Pyro-Velocity: Probabilistic RNA Velocity inference from single-cell data

Qian Qin[1,2,3,4,5], Eli Bingham[5,7], Gioele La Manno[6]*, David M. Langenau[1,2,3,4]*, Luca Pinello[1,2,5]*

[1] Molecular Pathology Unit, Massachusetts General Research Institute, Charlestown, MA 02129. [2] Massachusetts General Hospital Cancer Center, Harvard Medical School, Charlestown, MA 02129. [3] Center for Regenerative Medicine, Massachusetts General Hospital, Boston, MA 02114.[4] Harvard Stem Cell Institute, Cambridge, MA 02139. [5] Broad Institute of MIT and Harvard, Cambridge, MA 02142. [6] Laboratory of Neurodevelopmental Systems Biology, Brain Mind Institute, School of Life Sciences, École Polytechnique Fédérale de Lausanne (EPFL), 1015 Lausanne, Switzerland, [7] Basis Research Institute, Cambridge, MA 02142.

Correspondence to:

Luca Pinello, Ph.D. lpinello@mgh.harvard.edu
David M. Langenau, Ph.D. dlangenau@mgh.harvard.edu
Gioele La Manno, Ph.D. gioele.lamanno@epfl.ch

* These authors jointly supervised this work

Abstract

Single-cell RNA Velocity has dramatically advanced our ability to model cellular differentiation and cell fate decisions. However, current preprocessing choices and model assumptions often lead to errors in assigning developmental trajectories. Here, we develop, Pyro-Velocity, a Bayesian, generative, and multivariate RNA Velocity model to estimate the uncertainty of cell future states. This approach models raw sequencing counts with the synchronized cell time across all expressed genes to provide quantifiable and improved information on cell fate choices and developmental trajectory dynamics.

Main text

RNA Velocity is a powerful computational framework to estimate the time derivative of gene expression[1–4], model transcriptional dynamics, and predict cell fate from single-cell RNA sequencing (RNA-seq) data. The framework has been used to study developmental cell lineage trajectories over a broad spectrum of biological processes, such as haematopoiesis[5], pancreas development[6], and neural progenitors differentiation[3]. Yet, recent state-of-the-art RNA Velocity approaches such as *velocyto* and *scVelo*[1] have several limitations. First, they may predict developmental trajectories that do not exist in nature, as it has been shown in the analysis of fully mature blood cells[7,8].–Second, data preprocessing steps (e.g., selection of top variable genes, dimensionality reduction) affect velocity estimates that are difficult to characterize and might lead to artefactual predictions. In addition, non-linear spliced and unspliced imputation procedures could distort phase portrait geometry and recovered trajectories (**Fig. 1a**). Third, most RNA velocity algorithms perform estimates independently for each gene, and therefore cannot exploit the natural interdependence of gene expression. Fourth, a unified description of the temporal relation between cells can only be assigned based on postprocessing steps that aggregate heuristically the gene-wise fits into a shared latent time (**Fig. 1a**). Finally, current RNA velocity frameworks only output point estimates of the velocity vector field and do not provide uncertainty estimation. Importantly, the absence of statistical confidence estimates increases false discoveries as it leaves users without a way to evaluate the robustness of the methods' outputs. To overcome these limitations, we developed a probabilistic generative model, named *Pyro-Velocity*, that estimates RNA dynamics and infers cell trajectories using Pyro[9], a robust and efficient probabilistic programming language.

Pyro-Velocity recasts the velocity estimation problem into a latent variable posterior probability inference (**Methods**). The proposed model is generative and fully Bayesian, with the different parameters considered as latent random variables. Each parameter is assigned a prior distribution, and the model is then conditioned on the raw data (both spliced and unspliced molecule raw counts) to estimate posterior probability distributions and to enable uncertainty estimation of velocity. Central to the Pyro-Velocity model is a *shared latent time* (t): a global latent variable placing each cell at a particular instant of the modeled dynamical process. From this latent time, the expectation of spliced and unspliced RNA levels for each gene can be computed knowing gene-specific kinetics parameters (transcriptional (α), splicing (β), and degradation rates (γ) and two gene-specific switching times (t_0^k), and gene-specific transcriptional state *k*) and integrating the RNA velocity differential equations analogously to what proposed in the dynamical RNA velocity model by Bergen et al.[1,10]. However, we innovate the structure and formulation of the latent time given that our model is set to jointly fit the entire raw data matrix, not individual genes. We approach the task of estimating the posterior distribution by restricting the search within a family of a variational distribution approximating the true posterior, and taking advantage of the automatic differentiation variational inference (ADVI) capabilities of Pyro[9,11] (**Fig. 1a**, **Methods**). Notably, our generative Bayesian model enables to sample from the posterior distributions of the latent and dependent variables, allowing uncertainty estimation of phase portraits, high dimensional velocity vector field, and shared latent time. Around this model, we built several visualizations to assess uncertainty and functions to recover putative cell fate priming genes and to investigate how the denoised gene expression recapitulates known cell-type annotations and their developmental order (**Fig. 1b, Methods**). Although several non-peer-reviewed RNA velocity methods[12–16] have been recently proposed, here we showcase the advantages and innovations of Pyro-Velocity by comparing it with the current state-of-the-art dynamical RNA velocity model proposed by Bergen et al.[1], and implemented in scVelo.

One major biological hurdle for current formulations of RNA Velocity models has been miscalling cell trajectories in fully differentiated cell populations that lack dynamic cell state transitions. One typical example being human peripheral blood mononuclear cells (PBMCs)[8]. In this dataset, 65,877 cells were analyzed using the 10x platform that comprised 11 fully differentiated, mature immune cell types. This dataset lacked stem and progenitor cells or other signatures of an undergoing dynamical differentiation; thus, no consistent velocity flow should be detected. We reanalyzed this dataset using scVelo and Pyro-Velocity using a common set of genes for a fair comparison. Using scVelo, we reproduced the original results showing incorrect cell trajectory assignments (**Fig. 2a**)[7]. ScVelo uncovered an errant, strong linear trajectory across

a subset of mature cell types[7] that is exasperated by the inability to confidently assess lineage directionality or a baseline to assess significance. On the contrary, Pyro-Velocity failed to detect high-confidence trajectories in the mature blood cell states, consistent with what is known about the biology underlying these cells (**Fig. 2a,** panel 2 vs. panel 3). Importantly, the Bayesian nature of Pyro-Velocity enabled the generation of uncertainty estimates that can be easily visualized as likely vectors for individual cells or averaged across cells on a constantly spaced grid on the cell embedding (*averaged vector field*).

Pyro-Velocity was also able to define well-known developmental cell hierarchies in a scRNA-seq dataset of E15.5 mouse pancreas (**Fig. 2b-e**, n=3,696 cells comprising 8 cell types (**Fig. 2b,** panel 1)), identifying cell trajectories originating from ductal progenitor cells and culminated in the production of mature Alpha, Beta, Delta, and Epsilon cells. Though the scVelo and Pyro-Velocity models have qualitatively comparable vector fields, the results generated by scVelo were greatly influenced by gene selection, neighboring cell number for gene expression smoothing, and cannot provide confidence assessment of lineage directionality (**Supplementary Fig. 2a**). In addition, Pyro-Velocity uncertainty visualizations uncovered that differentiated cells had a larger vector field uncertainty than stem and progenitor cells, as would be expected for terminally differentiated cell states where progression and lineage trajectories would not be anticipated. Pyro-Velocity also recovered a developmental order in close agreement with what was estimated by Cytotrace (Spearman's r=0.95, p-value<1E-10 between latent time and developmental order) [17] - a state-of-the-art orthogonal method used to predict cell differentiation based on the number of expressed genes per cell.

We next used Pyro-Velocity to unbiasedly uncover potential cell fate determinant markers genes within the E15.5 mouse pancreas. To this end, we first selected the top 300 genes with the best goodness of fit (negative mean absolute errors across cells). Next, we ranked them based on the positive (or negative) correlation between inferred shared time and corresponding denoised spliced expression per cell (**Fig. 2d, Methods**). This procedure successfully recovered key developmental genes that are activated (or repressed) during endocrinogenesis (**Fig. 2d-e**, **Supplementary Fig. 2**). Among the top positively correlated genes identified by Pyro-Velocity, endocrine Beta cells expressed *insulin 1* and *2* (*Ins1, Ins2*) and *Nnat,* which is a key regulator Beta cell insulin generation[18] and *Ppp1r1a* which is a marker of Beta cells[19]. The ranking procedure proposed by scVelo failed to identify *Ins1, Ins2, Nnat,* and *Ppp1r1a* as lineage-restricted genes in their top 50 ranked gene list. The inability of scVelo to identify these key developmental genes could be likely explained by the kNN smoothing of read count and filtering

steps (**Supplementary Table 1, 2**). Visual inspection of these four top genes by Pyro-Velocity revealed a single activation phase during cell type transitions based on phase portrait curves. We also propose their visualization through “rainbow plots” to illustrate denoised single gene spliced expression across shared time and to investigate the inferred cell order based on known annotations. Further, UMAP embedding of cells colored based on denoised gene expression confirmed the cell fate markers’ smooth and progressive activation (or repression) towards endocrine cells (**Fig. 2e**, **Supplementary Figure 2**).

Ground truth velocity is not a quantity we can easily access. Therefore, no robust and quantitative benchmarking of RNA velocity methods has been proposed. However, recent single-cell assays allow recording lineage relationships and can be used to assess the ability of different methods in recapitulating cell fate progression. Therefore, we next systematically compared scVelo and Pyro-Velocity ability to recover single cell fates in Lineage And RNA RecoverY (LARRY) barcoded hematopoietic cells. LARRY leverages unique lentiviral barcodes inserted into 3’ untranslated region (UTR) of GFP reporter that is integrated as a single copy into parental cells. Then progeny can be clonally traced to follow the descendent cell fates over time. LARRY has been recently applied to *in vitro* differentiation of human blood and accurately recovered expected cell fate and lineage trajectories[20]. This dataset sampled differentiation over the course of 2, 4, and 6 days and contained 49,302 cells that each could be traced with at least one barcode. Because LARRY couples single-cell transcriptomics and direct cell lineage tracing, this data set provides a unique opportunity to benchmark RNA Velocity quantitatively and to provide insights into the correct interpretation of recovered Velocity vector fields and latent cell times. To quantitatively assess the agreement between observed and predicted fates, we considered two metrics: 1. average cosine similarity (*cs*) between vectors from the RNA Velocity vector field and the *clonal progression vector field,* i.e., a vector field derived directly from the lineage data (**Methods**), 2. Spearman’s correlation (ρ) between cell latent (shared) time and fate potency scores derived from Cytotrace and Cospar. Cospar, our gold standard, is a state-of-the-art method designed explicitly for predicting fate potency based on LARRY data, while Cytotrace leverages only transcriptomic data. We selected cell state changes associated with monocytes and neutrophils lineages because both lineages have the largest numbers of cells in the scRNA sequencing data set and well-known cell fate trajectories[20]. As expected, the Pyro-Velocity vector field accurately predicted cell state trajectories from hematopoietic progenitor cells into mature monocytes or neutrophils and had high cosine similarity between the inferred vector field and clonal progression vector field assigned by LARRY (monocytes *cs*=0.49, neutrophils *cs*=0.38). High Spearman’s correlation was also seen between shared latent time and Cospar fate potency

(monocytes ρ=0.60, neutrophils ρ=0.74) or shared latent time with Cytotrace fate potency (monocytes ρ=0.925, neutrophils ρ=0.925). By contrast, scVelo generated vector fields with reversed order or low correlation (cosine similarity: -0.38, monocytes, cosine similarity: 0.32, neutrophils). Moreover, Cospar fate potency scores were negative (ρ=-0.59, monocytes) or non-significantly correlated (ρ=-0.07, neutrophils). A similar low correlation was observed for scVelo and Cytotrace fate potency scores (monocytes ρ=0.309, neutrophils ρ=0.751) (**Fig. 3a-b, Supplementary Fig. 4c**).

