

RESEARCH ARTICLE

RNA velocity unraveled

Gennady Gorin¹, Meichen Fang¹, Tara Chari², Lior Pachter^{1,2,3*}

1 Division of Chemistry and Chemical Engineering, California Institute of Technology, Pasadena, California, United States of America, **2** Division of Biology and Biological Engineering, California Institute of Technology, Pasadena, California, United States of America, **3** Department of Computing and Mathematical Sciences, California Institute of Technology, Pasadena, California, United States of America

* lpachter@caltech.edu



OPEN ACCESS

Citation: Gorin G, Fang M, Chari T, Pachter L (2022) RNA velocity unraveled. PLoS Comput Biol 18(9): e1010492. <https://doi.org/10.1371/journal.pcbi.1010492>

Editor: Qing Nie, University of California Irvine, UNITED STATES

Received: March 17, 2022

Accepted: August 14, 2022

Published: September 12, 2022

Copyright: © 2022 Gorin et al. This is an open access article distributed under the terms of the [Creative Commons Attribution License](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Data Availability Statement: All analyses, including all figure generation, were performed using the Jupyter notebooks hosted at https://github.com/pachterlab/GFCP_2022 and deposited at Zenodo (DOI: [10.5281/zenodo.6360950](https://doi.org/10.5281/zenodo.6360950)). All simulated data were generated using custom, deterministic routines implemented in scripts and notebooks available at the GitHub repository. The human forebrain dataset used for numerical experiments was obtained from <http://pklab.med.harvard.edu/velocyto/hgForebrainGlut/hgForebrainGlut.loom> and rehosted at the GitHub repository. The ten datasets used for pre-processing comparisons were obtained from

Abstract

We perform a thorough analysis of RNA velocity methods, with a view towards understanding the suitability of the various assumptions underlying popular implementations. In addition to providing a self-contained exposition of the underlying mathematics, we undertake simulations and perform controlled experiments on biological datasets to assess workflow sensitivity to parameter choices and underlying biology. Finally, we argue for a more rigorous approach to RNA velocity, and present a framework for Markovian analysis that points to directions for improvement and mitigation of current problems.

Author summary

Single-cell sequencing data are snapshots of biological processes, making it challenging to infer dynamic relationships between cell types. RNA velocity attempts to bypass this challenge by treating the unspliced RNA content as a proxy for spliced RNA content in the near future, and using this “extrapolation” to build directional relationships. However, the method, as implemented in several software packages, is not yet reliable enough to be actionable, in part due to the large number of arbitrary, user-set hyperparameters, as well as fundamental incompatibilities between the biophysics of transcription in the living cell and the models used throughout the velocity workflows. In this study, we review these issues, and use existing results from the fields of stochastic modeling and fluorescence transcriptomics to develop an alternative theoretical framework. We show that our framework can facilitate the development and inference of physically consistent models for sequencing data, as well as the unification of single-cell analyses to self-consistently treat variation due to cell type dynamics and identities, the stochasticity inherent to single-molecule processes, and the uncertainty introduced by sequencing experiments.

Introduction

Background

The method of *RNA velocity* [1] aims to infer directed differentiation trajectories from snapshot single-cell transcriptomic data. Although we cannot observe the transcription rate, we can

<https://support.10xgenomics.com/single-cell-gene-expression/datasets> and the Sequence Read Archive (runs SRR14713295 for dmso and SRR14713295 for idu). The processed loom files generated by the three workflows are available at the CaltechData repository, at <https://data.caltech.edu/records/20030>. The scripts used to generate the loom files are available at the GitHub repository.

Funding: L.P. received the National Institutes of Health (nih.gov) award U19MH114830, administered by the National Institute of Mental Health (nimh.nih.gov). G.G., M.F., T.C., and L.P. were partially funded by this award. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing interests: The authors have declared that no competing interests exist.

count molecules of spliced and unspliced mRNA. The unspliced mRNA content is a leading indicator of spliced mRNA, meaning that it is a predictor of the spliced mRNA content in the cell's near future. This causal relationship can be usefully exploited to identify directions of differentiation pathways without prior information about cell type relationships: "depletion" of nascent RNA suggests the gene is downregulated, whereas "accumulation" suggests it is upregulated. This qualitative premise has profound implications for the analysis of single-cell RNA sequencing (scRNA-seq) data. The experimentally observed transcriptome is a snapshot of a biological process. By carefully combining snapshot data with a causal model, it is for the first time possible to reconstruct the dynamics and direction of this process without prior knowledge or dedicated experiments.

The bioinformatics field has recognized this potential, widely adopting the method and generating numerous variations on the theme. The roots of the theoretical approach date to 2011 [2], but the two most popular implementations for scRNA-seq were released in 2017–2018: *velocyto* by La Manno et al. [1], which introduced the method, and *scVelo* by Bergen et al. [3], which extended it to fit a more sophisticated dynamical model. Aside from these packages, numerous auxiliary methods have been developed, including *protaccel* [4] for incorporating newly available protein data, *MultiVelo* [5] and *Chromatin Velocity* [6] for incorporating chromatin accessibility, *VeTra* [7], *CellPath* [8], *Cytopath* [9], *CellRank* [10], and *Revelio* [11] for investigating coarse-grained global trends, *scRegulosity* [12] for identifying local trends, *Velo-Predictor* [13] for incorporating machine learning, *dyngen* [14] and *VeloSim* [15] for simulation, and *VeloViz* [16] and *evo-velocity* [17] for constructing velocity-inspired visualizations. This profusion of computational extensions has been accompanied by a much smaller volume of analytical work, including discussions of potential extensions and pitfalls [18–21], as well as theoretical studies based on optimal transport [22, 23] and stochastic differential equations [24]. However, at their core, these auxiliary methods are built on top of the theory and code base from *velocyto* or *scVelo*.

These two most popular software implementations emphasize usability and integration with standard visualization methods. The typical user-facing workflows, with internal logic abstracted away, are shown in Fig 1: a set of reads is converted to cell × gene matrices derived from spliced and unspliced mRNA molecule measurements, the matrices are processed to generate phase plots describing a dynamical transcription process, and finally the transcriptional dynamics are fit, extrapolated, and displayed in a low-dimensional embedding.

Despite the popularity of RNA velocity [13, 27] and increasingly sophisticated attempts to combine it with more traditional methods for trajectory inference [8, 10], there has been little comprehensive investigation of the modeling assumptions that underlie the seemingly simple workflow, with the sole dedicated critique to date largely focusing on the embedding process [28]. This is an impediment to applying, interpreting, and refining the methods, as problems arise even in the simplest cases. Consider, for example, the result displayed in Fig 1, where the outputs of the two most popular RNA velocity programs applied to human embryonic forebrain data generated by La Manno et al. [1] ("forebrain data") are qualitatively different. The inferred directions in the example should recapitulate a known differentiation trajectory from radial glia to mature neurons. However, *scVelo*, which "generalizes" *velocyto*, fails to identify, and even reverses the trajectory, suggesting totally different causal relationships between cell types. This type of problematic result has been reported elsewhere (Figs 2–3 of [3], Fig 2 of [21], Fig 4B of [5], Fig 5A of [9], Fig 5 of [10], and Fig 3 of [29]).

Motivated by such discrepancies, we wondered whether either *velocyto* or *scVelo* are reliable for standard use in applications where ground truth may be unknown. An examination of their theoretical foundations, and those of related methods, revealed that they are largely

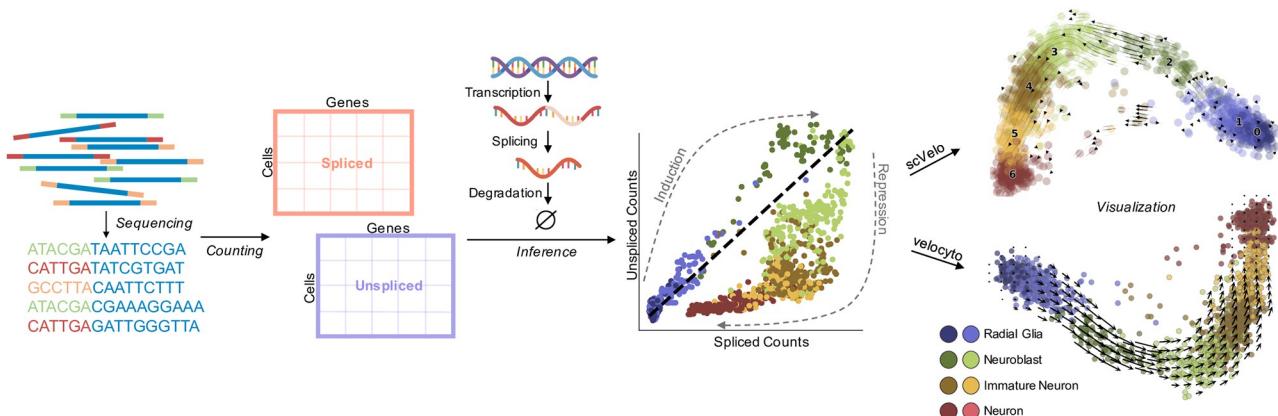


Fig 1. A summary of the user-facing workflow of a typical RNA velocity workflow. Initial processing of sequencing reads produces spliced and unspliced counts for every cell, across all genes. Inference procedures, implemented in *velocyto* and *scVelo*, fit a model of transcription, and predict cell-level velocities. The final embedding of cells and smoothed velocities is displayed in the top two principal component dimensions. Visualizations adapted from [25, 26]; dataset from [1]. The DNA and RNA illustrations are derived from the DNA Twemoji by Twitter, Inc., used under CC-BY 4.0.

<https://doi.org/10.1371/journal.pcbi.1010492.g001>

informal. Even the term “RNA velocity” is not precisely defined, and is used for the following distinct concepts:

- A generic method to infer trajectories and their direction using relative unspliced and spliced mRNA abundances by leveraging the causal relationship between the two RNA species, which is the interpretation in [24].
- A set of tools implementing this method or parts of it, as in an “RNA velocity workflow implemented in *kallisto|bustools*,” which is the interpretation in [30].
- A gene- and cell-specific quantity under a continuous model of transcription, as in “the RNA velocity of a cell is $\frac{ds}{dt} = \beta u - \gamma s$ ”, which is the interpretation in [18, 27].
- A gene- and cell-specific quantity under a probabilistic model of transcription, as in “the RNA velocity of a cell is $\frac{d\mathbb{E}[S_i]}{dt} = \beta u - \gamma s$ ”, which is the interpretation in [4].
- A gene-specific average quantity, as in “the total RNA velocity of a gene is $\sum_i (\beta u_i - \gamma s_i)$ ”, which is the interpretation in [12, 27].
- A cell-specific vector composed of gene-specific velocity components, as in “the vector RNA velocity of a cell is $\beta_j u_{ij} - \gamma_j s_{ij}$ ”, which is the interpretation in [7, 9, 27].
- The cell-specific linear or nonlinear embedding of a cell-specific vector in a low-dimensional space, which is the interpretation in [9].
- A local property, such as curvature, of a theorized cell landscape computed either from an embedding or a set of velocities, which is the interpretation in [22, 29].

These discrepancies and, more broadly, the limitations of current theory, stem from historical differences between sub-fields, which have calcified over the past twenty years of single-cell biology. On the one hand, fluorescence transcriptomics methods, including single-molecule fluorescence *in situ* hybridization and live-cell MS2 tagging, which target small, well-defined systems with a narrow set of probes [31–33], have motivated the development of interpretable stochastic models of biological variation [34, 35]. On the other hand, “sequence census” methods [36], such as scRNA-seq, provide genome-wide quantification of RNA, but the associated

challenges of exploratory, high-dimensional data analysis have not, for the most part, been addressed with mechanistic models. Instead, descriptive summaries, such as graph representations and low-dimensional embeddings, are the methods of choice [37]. Nevertheless, descriptive analyses, even if *ad hoc*, can still facilitate biological discovery: RNA velocity has been used to produce plausible trajectories [38–46], and our simulations show that it can recapitulate key information about differentiation trajectories in best-case scenarios (Fig A in S1 Text). These results highlight the potential of RNA velocity, and motivated us to review its assumptions, understand its current failure modes, and to solidify its foundations.

Towards this end, we found it helpful to contrast the sub-fields of fluorescent transcriptomics and sequencing, which have analogous goals, albeit disparate origins that have led to analytical methods with distinct philosophies and mathematical foundations. The sub-fields have, at times, interacted. Fluorescence transcriptomics can now quantify thousands of genes at a time, and this scale of data is now occasionally presented using visual summaries popular for RNA sequencing data, such as principal component analysis (PCA) [47], Uniform Manifold Approximation and Projection (UMAP) [48], and *t*-distributed stochastic neighbor embedding (*t*-SNE) [49, 50]. Conversely, the commercial introduction of scRNA-seq protocols with unique molecular identifiers (UMIs) has spurred the adoption of theoretical results from fluorescence transcriptomics for sequence census analysis [51–55]. Sequencing studies frequently use count distribution models that arise from stochastic processes, such as the negative binomial distribution, albeit without explicit derivations or claims about the data-generating mechanism [51, 56, 57]. These connections highlight the promise of mechanistic gene expression models: in principle, parameters can be fit to sequencing data to produce a physically interpretable, genome-scale model of transcriptional regulation in living cells, and some steps have been taken in this direction over the past decade [52–55, 58, 59].

