalisation is such that the random predictor would return a score of one). NeuroVelo returns the best performance in half of the cases, with a particularly marked improvement for two of the most relevant cell lines, embryonic erythroblasts and mESC derived erythroid cells. It should be noticed that NeuroVelo is agnostic with respect to which genes are transcription factors, while both Genie3 and GRNboost require as input the list of transcription factor genes, thus significantly narrowing the search space for their models. Figure 2b instead shows an enrichment analysis for NeuroVelo (using the ranked list of genes provided by the eigenvectors of the Jacobian) and for scVelo, using the list of driver genes in its dynamical model setting. The two analyses broadly

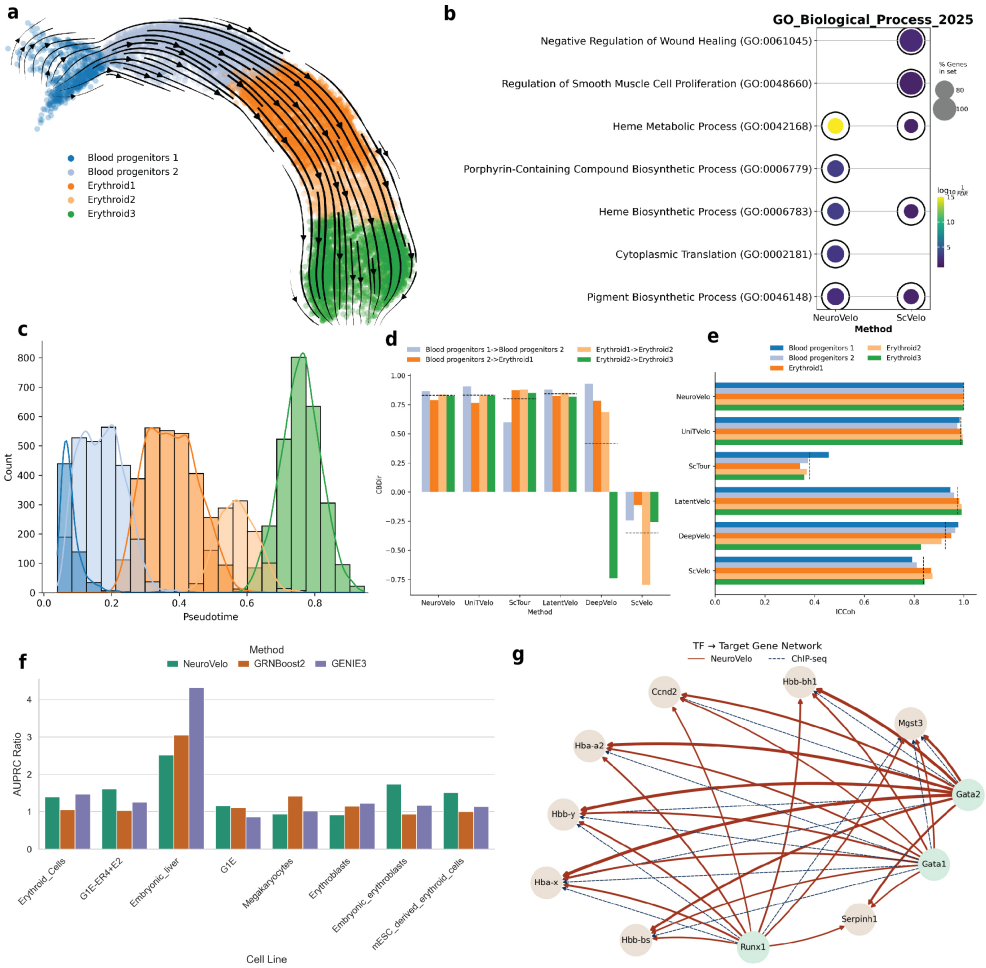

Fig. 2 (a) Velocity field in gene space on the mouse gastrulation dataset projected to the UMAP-based embedding. (b) Comparison on gene set enrichment analysis of the gene influencing the dynamics. (c) Histogram of Inferred pseudotime. (d) Comparison on the cross boundary direction. (e) Comparison on the inter-cluster coherence. (f) Comparison on AUPRC ratio for the gene regulatory network inference. (g) Gene regulatory network inferred by NeuroVelo along with connections from Chip-seq data.

overlap (see also Suppl. Fig. S3 for a comparison of the two gene rankings), although neuroVelo returns larger gene sets with a slightly higher significance level.

A similar picture emerges from the analysis of the human bone marrow data set in Figure 3. NeuroVelo provides the right cell-type transitions and a correct pseudotemporal ordering (Figure 3(a,c)), mirrored in state-of-the-art performance under both metrics (Figure 3 d-e). An enrichment analysis returns several highly enriched pathways related to immune cell proliferation and differentiation (Figure 3b), while in this

case the gene ontology analysis was not able to identify meaningful biological signals from the scVelo driver genes set.

Figure 3f again shows the agreement of the inferred networks with the ChIP-seq derived gold-standard in a number of relevant cell types. NeuroVelo achieves best performance in four out of eight cell types, including the most relevant cell types (hematopoietic cells and pre-B cells). Interestingly, in this well annotated data set we can also linearise about different points in latent space, producing cell-type specific networks which highlight the varying temporal dynamics in the process (see Supplementary Movies).