Next, we evaluated the ability of the Pyro-Velocity models to recover bi-fate or multi-fate trajectories by considering clones belonging to monocyte and neutrophil lineages or using the entire dataset. Not surprisingly, both methods showed their limitations, given that they are not designed to model and capture multiple fate trajectories (**Supplementary Fig. 5c-d**). The recovered vector fields and cell latent (shared) time show inconsistent directions for both methods. However, Pyro-Velocity identified higher uncertainty for some of the problematic regions. Furthermore, we discovered that vector field and shared time estimation could be significantly improved by extending the original kinetic model by allowing each gene a time lag for the transcriptional activation and a baseline expression level potentially different than 0 (Pyro-Velocity Model 2, **Methods**) (**Fig. 3c,d, Supplementary Fig. 1b,c**). This extended model recapitulated a vector field consistent with the *clonal progression vector field* for most branches with some spurious predictions within the intermediate cell state between stem cells and Megakaryocytes and most terminal Neutrophils (bi-fate cosine similarity: 0.62, multi-fate cosine similarity: 0.63). However, the proposed uncertainty estimation accurately flagged these subpopulations to warn users of potential false discoveries. Notably, this extended model significantly improved the correlation of cell shared time with the clonal fate potency (bi-fate ρ=0.65, multi-fate ρ=0.74) and also improved the correlation with Cytotrace fate potency (bi-fate ρ=0.931, multi-fate ρ=0.614). Remarkably, the shared time per cell from Pyro-Velocity also outperformed Cytotrace for developmental order prediction (**Supplementary Fig. 4a,c**). Importantly, this extended model retained its ability to accurately identify cell lineage trajectories and shared time for all the other datasets analyzed above **(Supplementary Fig. 3)**. Taken together, our approach using LARRY barcoded datasets established the first quantitative benchmark of RNA velocity-based cell fate trajectory inference and provides an exciting new framework for continued optimization of multi-fate predictions in the future.

In conclusion, Pyro-Velocity provides the first probabilistic, scalable, and end-to-end Bayesian inference framework for RNA Velocity that learns cell dynamics and recovers cell fate

from multimodal raw read counts. This approach provides uncertainty estimations of cell fate trajectories and novel visualization approaches to mitigate false discoveries. Pyro-Velocity recovered shared time and vector fields that are recapitulated in true developmental order and outperformed state-of-the-art methods like Cytotrace and scVelo (**Supplementary Fig. 4a**).

Software availability

We developed our Pyro-Velocity models with Pyro (version 1.6.0) probabilistic programming language and PyTorch (version 1.8.1). Pyro-Velocity source code is available at https://github.com/pinellolab/pyrovelocity. The version of Pyro-Velocity (v0.1.0) used to generate the results presented in this manuscript has been deposited on Zenodo: https://zenodo.org/record/7072876#.Yx-2e-zMLxg.

Data availability

All the data presented in this manuscript were collected from previously published studies. The fully mature PBMC, pancreas datasets are downloaded and loaded through the scVelo using scvelo.datasets.pancreas(),and scvelo.datasets.pbmc68k(). The LARRY inDrop sequencing data was processed by the LARRY authors using dropEst[21] and velocyto. This dataset can be accessed at https://figshare.com/articles/dataset/larry_invitro_adata_sub_raw_h5ad/20780344.

Acknowledgments

This project has been made possible in part by grant number 2022- 249212 from the Chan Zuckerberg Initiative DAF, an advised fund of Silicon Valley Community Foundation. L.P. received support from the U.S. NIH (R35 HG010717) and Silicon Valley Community Foundation (2022- 249212). D.M.L is supported by R01CA226926, R01 CA215118, R01CA154923, R24 OD031955, CureSearch Acceleration Initiative, and the MGH Scholars Award**.** We thank the Pinello and Langenau labs members for helpful discussions.

Author contributions

L.P., Q.Q., and E.B. conceived the work. Q.Q. and E.B created the software. Q.Q. analyzed the data, and prepared the figures and tables. L.P and D.L. supervised the project. L.P., D.L., E.B., G.L.M helped in designing experiments and analyses. All the authors contributed to the writing of the manuscript.

Competing financial interests statement

L.P. has financial interests in Edilytics, Inc., Excelsior Genomics, and SeQure Dx, Inc. L.P.'s interests were reviewed and are managed by Massachusetts General Hospital and Partners HealthCare in accordance with their conflict of interest policies.

Methods

Model formulation

We assume the dynamical gene expression is determined by the RNA splicing process, and infer the unspliced and spliced gene expression level from the differential equations proposed in *velocyto* and scVelo[1,3]:

$$\frac{du\left(\tau^{(k_{cg})}\right)}{d\tau^{(k_{cg})}} = \alpha^{(k_{cg})} - \beta_g u\left(\tau^{(k_{cg})}\right)$$

$$\frac{ds\left(\tau^{(k_{cg})}\right)}{d\tau^{(k_{cg})}} = \beta_g u\left(\tau^{(k_{cg})}\right) - \gamma_g s\left(\tau^{(k_{cg})}\right)$$

In the equation, the subscript *c* is the cell dimension, *g* is the gene dimension, $\left(u\left(\tau^{(k_{cg})}\right), s\left(\tau^{(k_{cg})}\right)\right)$ are the unspliced and spliced expression functions given the change of time per cell and gene. τ_{cg} represents the displacement of time per cell and gene with $\tau^{(k_{cg})} = softplus\left(t_c - t_0^{(k_{cg})}{}_g\right) = log(1 + exp(t_c - t_0^{(k_{cg})}{}_g))$, in which t_c is the shared time per cell, $t_0^{(k_{cg})}{}_g$ is the gene-specific switching time. Each cell and gene combination has its transcriptional state $k_{cg} \in \{0,1\}$, where 0 indicates the activation state and 1 indicates the expression state. Each gene has two switching times for representing activation and repression: $t_0^{(0)}{}_g$ is the first switching time corresponding to when the gene expression starts to be activated, $t_0^{(1)}{}_g$ is the second switching time corresponding to when the gene expression starts to be repressed.

We note that $\alpha^{(1)}$ is shared for all the genes, while $\alpha^{(0)}{}_g$is learned independently for each gene.

The analytic solution of the differential equations to predict spliced and unspliced gene expression given their parameters is derived by the authors of scVelo[1] and a theoretic RNA Velocity study[10].

$$u\left(\tau^{(k_{cg})}\right) = u_0^{(k_{cg})}{}_g e^{-\beta_g \tau^{(k_{cg})}} + \frac{\alpha^{(k_{cg})}}{\beta_g}\left(1 - e^{-\beta_g \tau^{(k_{cg})}}\right)$$

If $\beta \neq \gamma$, then

$$s\left(\tau^{(k_{cg})}\right) = s_0^{(k_{cg})}{}_g e^{-\gamma_g \tau^{(k_{cg})}} + \frac{\alpha^{(k_{cg})}}{\gamma_g}\left(1 - e^{-\gamma_g \tau^{(k_{cg})}}\right) + \frac{\alpha^{(k_{cg})} - \beta_g u_0^{(k_{cg})}{}_g}{\gamma_g - \beta_g}\left(e^{-\gamma_g \tau^{(k_{cg})}} - e^{-\beta_g \tau^{(k_{cg})}}\right)$$

If $\beta = \gamma$, then

$$s\left(\tau^{(k_{cg})}\right) = s_0^{(k_{cg})}{}_g e^{-\beta_g \tau^{(k_{cg})}} + \frac{\alpha^{(k_{cg})}}{\beta_g}\left(1 - e^{-\beta_g \tau^{(k_{cg})}}\right) - \left(\alpha^{(k_{cg})} - \beta_g u_0^{(k_{cg})}{}_g\right)\tau^{(k_{cg})} e^{-\beta_g \tau^{(k_{cg})}}$$

To simplify the above equation, we consider the case when k_{cg} =0 and $\beta_g \neq \gamma_g$, then

$$u\left(\tau^{(0)}\right) = u_0^{(0)}{}_g e^{-\beta_g \tau^{(0)}} + \frac{\alpha^{(0)}{}_g}{\beta_g}\left(1 - e^{-\beta_g \tau^{(0)}}\right)$$

$$s\left(\tau^{(0)}\right) = s_0^{(0)}{}_g e^{-\gamma_g \tau^{(0)}} + \frac{\alpha^{(0)}{}_g}{\gamma_g}\left(1 - e^{-\gamma_g \tau^{(0)}}\right) + \frac{\alpha^{(0)}{}_g - \beta_g u_0^{(0)}{}_g}{\gamma_g - \beta_g}\left(e^{-\gamma_g \tau^{(0)}} - e^{-\beta_g \tau^{(0)}}\right)$$

When k_{cg} =0 and $\beta_g = \gamma_g$, $u\left(\tau^{(0)}\right)$ has the same solution, $s\left(\tau^{(0)}\right)$ becomes:

$$s\left(\tau^{(0)}\right) = s_0^{(0)}{}_g e^{-\gamma_g \tau^{(0)}} + \frac{\alpha^{(0)}{}_g}{\gamma_g}\left(1 - e^{-\gamma_g \tau^{(0)}}\right) - \left(\alpha^{(0)}{}_g - \beta_g u_0^{(0)}{}_g\right)\tau^{(0)} e^{-\beta_g \tau^{(0)}}$$

When k_{cg} =1 and $\beta_g \neq \gamma_g$, then

$$u\left(\tau^{(1)}\right) = u_0^{(1)}{}_g e^{-\beta_g \tau^{(1)}}$$

$$s\left(\tau^{(1)}\right) = s_0^{(1)}{}_g e^{-\gamma_g \tau^{(1)}} + \frac{-\beta_g u_0^{(1)}{}_g}{\gamma_g - \beta_g}\left(e^{-\gamma_g \tau^{(1)}} - e^{-\beta_g \tau^{(1)}}\right)$$

When k_{cg} =1 and $\beta_g = \gamma_g$, $u\left(\tau^{(1)}\right)$ has the same solution, $s\left(\tau^{(1)}\right)$ becomes:

$$s\left(\tau^{(1)}\right) = s_0^{(1)}{}_g e^{-\gamma_g \tau^{(1)}} + \beta_g u_0^{(1)}{}_g \tau^{(1)} e^{-\beta_g \tau^{(1)}}$$

We use these solutions to formulate an end-to-end probabilistic generative model that relates prior distributions on kinetic parameters to a distribution on pairs of observed unspliced and spliced read count matrices:

$$\alpha^{(0)}{}_g \sim LogNormal(0,1)$$

$$\beta_g \sim LogNormal(0,1)$$

$$\gamma_g \sim LogNormal(0,1)$$

$$t_0^{(k_{cg})}{}_g = \begin{cases} t_0^{(0)}{}_g \sim Normal(0,1), k_{cg} = 0 \\ t_0^{(1)}{}_g = t_0^{(0)}{}_g + \Delta switching_g, \quad \Delta switching_g \sim LogNormal(0,1), k_{cg} = 1 \end{cases}$$

$$t_c \sim LogNormal(0,1)$$

$$k_{cg} \sim Bernoulli(logits = t_c - t_0^{(1)})$$

$$\tau^{(k_{cg})} = softplus\left(t_c - t_0^{(k_{cg})}{}_g\right)$$

$$u_{cg} = Measurement_u\left(u\left(\tau^{(k_{cg})}\right);\ u_{cg}^{obs}\right)$$

$$s_{cg} = Measurement_s(s\left(\tau^{k_{cg}}\right); s_{cg}^{obs})$$

$u\left(\tau^{(k_{cg})}\right)$ and $s\left(\tau^{(k_{cg})}\right)$ are called the denoised gene expression calculated from the velocity analytic solution input with the kinetics random variables. u_{cg} and s_{cg} are the spliced and unspliced read count sampled from the Poisson models. u_{cg}^{obs} and s_{cg}^{obs} are the observed spliced and unspliced read count tables.