RNA velocity methods are products of the sequence census paradigm: they draw heavily on low-dimensional embeddings and graphs derived from the raw data. Their current limitations stem from viewing biology through the lens of signal processing, where noise is something to be eliminated or smoothed out. We posit that it is more appropriate to view the data through the lens of quantitative fluorescence transcriptomics, in which noise is a biophysical phenomenon in its own right. Through this lens, modeling that decomposes variation into single-molecule (intrinsic) and cell-to-cell (extrinsic) [60] components, in addition to technical noise [61], is key. Beyond this conceptual issue, we find that an assessment of the impact of hyper-parameterized, heuristic data pre-processing and visualization in current RNA velocity workflows is useful for developing more reliable analyses.

Goals and findings

To fully describe what RNA velocity does, why it may fail, and how it can be improved, requires work on several fronts:

In the section “Workflow and implementations,” we describe an idealized “standard” RNA velocity workflow. We introduce the biophysical foundations presented in the original publication, outline the methodological choices implemented in the software packages, and enumerate the tunable hyperparameters left to the user.

In the section “Logic and methodology,” we probe the logic of the assumptions made in the workflow and describe potential failure points. This analysis revisits the outline through complementary critical lenses, adapted to the mechanistic and phenomenological steps. To characterize its biological coherence, we compare the concrete and implicit biophysical models to those standard in the field of fluorescence transcriptomics, and discuss the implications of assumptions that do not appear to be backed by a biophysical or mathematical argument.

To characterize its stability, we test the quantitative effects of tuning hyperparameters and using different software implementations on real datasets.

Our findings on RNA velocity have implications for other scRNA-seq analyses. On one hand, the theory behind RNA velocity is not sufficiently robust. The models disagree with known biophysics: they do not recapitulate bursty production [62], and place needlessly restrictive constraints on regulatory trends. They are also internally inconsistent, as they do not preserve cell identities: genes are fit independently, so the same cells' placement along putative trajectories differs between genes. Furthermore, the embedding processes are *ad hoc* and heavily rely on error cancellation, apparently discarding much of the data in the process. These problems are intrinsic, and derived methods inherit them.

Fortunately, better options, inspired by fluorescence transcriptomics models, are available. In order to develop a meaningful foundation for RNA velocity, we formalize its stochastic model and describe an inferential procedure that can be internally coherent and consistent with transcriptional biophysics. Furthermore, by examining the assumptions underpinning RNA velocity and reframing them in terms of stochastic dynamics, we find that the *velocyto* and *scVelo* procedures naturally emerge as approximations to our solutions. Our approach, presented in the section “Prospects and solutions,” provides an alternative to current trajectory inference methods: instead of using physically uninterpretable adjacency metrics and fitting a narrow set of topologies, it is relatively straightforward to solve many combinations of transient or stationary topologies and apply standard Bayesian methods to identify the best fit. Conceptually, instead of “denoising” data, our approach proposes fitting the molecule distributions and preserving the uncertainty inherent in noisy biological and experimental processes.

Workflow and implementations

We begin with a conceptual overview of an idealized RNA velocity workflow, with a description of implementation-specific choices. We focus on datasets with cell barcodes and UMIs, such as those generated by the 10x Genomics Chromium platform [63], as they provide the most natural comparison to discrete stochastic models later in the discussion (“Occupation measures provide a theoretical framework for scRNA-seq” under “Prospects and solutions”). We summarize the workflow in Fig 2, giving particular attention to the parameter choices required at each step. To clarify the information transfer in the process, we report the manipulations performed and the variables defined in a single run of the processing workflow in Fig B in S1 Text (as used to generate Fig 4 of [1]).

Pre-processing

RNA velocity analysis begins by processing raw sequencing data to distinguish spliced and unspliced molecules. This is a genomic alignment problem. For example, reads aligning to intronic references are assigned to unspliced molecules, whereas reads spanning exon-exon splice junctions are assigned to spliced molecules. Data from reads associated with a single UMI are combined to generate a label of “spliced,” “unspliced,” or “ambiguous” for each read. “Ambiguous” reads are omitted from downstream analysis, so the assignments are effectively binary.

Until recently, traditional alignment and UMI counting software, such as *Cell Ranger* from 10x Genomics, discarded intronic information [63]. The same was true of pseudoalignment methods, as they identify transcript classes consisting of annotated, and presumably terminal, isoforms [64]. The explicit quantification of transient intron-containing molecules appears to have been introduced in the *velocyto* command-line interface [1]. Since then, existing workflows have added functionality for unspliced transcript quantification [27]. In particular,

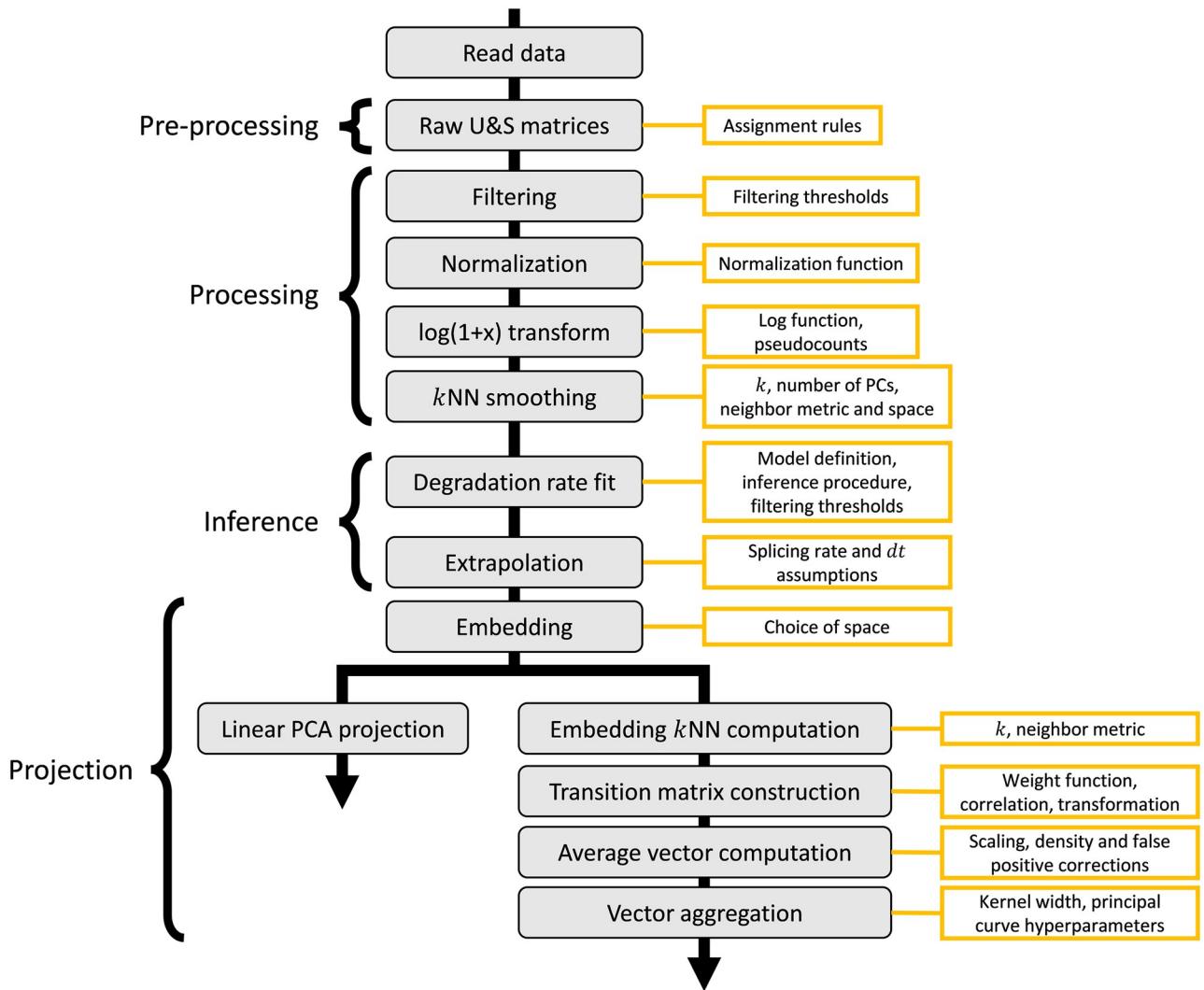


Fig 2. An RNA velocity workflow, beginning with read processing and ending with two-dimensional projection, and the parameters that must be specified by the user.

<https://doi.org/10.1371/journal.pcbi.1010492.g002>

alignment can be performed via *STARsolo* [65] and *dropEst* [66], whereas pseudoalignment can be performed via *kallisto|bustools* [30] or *salmon* [27]. Benchmarking has shown discrepancies between the outputs of these workflows [27, 30], apparently due to differences in filtering, thresholding, and counting ambiguous reads. However, there is currently little principled reason to prefer one program's results to another, as quantification rules largely follow *velocyto*, and assume a two-species model is sufficient.

Count processing

The raw count data are processed to smooth out noise contributions that can skew the downstream analysis. This step is generally combined with the standard quality control techniques for scRNA-seq [37]. First, cells with extremely low expression are filtered out. Then, a subset of several thousand genes with the highest expression and variation are selected. The counts are normalized by the number of cell UMIs to counteract technical and cell size effects. At this

point, the PCA projection is computed from log-transformed spliced RNA counts. Finally, the normalized counts are smoothed out by nearest-neighbor pooling. To accomplish this, the algorithm computes the k nearest cell neighbors in a PCA space for each cell, then replaces the abundance with the neighbors' average. This step is crucial, as it produces the cyclic or near-cyclic “phase portraits” used in the inference procedure.

The implementation specifics vary even between the two most popular packages, the Python versions of *velocyto* and *scVelo*. For example, there appears to be no consensus on the appropriate k or neighborhood definition for imputation. The original publication reports k between 5 and 550, calculated using Euclidean distance in 5–19 top PC dimensions [1]. By default, *scVelo* uses $k = 30$ in the top 30 PC dimensions [3].

Inference

The normalized and smoothed count matrices are fit to a biophysical model of transcription. The model structure for a single gene is outlined in Fig 3a. $\alpha(t)$ is a transcription rate, which has pulse-like behavior over the course of the trajectory. The constant parameters are β , the splicing rate, and γ , the degradation rate. Driving by $\alpha(t)$ induces continuous trajectories $\mu_u(t)$ and $\mu_s(t)$, which informally represent instantaneous averages, μ , of the unspliced, u , and spliced, s , species, governed by the following ordinary differential equations (ODEs):

$$\begin{aligned} \frac{d\mu_u(t)}{dt} &= \alpha(t) - \beta\mu_u(t), \\ \frac{d\mu_s(t)}{dt} &= \beta\mu_u(t) - \gamma\mu_s(t). \end{aligned} \quad (1)$$

The qualitative behaviors of these functions are shown in Fig 3b. By fitting smoothed count data for a single gene, now interpreted as samples from a dynamical phase portrait governed by Eq 1 (Fig 3c), it is possible to estimate the ratio γ/β . Finally, with this ratio in hand, the

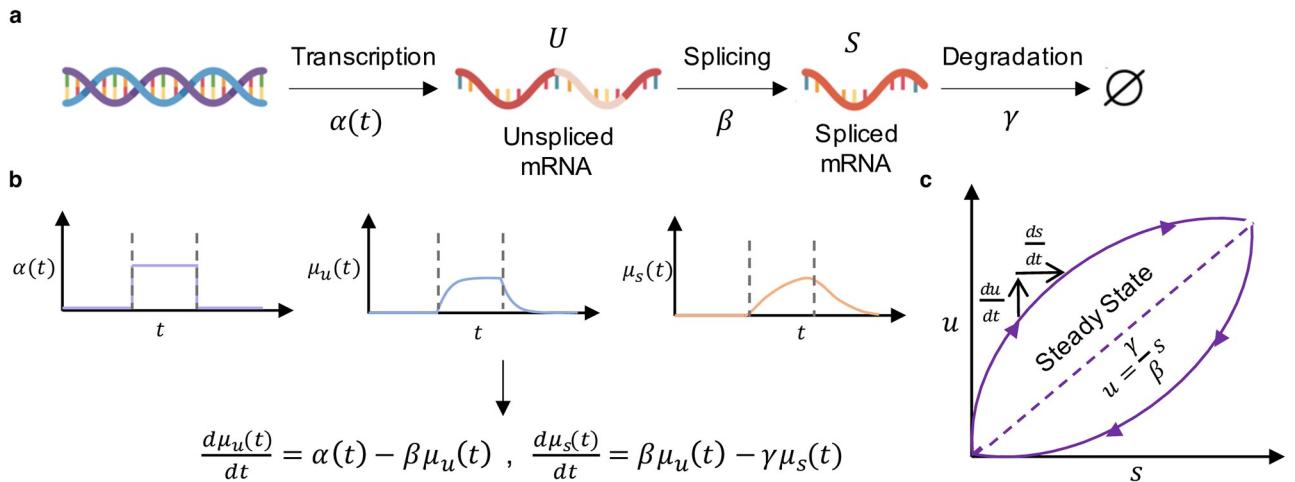


Fig 3. **a.** The continuous model of transcription, splicing, and degradation used for RNA velocity analysis. **b.** Plots of $\alpha(t)$, $\mu_u(t)$, and $\mu_s(t)$ over time t and the corresponding governing equations for the system. Dashed lines indicate time of switching event. **c.** Outline of the common phase portrait representation, with both steady state and dynamical models denoted. Adapted from [1]. The DNA and RNA illustrations are derived from the DNA Twemoji by Twitter, Inc., used under CC-BY 4.0.