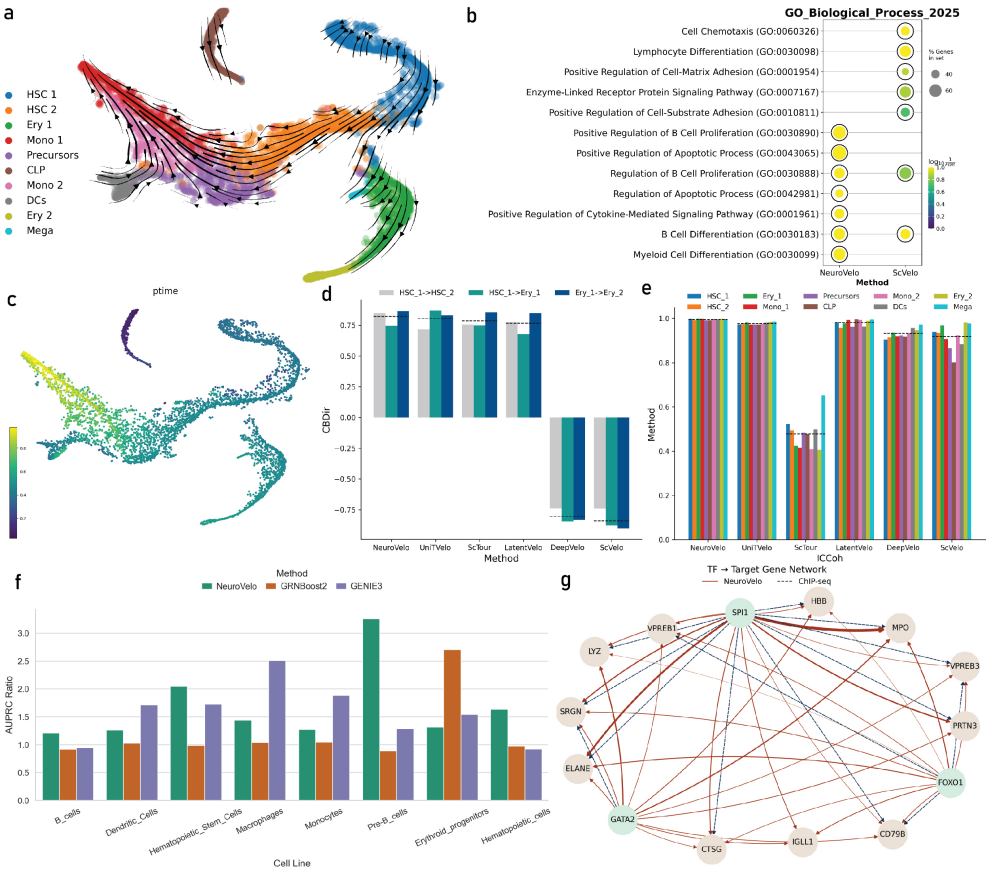

Fig. 3 (a) UMAP of NeuroVelo velocity field on the human bone marrow dataset, as well as (c) NeuroVelo pseudotime. (b) Comparison on gene set enrichment analysis of the gene influencing the dynamics. (d) Comparison on the cross boundary direction. (e) Comparison on the inter-cluster coherence. (f) Comparison on AUPRC ratio for the gene regulatory network inference. (g) Gene regulatory network inferred by NeuroVelo along with connections from Chip-seq data.

As a third example of neuroVelo analysis on standard scRNA-seq, we examined the pancreatic endocrinogenesis data set. Results are shown in Figure 4; once again, neuroVelo provides an excellent visualisation (Figure 4a), and performs best or among the best for all cell types under the metrics considered (Figure 4c-d). An enrichment analysis (Figure 4 b,e) highlights highly relevant pathways connected with pancreas function with very high normalised enrichment score. An analysis of the interactions predicted again highlights some interactions already detailed in the literature (Figure 4 f-g), however it is noticeable that in this less well studied scenario fewer interactions can be derived from the literature, underscoring the potential for data-driven approaches like neuroVelo to uncover novel biological hypotheses.

Taken together, these results demonstrate that neuroVelo achieves state of the art or is competitive with the best RNA-velocity methods on a variety of data sets. Additionally, neuroVelo is able to predict (possibly time-varying) interaction mechanisms which can explain the observed transcriptome dynamics, and which frequently recapitulate experimentally observed interactions in well characterised systems.

2.4 Inference of fitness in lineage barcoded data

As a second set of benchmark data, we consider two lineage barcoded data sets which were recently published by [26]. This technology combines scRNA-seq with barcoding to determine simultaneously the fitness of a cell population and its transcriptomic profile. [20] developed also a specialised "velocity" method called PhyloVelo, which does not use splicing information but relies on the barcoding to infer cellular dynamics.

Results of the analyses on these two data sets are shown in Figure 5, which again show a very good visualisation on both data sets (panels a-b and d-e). To evaluate quantitatively the results, we focus on correlating the inferred pseudotime with the fitness signature, which is a specific gene module whose expression is associated with a cell's proliferating fitness and it is estimated from the phylogenetic tree (see [26] for details on how fitness signatures are computed). Panel c shows the scatterplot of fitness values versus inferred pseudotime for one of the data sets, while panel f shows a comparison of all methods (including PhyloVelo) in terms of Spearman correlation with fitness. This shows that most methods perform extremely well (in fact, comparably to PhyloVelo, which explicitly uses the barcoding information), with the exception of scVelo and DeepVelo, this latter method only on one of the data sets.

This analysis demonstrates that utilising splicing information can provide an effective way to reconstruct cellular ordering which validates against barcoding information. Once again, neuroVelo emerges as one of the strongest performers also on this task.

3 Discussion

NeuroVelo innovates over other RNA-velocity methods by bringing a new dimension of functional interpretability: while standard methods focus on defining a cell-level velocity field, depicting the evolution of individual cells, neuroVelo provides a data-driven mechanistic model which can be interpreted in terms of (time-varying) regulatory networks. Results on all data set demonstrated not only that neuroVelo is among the best

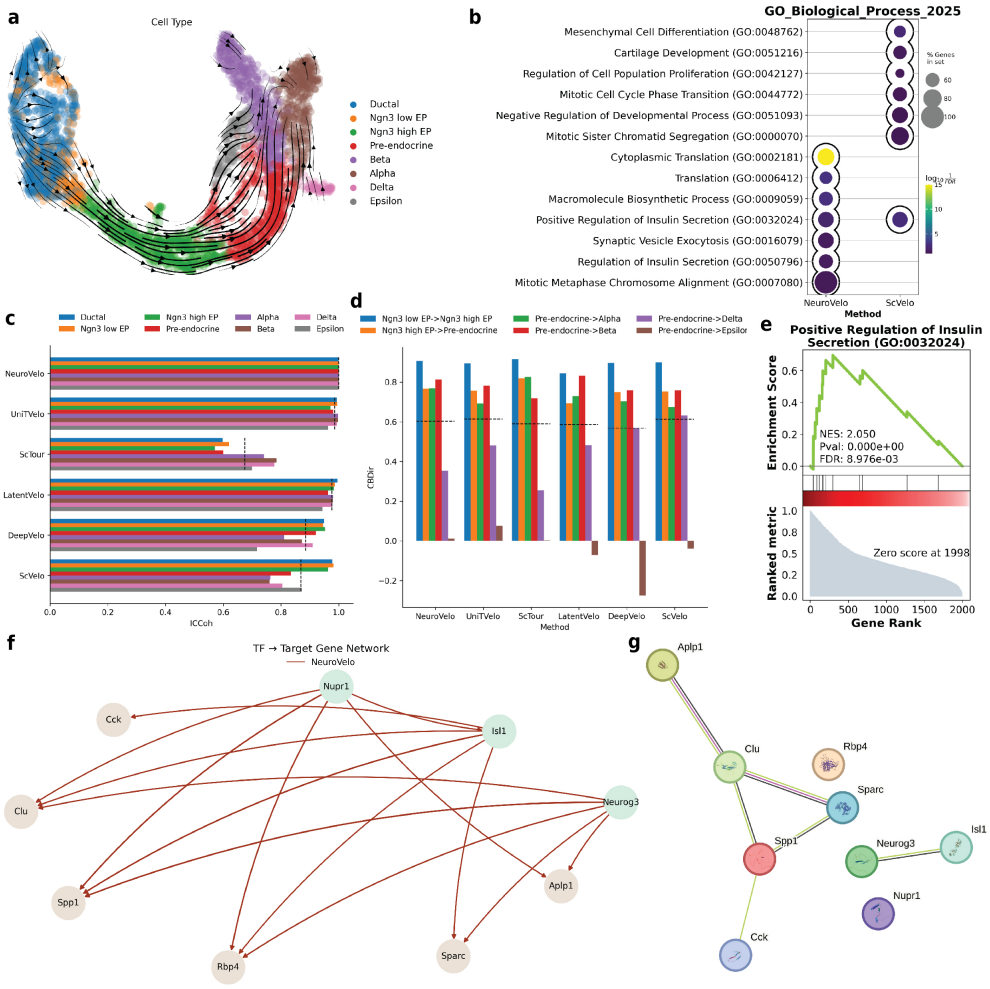

Fig. 4 (a) UMAP of NeuroVelo velocity field on pancreatic endocrinogenesis dataset.(b) Comparison on gene set enrichment analysis of the gene influencing the dynamics. (c) Comparison on the in-cluster coherence. (d) Comparison on the cross boundary direction. (e) Gene set enrichment analysis of insulin secretion pathway. (f) Gene regulatory network infered by NeuroVelo and (g) gene network structured using string-db (right)