The generative process $Measurement(\,.\,)$ for observed unspliced read counts given denoised unspliced gene transcript expression level $u\left(\tau^{(k_{cg})}\right)$ (and identical for observed spliced read counts) models the expected number of observed reads for a given gene in a given cell as the number of transcripts times the ratio of the cell's total reads to total transcripts:

$$\widehat{u_c^{obs}} = \sum_g u_{cg}^{obs}, \qquad \widehat{u_c} = \sum_g u\left(\tau^{(k_{cg})}\right),$$

$$\eta_c^{(u)} \sim Normal\left(\widehat{u_c^{obs}}, std_c\left(\widehat{u_c^{obs}}\right)\right)$$

$$\mu_{cg}^{(u)} = log\left(u\left(\tau^{k_{cg}}\right)\right) + log\left(\eta_c^{(u)}\right) - log(\widehat{u_c})$$

$$u_{cg}^{obs} \sim Poisson\left(\lambda = exp\left(\mu_{cg}^{(u)}\right)\right)$$

For the first Pyro-Velocity model (Model 1), we constrain the shared time to be strictly larger than $t_0^{(0)}$ by introducing auxiliary random variables $t_constraint_{cg} \sim Bernoulli(\text{logits} = t_c - {t_0^{(0)}}_g)$ and setting their values to 1, and we set the initial condition per gene to be:

$$\left({u_0^{(k_{cg})}}_g, {s_0^{(k_{cg})}}_g\right) = \begin{cases} (0,0), k_{cg} = 0 \\ \left(u\left(\Delta switching_g\right)_g, s(\Delta switching_g)_g\right), k_{cg} = 1 \end{cases}$$

For the extended Pyro-Velocity model (Model 2), we remove the shared time constraint $t_constraint_{cg}$, thus allow a time lag per gene that might be caused by delayed gene activation and set the initial condition per gene as random variables that are strictly positive $({u_0^{(0)}}_g, {s_0^{(0)}}_g)$, which allow genes having basal expression level before gene activation. Then, we compute the gene expression at the second switching time as $\left(u_{0(g)}^{(1)}, s_{0(g)}^{(1)}\right) = \left(u\left(\Delta switching_g\right)_g, s(\Delta switching_g)_g\right)$ which shares the same initial condition $({u_0^{(0)}}_g, {s_0^{(0)}}_g)$ where ${u_0^{(0)}}_g \sim LogNormal(0,1)$, ${s_0^{(0)}}_g \sim LogNormal(0,1)$.

Variational inference

Given observations $\tilde{X}_{cg} = (u_{cg}^{obs}, s_{cg}^{obs})$, we would like to compute the posterior distribution over the random variables $\theta = (t_c, \eta_c^{(u)}, \eta_c^{(s)})$, $\psi = ({t_0^{(0)}}_g, \Delta switching_g, {\alpha^{(0)}}_g, \beta_g, \gamma_g)$, but exact Bayesian inference is intractable in this model. We use Pyro to automatically integrate out the local discrete latent variables *k*, which is defined as the cell and gene transcriptional state (see above), and approximate the posterior over the remaining latent variables using variational inference[11], which converts intractable integrals into optimization problems that can be solved with off-the-shelf tools. In variational inference, we maximize a tractable bound (the evidence lower bound (ELBO)) on

the Kullback-Leibler (KL) divergence $KL(\, q_{\varphi}(\theta,\psi) \parallel p(\theta,\psi|\tilde{X}_{cg}))$ between a parametric family of tractable probability distributions *q* and the intractable true posterior:

$$\varphi^{*} = argmax_{\varphi}\, ELBO = argmax_{\varphi}\, \left\{ E_{q_{\varphi}} \left[\log\left(p\left(\tilde{X}_{cg},\theta,\psi\right)\right) - \log\left(q_{\varphi}(\theta,\psi)\right)\right]\right\}$$

We approximate our model's posterior distribution p($\theta,\psi|\tilde{X}$) with a tractable family of probability distributions $q_{\varphi}(\theta,\psi) = q_{\varphi_1}(\theta) q_{\varphi_2}(\psi)$, where $q_{\varphi_1}(\theta)$ is a product of independent univariate Gaussian distributions with learnable location and scale parameters and $q_{\varphi_2}(\psi)$ is a multivariate Gaussian distribution with a learnable location parameter and a learnable low-rank covariance matrix.

We solve the resulting stochastic optimization problem using a version of Automatic Differentiation Variational Inference (ADVI) [11]: we obtain gradient estimates by differentiating a Monte Carlo estimate of the ELBO and update the variational parameters φ using stochastic gradient ascent. Our implementation and experiments use generic variational families and Monte Carlo ELBO and gradient estimators provided by Pyro[9] , an open source library for probabilistic machine learning, and a version built into Pyro of the adaptive stochastic gradient ascent algorithm Adam augmented with gradient clipping to enhance numerical stability during training.

Model training

For the pancreas, PBMC, the uni-fate and bi-fate LARRY single-cell data, since the data dimension is relatively small, we run Model 1 and Model 2 with a minimum of 100 epochs and a maximum of 4000 epochs. For each epoch, we input spliced and unspliced read count from all the cells and determine the convergence condition with an early stopping strategy in which the patience is set to 45 and consumes one patience if the minimal ELBO improvement of training data per epoch is lower than 1e-4 of previous loss. The learning rate is set to be 1e-2 with a decay rate of $0.1^{1/4000}$ per epoch.

For the multi-fate LARRY dataset, since this is a large dataset with over 40K cells, we use mini batches of cells to train Model 1 and Model 2 with a minimum of 100 epochs and a maximum of 1000 epochs. Specifically, the batch size was set to be 4000 cells for both models with an early stopping patience 45 and consume one patience if the minimal ELBO improvement of training data per epoch lower than 1e-3 of previous loss. The learning rate is set to be 1e-2 with a decay rate of $0.1^{1/1000}$ per epoch.

All models were trained on a machine with an NVIDIA A100 graphic card and the CentOS 7 operating system.

Posterior prediction

To benchmark Pyro-Velocity performance in predicting cell fate, we generated the posterior samples measurement $x_{cg} = \left(u\left(\tau^{(k_{cg})}\right), s\left(\tau^{(k_{cg})}\right)\right)$ or $x_{cg} = (u_{cg}, s_{cg}), t_c, \beta_g, \gamma_g$ from the same single cell RNA-seq using $N = 30$ Monte Carlo samples from the posterior predictive distribution following: $p\left(x_{cg}\middle|\theta,\psi\right)p(\theta,\psi|) \approx p\left(x_{cg}\middle|\theta,\psi\right)q_{\varphi}(\theta,\psi)$. x_{cg} can be posterior samples of either denoised gene expression (used for phase portraits and vector field-based trajectory inference) or raw read counts (used for gene selection). Then we calculated the posterior samples of RNA velocity as $v_{cg} = \frac{ds\left(\tau^{(k_{cg})}\right)}{d\tau^{(k_{cg})}}$ based on posterior samples measurement of denoised gene expression x_{cg} and β_g, γ_g.

Prioritization of the cell fate markers

We prioritize the cell fate markers using two metrics: First, the Pearson correlation between each gene's posterior mean of the denoised spliced expression and posterior mean of the shared time; Second, the negative mean absolute errors of each gene's observed spliced, and unspliced read counts with posterior predictive samples of spliced and unspliced raw read counts, i.e.$\frac{1}{Nn_c}\sum_i \sum_c (x_{icg} - \tilde{X}_{cg})$, N is the posterior sample number that is set to 30, i is the posterior

sample index, n_c is the cell number. We first select the top 300 genes with the highest negative mean absolute errors and then rank the 300 genes based on the most positively correlated genes and the least negatively correlated genes. We use the same strategy for scVelo to rank the markers by the model likelihood to get the top 300 genes and then prioritize these genes by Pearson's correlation between scVelo latent time and normalized expression.

Single-cell data preprocessing

We used Scanpy and scVelo to handle the data input and output; thus, both h5ad and loom files generated by velocyto and kallisto[22] are supported.

The fully mature PBMC dataset was processed with the same procedure proposed in a recent review paper[7] (https://scvelo.readthedocs.io/perspectives/Perspectives/). We reproduced this procedure using the scVelo package and raw read counts of the same top three dynamical genes NKG7, IGHM, and GNLY with the best likelihoods.

The pancreas dataset was processed with the scVelo with the following options:

scv.pp.filter_and_normalize(adata, min_shared_counts=30, n_top_genes=2000)

scv.pp.moments(adata, n_pcs=30, n_neighbors=30)

The same top variable genes with raw spliced and unspliced read counts were used as input for the Pyro-Velocity model.

The original LARRY dataset of in vitro Hematopoiesis containing 130,887 cells was first filtered to remove cells without LARRY barcoding. 49,302 cells were recovered after this step with at least one LARRY barcode. For simplicity, we termed this filtered dataset with multiple cell fate (multi-fate) as the full dataset. Based on this dataset, we created two datasets with uni-fate progression toward monocyte or neutrophil based on the lineage LARRY barcodes and time information. Namely, we selected sets of cells with a single LARRY barcode, spanning three time points (day 2, 4, 6), and all the cells from the last time point (day 6) belong to a unique cell type (either

monocyte or neutrophil). The two uni-fate datasets were combined to represent the bi-fate LARRY dataset. The multi-fate full dataset was processed using the same options as the pancreas dataset; the rest of the uni-fate and bi-fate datasets were processed using the following parameters:

```
scv.pp.filter_and_normalize(adata, n_top_genes=2000, min_shared_counts=20)

scv.pp.moments(adata)
```

scVelo model

We benchmarked the dynamical RNA velocity model implemented in scVelo (version 0.2.4) for the pancreas and the four LARRY datasets using the same user options :

```
scvelo.tl.recover_dynamics(adata, n_jobs=30)

scvelo.tl.velocity(adata, mode='dynamical')
```

Then, we tested a set of user options, including the neighboring cell numbers and the top variable gene numbers, in the pancreas dataset to explore the stability of the scVelo dynamical model. For the fully mature PBMC dataset, we followed the notebook proposed by the original authors (https://scvelo.readthedocs.io/perspectives/Perspectives/), i.e., we used the stochastic RNA velocity model implemented in scVelo with the top three likelihood genes. The latent time from scVelo was computed using their provided function: scvelo.tl.latent_time(adata).