<https://doi.org/10.1371/journal.pcbi.1010492.g003>

velocity v_i may be computed for each cell i :

$$v_i := \frac{\Delta s_i}{\Delta t} = (\beta u_i - \gamma s_i) \propto u_i - \frac{\gamma}{\beta} s_i, \quad (2)$$

where s_i and u_i are cell-specific counts, Δt is an arbitrary small time increment, and Δs_i is the change in spliced counts achieved over that increment.

The popular packages differ on the appropriate way to fit the rate parameters. The *velocyto* procedure presupposes that the system reaches equilibria at the low- and high-expression states of $\alpha(t)$, and approximates them by the extreme quantiles of the phase plots. By computing the slope of a linear fit to these quantiles, it obtains the parameter γ/β (Fig 3c). On the other hand, *scVelo* relaxes the assumption of equilibrium and implements a “dynamical” model, which fits the solution of Eq 1 to the entire phase portrait to obtain γ and β separately. This methodological difference corresponds to conceptual differences in the interpretation of imputed data. In *velocyto*, imputation appears to be an *ad hoc* procedure for filtering technical effects, in line with the usual usage [67, 68]. On the other hand, in *scVelo*, the imputed data are called “moments” and treated as identical to the instantaneous averages $\mu_u(t)$ and $\mu_s(t)$ of the process. In addition, *scVelo* offers a “stochastic” model, which posits pooled second moments are equivalent to the instantaneous second moments (e.g., the sum of s^2 over neighbors is equal to $\sigma_s^2(t) + \mu_s^2(t)$).

The genes are analyzed independently, generating a velocity v_{ij} for each cell i and gene j . As the *velocyto* procedure cannot separately fit β_j and γ_j , its velocities have different units for different genes. On the other hand, the *scVelo* procedure does separately fit the rate parameters, albeit by assigning a latent time t_{ij} to each cell, distinct for each gene’s fit.

Embedding

Low-dimensional representations are generated using one of the conventional algorithms, such as PCA, *t*-SNE, or UMAP. These algorithms can be conceptualized as functions that map from a high-dimensional vector s_i to a low-dimensional vector $E(s_i)$. The original publication offers two methods to convert cell’s velocity vector v_i to a low-dimensional representation [1].

If the embedding is deterministic (e.g., E is PCA on log-transformed counts), one can define a source point $E(s_i)$, compute a destination point $E(s_i + v_i \Delta t) = E(s_i + \Delta s_i)$, and take the difference of these two low-dimensional vectors to obtain a local vector displacement:

$$V_i = \Delta E(s_i) = E(s_i + \Delta s_i) - E(s_i). \quad (3)$$

This displacement is then interpreted as a scalar multiple of the cell-specific embedded velocity.

If the embedding is non-deterministic, one can apply an *ad hoc* nonlinear procedure. This procedure essentially computes an expected embedded vector by weighting the directions to k embedding neighbors; neighbors that align with Δs_i are considered likely destinations for cell state transitions in the near future:

$$V_i = \sum_{q=1}^k (E(s_q) - E(s_i)) w(s_q - s_i, \Delta s_i), \quad (4)$$

where w is a composition of the softmax operator (with a tunable kernel width parameter) with a measure of concordance between the arguments. Once an average direction is computed, it undergoes a set of corrections, e.g., to remove bias toward high-density regions in the

embedded space. Finally, the cell-specific embedded vectors are aggregated to find the average direction over a region of the low-dimensional projection.

The packages almost exclusively use the nonlinear embedding procedure. There is no consensus on the appropriate choice of embedding, number of neighbors, or measure of concordance. PCA, t-SNE, and UMAP have been used to generate low-dimensional visualizations [1, 3]. The original publication uses k between 5 and 300 and applies square-root or logarithmic transformations prior to computing the Pearson correlation between the velocity and neighbor directions [1]. In contrast, *scVelo* uses a recursive neighbor search by averaging over neighbors and neighbors of neighbors (with $k = 30$), and implements several variants of cosine similarity [3]. An optional step adjusts the embedded velocities by subtracting a randomized control; this correction is usually omitted in demonstrations of *velocyto* and implemented but apparently undocumented in *scVelo*.

As demonstrated in Fig 2, the linear PCA embedding is the simplest dimensionality reduction technique; it consists of a projection and requires fewer parameter choices than other methods. However, it is only consistently used in *Revelio* [11]. The *velocyto* package does not appear to have a native implementation of this procedure, although it is briefly demonstrated in the original article (Fig 2d and S2 Figs 8–9 of [1]). On the other hand, *scVelo* does implement the PCA velocity projection, but disclaims the results of using it as unrepresentative of the high-dimensional dynamics.

Logic and methodology

To understand the implications of the choices implemented in various RNA velocity workflows, we examined the procedures from a biophysics perspective, with a view towards understanding the mechanistic and statistical meaning of methods implemented. In this section, we broadly discuss potential challenges, problematic assumptions, and contradictory results. In the following section, we draw on lessons learned and propose a modeling approach of our own.

Pre-processing

As outlined in “Pre-processing” under “Workflow and implementations,” several workflows are available for converting raw reads to molecule counts. These workflows largely follow the logic set out in the original implementation [1]; however, as pointed out by Soneson et al. [27], they produce different outputs from the same data. We reproduced their analysis on a broader selection of datasets (as reported in “Data availability”) in Fig C in S1 Text, according to the procedure outlined under “Pre-processing concordance.” The performance was broadly consistent with the previous benchmarking and the description under “Pre-processing” in the previous section: the methods agreed on the definition of “spliced” molecules, but different rules for the assignment of “unspliced” molecules led to discrepancies in counts. These discrepancies were particularly pronounced when comparing datasets one gene at a time, likely due to noise in tens of thousands of low-expressed genes (ρ by cell in Fig C in S1 Text; cf. lower triangle of Fig S10 in [27]).

However, a simple comparison between the software outputs obscures a far more fundamental challenge: the binary classification of transcripts as either spliced or unspliced is necessarily incomplete. The average human transcript has 4–7 introns [69], and a combinatorial number of potential transient and terminal isoforms. The vast majority of genes are alternatively spliced [70–72].

We can consider the hypothetical example of a nascent transcript with the structure $E_1I_1E_2I_2E_3$, where I_i are introns and E_i are exons, as shown in Fig 4. If we place all UMIs with

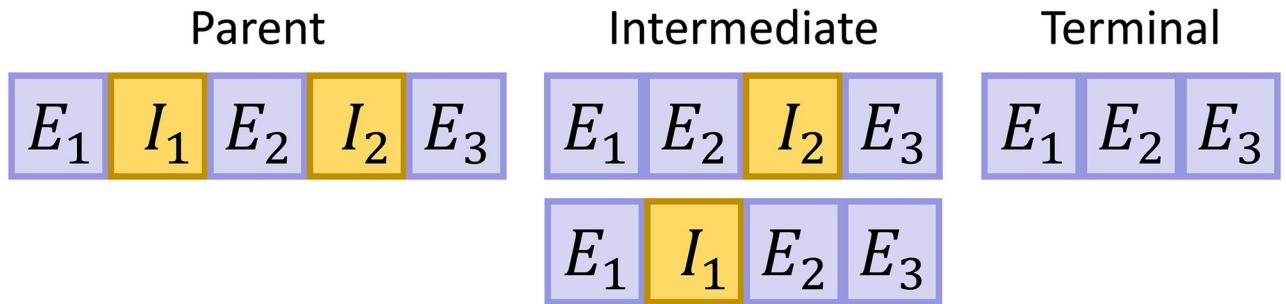


Fig 4. A two-intron mRNA species may not have well-defined “unspliced” and “spliced” forms.

<https://doi.org/10.1371/journal.pcbi.1010492.g004>

intronic reads into the “unspliced” category, we conflate the parent and intermediate transcripts. On the other hand, if we place all UMIs with splice junctions into the “spliced” category, we conflate the intermediate and terminal transcripts. Adding more complexity, some isoforms may retain introns through alternative splicing mechanisms; for example, the intermediate transcripts may be exported, translated, and degraded alongside the terminal.

The binary model is not large enough to include the diversity of possible splicing dynamics, but approximately holds under fairly restrictive conditions: the predominance of a single terminal isoform, as well as the existence of a single rate-limiting step in the splicing process. Previous work reports that minor isoforms are non-negligible [70, 72], differential isoform expression is physiologically significant [72–74], and intron retention in particular is implicated in regulation and pathology [75–78]. Splicing rate data are more challenging to obtain, but targeted experiments [79], genome-wide imaging [80], and our preliminary mechanistic investigations [81] suggest that selection and removal of individual introns is stochastic, but the overall splicing process has rather complex kinetics, not reducible to a single step.

RNA velocity biophysics

We will first inspect the complexity obscured by the simple schema given in Fig 3a. The velocity manuscripts use several distinct models for the transcription rate $\alpha(t)$. Furthermore, the amounts of molecular species \mathcal{U} and \mathcal{S} (previously denoted informally by u and s) have incompatible interpretations. The following models make fundamentally different claims about the data-generating process and imply fundamentally different inference procedures.

1. $\alpha(t)$ is piecewise constant over a finite time horizon; u and s are discrete (SN2 pp. 2–3, Fig 1a-b of [1]).
2. $\alpha(t)$ is continuous and periodic; u and s are discrete (SN2 pp. 2–3, Fig 1e of [1]).
3. α , β , and γ all smoothly vary over a finite time horizon according to an undisclosed function, with α exhibiting pulse behavior; u and s are discrete (SN2 Fig 5 of [1]).
4. α , β , and γ all smoothly vary over a finite time horizon according to an arbitrary function; u and s are continuous (Fig 3 of [21]).
5. $\alpha(t)$ is piecewise constant over a finite time horizon; u and s are continuous (Fig 1 of [1], Methods of [3]).
6. $\alpha(t)$ is piecewise constant over a finite time horizon; u and s are continuous-valued but may contain discontinuities (Methods of [3]).

7. α is constant; u and s are continuous (Fig 1b and SN1 pp. 1–2 of [1]). This formulation yields the reaction rate equation, and cannot produce the bimodal phase plots of interest.
8. α is constant; u and s are discrete (SN1 pp. 2–3 of [1]). This is the stochastic extension [82] of the previous model, and cannot produce the bimodal phase plots of interest, as explicitly shown on page 3 of SN1 in [1].

These discrepancies make a comprehensive analysis challenging. Models 7–8 do not contain differentiation dynamics. Certain models are contrived; models 3–4 propose transcription rate variation without motivating the specific form, and model 6 introduces nonphysical discontinuities. Model 2 alludes to limit cycles in stochastic systems under periodic driving, an intriguing phenomenon in its own right [83, 84], but not otherwise explored in the *scVelo* and *velocyto* publications. For the rest of this report, we focus on the discrete formulation (model 1) and its continuous analog (model 5).

For the discrete formulation, the RNA velocity v should be interpreted as the time derivative of the expectation of a random variable S_t that tracks the number of spliced RNA, conditional on the current state (Section A in S1 Text):

$$v = \partial_t \mathbb{E}[S_t | S_0 = s, U_0 = u] |_{t=0} = \beta u - \gamma s. \quad (5)$$

For the continuous formulation, it should be interpreted as the time derivative of the deterministic variable s_t that tracks the amount of spliced RNA, initialized at the current state:

$$\begin{aligned} v &= \partial_t s_t |_{t=0} \text{ [such that } s_0 = s, u_0 = u] \\ &= \beta u - \gamma s. \end{aligned} \quad (6)$$

These formulations happen to be mathematically identical, which creates ambiguity. Nevertheless, both are legitimate, if narrow, statements about the near future of a process initialized at a state with u unspliced and s spliced molecules. The questions that arise immediately before, and immediately after, the velocity computation procedure, are (1) what generative model should be fit to obtain β and γ and (2) even with a v , how much use can one make of it?

Model definition

Continuous, deterministic models are fundamentally inappropriate in the low-copy number regime, which is predominant across the transcriptome [85–87]. Although continuous equations such as Eq 1 can represent the evolution of moments, they are insufficient for inference, as fitting average mRNA abundance amounts to invoking the central limit theorem for a very small, strictly positive quantity [88–92]. A comprehensive understanding of the stochastic noise model is necessary prior to making such approximations. Therefore, simulation methods that use a continuous model are immediately suspect [15]. We describe implementation-specific concerns in the subsections “Count processing” and “Inference” under “Logic and methodology.”

The motivation behind a pulse model of transcriptional regulation is obscure. Although dynamic processes certainly have transiently expressed genes [93–95, 95, 96], it is far from clear that this model applies across the transcriptome, to thousands of potentially irrelevant genes. Indeed, it is not even coherent with genes showcased in the original report (Fig 4d and Extended Data Fig 8b of [1]): only ELAVL4 appears to show a symmetric pulse of expression. Finally, even when this model does apply, the assumption of constant splicing and degradation rates across the entire lineage is a potentially severe source of error, with no simple way to diagnose it [21].

Most problematic is that even the discrete model is incoherent with known mammalian transcriptional dynamics. If we suppose induction and repression periods are relatively long,

as for a stationary, terminal, or unregulated cell population, we arrive at genome-wide constitutive transcription, in which the rate of RNA production is constant. This contradicts numerous sources that suggest transcriptional activity varies with time even in stationary cells [62, 90, 97–103], and is effectively described by a telegraph model that stochastically switches between active and inactive states [104, 105].