performing tools in terms of standard RNA-velocity metrics, but also that the mechanisms it captures are biologically meaningful and extend known interactions from the literature. Importantly, a quantitative assessment of the reconstructed GRNs against experimental ChIP-seq data showed NeuroVelo to perform competitively with bespoke GRN inference methods which were supplied the list of transcription factor genes, a notable advantage which neuroVelo evidently could reconstruct independently from data.

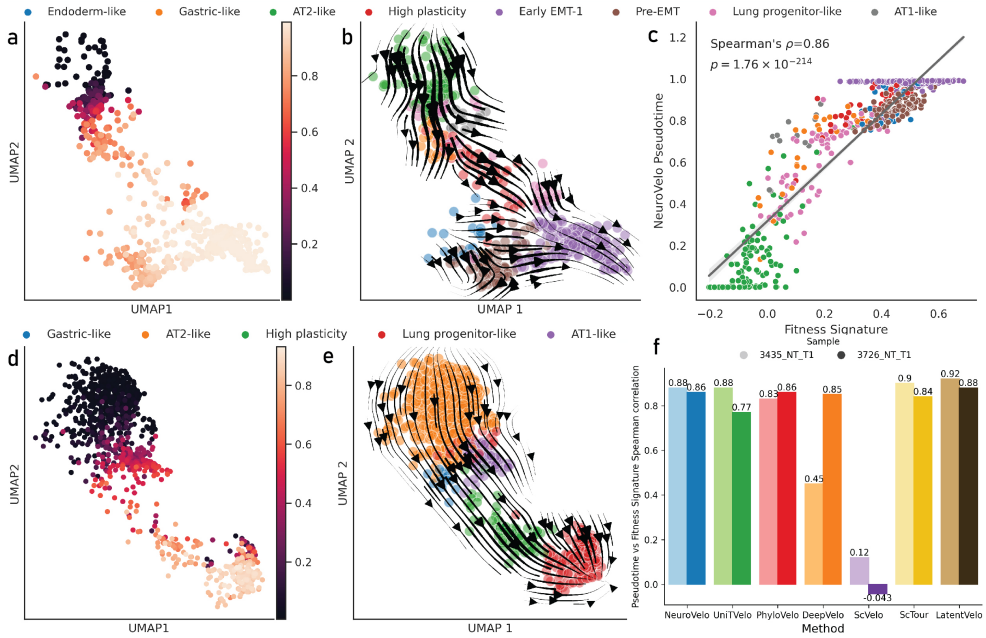

Fig. 5 (a) UMAP of NeuroVelo infered pseudotime and (b) NeuroVelo velocity field on the sample **3726_NT_T1** of the mouse cancer data set, as well as (c) scatterplot of the neuroVelo pseudotime against the fitness signature (inferred from barcoding information). (d) UMAP of NeuroVelo infered pseudotime and (e) NeuroVelo velocity field on the sample **3435_NT_T1** of the mouse cancer data set. (f) Comparison on pseudotime and fitness signature spearman correlation for 2 different samples.

Methodologically, neuroVelo attempts to strike a balance between complexity and interpretability. It does so by limiting its non-linearity to the dynamical system in the low-dimensional latent space, which enables both a very direct interpretation in terms of genes and the imposition of the RNA-velocity constraint as physics-informed loss term. An additional benefit of the relative simplicity of neuroVelo is its robustness.

Defining a semi-mechanistic model for single-cell dynamics opens a number of exciting opportunities to expand the scope of neuroVelo. First of all, principled ways to incorporate known mechanisms might be devised using ideas from Bayesian statistics. Secondly, other data modalities, such as chromatin accessibility, could be incorporated when available, thus providing a mechanistic extension to current methods for multi-omics analysis [27]-[28, 29]. Finally, different types of underlying dynamics might be considered, for example to accomodate mutational events which might lead to non-smooth dynamics, of to incorporate spatial correlation in spatial transcriptomic data.

References

[1] Regev, A., Teichmann, S.A., Lander, E.S., Amit, I., Benoist, C., Birney, E., Bodenmiller, B., Campbell, P., Carninci, P., Clatworthy, M., *et al.*: The human cell atlas. elife **6**, 27041 (2017)

[2] Trapnell, C., Cacchiarelli, D., Grimsby, J., Pokharel, P., Li, S., Morse, M., Lennon, N.J., Livak, K.J., Mikkelsen, T.S., Rinn, J.L.: The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering of single cells. Nature biotechnology **32**(4), 381–386 (2014)

[3] Haghverdi, L., Büttner, M., Wolf, F.A., Buettner, F., Theis, F.J.: Diffusion pseudotime robustly reconstructs lineage branching. Nature methods **13**(10), 845–848 (2016)

[4] Campbell, K.R., Yau, C.: Uncovering pseudotemporal trajectories with covariates from single cell and bulk expression data. Nature communications **9**(1), 2442 (2018)

[5] Setty, M., Kiseliovas, V., Levine, J., Gayoso, A., Mazutis, L., Pe'Er, D.: Characterization of cell fate probabilities in single-cell data with palantir. Nature biotechnology **37**(4), 451–460 (2019)

[6] La Manno, G., Soldatov, R., Zeisel, A., Braun, E., Hochgerner, H., Petukhov, V., Lidschreiber, K., Kastriti, M.E., Lönnerberg, P., Furlan, A., *et al.*: Rna velocity of single cells. Nature **560**(7719), 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**(12), 1408–1414 (2020)

[8] Chen, Z., King, W.C., Hwang, A., Gerstein, M., Zhang, J.: Deepvelo: Single-cell transcriptomic deep velocity field learning with neural ordinary differential equations. Science Advances **8**(48), 3745 (2022)

[9] Gao, M., Qiao, C., Huang, Y.: Unitvelo: temporally unified rna velocity reinforces single-cell trajectory inference. Nature Communications **13**(1), 6586 (2022)

[10] Gayoso, A., Weiler, P., Lotfollahi, M., Klein, D., Hong, J., Streets, A., Theis, F.J., Yosef, N.: Deep generative modeling of transcriptional dynamics for rna velocity analysis in single cells. Nature Methods, 1–10 (2023)