Trajectories inference by vector fields

Velocity vector field

The velocity-based vector fields were generated using the scVelo function as:

```
scvelo.tl.velocity_graph(adata)

scvelo.tl.velocity_embedding(adata, vkey='velocity', basis='embed')
```

We used the default options for projecting the vector fields from scVelo models. Unlike scVelo, Pyro-Velocity uses posterior samples (or posterior mean) of the denoised spliced gene expression and posterior samples (or posterior mean) of the velocity estimation for building the transition matrix with cosine similarity. Pyro-Velocity uses the same projection method as scVelo for projecting the transition matrix into the two-dimensional vector field on the user-provided embedding spaces.

Clonal progression vector field

No method exists for projecting clonal cell fates from LARRY datasets into a vector field. Thus, we designed a strategy for projecting *clonal progression vector fields:* based on the cells sharing the same LARRY barcode, we considered the vectors connecting the embedding centroid of cells at consecutive time points and then leveraged Gaussian smoothing of nearest centroid directions to represent on a regular grid over the cell manifold the *clonal progression vector field* informed by LARRY barcodes.

Cell fate uncertainty estimation

Pyro-Velocity evaluates the cell fate uncertainty by four statistics. The first statistic is the uncertainty of shared time which is evaluated using the standard deviation of posterior samples of t_c for each cell. The second statistic is the angular standard deviation[23] across posterior samples of the velocity-based vector field. The value of this statistic was mapped linearly to the range [0, 360] for interpretability. The third statistic is the Rayleigh test[23] for posterior samples of the velocity-based vector fields for each cell, whose p values have been corrected for multiple tests using Benjamini-Hochberg FDR methods; if FDR is less than 0.05, the cell has one statistically significant future direction. The second and third circular statistics were implemented using Astropy package (version 5.1)[24]. The last statistic is state uncertainty which evaluates the mean Euclidean distance between posterior mean and posterior samples of the raw read count prediction: $\frac{1}{N}\sum_i \sqrt{\sum_g (x_{icg} - \frac{1}{N}\sum_i x_{icg})^2}$ for each cell.

Cospar model

We computed the Cospar(version 0.1.9)[25] fate potency score for the above processed multi-fate LARRY dataset, and also selected the subset of the fate potency score for evaluating bi-fate and uni-fate datasets. The following functions achieve this:

```
cospar.tmap.infer_Tmap_from_multitime_clones(adata, compute_new=True,
                                             clonal_time_points=[2,4,6], later_time_point=6,
                                             smooth_array=[20,15,10],
                                             sparsity_threshold=0.2, max_iter_N=3)
cospar.pl.fate_potency(adata, used_Tmap='transition_map',
                       map_backward=True,method='norm-sum',color_bar=True,fate_count=True)
```

Cytotrace model

We implemented a faithful Cytotrace Python package based on the R version for benchmarking the Pyro-Velocity and scVelo's shared (latent) time prediction in the Python environment. This implementation is available in the Pyro-Velocity GitHub repository (https://github.com/pinellolab/pyrovelocity/blob/master/pyrovelocity/cytotrace.py).

Model evaluation and selection

Our Pyro-Velocity provides two model formulations as described above. Users can compare their predictive performance to decide which of the two models to use for a given dataset. Namely, the mean absolute errors of posterior samples prediction of spliced and unspliced raw read counts x_{icg} from its observation $\tilde{X}_{cg}$ per gene and cell combinations: $\frac{1}{N_i N_c N_g}\sum_i \sum_c \sum_g |x_{icg} - \tilde{X}_{cg}|$. i is the posterior sample index, N_i is the posterior sample number, N_c is the cell number, N_g is the gene number. Then it is possible to evaluate the cell fate inference performance of the models based on two metrics: 1. the consistency between the velocity vector field and the *clonal progression vector field* using grid-based cosine similarity; 2. Spearman's correlation of velocity-based shared

(latent) time with Cospar fate potency and the silver standard Cytotrace score. Both metrics are implemented in Scipy.

References

1. Bergen V, Lange M, Peidli S, Wolf FA, Theis FJ. Generalizing RNA velocity to transient cell states through dynamical modeling. *Nat Biotechnol*. 2020;38(12):1408-1414. doi:10.1038/s41587-020-0591-3

2. Qiu X, Zhang Y, Yang D, et al. Mapping Vector Field of Single Cells. *bioRxiv*. Published online 2019:696724. doi:10.1101/696724

3. La Manno G, Soldatov R, Zeisel A, et al. RNA velocity of single cells. *Nature*. 2018;560(7719):494-498. doi:10.1038/s41586-018-0414-6

4. Svensson V, Pachter L. RNA Velocity: Molecular Kinetics from Single-Cell RNA-Seq. *Mol Cell*. 2018;72(1):7-9. doi:10.1016/j.molcel.2018.09.026

5. Qiu X, Zhang Y, Martin-Rufino JD, et al. Mapping transcriptomic vector fields of single cells. *Cell*. 2022;185(4):690-711.e45. doi:10.1016/j.cell.2021.12.045

6. Bastidas-Ponce A, Tritschler S, Dony L, et al. Comprehensive single cell mRNA profiling reveals a detailed roadmap for pancreatic endocrinogenesis. *Dev*. 2019;146(12). doi:10.1242/dev.173849

7. Bergen V, Soldatov RA, Kharchenko P V, Theis FJ. RNA velocity—current challenges and future perspectives. *Mol Syst Biol*. 2021;17(8):1-9. doi:10.15252/msb.202110282

8. Zheng GXY, Terry JM, Belgrader P, et al. Massively parallel digital transcriptional profiling of single cells. *Nat Commun*. 2017;8. doi:10.1038/ncomms14049

9. Bingham E, Chen JP, Jankowiak M, et al. Pyro: Deep universal probabilistic programming. *J Mach Learn Res*. 2019;20:1-6.

10. Li T, Shi J, Wu Y, Zhou P. On the Mathematics of RNA Velocity I: Theoretical Analysis. *bioRxiv*. Published online 2020:2020.09.19.304584. https://doi.org/10.1101/2020.09.19.304584

11. Kucukelbir A, Blei DM, Gelman A, Ranganath R, Tran D. Automatic Differentiation Variational Inference. *J Mach Learn Res*. 2017;18:1-45.

12. Gayoso A, Weiler P, Lotfollahi M, et al. Deep generative modeling of transcriptional dynamics for RNA velocity analysis in single cells. *bioRxiv*. Published online January 1, 2022:2022.08.12.503709. doi:10.1101/2022.08.12.503709

13. Gu Y, Blaauw D, Welch JD. Bayesian Inference of RNA Velocity from Multi-Lineage Single-Cell Data. *bioRxiv*. Published online January 1, 2022:2022.07.08.499381. doi:10.1101/2022.07.08.499381

14. Farrell S, Mani M, Goyal S. Inferring single-cell dynamics with structured dynamical representations of RNA velocity. *bioRxiv*. Published online January 1, 2022:2022.08.22.504858. doi:10.1101/2022.08.22.504858

15. Gorin G, Fang M, Chari T, Pachter L. RNA velocity unraveled. *bioRxiv*. Published online January 1, 2022:2022.02.12.480214. doi:10.1101/2022.02.12.480214

16. Gao M, Qiao C, Huang Y. UniTVelo: temporally unified RNA velocity reinforces single-cell trajectory inference. *bioRxiv*. Published online January 1, 2022:2022.04.27.489808. doi:10.1101/2022.04.27.489808

17. Gulati GS, Sikandar SS, Wesche DJ, et al. Single-cell transcriptional diversity is a hallmark of developmental potential. *Science (80-)*. 2020;411(January):649848. doi:10.1101/649848

18. Millership SJ, Da Silva Xavier G, Choudhury AI, et al. Neuronatin regulates pancreatic β cell insulin content and secretion. *J Clin Invest*. 2018;128(8):3369-3381.

doi:10.1172/JCI120115

19. Jiang L, Brackeva B, Ling Z, et al. Potential of protein phosphatase inhibitor 1 as biomarker of pancreatic β-cell injury in vitro and in vivo. *Diabetes*. 2013;62(8):2683-2688. doi:10.2337/db12-1507

20. Weinreb C, Rodriguez-Fraticelli A, Camargo FD, Klein AM. Lineage tracing on transcriptional landscapes links state to fate during differentiation. *Science (80-)*. 2020;367(6479):eaaw3381. doi:10.1126/science.aaw3381

21. Petukhov V, Guo J, Baryawno N, et al. dropEst : pipeline for accurate estimation of molecular counts in droplet-based single-cell RNA-seq experiments. Published online 2018:1-16.

22. Melsted P, Booeshaghi AS, Liu L, et al. Modular , efficient and constant-memory single-cell RNA-seq preprocessing. *Nat Biotechnol*. 2021;39(July). doi:10.1038/s41587-021-00870-2

23. Berens P. CircStat : A MATLAB Toolbox for Circular Statistics. *J Stat Softw*. 2009;31(10):293-295. doi:10.18637/jss.v031.i10

24. Price-Whelan AM, Sipőcz BM, Günther HM, et al. The Astropy Project: Building an Open-science Project and Status of the v2.0 Core Package. *Astron J*. 2018;156(3):123. doi:10.3847/1538-3881/aabc4f

25. Wang SW, Herriges MJ, Hurley K, Kotton DN, Klein AM. CoSpar identifies early cell fate biases from single-cell transcriptomic and lineage information. *Nat Biotechnol*. Published online 2022. doi:10.1038/s41587-022-01209-1

Figures

Figure 1 – Pyro-Velocity is a fully generative Bayesian method with uncertainty estimation of velocity vector fields and shared latent time, based on raw counts and without ad-hoc preprocessing steps. a. Schematic comparing current RNA Velocity and Pyro-Velocity workflows. Both methods take in input spliced and unspliced read count tables and output RNA velocity vector fields and latent (shared) time. Previous methods require ad-hoc preprocessing steps, including the arbitrary selection of top variable genes, number of principal components, and kNN-pooling to impute both spliced and unspliced expression within the PCA-reduced spliced expression space. In addition, manually tuned ad-hoc postprocessing steps are used to compute the final vector field and latent time based on aggregation and smoothing of the independent velocity estimates for each gene which can dramatically change the velocity estimates. In contrast, Pyro-Velocity is a fully generative and probabilistic model that uses raw read counts. Our model requires minimal preprocessing and postprocessing steps and outputs a common *shared* time based on all the genes without additional aggregation steps. It also provides uncertainty estimation based on posterior samples. **b.** The main outputs of the Pyro-Velocity framework. Visualization of single gene expression across inferred developmental cell order using rainbow plots (right panel).