Thus, we must impose basic consistency criteria. Using the models outlined under “RNA velocity biophysics” requires the assumption that stationary, homogeneous cell populations are Gaussian or Poisson-distributed. This assumption contradicts at least thirty years of evidence for widespread bursty transcription [98, 105]. We have obtained the answer to question (1) in the previous subsection: the model must be coherent with known biophysics, and provide a robust way to identify cases when its assumptions fail.

Count processing

Some standard properties of constitutive systems appear to at least qualitatively motivate gene filtering. Only genes with spliced–unspliced Pearson correlation above 0.05 are used for fitting parameters (as on p. 4 of SN2 in [1]); if the correlation is below this threshold, the gene is removed from procedure and presumed stationary. This is valid for the constitutive model, but inappropriate for broader model classes: for example, bursty transcription yields strictly positive correlations, making this statistic ineffective for identifying dynamics [81, 106].

Normalization relative to the cell’s molecular count is a standard feature of sequencing workflows [37, 107, 108], but reduces interpretability. Normalization converts absolute discrete count data to a proportion of the total cellular counts, ostensibly to account for the compositional nature of read-data data [109]. Several recent studies strongly discourage normalization of UMI-based counts [110, 111], although this perspective is not universal [112, 113]. It is clear that continuous-valued normalized data are incompatible with discrete mechanistic models. Moreover, the suitability of continuous models (such as Eq 1) is never explicitly justified, but merely assumed. Since normalization nonlinearly transforms the molecule distributions [68, 111] and introduces a coupling even between independent genes, the precise interpretation of single-gene ODE models is unclear.

Nearest-neighbor averaging is used to smooth the data after normalization. Though it effaces much of the stochastic noise to give an “averaged” trajectory, it introduces distortions of unknown magnitude. As discussed in the subsection “Inference” under “Workflow and implementations,” the imputation step does not have a consistent interpretation. The original report [1] defines it as “kNN pooling” in the manuscript and “imputation” in the documentation (and Fig 17 of SN2), placing the emphasis on denoising. On the other hand, *scVelo* interprets the local average as an estimate of the expectations $\mu_u(t)$, $\mu_s(t)$. Neither approach appears to be justified by previous studies or benchmarking on ground truth, and both are circular as the neighborhood is computed based on the observed counts. A probabilistic analysis in Section B in S1 Text formalizes more deep-seated issues with using model-agnostic point estimates to “correct” data. Although these claims may hold and simply require more theoretical work to prove, our simulations in “Count processing” under “Prospects and solutions” strongly suggest they are invalid even in the best-case scenario: the phase portraits are smoothed out, but fail to capture the underlying dynamics in a way coherent with those claims.

To illustrate these problems, we performed a simple test of self-consistency, illustrated in Fig 5. We reprocessed the forebrain dataset (Fig 4 of [1]) using the *velocyto* workflow, varying k , and investigated its effect on the appearance of the phase plots and the inferred parameters. As the neighborhood size was increased, the phase plot was distorted, with no apparent “optimal” choice of k .

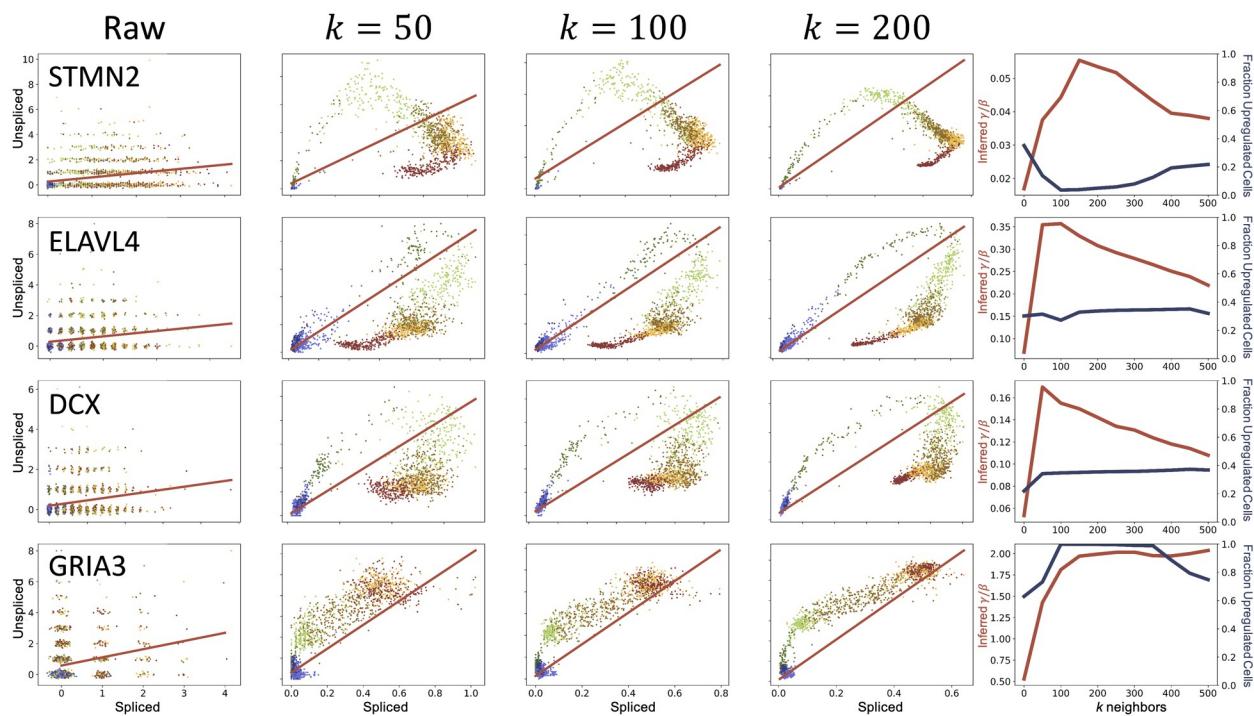


Fig 5. Distortions in data and instabilities in the inferred γ/β values introduced by the imputation procedure on the forebrain data from [1]. Column 1: Raw data (points: spliced and unspliced counts with added jitter; color: cell type, as in Fig 1; line: best fit line $u = \gamma s/\beta + q$, estimated from the entire dataset). Columns 2–4: Normalized and imputed data under various values of k (points: spliced and unspliced counts; color: cell type, as in Fig 1; line: best linear fit $u = \gamma s/\beta + q$, estimated from extreme quantiles). Column 5: Inferred values of γ/β (red, left axis) and inferred fraction of upregulated cells, defined as $\sum_i u_i - (\gamma s_i/\beta + q) > 0$ (blue, right axis).

<https://doi.org/10.1371/journal.pcbi.1010492.g005>

Inference

Broadly speaking, *velocyto*-like moment estimates for γ/β are legitimate if the system has time to equilibrate (as outlined under “The ‘deterministic’ *velocyto* model as a special case”). However, moment-based estimation underperforms maximum likelihood estimation in general. The two approaches are in concordance under the highly restrictive assumptions of error normality and homoscedasticity. These assumptions are routinely violated in the low-copy number regime [88].

Regression on top and bottom quantiles inherits all of the issues of regression on the entire dataset, but compounds them by discarding a large fraction of data. Extremal quantile regression is otherwise a well-developed method [114–117], but it is generally applied to processes with nontrivial tail effects. For the quantile computation the filtering criterion is *ad hoc*, and not amenable to theoretical investigation. The order statistics of discrete distributions are notoriously challenging to compute [118–120], and even the simplest Poisson case exhibits complex trends [121]. In other words, the extrema themselves may be affected by noise, introducing more uncertainty into inference. Although the original article does perform some validation (SN2, Sec. 3 of [1]), it focuses on cell-specific velocities rather than parameter values, and only provides relative performance metrics rather than actual comparisons to simulated ground truth.

Even without testing the inference procedures against simulations, we can characterize their performance in terms of internal controls. As we demonstrate in Fig 5, the inferred γ/β values were unstable under varying k : the *velocyto* parameter inference procedure was highly sensitive to a user-defined neighborhood hyperparameter. On the other hand, using a simple ratio of the means (as in the first column and the $k = 0$ case in the fifth column of Fig 5) produced biases [1].

Interestingly, the fraction of cells predicted to be upregulated is qualitatively more stable, suggesting that the inference step is best understood as an *ad hoc* binary classifier, rather than a quantitative descriptor of system state. Given the stability of this classifier, as well as our preliminary discussion of similar results in the context of validating *protacel* [4], we used this binary classifier as a benchmark in the subsections named “Embedding” under “Logic and methodology” and “Prospects and solutions.”

Regression of the piecewise deterministic “dynamical” model in *scVelo* asserts the imputed counts have normal noise with equal residuals for spliced and unspliced species, once again implausible in the low-copy number regime. More fundamentally, it fails to preserve gene-gene coherence. If a cell is predicted to lie at the beginning of a trajectory for one gene, this estimate does not inform fitting for any other gene. The “dynamical” model appears to address this discrepancy in a *post hoc* fashion. First, the algorithm identifies putative “root cells,” which are themselves computed from the velocity embedding. Then, the disparate gene-specific times are aggregated into one by computing a quantile near the median. This procedure presupposes that the velocity graph is self-consistent and physically meaningful, and that the point estimate of process time is sufficient, but does not mathematically prove these points or test them by simulation.

Embedding

After inference and evaluation of Δs for every cell and gene, the array is converted to an embedding-specific representation. In the single-cell sequencing field, low-dimensional projections are more than a visualization method: they are ubiquitous tools for inference and discovery. Transcriptomics workflows convert large data arrays to human-parsable visuals; these visuals are then used to explore gene expression and validate cell type relationships, under the assumption that they represent the underlying data well enough to draw conclusions. However, the embedding procedures involve several distortive steps, which should be recognized and questioned.

For such visuals, the goal is to recapitulate local and global cell-cell relationships. However, accurately representing desired properties such as pairwise relationships between many points is inherently difficult, requiring dimensions several orders of magnitudes larger than two to faithfully represent the data [122]. Thus distortion of cell-cell relationships is naturally induced in two-dimensional embeddings, and grows as $\mathcal{O}(\sqrt{M})$ for M cells [122, 123]. Both linear (PCA) and nonlinear (*t*-SNE/UMAP) methods exhibit these distortions, and warp existing cell-cell relationships or suggest new ones not present in the underlying data [122, 124]. Tuning algorithm parameters can slightly improve some distortion metrics, though often at the expense of others [125]. Essentially, nonlinear embeddings utilize sensitive hyperparameters that can be tuned, but do not provide well-defined criteria for an “optimal” choice [124, 125]. Using visualizations for discovery thus risks confirmation bias.

The velocity algorithms present a particularly natural criterion for quantifying the embedding distortion. The nonlinear embedding procedure generates weights for vectors defined with reference to embedding neighbors. Therefore, we can reasonably investigate the effect of the embedding on the neighborhood definitions. In other words, if the velocity arrows quantify the probability of transitioning to a set of cells, what relationship does this set have to the set of neighbors in the pre-embedded data?

This relationship is conventionally [124] quantified by the Jaccard distance, defined by the following formula:

$$d_j = \frac{|A \cup B| - |A \cap B|}{|A \cup B|}, \quad (7)$$

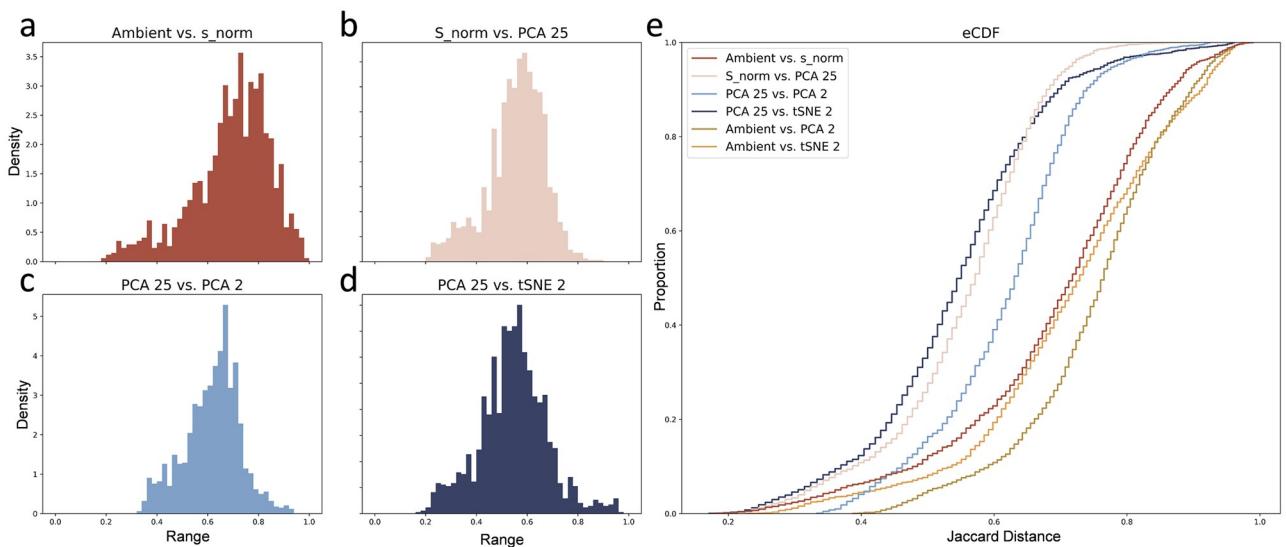


Fig 6. Normalization followed by two rounds of dimensionality reduction introduce distortions in the local neighborhoods. a.–d. Histograms of Jaccard distances between intermediate embeddings. e. Empirical cumulative distribution functions of Jaccard distances between intermediate embeddings, as well as the overall distortion (Ambient vs. PCA 2 and Ambient vs. tSNE 2). The palette used is derived from dutchmasters by EdwinTh.