[11] Li, S., Zhang, P., Chen, W., Ye, L., Brannan, K.W., Le, N.-T., Abe, J.-i., Cooke, J.P., Wang, G.: A relay velocity model infers cell-dependent rna velocity. Nature biotechnology, 1–10 (2023)

[12] Qiao, C., Huang, Y.: Representation learning of rna velocity reveals robust cell transitions. Proceedings of the National Academy of Sciences **118**(49), 2105859118 (2021)

[13] Barile, M., Imaz-Rosshandler, I., Inzani, I., Ghazanfar, S., Nichols, J., Marioni, J.C., Guibentif, C., Göttgens, B.: Coordinated changes in gene expression kinetics underlie both mouse and human erythroid maturation. Genome biology **22**(1),

1–22 (2021)

[14] Gorin, G., Fang, M., Chari, T., Pachter, L.: Rna velocity unraveled. PLOS Computational Biology **18**(9), 1010492 (2022)

[15] Singh, R., Wu, A.P., Mudide, A., Berger, B.: Causal gene regulatory analysis with RNA velocity reveals an interplay between slow and fast transcription factors. Cell Syst. **15**(5), 462–4745 (2024)

[16] Chen, R.T., Rubanova, Y., Bettencourt, J., Duvenaud, D.K.: Neural ordinary differential equations. Advances in neural information processing systems **31** (2018)

[17] Li, Q.: sctour: a deep learning architecture for robust inference and accurate prediction of cellular dynamics. Genome Biology **24**(1), 1–33 (2023)

[18] Pijuan-Sala, B., Griffiths, J.A., Guibentif, C., Hiscock, T.W., Jawaid, W., Calero-Nieto, F.J., Mulas, C., Ibarra-Soria, X., Tyser, R.C.V., Ho, D.L.L., Reik, W., Srinivas, S., Simons, B.D., Nichols, J., Marioni, J.C., Göttgens, B.: A single-cell molecular map of mouse gastrulation and early organogenesis. Nature **566**(7745), 490–495 (2019) https://doi.org/10.1038/s41586-019-0933-9

[19] Bastidas-Ponce, A., Tritschler, S., Dony, L., Scheibner, K., Tarquis-Medina, M., Salinno, C., Schirge, S., Burtscher, I., Böttcher, A., Theis, F.J., Lickert, H., Bakhti, M.: Comprehensive single cell mrna profiling reveals a detailed roadmap for pancreatic endocrinogenesis. Development **146**(12) (2019) https://doi.org/10.1242/dev.173849

[20] Wang, K., Hou, L., Wang, X., Zhai, X., Lu, Z., Zi, Z., Zhai, W., He, X., Curtis, C., Zhou, D., et al.: Phylovelo enhances transcriptomic velocity field mapping using monotonically expressed genes. Nature Biotechnology, 1–12 (2023)

[21] Farrell, S., Mani, M., Goyal, S.: Inferring single-cell transcriptomic dynamics with structured latent gene expression dynamics. Cell Reports Methods **3**(9), 100581 (2023) https://doi.org/10.1016/j.crmeth.2023.100581

[22] Huynh-Thu, V.A., Irrthum, A., Wehenkel, L., Geurts, P.: Inferring regulatory networks from expression data using tree-based methods. PLoS ONE **5**(9), 12776 (2010) https://doi.org/10.1371/journal.pone.0012776

[23] Moerman, T., Aibar Santos, S., Bravo González-Blas, C., Simm, J., Moreau, Y., Aerts, J., Aerts, S.: Grnboost2 and arboreto: efficient and scalable inference of gene regulatory networks. Bioinformatics **35**(12), 2159–2161 (2018) https://doi.org/10.1093/bioinformatics/bty916

[24] Pratapa, A., Jalihal, A.P., Law, J.N., Bharadwaj, A., Murali, T.M.: Benchmarking algorithms for gene regulatory network inference from single-cell transcriptomic data. Nature Methods **17**(2), 147–154 (2020) https://doi.org/10.1038/s41592-019-0690-6

[25] Zou, Z., Ohta, T., Oki, S.: Chip-atlas 3.0: a data-mining suite to explore chromosome architecture together with large-scale regulome data. Nucleic Acids Research **52**(W1), 45–53 (2024) https://doi.org/10.1093/nar/gkae358 https://academic.oup.com/nar/article-pdf/52/W1/W45/58436223/gkae358.pdf

[26] Yang, D., Jones, M.G., Naranjo, S., Rideout, W.M., Min, K.H.J., Ho, R., Wu, W., Replogle, J.M., Page, J.L., Quinn, J.J., *et al.*: Lineage tracing reveals the phylodynamics, plasticity, and paths of tumor evolution. Cell **185**(11), 1905–1923 (2022)

[27] 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**(3), 387–398 (2022) https://doi.org/10.1038/s41587-022-01476-y

[28] Kazwini, N.E., Sanguinetti, G.: Share-topic: Bayesian interpretable modeling of single-cell multi-omic data. Genome Biology **25**(1) (2024) https://doi.org/10.1186/s13059-024-03180-3

[29] El Kazwini, N., Gao, M., Boudjelthia, I.K., Cai, F., Huang, Y., Sanguinetti, G.: Crak-velo: Chromatin accessibility kinetics integration improves rna velocity estimation. bioRxiv, 2024–09 (2024)

Methods

RNA velocity

ScRNA-seq captures a static snapshot of gene expression at a single time point, hence modeling cellular dynamics from single-cell RNA-seq data is not straightforward. La Manno et al. [6] leverage the presence of unspliced pre-mRNA to model transcriptional dynamics from single cell data. The coexistence of unspliced and spliced transcripts enables the inference of RNA velocity by modeling the underlying splicing kinetics. A simple model was assumed by [6] describing the evolution of unspliced and spliced mRNA molecules over time given by the system of equations 1

$$\begin{aligned} \dot{u} &= \alpha - \beta u(t) \\ \dot{s} &= \beta u(t) - \gamma s(t) \end{aligned} \tag{1}$$

$\dot{u}$ and $\dot{s}$ are time derivatives of the abundance of unspliced and spliced RNA (also known as RNA velocity), respectively. α is the transcription rate, β is the splicing rate, and γ is the degradation rate of the spliced RNA. By estimating parameter β and γ, one can make RNA velocity estimate $\dot{s}$, which is a vector in gene space that gives the infinitesimal change in abundance of spliced RNA amount (hence the analogy with the concept of velocity from classical mechanics).

Model architecture

NeuroVelo is a neural network architecture that consists of a linear auto-encoder and neural ODEs architecture in the latent representation of the data. The main idea of NeuroVelo is to construct cell dynamics from scRNA-seq data in the latent space.