Figure 2 - Pyro-Velocity outperforms other RNA-velocity methods and accurately predicts cell lineage trajectories. a,b. RNA Velocity model comparisons from fully mature peripheral blood mononuclear cells (PBMCs, t-SNE embedded plots, a) or E15.5 mouse pancreas (UMAP embedded plots, b-c). From left to right in panels a and b: 1. Cells colored by cell type; 2. Stream plot of velocity vector field defined by scVelo; 3. Stream plot of the posterior mean vector field from Pyro-Velocity; 4. Pyro-Velocity visualization of cell dynamics for 3 arbitrarily selected single cells; 5. *Single-cell vector field uncertainty* where each cell is colored based on the scaled angular standard deviation to visualize uncertainty based on n=30 iterated, posterior samples depicted as

arrows noted throughout our analysis; 6. *Averaged vector field uncertainty* where each vector is colored by the averaged vector field uncertainty. The scaled angular standard deviation range is [0 360], where 0 corresponds to the highest confidence for a single direction, 360 corresponds to the highest uncertainty, and no directionality preference based on 30 posterior samples (4-6). Vectors were projected into the UMAP space using the scVelo API based on cosine similarity-derived transition matrix. **c.** UMAP embedded cells colored by the average cell shared time across 30 posterior samples. The Spearman's correlation between Pyro-Velocity and Cytotrace is provided. **d.** Pyro-Velocity automatic gene selection and ranking for predicting important genes involved in pancreas development. **e.** Phase portraits of posterior samples average spliced and unspliced gene expression (left), rainbow plots showing denoised spliced gene expression (middle), and gene expression within UMAP embedded space. Single dots correspond to individual cells (left, right) or vertical lines (middle). Cell type annotation coloring is the same as in panel b.

Figure 3 - Pyro-Velocity accurately predicts cell fate choices and lineage trajectory in human LARRY barcoded hematopoietic cells. Systematic benchmarking of cell fate prediction methods - predicting uni-lineage cell fates (**a-b**, Model 1), bifurcated lineages (**c**, Model 2), or all lineages together (**d**, Model 2). Panels from left to right show on the same SPRING embedding: 1. Cell types; 2. Clonal progression vector field derived based on the observed clonal barcodes (**Methods**); 3. scVelo recovered vector field; 4. Pyro-Velocity single cell level vector field uncertainty (angular standard deviation from 30 posterior samples); 5. Pyro-Velocity recovered vector field; vectors are colored by the averaged vector field uncertainty; 6. Clonal fate potency score using Cospar[25]; 7. scVelo recovered latent time; 8. Pyro-Velocity inference of posterior mean shared time per cell. Cosine similarity and Spearman's correlation for the different methods with the clonal progression and fate potency are reported in the titles.

Supplementary Figures

Supplementary Figure 1 – Graphical representations detail the generative process of the Pyro-Velocity models and highlight their differences. a. Pyro-Velocity Model 1. Each rectangle represents one data dimension (e.g., gene or cell). The arrows represent the directed hierarchical dependency among random variables. The white circles correspond to the modeled random variables. The probability distributions of these variables are denoted on the right legend. The gray circles correspond to intermediate results computed from random variables. In this model, the kinetics random variables for the cell shared time and gene are modeled with two constraints: 1. the shared time is strictly larger than the gene-specific activation time $t_0^{(0)}$, 2. the gene expression in the activate state starts with the initial condition ($u_0 = 0, s_0 = 0$), as in previous methods[1,3] (**Methods**). **b.** Pyro-Velocity Model 2. This figure follows the same graphical convention as in a. In this model $t_0^{(0)}$ is independent from the shared time, thus allowing a time lag for the gene activation phase. In addition, a positive starting basal gene expression level (u_0, s_0) different than ($u_0 = 0, s_0 = 0$) is allowed. (Methods). **c.** Schematic illustrating how the constraints proposed in Model 1 (left) and Model 2 (right) are reflected in the modeling of individual gene expression**.** The x-axis corresponds to the shared time, the y-axis corresponds to the posterior mean of the spliced expression level, the vertical lines with arrows show the start and end time of the observed cell sampling, t0 represents the gene-specific activation time $t_0^{(0)}$, the red lines highlight the differences for the potential gene-specific time lag when $t_0^{(0)} > \min(t)$, and the initial spliced gene expression. The same differences can also be illustrated for the unspliced gene expression profiles (not shown).

Supplementary Figure 2 - Extended figure of Figure 2 for evaluating Pyro-Velocity (Model 1) and scVelo on the PBMC and pancreas datasets. a. Left figure shows the scVelo vector field using the default user options, right figure shows the evaluation of the scVelo vector field

using different combinations of user options within Epsilon cells. **b.** Pyro-Velocity additional single cell level uncertainty diagnostic plots on the fully mature PBMC dataset**,** from left to right: 1. standard deviation of posterior samples of shared time, 2. Rayleigh test of posterior samples of the vector field (the title reports false discovery rate after p-value correction for multiple tests across cells), 3. transcriptional state uncertainty of predicted raw read count per cell. The contour lines show the top 10% of the most uncertain cells. **c.** For the pancreas dataset, the figures show from left to right: UMAP rendering of 1. Cytotrace, 2. scVelo latent time, 3. Posterior mean of Pyro-Velocity shared time averaged across 30 posterior samples, 4. Top 50 markers overlapping between Pyro-Velocity and scVelo (**Methods**, **Supplementary Table 1-2**), 5. The same uncertainty diagnostic plots per cell as in **b. d.** The same plot as **Fig. 2d**. **e.** Selection of the top negatively correlated genes with the cell shared time based on the posterior mean on denoised spliced expression and negative mean absolute error of gene expression prediction (**Methods**). **f.** Phase portraits, rainbow plots, and UMAP rendering of posterior mean of gene expression for the genes selected in e.

Supplementary Figure 3 - Extended figure of Figure 2, Pyro-Velocity (Model 2) outperforms scVelo on PBMC and pancreas datasets. Same format as in **Fig. 2** and **Supplementary Fig. 2**.

Supplementary Figure 4 - Extended figure of Figure 3, Pyro-velocity (Model 1 for uni-fate data, Model 2 for bi-fate and multi-fate data) outperforms Cytotrace on scRNA-seq datasets with lineage barcoding information (LARRY dataset). a. SPRING embedding rendering of Cytotrace scores, the title reports the Spearman's correlation between the Cytotrace and Cospar fate potency scores, **b.** Pyro-Velocity additional single cell level uncertainty diagnostic plots based on the multi-fate LARRY dataset, from left to right: 1. Standard deviation of posterior samples of shared time per cell, 2. Rayleigh test of posterior samples of vector field per cell (the title reports the false discovery rate after p-value correction for multiple tests across cells), 3. Transcriptional

state uncertainty of predicted raw read count per cell. The contour lines show the top 10% of the most uncertain cells (**Methods**)**, c.** Pairwise Spearman's correlation between Cytotrace/Cospar fate potency scores and velocity methods for the four different benchmarking datasets selected based on the lineage barcoding information from LARRY data. **d.** SPRING embedding rendering of posterior mean of shared time across 30 posterior samples, the title shows the Spearman's correlation of mean shared time with Cytotrace. **e.** Selection of the top positively correlated genes with the cell shared time based on the posterior mean of spliced expression. **f.** Phase portraits, rainbow plots, and UMAP rendering with splicing gene expression level for the selected genes in e.

Supplementary Figure 5 - Extended figure of Figure 3 presenting an extended comparison of Pyro-Velocity and scVelo on scRNA-seq datasets with lineage barcoding information (LARRY dataset). a. uni-fate Monocyte lineage analysis with Pyro-Velocity Model 2, **b.** uni-fate Neutrophil lineage analysis with Pyro-Velocity Model 2, **c.** Neutrophil and Monocyte bi-fate lineage analysis with Pyro-Velocity Model 1, and **d.** multi-fate lineage analysis with Pyro-Velocity Model 1**.** The panel descriptions are the same as in Figure 3.

Supplementary Table 1: Pyro-Velocity Model 1 gene selection comparison with scVelo on the scRNA-seq pancreas dataset. Sheet 1: Top 50 positively correlated genes with shared time from Pyro-Velocity; Sheet 2: Top 50 positively correlated genes with latent time from scVelo; Sheet 3: Top 50 negatively correlated genes with shared time from Pyro-Velocity; Sheet 4: Top 50 negatively correlated genes with latent time from scVelo.

Supplementary Table 2: Pyro-Velocity Model 2 gene selection comparison with scVelo on the scRNA-seq pancreas dataset. Sheet 1: Top 50 positively correlated genes with shared time from Pyro-Velocity; Sheet 2: Top 50 positively correlated genes with latent time from scVelo; Sheet 3: Top 50 negatively correlated genes with shared time from Pyro-Velocity; Sheet 4: Top 50 negatively correlated genes with latent time from scVelo.

Figure 1

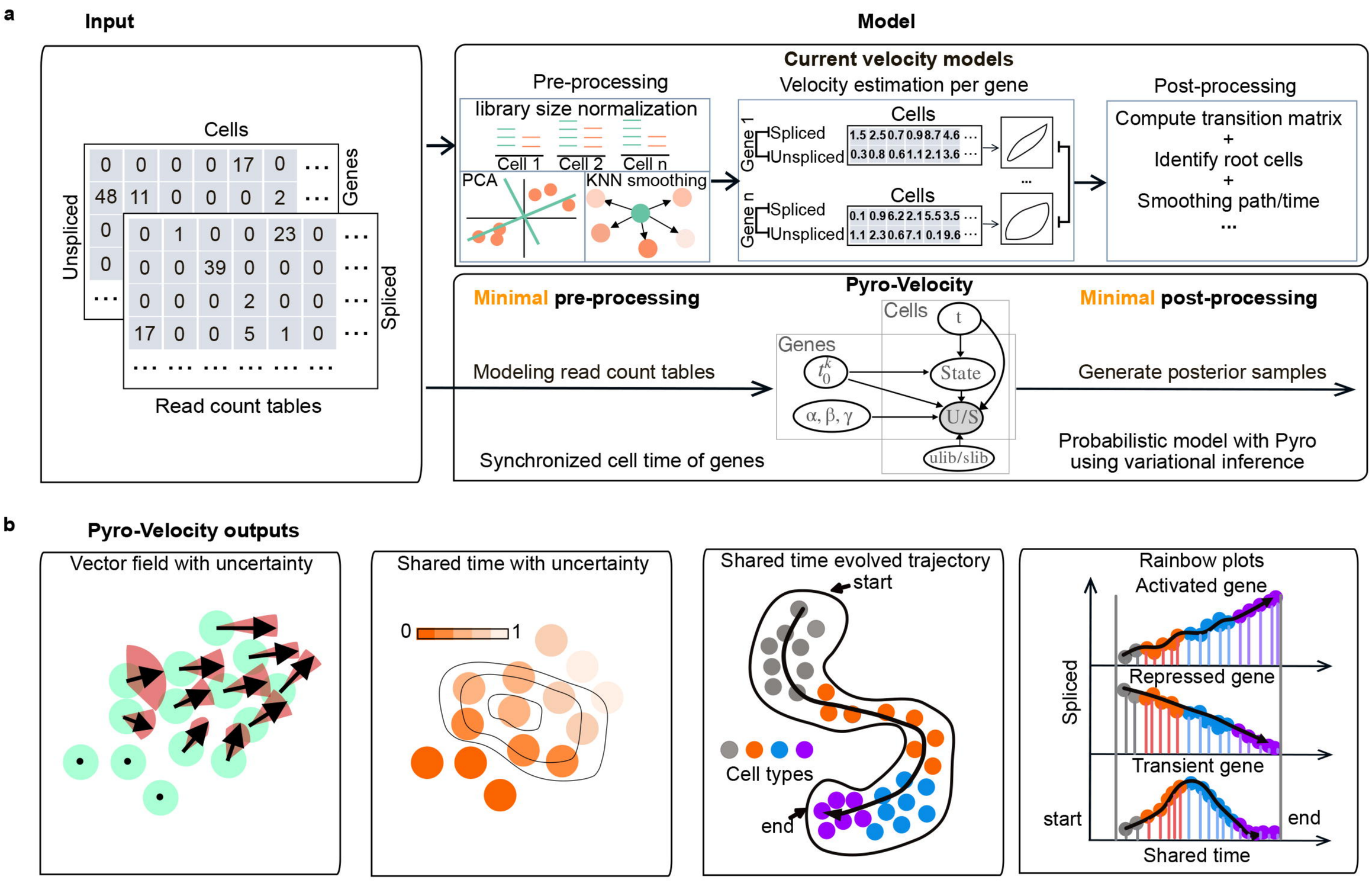

a
Input
Model
Cells
Genes
Unspliced
Spliced
Read count tables
Current velocity models
Pre-processing
library size normalization
Cell 1
Cell 2
Cell n
PCA
KNN smoothing
Velocity estimation per gene
Gene 1
Spliced
Unspliced
Gene n
Post-processing
Compute transition matrix
+
Identify root cells
+
Smoothing path/time
...
Minimal pre-processing
Pyro-Velocity
Minimal post-processing
Modeling read count tables
Genes
Cells
State
U/S
ulib/slib
Synchronized cell time of genes
Generate posterior samples
Probabilistic model with Pyro
using variational inference
b
Pyro-Velocity outputs
Vector field with uncertainty
Shared time with uncertainty
0
1
Shared time evolved trajectory
start
Cell types
end
Rainbow plots
Activated gene
Repressed gene
Transient gene
Spliced
start
end
Shared time