<https://doi.org/10.1371/journal.pcbi.1010492.g006>

which reports the normalized overlap between the original sets of cell neighbors (A) and embedded cell sets (B). This dissimilarity metric ranges from 0 to 1, where 1 (or 100%) denotes completely non-overlapping sets. We applied standard steps of dimensionality reduction and smoothing to the forebrain dataset (Fig 4 of [1]) and computed their effect on the neighborhoods (taken to be $k = 150$ for consistency with the velocity embedding process). We report the Jaccard distance distributions in Fig 6, and observe the gradual degradation of neighborhoods. On average, moving from the ambient high-dimensional space to a two-dimensional representation induced a d_J of 70–75%. Therefore, cell embedding substantially distorts precisely the local structure relevant to velocity embedding.

The two-dimensional arrows in Fig 1 combine three sources of error: the intrinsic information loss of low-dimensional projections, the instabilities in upstream processing and inference, and any additional error incurred by the nonlinear procedure outlined in “Embedding” under “Workflow and implementations.” The softmax kernel-based procedure exhibits an inherent tension that merits closer inspection. On one hand, it is explicitly designed [1] to mitigate error incurred by cell- and gene-specific noise by performing several steps of pooling and smoothing. On the other hand, it is technically questionable: if we assume differentiation processes are largely governed by a small set of “marker genes,” pooling them with thousands of non-marker genes amounts to hoping that variation in an orthogonal data-generating process cancels out well enough to recapture latent dynamics. Certain processes may involve the modulation of large sets of genes, e.g., if expression overall increases over the transcriptome. However, the velocity workflows are intrinsically unable to identify such trends, as they use normalized data. As we demonstrate later, a model with no latent dynamics at all (Fig F in S1 Text) can generate apparent signal in the embedded space, illustrating the dangers of relying on error cancellation. When multiple data-generating processes are present, naïve aggregation risks obscuring rather than revealing signal.

Aside from this high-level inconsistency, other problems emerge upon investigating the embedding procedure closer, even prior to performing any numerical controls. The nonlinear

embedding approach introduced by La Manno et al. ([Eq 4](#)) is highly hyperparametrized, not motivated by any previous theory, has no physical interpretation, and does not appear to have been formally validated against linear or simulated ground truth. Just as with the cell embedding, the procedure is dependent on an arbitrary number of nearest neighbors and velocity transformation functions, with no clearly optimal choices. These hyperparameters can be tuned to correct for such instabilities, potentially resulting in overfitting to a pre-determined hypothesis. Since the procedure has no physical basis, potential false discoveries are challenging to diagnose. Furthermore, it reduces the limited biophysical interpretability of the result, particularly because the relationship between cell state graphs and the underlying physical process is obscure and subject to distortions (Section C in [S1 Text](#)). The velocity derivation is model-informed and, as discussed under “The ‘deterministic’ *velocyto* model as a special case” and “The ‘dynamical’ *scVelo* model as a special case” in the next section, can be informally viewed as an approximation under several strong assumptions about the process biophysics. The embedding, on the other hand, is *ad hoc* and can only degrade the information content.

A final theoretical point remains before we can begin quantitatively validating the embeddings: as suggested by [Eq 2](#), and discussed under “Inference” in the previous section, the *velocyto* gene-specific v_j have different units. Therefore, the aggregation in [Eq 4](#) is questionable. The standard *velocyto* workflow assumes that the splicing rates are close enough to neglect differences, which appears to contradict other results reported in the same paper (Extended Data Fig 2f of [\[1\]](#)).

To bypass this limitation in a self-consistent way, we implemented a “Boolean” or binary measure of velocity, as motivated by validation in the original manuscript (Sec. 3 in SN2 of [\[1\]](#)), introduced in the context of validating *protaccel* [\[4\]](#), and implied by resampling β values from a uniform distribution in an investigation of latent landscapes (p. 3 in Supplementary Methods of [\[29\]](#)). Essentially, instead of computing transition probabilities based on the velocity values, we computed them based on signs, bypassing the unit inconsistency. The algorithm used to produce this embedding is described under “Velocity embedding.”

The Boolean procedure offers a natural internal benchmark. If this approach largely recapitulates findings from the standard methods, the embedding process serves as an information bottleneck: the inference procedure performs as well as a binary classifier, and the complexities of the dynamics are effaced by embedding. We used this approach as a trivial baseline and compared it to the standard suite of variance-stabilizing transformations implemented in *velocyto*. In addition, we tested the effects of neighborhood sizes, in the vein of the stability analysis performed in the original manuscript (Sec. 11 in SN2 of [\[1\]](#)). In Fig D in [S1 Text](#), we plot the distributions of angle deviations between a linear baseline, obtained by projecting the extrapolated cell state and computing $E(s_i + \Delta s_i) - E(s_i)$ using PCA, and the nonlinear velocity embedding. This control has not previously been investigated in any detail, but seems key to the claim that the nonlinear velocity embedding is meaningful: intuitively, we expect it to recapitulate the simplest baseline, at least on average. To avoid any confusion, we reiterate that the linear embedding is given by [Eq 3](#), and not the identity nonlinear embedding implemented in *velocyto* (i.e., $\varrho(x) = x$ in SN1, pp. 9–10 of [\[1\]](#)).

The angle deviations in arrow directions were all severely biased relative to the linear baseline. The different normalization methods were distortive to approximately the same degree. The performance of the Boolean embedding, which discards nearly all of the quantitative information, was nearly identical to the built-in methods, which suggests that the choice of normalization methods is a red herring: quantitative velocity magnitudes have little effect on the embedding quality. This is consistent with previous investigations (cf. Fig S52 in [\[4\]](#)). On the other hand, the neighborhood sizes did not appear to matter much, at least over the modest range explored here (in contrast to Sec. 11 in SN2 of [\[1\]](#)). Therefore, the directions reported in

embeddings were unrepresentative of the actual velocity magnitudes in high-dimensional space, as well as severely distorted relative to the linear projection. These discrepancies are a potential cause for concern. Observing the qualitative similarity of Figs 2d and 2h in the original report [1], the reasonable performance of the linear extrapolation in t-SNE in its supplement (SN2 Fig 9a of [1]), as well as the cell cycle dynamics explored with the linear embedding in *Revelio* [11], a casual reading of the RNA velocity literature would suppose these embeddings to be largely interchangeable.

Finally, we visualized the aggregated velocity vectors in Fig 7 to assess the local and global structures. This visualization served as both an internal and an external control. The internal control demonstrated the local structure and the stability of the velocity-specific methods, i.e., the actual directions of the arrows on the grid. We compared the conventional nonlinear projection to the Boolean method, as well as the linear embedding. The external control concerned the global structure, which can be analyzed in light of known physiological relationships: radial glia differentiate through neuroblasts into neurons [1]. If this global relationship is not captured by the embedding, the inferred trajectories are *a priori* physically uninterpretable, in a way that is particularly challenging to diagnose.

In the PCA embedding, the global structure was retained and the arrows were fairly robust, even when the non-quantitative Boolean method was used. However, the various projection options suggested drastically different relationships between the cell types, with PCA presenting more continuous representations of cell relationships faithful to ground truth, and UMAP and t-SNE presenting more local images, with distinct and discrete clusters of cells. Clearly, if the relationship between progenitor and descendant is lost, the velocity workflow cannot infer it. The t-SNE and UMAP parameters can be adjusted by the user; however, adding a new set of tuning steps and optimizations provides an opportunity for confirmation bias to overrule the data.

Summary

The standard RNA velocity framework presupposes that the evolution of every gene's transcriptional activity throughout a differentiation or cycling process can be described by a continuous model with a single upregulation event and a single downregulation event. It proceeds to normalize and smooth the data until the rough edges of single-molecule noise are filed off, and fitting the continuous model assuming Gaussian residuals.

In the process, the stochastic dynamics that predominate in the low-copy number regime, and that characterize nearly all of mammalian transcription, are lost and cannot be recovered. Although parameters can be fit, they are distorted to an unknown extent, due to a combination of data transformation, suboptimal inference, and unit incompatibilities. The gene-specific components of velocity are underspecified due to their direct dependence on the imputation neighborhood and splicing timescale. In *scVelo*, parameters are estimated under a highly restrictive model, yet applied to make broad claims about complex topologies. In *velocyto*, only the sign of the velocity is physically interpretable. It can still be used to calculate low-dimensional directions, and this binary velocity embedding is seemingly as good as any other, suggesting that other methods fail to fully utilize valuable information. However, the embedding process itself is not based on biophysics, and is not guaranteed to be stable or robust. Fortunately, the natural match between stochastic models and UMI-aided molecule counting offers hope for quantitative and interpretable RNA velocity.

Prospects and solutions

Is there no balm in Gilead? Given the foundational issues we have raised, how can the RNA velocity framework be reformulated to provide meaningful, biophysically interpretable

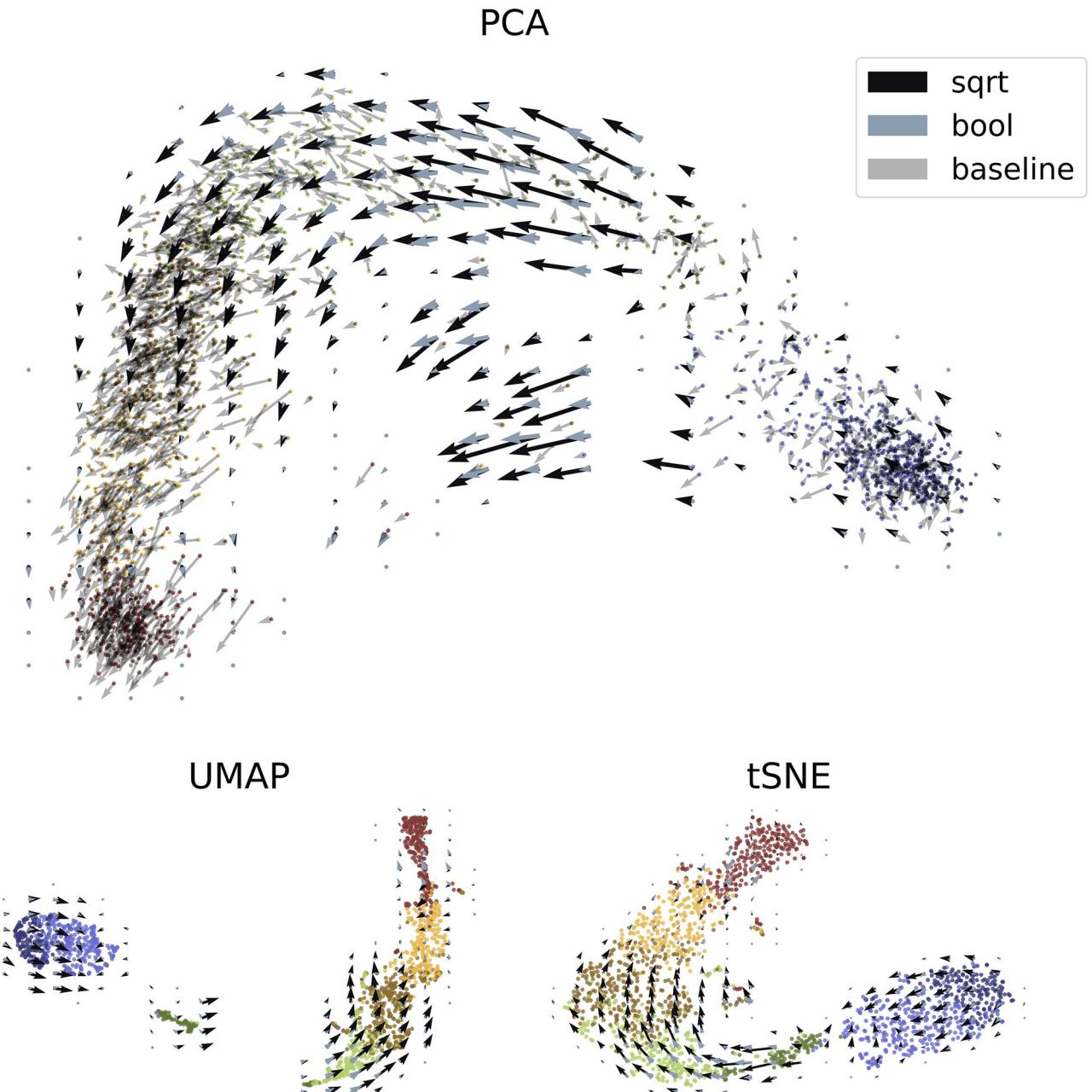


Fig 7. Performance of cell and velocity embeddings on the forebrain data. Top: PCA embedding with linear baseline and nonlinear aggregated velocity directions. Bottom: UMAP and *t*-SNE embeddings with nonlinear velocity projections. The palette used is derived from dutchmasters by EdwinTh.

<https://doi.org/10.1371/journal.pcbi.1010492.g007>

insights? We propose that discrete Markov modeling can directly and naturally address the fundamental issues. In particular, transient and stationary physiological models can be defined and solved via the chemical master equation (CME), which describes the time evolution of a discrete stochastic process. Since the “noise” is the data of interest, in such an approach smoothing is not required. Rather, technical and extrinsic noise sources can be treated as stochastic processes in their own right, and explicit modeling of them can improve the

understanding of batch and heterogeneity effects. Finally, within this framework, parameters can be inferred using standard and well-developed statistical machinery.