Encoder/ decoder architecture

We used spliced and unspliced reads of scRNA-seq data with g number of genes, and embedded them using the same 2 layers linear encoder function $E : \mathbb{R}^g \to \mathbb{R}^l$, into a latent space $\boldsymbol{z} = (\boldsymbol{z}_s, \boldsymbol{z}_u)$ of dimension l.

$$\begin{aligned} \boldsymbol{z}_s &= W_e^{(2)}(W_e^{(1)} X_s + b_e^{(1)}) + b_e^{(2)} \\ \boldsymbol{z}_u &= W_e^{(2)}(W_e^{(1)} X_u + b_e^{(1)}) + b_e^{(2)} \end{aligned}$$

This step is mainly for dimensionality reduction and feature extraction of the highly dimensional gene space to a much smaller latent space (50 by default).

The encoder latent representation $\boldsymbol{z}$ is decoded with a 2 layers linear decoder network $D : \mathbb{R}^l \to \mathbb{R}^g$ to create a reconstruction of the original spliced and unspliced inputs ($\hat{X}_s$, $\hat{X}_u$) learning a function that maps from latent space to gene space.

$$\begin{aligned} \hat{X}_s &= W_d^{(2)}(W_d^{(1)} \boldsymbol{z}_s + b_d^{(1)}) + b_d^{(2)} \\ \hat{X}_u &= W_d^{(2)}(W_d^{(1)} \boldsymbol{z}_u + b_d^{(1)}) + b_d^{(2)} \end{aligned}$$

Pseudotime encoding

The output of the first layer of the encoder is also used to estimate the pseudotime t of the cells, such that:

$$t = S(W_e^{(1)} X_s + b_e^{(1)})$$

Where $S(x)$ is the sigmoid function. Both pseudotime and latent representation of the cell are used to train nonlinear neural ODEs.

Neural ODE

Once we project the cells into a lower-dimensional space, we use a set of neural ODEs to describe the cellular dynamics in this latent space

$$\frac{d\boldsymbol{z}_s^{(d)}(t)}{dt} = f^{(d)}(\boldsymbol{z}_s^{(d)}(t), t), \tag{2}$$

where $f^{(d)} : \mathbb{R}^l \to \mathbb{R}^l$ is a two layers neural network that takes $z_s \in \mathbb{R}^l$ (with $l = 50$ by default) as input. The network uses the Exponential Linear Unit (ELU) function as the nonlinear activation function, and has a hidden layer dimension of 100. This neural ODE describes the dynamics of cells from particular sample d. Using the nonlinear ODEs in the latent space helps to capture the essential and complex dynamics of cells in this space.

Notice that, since we are learning both the pseudotime and the generator function of the dynamics (the f function), one could reverse simultaneously the pseudotime and the sign of f without altering the trajectories. This ambiguity may be resolved either by prior knowledge or by inverting both signs *a posteriori*.

Linearisation and interpretation

For any dynamical process governed by a nonlinear ODE, we can understand the local behavior of the system by linearizing the ODE around a given point. In this case we choose as the point about which to linearise the average of the encoded state of (a group of) cells; the linearized system becomes:

$$\dot{\boldsymbol{z}}_s(t) = \mathbf{J}\boldsymbol{z}_s(t), \quad \mathbf{J} = \begin{bmatrix} \frac{\partial f_1}{\partial z_1} & \cdots & \frac{\partial f_1}{\partial z_l} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_l}{\partial z_1} & \cdots & \frac{\partial f_l}{\partial z_l} \end{bmatrix} \tag{3}$$

where the matrix of partial derivatives $\mathbf{J}$ is the Jacobian matrix. We perform *eigenanalysis* on the Jacobian matrix. The eigenvalues give the important directions of the correspondent eigenvectors in the latent space, then we use the decoder function to map these important directions in the latent space to the gene space. The output of this decoder is a list of ranked genes based on their importance in a particular direction.

The main issues related to using neural networks are interpretability and robustness, the former is not a problem because NeuroVelo uses both linear embedding and projection, thus the interpretation of the latent space into the gene space is straightforward. The robustness part is more of a concern because a different initialization leads to a different local minima which makes biological interpretation uncertain. Here we propose a geometrical and statistical approach that gives a solid gene list by using many trained models.

The first step is to find the set of eigenvectors from other models that align the most with an eigenvector of a current model $\vec{v}_i^{(m)}$, the alignment is based on the absolute value of the cosine similarity. This set of the most aligned eigenvectors is given by the following equation:

$$U\left(\vec{v}_i^{(m)}\right) = \left\{\max_j \left|S_c\left(\vec{v}_i^{(m)}, \vec{v}_j^{(m')}\right)\right|, \text{for } m' \neq m\right\}$$

where S_c is the cosine similarity function and m' is the index of the trained model.

We take the ranked genes of the aligned eigenvectors, and we average the ranks gene-wise. The output is a list of genes ranked based on the average rank across the aligned eigenvectors from different trained models. This indicates that the genes that are at the top of the list hold significant importance for the dynamics learned by various models and vice versa. A set of different analyses can be performed on these average-ranked genes for validation.

GRN construction

As shown in equation 3, the non-linear latent dynamics are approximated by the linearized system $\dot{\boldsymbol{z}}_s(t) = \mathbf{J}\boldsymbol{z}_s(t)$.

This $\mathbf{J}$, is obtained by linearizing the latent dynamical system around an average of group of cells, this could be all the cells -as done for benchmarking against other GRN methods- or by cell-type or cell pseudotime ordering -as done for the time dependent GRN in supplementary material-.

By mapping the latent representation to the gene space, we can show that this system can be written as:

$$\dot{\boldsymbol{X}}_s(t) = \hat{\mathbf{J}}\boldsymbol{X}_s(t) + \mathbf{C}$$

where $\mathbf{C}$ is a bias term that does not depend on X. We call $\hat{\mathbf{J}} = \mathbf{W_d}\mathbf{J}\mathbf{W_e}$ the decoded Jacobian, $\mathbf{W_d}$ and $\mathbf{W_e}$ are the weight of the decoder and encoder networks respectively.

The matrix $\hat{\mathbf{J}} \in \mathbb{R}^{g\times g}$ (where g is the number of genes) captures the effective interactions between gene expression levels- i.e., the element $\hat{J}_{ij}$ tell us how a change in the expression of gene j influences the rate of change of gene i. A positive $\hat{J}_{ij}$ suggests that gene j activates gene i, while a negative value suggests that gene j represses gene i

To construct a graph from the matrix $\hat{\mathbf{J}}$ we take a directed graph $G = (V, E)$ approach, such that each node v_i correspond to a gene i and a directed edge $(v_j \to v_i)$ is weighted based on the magnitude of $\hat{J}_{ij}$.

GRN evaluation: ChIP-seq data processing

The main challenge with benchmarking GRN is to establish the ground truth network to compare it with, and taking into consideration that gene regulatory network depends on the cell-type/cell-line. We create a *gold standard* gene regulatory network using chip-seq data from ChIP-Atlas [25], and we take the MACS2 score as a signal of the transcription factor binding to a target gene. We look at potential genes where the transcription factor binds at $\pm 10k$ distance from TSS of that gene.
To compare the inferred GRN by the 3 methods with the gold standard one, we take a sub-graph containing only TFs-Target genes pairs that exist in the our ground truth and we compute the area under precision recall curve (AUPRC). We use different cell lines and different threshold on the MACS2 score to evaluate the AUPRC, and we report AUPRC ratio, which is AUPRC/Baseline(Random Guess).