Figure 2

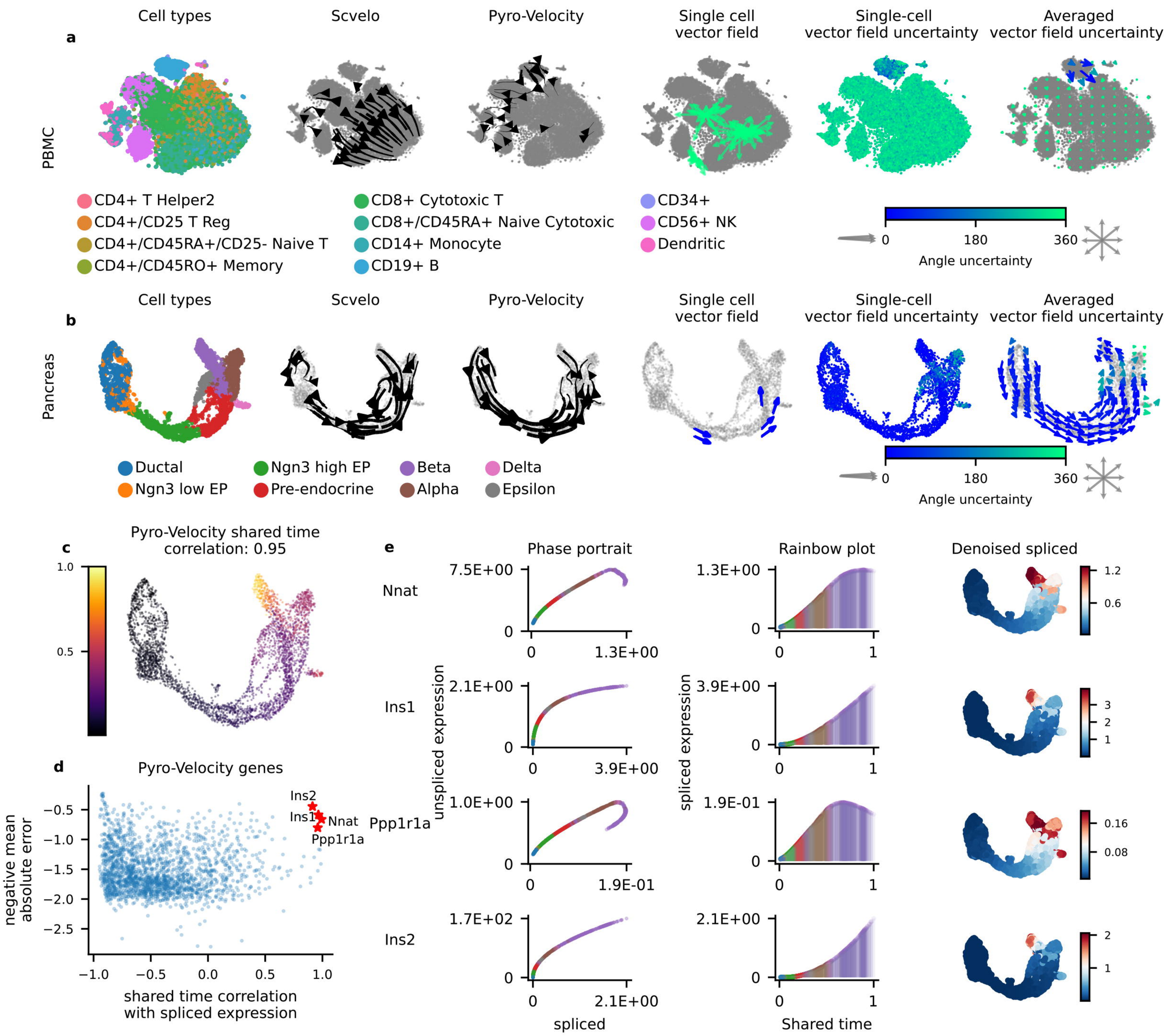
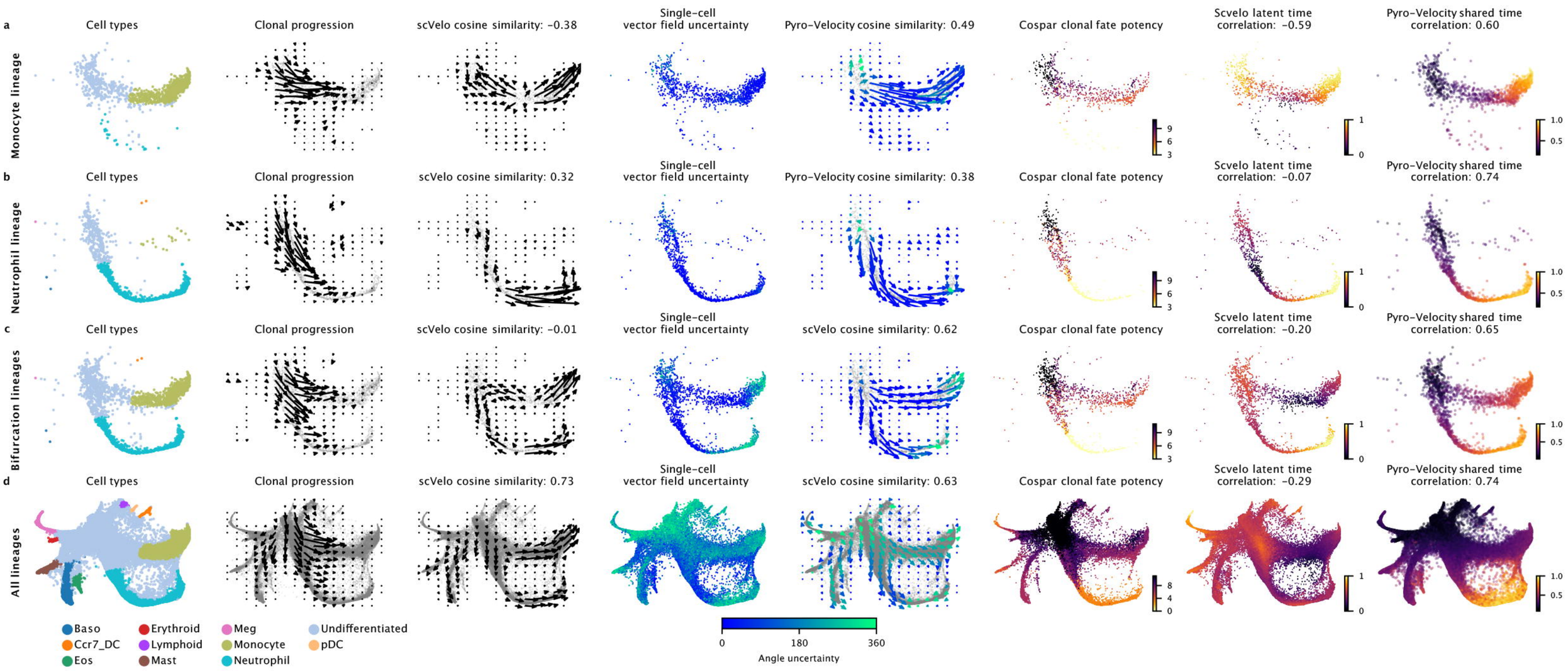
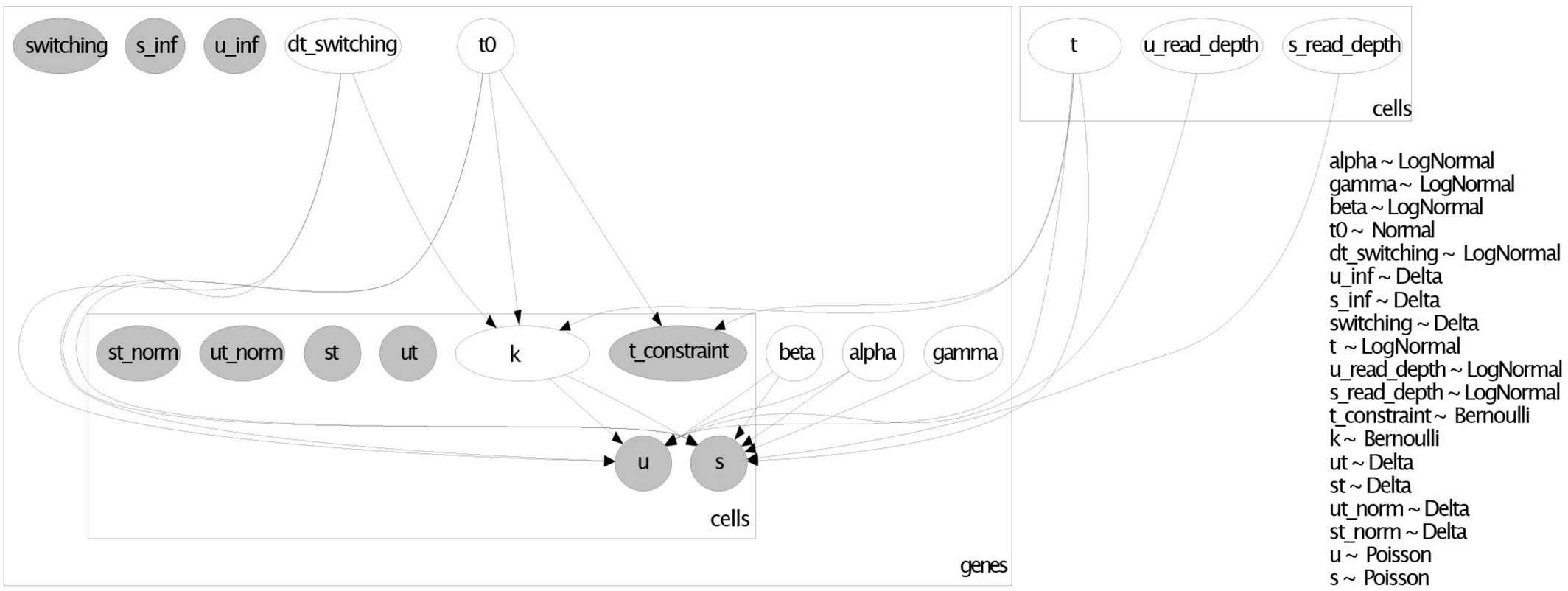

a
Cell types
Scvelo
Pyro-Velocity
Single cell vector field
Single-cell vector field uncertainty
Averaged vector field uncertainty
PBMC
CD4+ T Helper2
CD4+/CD25 T Reg
CD4+/CD45RA+/CD25- Naive T
CD4+/CD45RO+ Memory
CD8+ Cytotoxic T
CD8+/CD45RA+ Naive Cytotoxic
CD14+ Monocyte
CD19+ B
CD34+
CD56+ NK
Dendritic
0
180
360
Angle uncertainty
b
Pancreas
Ductal
Ngn3 low EP
Ngn3 high EP
Pre-endocrine
Beta
Alpha
Delta
Epsilon
c
Pyro-Velocity shared time correlation: 0.95
1.0
0.5
d
Pyro-Velocity genes
Ins2
Ins1
Nnat
Ppp1r1a
negative mean absolute error
shared time correlation with spliced expression
e
Phase portrait
Rainbow plot
Denoised spliced
Nnat
Ins1
Ppp1r1a
Ins2
unspliced expression
spliced
spliced expression
Shared time