Pre-processing

The diversity of potential intermediate and terminal transcripts suggests that simplistic splicing models are inadequate for physiologically faithful descriptions of transcription dynamics. What is needed is a treatment of the types of transcripts listed in “Pre-processing” under “Logic and methodology” as distinct species. This approach immediately leads to several significant challenges, relating to quantification, biophysics, and identifiability.

Transient, low-abundance intermediate transcripts are substantially less characterized than coding isoforms. Some data are available from fluorescence transcriptomics with intron-targeted probes [47], but such imaging is impractical on a genome-wide scale. Unfortunately, the references and computational infrastructure necessary to identify intermediate transcripts do not yet exist.

Even if intermediate isoforms could be perfectly quantified, single-cell RNA-seq data do not generally contain enough information to identify the order of intron splicing. The problem of splicing network inference has been examined; however, experimental approaches [126, 127] are challenging to scale, whereas computational approaches [128] do not generally have enough information to resolve ambiguities.

Furthermore, even with complete annotations and a well-characterized splicing graph at hand, large-scale short-read sequencing cannot fully resolve transcripts. This limitation gives rise to a challenging inference problem. For example, if transcripts $A := E_1 I_1 E_2 I_2 E_3$ and $B := E_1 E_2 I_2 E_3$ are indistinguishable whenever only the 3' end of each molecule is sequenced, it is necessary to fit parameters through the random variable $X_A + X_B$, i.e., from aggregated data. The functional form of this random variable's distribution is not yet analytically tractable.

We have described a preliminary method that can partially bypass these problems [81]. Sequencing “long” reads, which at this time is possible with technologies such as Oxford Nanopore [72], or sequencing of “full-length” libraries produced with methods such as Smart-seq3 [129], enhances identifiability and facilitates the construction of new annotations based on presence or absence of intron sequences. Finally, even though additional data are required to specify entire splicing networks, sequencing data are sufficient to constrain parts of these networks; for example, if two transcripts differ by one intron, the longer one cannot possibly be generated from the shorter.

Defining more species leads to inferential challenges in downstream analysis. Even if sequencing data are available, their relationship to the biological counts is nontrivial: some intermediate transcripts may not be observable using certain technologies because they do not contain sequences necessary to initiate reverse priming, whereas others may be overrepresented in the data because they contain many. In a preliminary investigation [130], which adopts the binary categories of “spliced” and “unspliced” defined in the original RNA velocity publication, we found that unspliced molecules originating from long genes are overrepresented in short-read sequencing datasets. This suggests that multiple priming occurs at intronic poly(A) sequences. To “regress out” this effect, a simple length-based proxy for the number of poly(A) stretches can be used, but a more granular description would require a sequence-based kinetic model for each intermediate transcript's capture rate.

Occupation measures provide a theoretical framework for scRNA-seq

The data simulated in the exposition of RNA velocity [1] comes from a particular set of what are called Markov chain occupation measures. As an illustration of what this means, we

consider the simplest, univariate model of transcription, a classical birth-death process:



where α is a constant transcription rate and β is a constant efflux rate. Depending on the system, β may have various biophysical interpretations, such as splicing, degradation, or export from the nucleus [81, 106].

Formally, the exact solution to this system is given by the chemical master equation (CME), an infinite series of coupled ordinary differential equations that describe the flux of probability between microstates x , which specify the integer abundance of \mathcal{X} , defined on \mathbb{N}_0 :

$$\frac{dP(x; t)}{dt} = \alpha[P(x - 1; t) - P(x; t)] + \beta[(x + 1)P(x + 1; t) - xP(x; t)]. \quad (9)$$

This equation encodes a full characterization of the system: transcription is zeroth-order, efflux is first-order, and the dynamics are memoryless, i.e., depend only on the state at t . We define the quantity y , namely the solution to the underlying reaction rate equation that governs the average of the \mathcal{X} copy number distribution:

$$\begin{aligned} \frac{dy}{dt} &= \alpha - \beta y, \\ \Rightarrow y(t) &= y_0 e^{-\beta t} + \frac{\alpha}{\beta} (1 - e^{-\beta t}), \end{aligned} \quad (10)$$

where y_0 is the average at time $t = 0$. To simplify the analysis, we assume that the initial condition is Poisson-distributed. Per classical results [82], the distribution of counts $P(x; t)$ is described by a Poisson law for all t , and converges to $\text{Poisson}(\alpha/\beta)$ as $t \rightarrow \infty$. Quantitatively, the time-dependent distribution is given by $P(x; t) \sim \text{Poisson}(y(t))$:

$$P(x; t) = \frac{1}{x!} y(t)^x e^{-y(t)}. \quad (11)$$

However, this is not the correct model class for distributions observed in scRNA-seq datasets. To appreciate these subtleties, we delve into and interrogate assumptions that underpin the use of such distributions.

The sequencing process does not “know” anything about the transcriptional dynamics and their time t . This stands in contrast to transcriptomics performed *in vitro*, with a physically meaningful experiment start time. For example, in many standard protocols, a stimulus is applied to the cells at time $t = 0$, and populations of cells are chemically fixed and profiled at subsequent time points, potentially up to a nominal equilibrium state [2, 34, 88, 131]. However, if there is no experimentally imposed timescale, and we adopt the standard assumption that cell dynamics are mutually independent, the process time decouples from experiment time. Although cells are sampled simultaneously, their process times t are draws from a random variable that must be defined.

Formalizing this framework requires introducing the notion of occupation measures. Considering a single cell, we designate its process time t as a latent process time (Fig 8a), the mechanistic analogue to the phenomenological pseudotime. In brief, “pseudotime” conventionally denotes a one-dimensional coordinate along a putative cell trajectory, which parametrizes a principal curve in a space based on RNA counts [37] (Fig 8b and 8c). On the other hand, the process time is a real time coordinate, which governs the “clock” of the stochastic process. Broadly speaking, this is the physical quantity which the “latent time” discussed in the exposition of *scVelo* attempts to approximate. The difference between pseudotimes and process

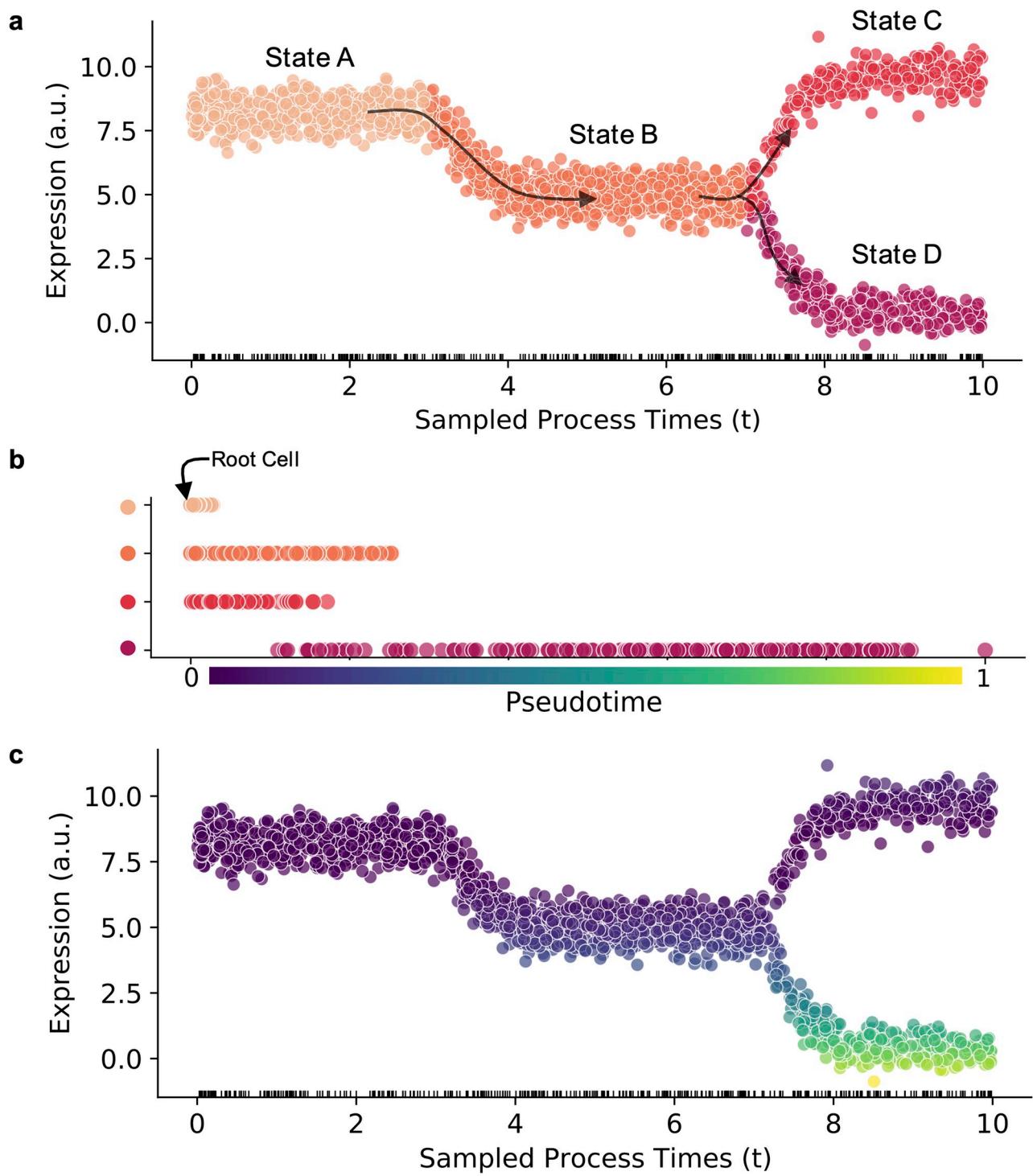


Fig 8. Markov Chain process time versus expression pseudotime. **a.** Simulated gene expression for 2000 cells over 4 states (states A, B, C, and D) with a bifurcation at C/D showing spliced counts of a single gene at the sampled process times. The abbreviation a.u. denotes arbitrary units. **b.** Ordering of all cells by expression pseudotime coordinate, calculated as the Euclidean distance between each cell and the root cell (cell at time $t = 0$) and scaled to between 0 and 1. **c.** The sampled cells colored by the calculated pseudotime value.

<https://doi.org/10.1371/journal.pcbi.1010492.g008>

times is fundamental. The Markov chain process time is physically interpretable as the progress of a process that induces the observations in expression space. Conversely, the expression pseudotime is purely phenomenological (Fig 8c), and we are unaware of any trajectory inference methods that explicitly parameterize the underlying stochastic model using the CME; instead, all available implementations appear to use isotropic or continuous noise models [37, 108, 132–140]. As we emphasize in “Model definition” under “Logic and methodology,” these models are inappropriate for low-abundance molecular species.

By construction, cell trajectories are observed at times $t \in \mathbb{R}$ (Fig 8a). This requires introducing a sampling distribution $f(t)$, which describes the probability of observing a cell at a particular underlying process time. Therefore, the probability of observing x molecules of \mathcal{X} in the constitutive case takes the following form:

$$P(x) = \int P(x; t) df(t) = \mathbb{E}_f[P(x; t)], \quad (12)$$

i.e., the expectation of $P(x; t)$ under the sampling law. $P(x)$ is called the occupation measure of the process, and reports the probability that a trajectory is observed to be in state x , a slight generalization of the usual definition [141–143].

Next, we must encode the assumption that cell observations are desynchronized from the sequencing process and each other. This assumption leads us to a choice consistent with the previous reports [1, 21], namely $df = T^{-1}dt$, where $[0, T]$ is the process time interval observable by the sequencing process. This constrains the probability of observing state x to be the actual fraction of time the system spends in that state. Then, we take $T \rightarrow \infty$, yielding

$$P(x) = \lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T P(x; t) dt = \lim_{t \rightarrow \infty} P(x; t), \quad (13)$$

which is a statement of the ergodic theorem [144]. Under mild conditions, this theorem guarantees that samples from unsynchronized trajectories converge to the same distribution as the far more tractable ensembles of synchronized trajectories.

With this discussion, we have clarified that the application of stationary distributions $\lim_{t \rightarrow \infty} P(x; t)$ to describe ostensibly unsynchronized cells naturally emerges from assumptions about biophysics and the nature of the sampling process. However, these assumptions may be violated; for example, RNA velocity describes molecules sampled from a transient process. This distinction is key: limits such as $\lim_{t \rightarrow \infty} P(x; t)$ may not even exist, and we expect to capture only a portion of the trajectory. A rigorous probabilistic model must treat the occupation measure directly, which remains valid without those assumptions. Formally, this amounts to relaxing the assumption of desynchronization: the sequencing process is time-localized to a particular interval of the underlying biological process.

To stay consistent, we continue using the sampling law $df = T^{-1}dt$ on $[0, T]$, but infinite supports are valid so long as they decay rapidly enough to be integrable. As scRNA-seq data are atemporal, this time coordinate is unitless and cannot be assigned a scale without prior information, so T can be defined arbitrarily without loss of generality.