For the case of NeuroVelo and because we are taking MACS2 score as ground truth, we cannot benchmark for the type of interaction -whether it is activation or repression-, thus we take the magnitude as weighted graph edge.

Model training

The tasks learnt by neural networks are based on what loss function the network is optimizing. To ensure that the cellular dynamics learned by the network have biological meaning, we propose a loss function that puts biophysical constraints on the data and does not require any parameter fine-tuning.

The first part of the loss is the usual mean squared error (MSE) of autoencoders network, we want our spliced and unspliced inputs to match their reconstructed counterparts from the decoder and so the first loss part is a sum of MSE of spliced and unspliced reads.

The second part of the loss is what puts a biophysical constraint on the neural ODEs in the same fashion of physics-informed neural networks. The idea is to make the derivative of latent spliced reads $\dot{\boldsymbol{z}}_s^d(t)$ align with the splicing dynamics from the RNA velocity but in latent representation $(e^{\beta}\boldsymbol{z}_u - e^{\gamma}\boldsymbol{z}_s)$, There is an exponential of β (and γ) because they are analogy of splicing and degradation rate and they must be positive.

The final loss is written as follows:

$$\begin{aligned} L =& \mathrm{MSE}(X_s, \hat{X}_s) + \mathrm{MSE}(X_u, \hat{X}_u) + \sum_{d=1}^{D} \\ & \mathrm{MSE}\left(f^{(d)}(\boldsymbol{z}_s^{(d)}(t), t), e^{\beta}\boldsymbol{z}_u^{(d)} - e^{\gamma}\boldsymbol{z}_s^{(d)}\right) + \mathrm{MSE}\left(z_s(t), z_s\right) \end{aligned} \tag{4}$$

Where the sum over d is the sum over specific group of cells or treatments. $\dot{z}(t)_s^d$ is the function that describes the splicing dynamics of a sample d in the latent space. z_s^d and z_u^d are the projected spliced and unspliced reads of the sample d. Notice that the third term in the loss enforces the RNA-velocity constraint, so that the role of the transcription rate is modelled flexibly and learnt directly from the data. The last

term in the loss aim to align the latent dynamics with the latent representation, due to the linearity of the model, this is equivalent to MSE $\left(\hat{X}_s(t), X_s\right)$ where $\hat{X}_s(t)$ is the decoded latent dynamics $z_s(t)$.

In this way, we are able to investigate complex and specific cellular dynamics by learning splicing dynamics constrained nonlinear ODEs in a reduced space and without the need to fine tune parameters on the level of the loss function as well.

Parameters selection

RNA velocity estimation can be very sensitive to the processing done such as constructing kNN graph as shown in [14] to use "moments" instead of raw reads. In addition, using neural network to estimate RNA velocity increases the number of parameter combinations, such as learning rate, optimizer, batch size and network size (dimensions). Moreover, for the same set of parameters, different initializations can be stuck in different local minima, leading to convergence in any case, but to different velocity estimates.

In address this issue, we propose a metric, *Velocity Alignment* which uses the same cosine similarity measure as *ICCoh* but is applied to velocities estimated using models trained with different random seeds.

$$\mathrm{VA}^{(n,m)} = \frac{1}{N}\mathbf{diag}\left(S_c(\mathbf{V}^{(n)}, \mathbf{V}^{(m)})\right)$$

$V^{(n)}$ and $V^{(m)}$ are RNA velocity estimated from different trained model (n) and (m) and N is the total number of cells. The **diag** takes the diagonal elements of the cosine similarity measure, meaning that we are measuring how the same cell velocity from 2 different models is aligned.

The idea is to select the set of parameters with the highest cosine similarity. This approach ensures that even if the model converges to a local minimum, the velocity estimated from different seeds is aligned.

It is important to point that this metric should be computed for models with different random initializations but for the same set of other parameters.

Implementation

NeuroVelo is implemented in Python machine learning framework `PyTorch 2.0` and using `torchdiffeq` package for ODE solvers in PyTorch.
Velocity field visualizations are done with `scvelo`
except for the mouse lung cancer dataset it was done using `PhyloVelo` package. Gene networks visualizations are done with `networkx`
Gene set enrichment analysis is done with *"GSEApy: Gene Set Enrichment Analysis in Python"* `gseapy`. Passing a dataframe of genes and their average rank as a ranking metric to `prerank` function. The enrichment analysis plots are all generated by the same package.
The metrics *ICCoh* and *CBDir* are borrowed from [12]

Datasets

Mouse erythroid

The human bone marrow dataset is downloaded from ScVelo package `scv.datasets.gastrulation_erythroid()`. The dataset is already processed and contains spliced and unspliced counts. We picked the top 2000 highly variables gene and we used counts training.It takes 524 seconds, 100 epoch for training and using CPU model: Intel(R) Xeon(R) Gold 6238R CPU @ 2.20GHz.

Mouse lung cancer

This dataset is CRISPR/Cas9-based lineage tracing dataset from a genetically-engineered mouse model (GEMM) of lung adenocarcinoma. We used it to compare NeuroVelo to PhyloVelo, mainly using two tumors (3726_NT_T1 and 3435_NT_T1).

We requested the spliced/unspliced reads from [20]. We picked the top 2000 highly variables gene and we used moments instead of counts for training.

Human bone marrow hematopoiesis

The human bone marrow dataset is downloaded from ScVelo package `scv.datasets.bonemarrow()`. The bone marrow is the primary site of new blood cell production or haematopoiesis. It is composed of hematopoietic cells, marrow adipose tissue, and supportive stromal cells. The dataset is already processed and contains spliced and unspliced counts. We picked the top 2000 highly variables gene and we used moments instead of counts for training.It takes 451 seconds, 100 epoch for training and using CPU model: Intel(R) Xeon(R) Gold 6238R CPU @ 2.20GHz

Pancreatic endocrinogenesis

Pancreatic endocrinogenesis dataset is downloaded from ScVelo package `scv.datasets.pancreas()`. Endocrine cells originate from endocrine progenitors found within the pancreatic epithelium. These progenitors differentiate into four primary cell types: α-cells that produce glucagon, β-cells that produce insulin, δ-cells that produce somatostatin, and ε-cells that produce ghrelin. The dataset is already processed and contains spliced and unspliced counts. We picked the top 2000 highly variables gene and we used counts training.It takes 185 seconds, 100 epoch for training and using CPU model: Intel(R) Xeon(R) Gold 6238R CPU @ 2.20GHz

Code availability

The source code for NeuroVelo is available on GitHub: https://github.com/idriskb/NeuroVelo. Along with tutorial notebooks on how to run the method for different datasets, posts analysis for extracting the most important genes and gene regulatory network construction.