Figure 3

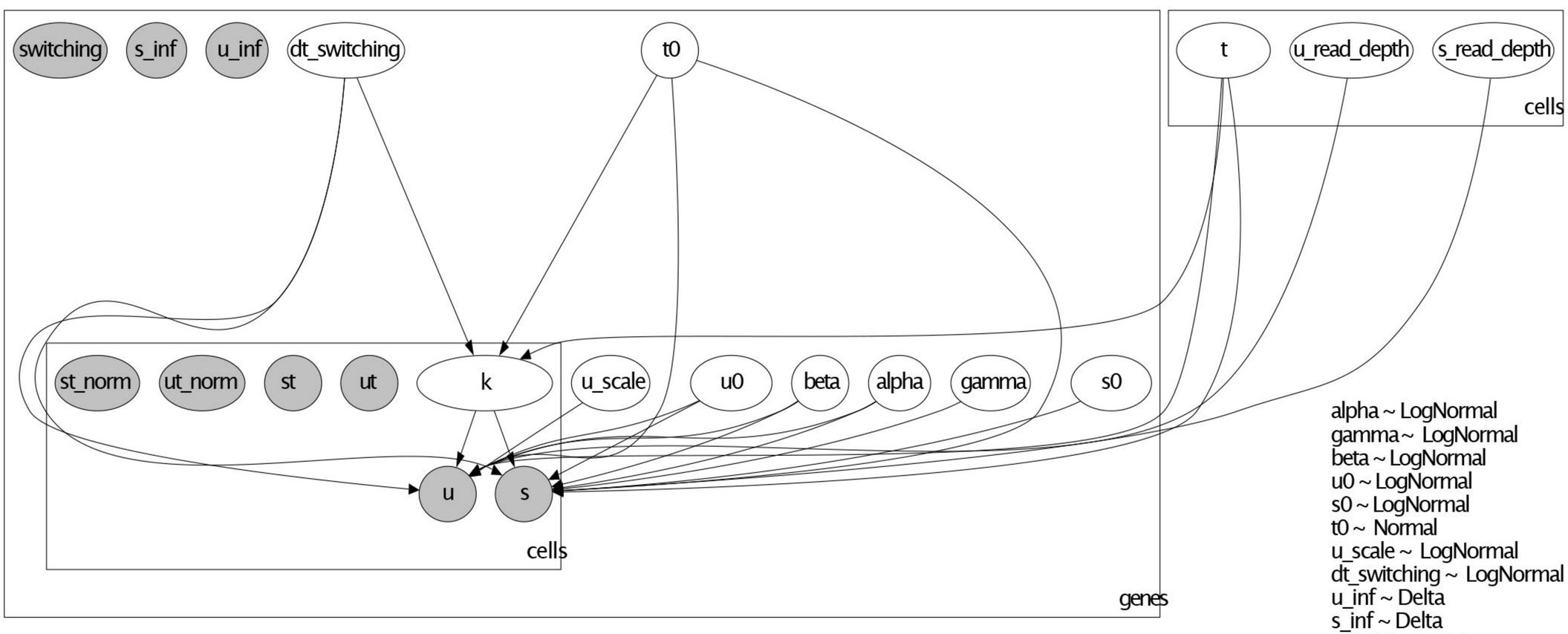

a
Monocyte lineage
Cell types
Clonal progression
scVelo cosine similarity: -0.38
Single-cell vector field uncertainty
Pyro-Velocity cosine similarity: 0.49
Cospar clonal fate potency
Scvelo latent time correlation: -0.59
Pyro-Velocity shared time correlation: 0.60
b
Neutrophil lineage
Cell types
Clonal progression
scVelo cosine similarity: 0.32
Single-cell vector field uncertainty
Pyro-Velocity cosine similarity: 0.38
Cospar clonal fate potency
Scvelo latent time correlation: -0.07
Pyro-Velocity shared time correlation: 0.74
c
Bifurcation lineages
Cell types
Clonal progression
scVelo cosine similarity: -0.01
Single-cell vector field uncertainty
scVelo cosine similarity: 0.62
Cospar clonal fate potency
Scvelo latent time correlation: -0.20
Pyro-Velocity shared time correlation: 0.65
d
All lineages
Cell types
Clonal progression
scVelo cosine similarity: 0.73
Single-cell vector field uncertainty
scVelo cosine similarity: 0.63
Cospar clonal fate potency
Scvelo latent time correlation: -0.29
Pyro-Velocity shared time correlation: 0.74
Baso
Ccr7_DC
Eos
Erythroid
Lymphoid
Mast
Meg
Monocyte
Neutrophil
Undifferentiated
pDC
0
180
360
Angle uncertainty