The occupation measure of the birth-death process takes the following form:

$$P(x) = \frac{1}{T} \int_0^T P(x; t) dt = \frac{1}{Tx!} \int_0^T y(t)^x e^{-y(t)} dt. \quad (14)$$

This integral can be solved exactly; however, this solution does not easily generalize to more complex systems. Instead, we can consider the probability-generating function (PGF), which

also takes a remarkably simple form:

$$\begin{aligned}
 G(z; t) &:= \mathbb{E}[z^x] = \sum_{x=0}^{\infty} P(x; t) z^x, \\
 H(z) &= \mathbb{E}_f[G(z; t)] = \sum_{x=0}^{\infty} \int P(x; t) z^x df \\
 &= \frac{1}{T} \int_0^T \sum_{x=0}^{\infty} P(x; t) z^x dt = \frac{1}{T} \int_0^T G(z; t) dt.
 \end{aligned} \tag{15}$$

By linearity, the generating function $H(z)$ of the occupation measure is the expectation of the generating function $G(z; t)$ of the original process with respect to the sampling measure f . From standard properties of the birth-death process, this yields:

$$\begin{aligned}
 y(t) &= (y_0 - \alpha/\beta)e^{-\beta t} + \alpha/\beta := ce^{-\beta t} + \alpha/\beta, \\
 G(z; t) &= e^{(z-1)y(t)} := e^{uy(t)}, \\
 H(z) &= \mathbb{E}_f[G(z; t)] = \frac{1}{T} \int_0^T e^{uy(t)} dt \\
 &= \frac{1}{T} e^{uz/\beta} \int_0^T \exp(uce^{-\beta t}) dt \\
 &= \frac{e^{uz/\beta}}{\beta T} [\text{Ei}(uc) - \text{Ei}(uce^{-\beta T})],
 \end{aligned} \tag{16}$$

where Ei is the exponential integral [145]. This is the solution to the system of interest. Although straightforward to evaluate, it does not appear to belong to any well-known parametric family.

Modular extensions to broader classes of biological phenomena

Extending this approach to more complicated biological models requires positing a hypothesis about the dynamics of transcription and the mRNA life-cycle, formalizing it as a CME, then solving that CME. This is the “forward” problem of statistical inference, which is typically intractable. However, in the current subsection, we summarize the model components that can be assembled to produce solvable systems, and outline potential challenges.

The canonical velocity model. To begin, we would like to fully recapitulate the model introduced and simulated in the original publication [1]. This model has the following structure:

$$\emptyset \xrightarrow{\alpha(t)} \mathcal{U} \xrightarrow{\beta} \mathcal{S} \xrightarrow{\gamma} \emptyset, \tag{17}$$

where $\alpha(t)$ is a piecewise constant transcription rate. The instantaneous distribution of this

process over copy number states (x_u, x_s) is well-known [81, 82]:

$$\begin{aligned}
 \mathbf{x} &= (x_u, x_s) \\
 G(u_u, u_s; t) &= \sum_{\mathbf{x}} P(\mathbf{x}; t)(u_u + 1)^{x_u}(u_s + 1)^{x_s} \\
 U_1(u_u, u_s; s) &= \frac{u_s \beta}{\beta - \gamma} e^{-\gamma s} + \left(u_u - \frac{u_s \beta}{\beta - \gamma} \right) e^{-\beta s} \\
 \phi(u_u, u_s; t) := \ln G(u_u, u_s; t) &= \int_0^t \alpha(t-s) U_1(u_u, u_s; s) ds \\
 &= \mu_u(t) u_u + \mu_s(t) u_s \\
 P(\mathbf{x}; t) &= \frac{\mu_u(t)^{x_u} e^{-\mu_u(t)}}{x_u!} \frac{\mu_s(t)^{x_s} e^{-\mu_s(t)}}{x_s!} \\
 P(\mathbf{x}) &= \frac{1}{T} \int_0^T P(\mathbf{x}; t) dt
 \end{aligned} \tag{18}$$

where U_1 is the characteristic of the unspliced mRNA solution, whereas $\mu_u(t)$ and $\mu_s(t)$ are the instantaneous averages, which can be written down in closed form for arbitrary piecewise constant $\alpha(t)$. The generating function derivation presupposes that the system starts at steady state. By defining N genes whose RNA abundance evolves on $[0, T]$, we can extend this univariate distribution to a multivariate occupation measure in $2N$ dimensions. This occupation measure is the putative source distribution for cells observed in experiment.

[Eq 18](#) recapitulates the system proposed in the original publication [1], and we use it to set up and solve transient biophysical systems throughout the rest of this report. Restricting analysis in this way allows us to speculate about how such models could be fit, while keeping the mathematics in closed form. However, the schema in [Eq 17](#) omits important physiological phenomena. Although these phenomena can be modeled, very few of them afford closed-form solutions, and their range can be classified according to their complexity.

PGF-tractable models. The first class includes models that afford transient single-gene solutions in terms of the generating function, and can be combined with the definition of an occupation measure to produce a generative model for observed data. These models can be solved using quadrature, and extend the dynamics in simple, Markovian ways, although strategies for fitting them are not yet well-developed.

The solution in [Eq 18](#) generalizes to an arbitrary number of species, merely by converting the stoichiometry and rate matrix to the appropriate $U_1(\mathbf{u}; s)$. The solution for a generic splicing and degradation network, under constitutive transcription, stays Poisson, and makes up one of the few closed-form cases [81, 82].

Transcriptional bursting at a gene locus can be represented by slightly modifying [Eq 18](#):

$$\phi(\mathbf{u}; t) = \int_0^t \alpha(t-s) [M(U_1(\mathbf{u}; s); t-s) - 1] ds, \tag{19}$$

where M is the generating function of the burst distribution, which governs the number of unspliced mRNA molecules produced per transcriptional event. This distribution can be time-dependent and reflect the transient modulation of bursting dynamics. As in [Eq 18](#), we omit the explicit representation of initial conditions, described in [81].

In principle, we need not assume $\alpha(t)$ in [Eq 17](#) is piecewise constant. If $\alpha(t)$ is deterministic, the solution can be obtained by quadrature. More sophisticated regulatory dynamics, such as transcriptional rates governed by a continuous- or discrete-valued stochastic process [146],

[147], can be treated analogously. If the process is continuous-valued, e.g., representing the concentration of a rapidly-binding regulator, it requires solving a single, potentially nonlinear ODE. If it is Markovian and discrete-valued, e.g., representing transitions between distinct promoter stats, it requires solving a set of coupled ODEs.

Gene-gene coexpression, or the synchronization of transcriptional events induced by a common regulator, can be solved by setting M in Eq 19 to a multivariate generating function whose arguments are the characteristics of unspliced mRNA species [81].

The model is broad enough to describe cell type distinctions and cell fate stochasticity simply by defining a discrete mixture model over transcription rate trajectories. For example, if a cell can choose to enter cell fate A with probability w_A or cell fate B with probability $1 - w_A$, the overall generating function takes the following form:

$$\begin{aligned} G(\mathbf{u}; t) &= w_A G_A(\mathbf{u}; t) + (1 - w_A) G_B(\mathbf{u}; t) \\ G(\mathbf{u}) &= w_A G_A(\mathbf{u}) + (1 - w_A) G_B(\mathbf{u}), \end{aligned} \quad (20)$$

where each branch's generating function is induced by distinct driving functions, $\alpha_A(t) \neq \alpha_B(t)$. This formulation is equivalent to defining a finite mixture of branches, with a categorical distribution of parameters as the mixing law. As previously discussed [81], it can be extended to continuous mixtures, although such systems are not typically tractable.

Certain models of technical noise are modular with respect to the biological processes. Specifically, if we suppose that sequencing is a random process that is independent and identically distributed for every molecule of a particular species i , with a distribution on \mathbb{N}_0 , the PGF of the observed UMIs is G evaluated at $x_i = G_{t_i, i}(x_i)$, where $G_{t_i, i}(x_i)$ is the PGF of the sampling distribution for species indexed by i . We have previously considered this model for Bernoulli and Poisson distributions, which presuppose that the cDNA library construction is sequestering and non-sequestering, respectively [130, 148].

Finally, it is useful to consider the meaning behind the choice of f in Eq 12. Thus far, we have assumed f is a uniform law on $[0, T]$. This choice is underpinned by a particular conceptual model of cell dynamics relative to the sequencing process. Formalizing such a conceptual model requires introducing several fundamental principles from the field of chemical reaction engineering and interpreting them through the lens of living systems. A reactor is a vessel that contains a reacting system; in the current context, it is a living tissue which is isolated for sequencing.

Three idealized extremes of reactor configurations exist: the plug flow reactor (PFR), the continuous stirred tank reactor (CSTR), and the batch reactor (BR). The PFR has no internal mixing: the reaction stream enters in the influx and exits from the efflux after a deterministic amount of time. Therefore, the PFR has memory: if the total residence time is T and reactor length is L , and the fluid element is localized at position Lt/T within the PFR, it will exit after a delay of $T - t$. Conversely, the CSTR is fully mixed: the reaction stream enters the vessel and combines with its existing contents. Therefore, the CSTR is memoryless or Markovian: if the average residence time is T , a given fluid element within the reactor will exit after a delay distributed according to $\text{Exp}(T^{-1})$, independent of time since its entrance [149]. The CSTR and PFR can operate at steady state, whereas the BR accumulates reaction products: the BR is charged at $t = 0$, and the reaction proceeds until a predetermined time T .

We can translate these basic configurations to the design of transcriptomics experiments, treating individual cells as fluid elements, and assuming the variation in transcriptional dynamics begins as the cells enter the reactor at $t = 0$. If the sampling process is independent from a PFR in space or a BR in time, we obtain the uniform sampling measure $df = T^{-1}dt$. If the process samples from a CSTR, we obtain the exponential sampling measure

$df = T^{-1}e^{-tT^{-1}}dt$. If the process samples from the efflux of a PFR or the product of a BR, we obtain the Dirac sampling measure $df = \delta(T - t)dt$. The first two scenarios appear to be more appropriate for describing scRNA-seq experiments, which can sample across entire processes, whereas the final scenario appears to correspond to fluorescence transcriptomics experiments, which have a well-defined start time. Assuming there are no cyclostationary dynamics, all of the reactor and sampling configurations converge to the ergodic limit as $T \rightarrow \infty$.

This conceptual model is highly simplified, but serves to motivate the need for the sampling measure formulation, as well as provide an intuition into the physical assumptions encoded by the choice of df . To summarize, if we wish to describe a transient process observed by sequencing, we need to begin with one of two assumptions. On one hand, the transient gene program may be triggered externally, but fully executed within the cell, allowing us to define “time-desynchronized” sampling from a PFR, CSTR, or BR. On the other hand, it may be controlled by external factors, allowing us to define “space-desynchronized” sampling from a PFR. However, considerable further theoretical and experimental work is necessary to understand when such assumptions are justified.

Simulation-tractable models. The second class includes models which can be cast into a Markovian form and simulated [150], or partially solved using matrix algorithms [151].

Although we may be able to write down a CME and its formal solution, numerical evaluation typically requires considerable computational expense, and the appropriate inference strategies are unclear.

We abstract away the mechanistic details of regulation. However, the variation in transcriptional parameters should be understood as the outcome of a regulatory process that is tightly controlled in time (to justify step changes in rates) and concentration (to justify deterministic parameter values). It is possible to explicitly represent regulatory networks or feedback, in line with *dyngen* [14] and standard systems biology studies [152]. Further, we assume all reactions are zeroth- or first-order, although it is possible to model the kinetics of enzyme binding [153]. Unfortunately, such systems are intractable in the current context, due to high dimensionality, mathematical challenges, lack of protein data, and complexity of regulatory networks on a genome-wide scale.

We described a method to model sequencing as a random process, if the fairly restrictive conditions of independence and identical distribution hold. The distributional assumption can be relaxed for a particular gene and molecular species [130, 148]. It is also possible to write down—but challenging to fit—a hierarchical model, such that each cell’s sequencing model parameters are random. However, it does not yet appear feasible to solve and fit models which impose coupling between cells and genes through the compositional nature of the sequencing process [109]. At the same time, generating realizations from such models is trivial, and amounts to sampling without replacement. Analogously, it is straightforward to set up a model with “cell size” variation, such that each cell separately regulates its average transcription level for the entire genome [154]. It is unclear how such models should be fit, although neural [155] and perturbative [154] approaches have seen use.

We have restricted our discussion to regulatory modulation in terminally differentiated cells. In a variety of contexts, cell division, which involves molecule partitioning and transcriptional dosage compensation [93, 156], plays a significant role in differentiation, and it is inappropriate to omit its effects. Considerable literature exists on cell cycle models [93, 157, 158], but they are fairly challenging to solve, and the appropriate way to integrate them with occupation measures is unclear at this time.

For mathematical simplicity, we have assumed mRNA molecules have no internal structure. This model implies that transcription is binary, rather than stepwise: when a transcriptional

event occurs, a new molecule springs into existence, fully formed. In actuality, mRNA molecules are polymers, and their production often takes considerable time [159]; on an even finer level of detail, transcriptional elongation is neither infinitely fast nor constant, and exhibits pauses or “pile-ups,” which may themselves have a regulatory role [160]. Adding yet more complexity, splicing may occur co-transcriptionally or post-transcriptionally [161], with deterministic or stochastic intron removal order [80], making the categories of “spliced” and “unspliced” molecules even less tenable. Certain models of elongation can be solved [61, 162] and simulated [163], but rapidly become mathematically intractable. In sum, characterizing the amount of information about granular *in vivo* processes which can be obtained from potentially incomplete annotations and potentially ambiguous transcript equivalence class data [164] is a considerable undertaking, but will be essential for scRNA-seq analysis in a longer perspective.