	Erythroid Mouse		3435_NT_T1	3726_NT_T1
	Mean CBDir	**Mean ICCoh**	**Spearman correlation**	**Spearman correlation**
NeuroVelo	0.8304	0.9999	0.88	0.86
UniTVelo	0.832575	0.98776	0.88	0.77
ScTour	0.801175	0.38038	0.90	0.84
LatentVelo	0.84415	0.97412	0.92	0.88
DeepVelo	0.41515	0.92614	0.45	0.85
ScVelo	-0.3519	0.83798	0.12	-0.043
PhyloVelo	-	-	0.83	0.86

	Human Bone Marrow		Pancreatic endocrinogenesis	
	Mean CBDir	**Mean ICCoh**	**Mean CBDir**	**Mean ICCoh**
NeuroVelo	0.81943	0.996017	0.603533	0.9997
UniTVelo	0.804933	0.97743	0.614	0.9866
ScTour	0.786033	0.47849	0.5898	0.67406
LatentVelo	0.7664	0.98141	0.5854	0.9763
DeepVelo	-0.804933	0.93207	0.5674	0.8854
ScVelo	-0.839467	0.91819	0.6131	0.8698
PhyloVelo	-	-	-	-

Table 1 Metrics table for different methods and datasets. For method that allows different random seeds (NeuroVelo, ScTour and LatentVelo) the table shpws the best performance of running 10 seeds. More on stability of velocity is shown in figure S1

Supplementary Figures

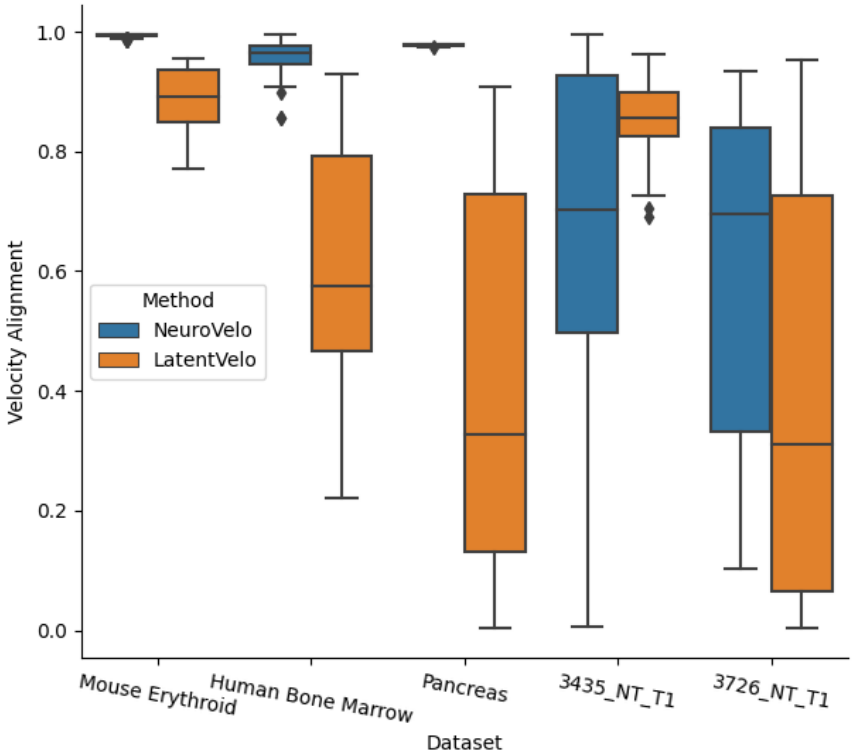

Fig. S1 Comparison on velocity alignment, each point is the velocity alignment of a pair of trained models.

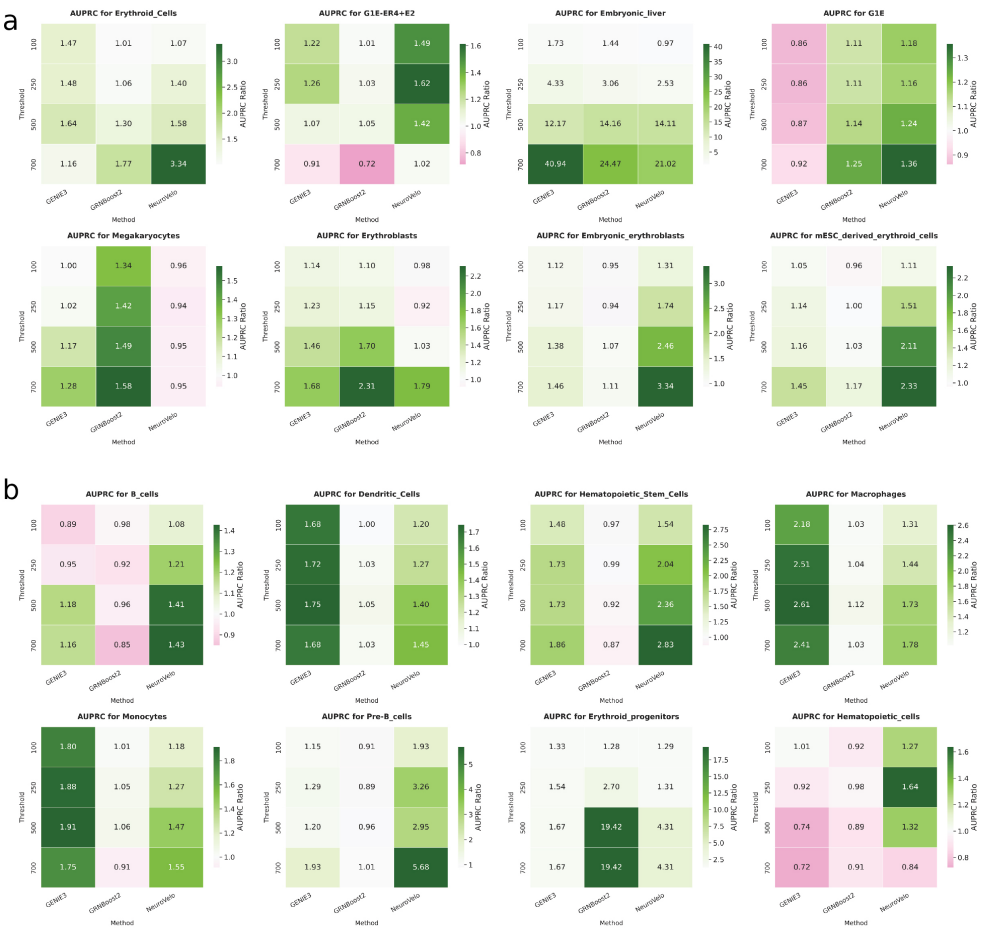

Fig. S2 Comparison on AUPRC ratio for GRN inference performance at different threshold of MACS2 score. (a) Mouse erythroid dataset, (b) Human bonemarrow dataset