Supplementary Figure 1

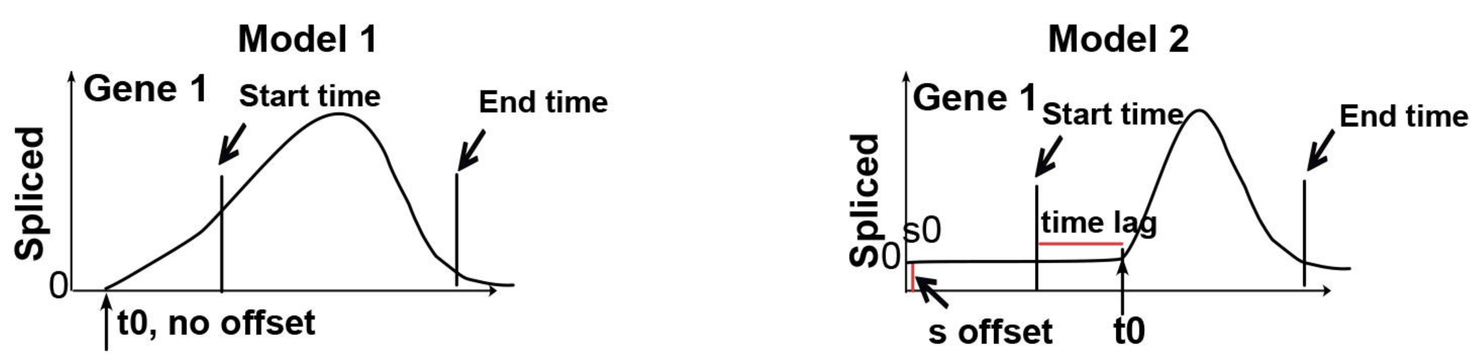
a **Model 1**



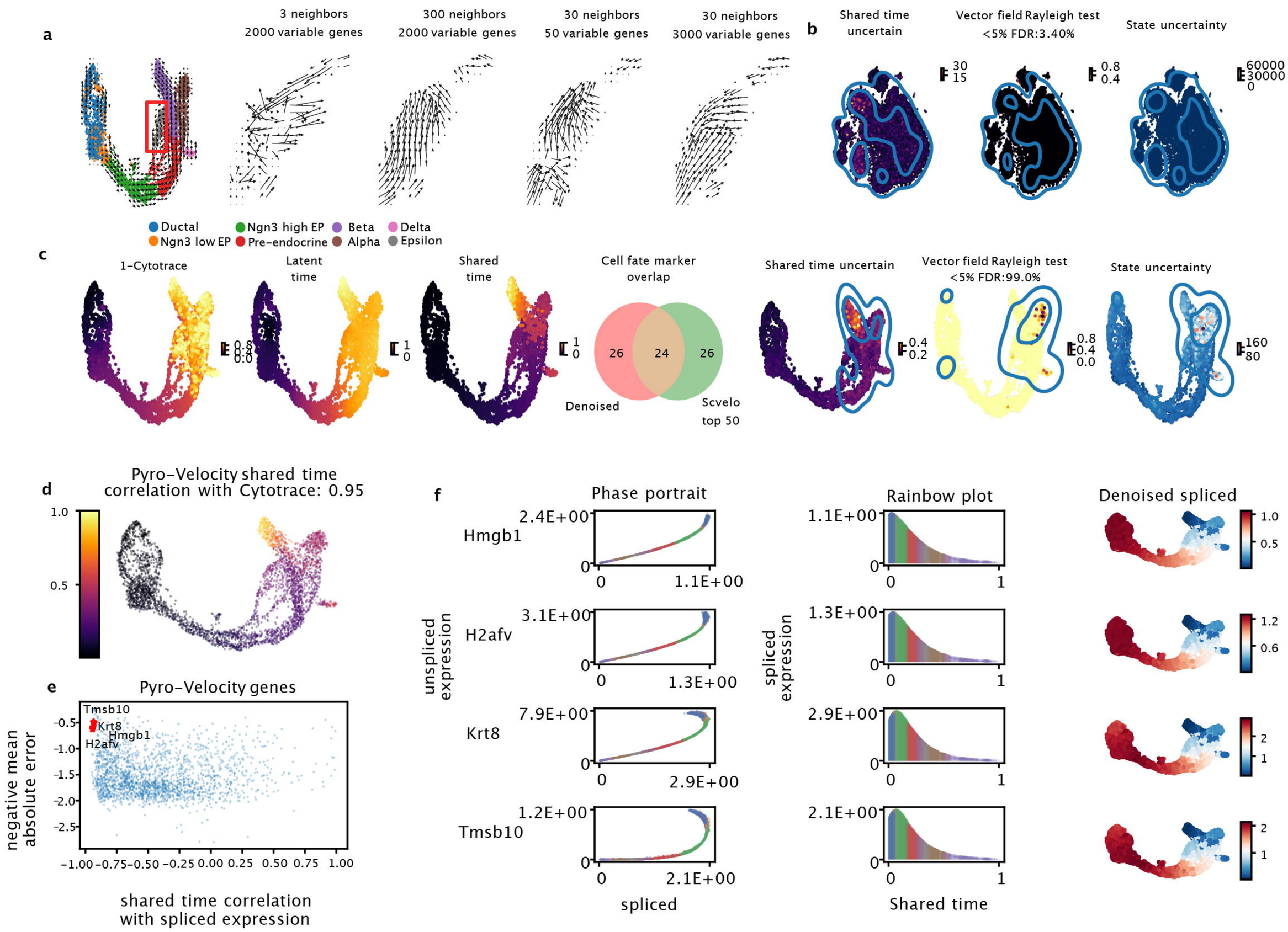
b **Model 2**



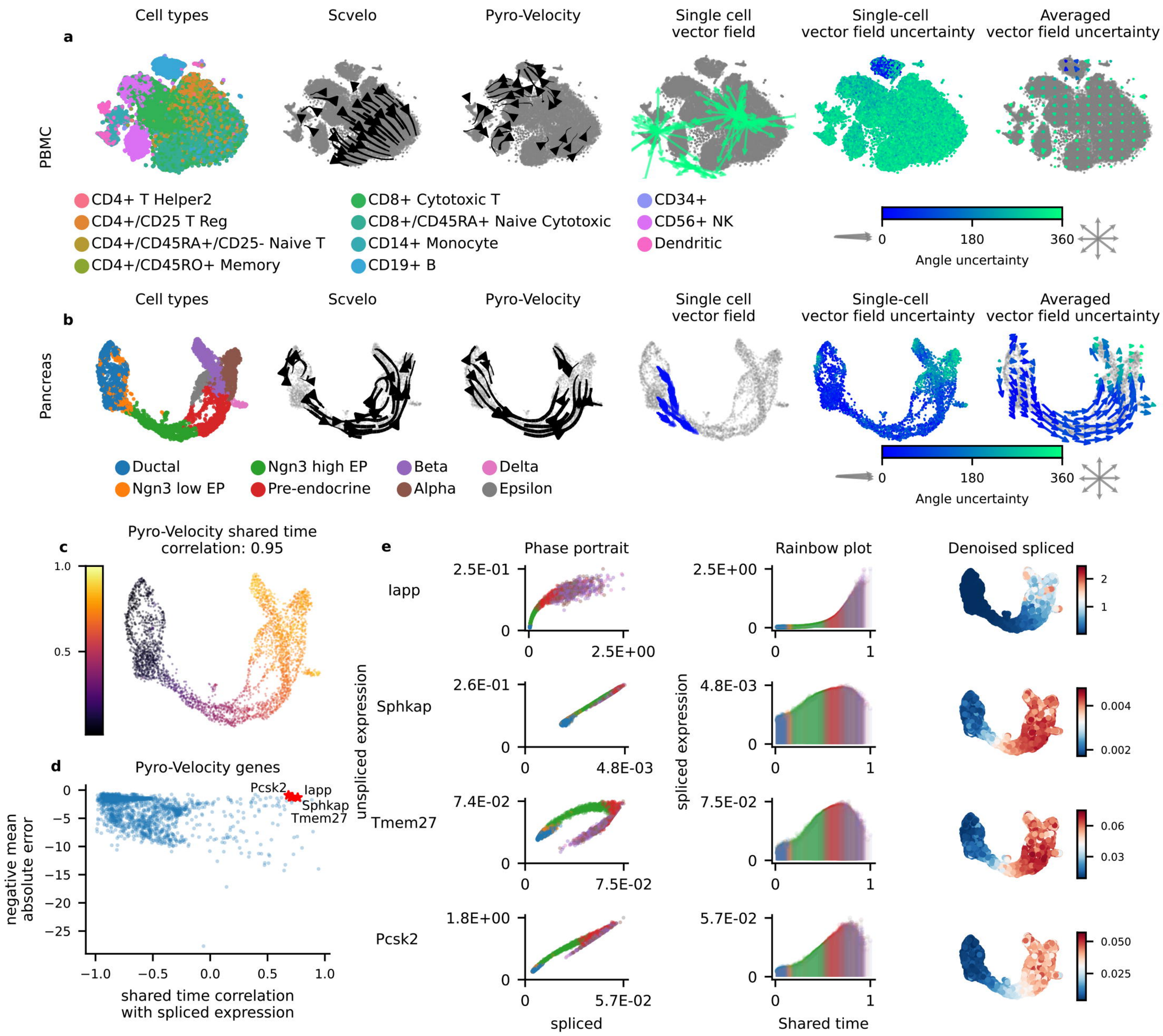
c



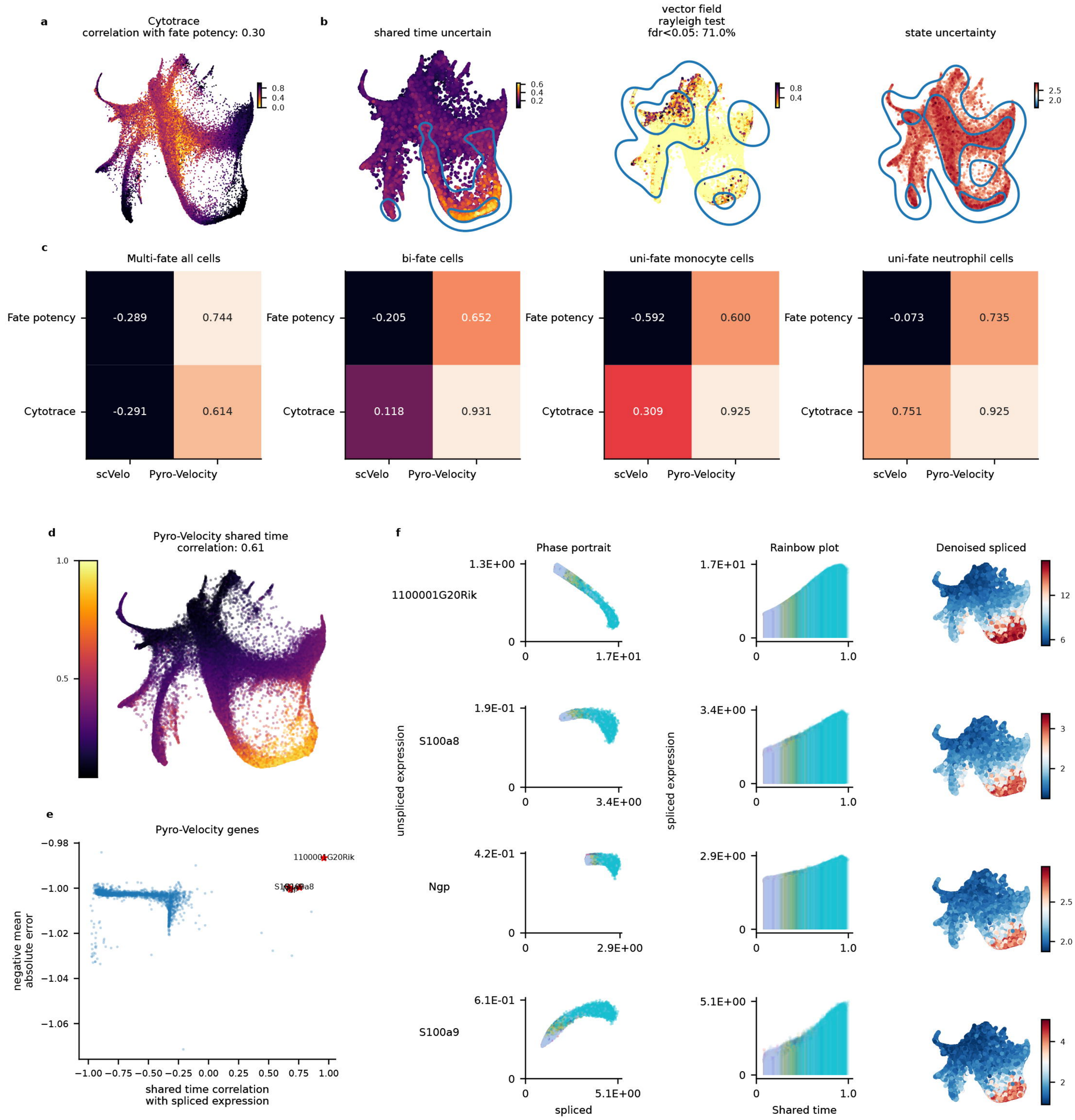
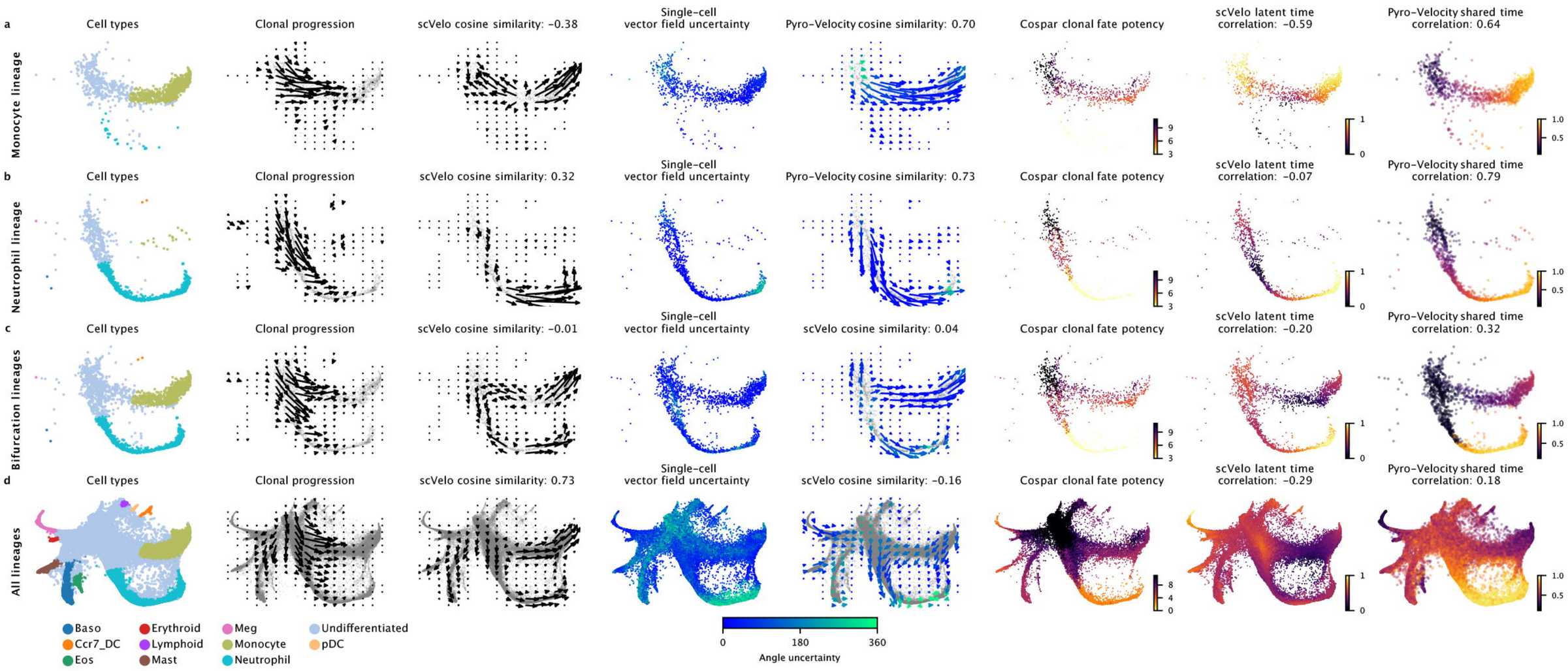
Supplementary Figure 2



Supplementary Figure 3



Supplementary Figure 4



Supplementary Figure 5