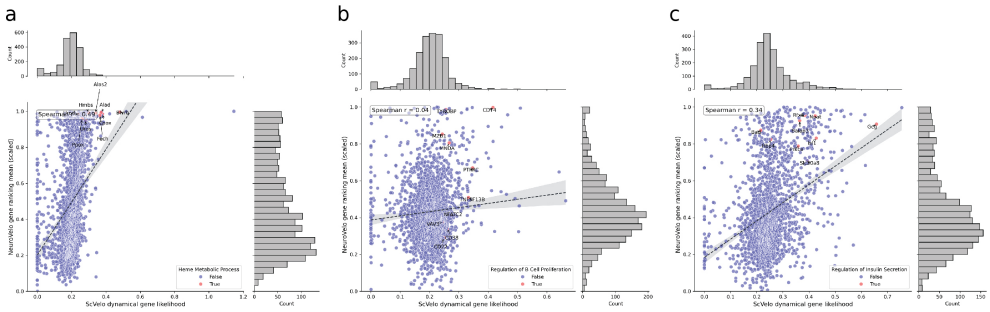

Fig. S3 Comparison on spearman correlation for genes driving dynamics from ScVelo and NeuroVelo on (a) Mouse erythroid, (b) Human bone marrow, (c) Pancreas datasets.

Supplementary Tables

Cell Line	TF	SRX Number
Embryonic erythroblasts	Gata1	SRX079887
		SRX079888
		SRX145055
		SRX3889813
		SRX3889814
Embryonic liver	Ebf1	SRX14998422
		SRX14998423
		SRX7807060
		SRX7807062
		SRX9204114
		SRX9204116
	Klf1	SRX2530433
	Myb	SRX3889819
		SRX3889820
	Smarca4	SRX12298985
		SRX12298986
		SRX12298989
		SRX12298990
		SRX7807050
		SRX7807051
		SRX7807052
		SRX7807053
		SRX7807054
		SRX7807056
		SRX7807058
	Tcf3	SRX9599934
		SRX9599937
		SRX9599940
Erythroblasts	Gata1	SRX6797012
		SRX6797015
		SRX6797018
		SRX6797021
		SRX6797024
		SRX6797027
		SRX6797030
		SRX6797033
		SRX6797036
		SRX6797039
		SRX744012
		SRX744013

		SRX744014
		SRX744015
	Nfe2	SRX203211
Erythroid Cells	Ezh2	SRX2741794
		SRX2741795
		SRX2741810
		SRX2741811
	Gata1	SRX10916762
		SRX18530072
		SRX18530073
		SRX18530074
	Nfe2	SRX18530089
		SRX18530090
	Yy1	SRX7367706
		SRX7367707
G1E	Gata1	SRX100316
		SRX145061
		SRX9519540
		SRX9519541
	Gata2	SRX079863
		SRX079864
		SRX100313
		SRX145067
		SRX1901617
		SRX1901619
	Runx1	SRX1901618
		SRX1901620
	Smad1	SRX100314
		SRX100317
G1E-ER4+E2	Chd4	SRX19103401
		SRX19103402
		SRX19103403
		SRX19103404
		SRX19103405
		SRX19103406
		SRX19103407
		SRX19103408
		SRX19103409
		SRX19103410
		SRX19103411
		SRX19103412
	Fli1	SRX1030239

		SRX1030240
		SRX1030242
		SRX1030243
		SRX1030246
	Gata1	SRX013185
		SRX10714839
		SRX10714840
		SRX130172
		SRX130174
		SRX145052
		SRX181742
		SRX181743
		SRX181744
		SRX181745
		SRX181748
		SRX181749
		SRX8820360
		SRX8820361
	Gata2	SRX145068
		SRX8820362
		SRX8820363
	Yy1	SRX24368044
		SRX24368045
		SRX24368046
		SRX24368047
Megakaryocytes	Fli1	SRX203227
	Nfe2	SRX203207
		SRX203208
		SRX204382
	Runx1	SRX203228
		SRX253098
mESC derived erythroid cells	Gata1	SRX18530081
		SRX18530083
		SRX18530084

Table ST 1: Summary of transcription factors (TFs) and their associated SRX accession numbers across different cell lines for mouse erythroid dataset.

Cell Line	TF	SRX Number
B cells	CREBBP	SRX2339191
		SRX2339193

	FOXO1	SRX1012536 SRX1012537
	POU2AF1	SRX12118508
	POU2F2	SRX12118507
	SMARCA4	SRX170356
	SPI1	SRX5574340 SRX5574341
Dendritic Cells	NR3C1	SRX11513458 SRX11513459 SRX11513460 SRX11513461 SRX11513462 SRX11513463 SRX11513464 SRX11513465
	SPI1	SRX5095870 SRX5095871 SRX5574357 SRX5574359 SRX5574361 SRX5574362 SRX5574363 SRX5574364 SRX5574365 SRX5574367 SRX5574369 SRX5574370 SRX627428 SRX627430
Erythroid progenitors	GATA2	SRX5169134 SRX687556 SRX687557
Hematopoietic Stem Cells	IRF2	SRX1089831
	SMARCA4	SRX170346
	SPI1	SRX038910 SRX038911 SRX1089832 SRX1089833
Hematopoietic cells	ERG	SRX026659 SRX026660
Macrophages	CREBBP	SRX5839976 SRX5839977

		SRX5839978
		SRX5839979
	EGR1	SRX6763473
		SRX6763474
		SRX6763475
		SRX6763476
		SRX7902978
		SRX7902979
		SRX7902980
		SRX7902981
		SRX7902982
		SRX9279232
	HIF1A	SRX212351
		SRX212352
		SRX212353
		SRX212354
		SRX212359
		SRX212360
		SRX212361
		SRX212362
		SRX23900619
		SRX23900620
		SRX23900621
		SRX23900622
		SRX23900623
		SRX23900624
		SRX23900625
		SRX23900626
	NR3C1	SRX3586329
		SRX3586330
		SRX3586331
		SRX3586332
		SRX716944
	SPI1	SRX093189
		SRX1023790
		SRX1023791
		SRX1023792
		SRX1023793
		SRX1023794
		SRX1023795
		SRX1023796
		SRX1023797
		SRX2770854
		SRX2770855

		SRX2770856
		SRX2770857
		SRX286018
		SRX286019
		SRX3244991
		SRX4001818
		SRX4001819
		SRX4001820
		SRX4001821
		SRX5574375
		SRX5574376
		SRX5574377
Monocytes	EGR1	SRX6763478
		SRX7031581
		SRX7031582
		SRX7031583
		SRX7902977
		SRX7902984
		SRX7902985
	NR3C1	SRX3586325
		SRX3586326
		SRX3586327
		SRX3586328
	SPI1	SRX093183
		SRX5574342
		SRX5574343
		SRX5574344
		SRX5574345
		SRX5574354
		SRX5574355
		SRX5574356
		SRX5574371
		SRX5574372
		SRX5574373
Pre-B cells	POU2AF1	SRX3460920
	POU2F2	SRX3460921

Table ST 2: Summary of transcription factors (TFs) and their associated SRX accession numbers across different cell lines for human bone marrow dataset.