Currently intractable models. We omit several types of phenomena because they are challenging to formalize using the standard CME framework. Thus far, we have assumed cells are biologically independent under some measure, although weak coupling may take place due to sequencing. However, cells interact with their neighbors in living tissues, leading to complex and tightly controlled patterns of spatial expression. These interactions are particularly important in biomedical research, and are typically simulated through agent-based modeling (ABM) techniques [165]. However, even the simplest models of interaction, such as the Ising model in solid-state physics, are challenging to treat and do not typically afford transient solutions [166].

Finally, CME models typically treat the cell as spatially homogeneous, occasionally with Markovian export from a spatially homogeneous nucleus [106]. This assumption simplifies the construction of Markov chains, and amounts to assuming that the diffusion timescale is considerably faster than the reaction timescale. However, as the intracellular medium is neither a stirred tank nor a dilute gas, diffusion may act as a limiting step. Spatial heterogeneity within the cell has been treated in recent whole-simulations, using classical mechanics [167] as well as reaction-diffusion master equations on a lattice [168]. However, such systems are not amenable to analysis and require pre-defining the cell geometry.

Count processing

Selecting and solving a model allows for a quantitative comparison of the predictions of the inference process to a meaningful baseline. We use the model solved in Eq 18 with simple impulse dynamics: $\alpha(t)$ is piecewise constant over a finite time horizon, with its dynamics given by a single positive or negative pulse.

$$\begin{aligned} \alpha(t) = & \quad \alpha_1 \text{ if } t \in (0, \tau_1), \\ & \quad \alpha_2 \text{ if } t \in (\tau_1, \tau_2), \\ & \quad \alpha_1 \text{ if } t \in (\tau_2, T), \end{aligned} \tag{21}$$

and $\alpha_1 \neq \alpha_2$. Different genes have different α , but synchronization is enforced: all genes switch at identical times τ_1, τ_2 . Physically, this description corresponds to a perturbation being applied to, then removed from the cells, resulting in the modulation shown in Fig 9a. We simulated this model using the procedure outlined under “Transient constitutive model: Perturbation and reversion.”

We investigated this class of phenomenological models for transcription variation as opposed to fully mechanistic descriptions (e.g., *dyngen* [14]) for three reasons. First, they provide the best-case baseline scenario for the validation of RNA velocity, as the inference

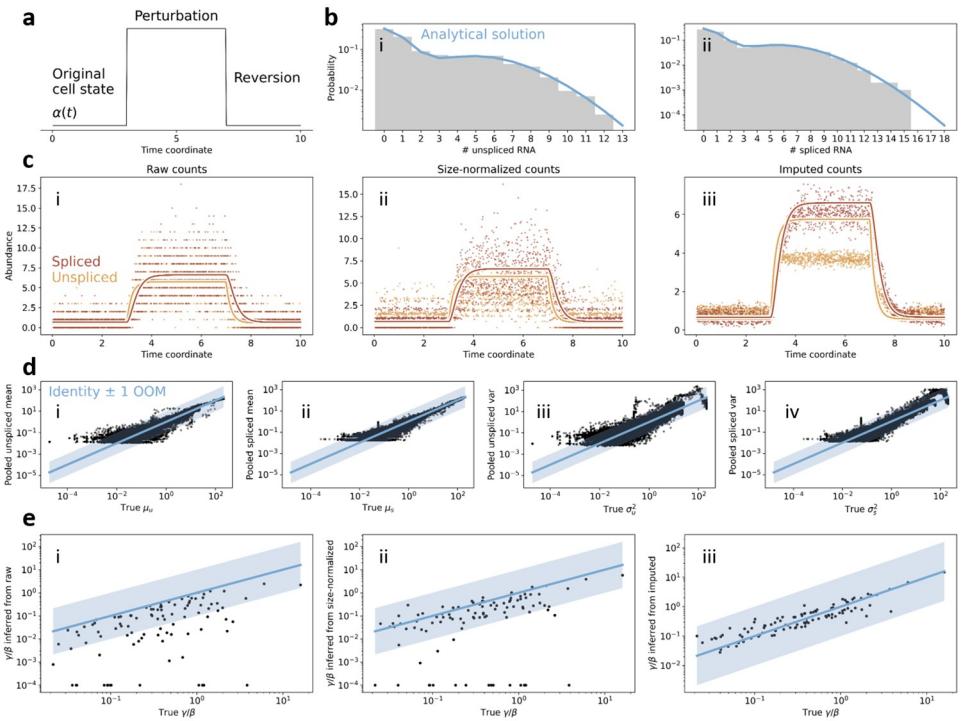


Fig 9. The RNA velocity count processing and inference workflow, applied to data generated by stochastic simulation. **a.** Schematic of the impulse model of gene modulation. **b.** Demonstration of the concordance between simulation and analytical solution for the occupation measure. **i.**: nascent mRNA counts; **ii.**: mature mRNA counts (gray: simulation; blue: occupation measure). **c.** Smoothing and imputation introduce distortions into the data. **i.**: raw data; **ii.**: data normalized to total counts; **iii.**: imputed data (points: raw or processed observations; lines: ground truth averages μ_s and μ_u ; red: spliced; yellow: unspliced). **d.** Local averages obtained by imputation are not interpretable as instantaneous averages. **i.**: mean unspliced; **ii.**: mean spliced; **iii.**: variance unspliced; **iv.**: variance spliced (black points: true moment vs. pooled moment; blue line: identity; blue region: factors of ten around identity). **e.** Smoothing and imputation improve the inference on extrema. **i.**: moment-based inference from raw data; **ii.**: extremal inference from normalized data; **iii.**: extremal inference from imputed data (black points: true vs. inferred values of γ/β ; blue line: identity; blue region: factors of ten around identity). The palette used is derived from dutchmasters by EdwinTh.

<https://doi.org/10.1371/journal.pcbi.1010492.g009>

procedure is predicated on this model. For example, Eq 21 is the pulse stimulus model proposed by La Manno et al. [1]. Second, they offer the analytical solutions discussed in “Occupation measures provide a theoretical framework for scRNA-seq,” whereas more complicated schema do not. Finally, mechanistic models of regulation are underdetermined relative to scRNA-seq data, as they rely on signal transfer through proteins. We anticipate that comprehensive future models will introduce further mechanistic details, as in the causal schema proposed in Fig 4 of [21]. However, here we abstract away the specific details of how this perturbation is effected, and focus on its effects on the observable transcriptome.

Predictably, the marginal distributions of the simulated data were bimodal, and matched the analytical solutions for the occupation measure (Fig 9b). The count processing workflow distorted the data in complex, nonlinear ways. Compared to the raw data, imputation did lower dispersion (Fig 9c). However, we did not find support for treating the imputed data as merely $\mu_u(t)$ and $\mu_s(t)$ with a Gaussian perturbation. For any particular gene, the relationship between the two was nontrivial and exhibited biases. The sample gene demonstrated in the figure produced an approximately smooth trace that did not resemble the true process average. At best, the imputed estimate appeared to be unbiased over the entire dataset, and tended to fall within of factor of ten of the true mean (Fig 9di and 9dii). Interpreting the local observed

variance as the true process moment σ^2 is even less appropriate: the relationship between local and true variance exhibited unintuitive, nonlinear effects ([Fig 9diii](#) and [9div](#)).

Normalization for the total number of UMI counts per cell is standard, and La Manno et al. do use it in their validation of *velocyto* (p. 6 in SN2 of [1]). On the other hand, it could be argued that such scaling is an *ad hoc* procedure intended to eliminate underlying variation in UMI counts due to differences in which genes are expressed in cells, or technical noise. As such, it may be inappropriate to apply it to simulated datasets that do not have these phenomena. We repeated the analysis without scaling counts by total molecule counts per cell in Fig E in [S1 Text](#). This means that raw counts were pooled, using the k nearest neighbors obtained from PCA, which was itself computed from the log-transformed raw, spliced counts. Qualitatively, the gene-specific imputed trajectories did not exhibit the severe biases of [Fig 9](#). However, they still produced errors in the transient regimes of interest, and scale- and species-dependent errors elsewhere, violating the assumptions of the *scVelo* “dynamical” inference procedure (as outlined in “Inference” under “Logic and methodology”). Furthermore, the relationship between true moments and pooled moments was nearly identical in [Fig 9d](#) and Fig Ed in [S1 Text](#).

We can conclude that even the best-case scenario, with matching model assumptions and no technical noise, does not justify using imputed data in place of the process average: the imputation procedure is circular, and can give rise to biases (as in Section B in [S1 Text](#)). Although these biases may cancel on average, this cannot be relied on for any particular gene. Therefore, instead of smoothing, which is fundamentally unstable and challenging to validate, we recommend explicitly constructing and solving error models (as in [130]). Such an approach provides insights into the system biophysics and enables the quantification of uncertainty through standard statistical methods.

Inference from occupation measure data

With the groundwork outlined immediately above, one can begin to tackle the problem of inference from sequencing data. We start by carefully inspecting the models set up in the velocity publications, enumerating their assumptions about the transcriptional processes, and writing down their formal solutions without making any further approximations. The resulting “inverse” problem of identifying biological parameters from data is currently intractable. We are unaware of any studies which treat this problem in full generality, so new inferential strategies need to be developed. However, it is instructive to write down the exact forms, as they clarify the origin of the complexity and may suggest satisfactory approximations.

First, we define the global structure of the transient system, which represents the parameters shared between different genes. This global structure is encoded in the vector θ_G , which we assume to be finite-dimensional. For example, if the differentiation process is linear and deterministic, with no branching, with K distinct cell types, θ_G gives the times τ_k of the cell type transitions (where $\tau_0 := 0$ and $\tau_K := T$, which we set to 1 with no loss of generality). On the other hand, if it is non-deterministic, with diverging cell fates, θ_G can encode topologies and fates’ probabilities (such as w_A from [Eq 20](#)). Conversely, a unipotent differentiation trajectory can be encoded by setting $w_A = 1$, i.e., simpler topologies are degenerate cases of more complex topologies. Finally, this vector may also encode non-biological phenomena, such as batch-specific technical noise parameters [130] and the specific form of the generalized occupation measure f .

In addition to θ_G , the system also involves vectors $\theta_j, j \in \{1, \dots, N\}$, where j ranges over the genes. The θ_j are gene-specific parameter vectors that parameterize physiological processes, such as transcriptional bursting, splicing, and degradation, as well as initial conditions. For

simplicity, we make two crucial assumptions, consistent with the previous velocity publications. First, we suppose the transcriptional parameters are piecewise constant throughout the process, and all other parameters are strictly constant. Second, we suppose that different genes' transcription and processing reactions are statistically independent. We define the full parameter set as $\Theta := \{\theta_1, \dots, \theta_N, \theta_G\}$. For completeness, we note that in previous publications the cell type transition times are gene-specific parameters, i.e., $\tau_{jk} \in \theta_j$, but since we assume that all genes switch at identical times they are global parameters, i.e., $\tau_k \in \theta_G$, in the model we propose here.

Finally, we formalize the data variables. The full dataset is given by the data matrix \mathcal{D} , with cell-specific arrays \mathcal{D}_i and gene-specific arrays \mathcal{D}_j . Thus, \mathcal{D}_{ij} reports the spliced and unspliced counts for gene j in cell i . Under this model and the assumption of gene independence, the following equation gives the likelihood of a particular cell's observation:

$$\mathcal{L}(\Theta; \mathcal{D}_i) = P(\mathcal{D}_i; \Theta) = \int P(\mathcal{D}_i; t, \Theta) df(t; \Theta) = \int \prod_j P(\mathcal{D}_{ij}; t, \theta_j, \theta_G) df(t; \theta_G). \quad (22)$$

The assumption of cell independence suggest the following total likelihood:

$$\mathcal{L}(\Theta; \mathcal{D}) = \prod_i \mathcal{L}_i(\Theta; \mathcal{D}_i) = \prod_i \int \prod_j P(\mathcal{D}_{ij}; t, \theta_j, \theta_G) df(t; \theta_G). \quad (23)$$

To fully characterize this system under the foregoing model assumptions, we need to optimize the likelihood with respect to the parameters. This is generally intractable; even evaluating Eq 23 can be non-trivial. However, we can make a series of simplifying approximations.

A combinatorial optimization approach to inference

Ultimately, we would like to understand the behavior of Eq 23 across the entire domain of parameters, compute the maximum likelihood estimate (MLE) of Θ , and characterize its stability by calculating its confidence region. However, this is not yet feasible. Therefore, we restrict the discussion to writing down a formula for the MLE that can be treated using standard algorithms.

Our strategy is to exploit the “latent” distribution of cell-specific process times, condition on this distribution, and find an approximate MLE. Assuming that cells are observed uniformly across process time, i.e., $df = dt$, the latent time of each cell i is given by $t_i \in (0, 1)$. These times are almost surely distinct, and induce a cell ranking in order of increasing process time. This ranking is unknown and has to be inferred from the data.

Assume, for the moment, that the ranking is known and given by σ , a permutation of the M cell indices $\{1, 2, \dots, M - 1, M\}$ corresponding to their process time order statistics. Given a cell's order statistic σ_i , we can use f to estimate its latent time t_i , which is distributed according to a rather complex multivariate Beta distribution [169]: even if each cell's rank in the total order is known, we have to account for the uncertainty in process time. However, we can exploit the fact that this uncertainty decreases as the number of cells grows. The marginal order statistics are distributed according to $Beta(\sigma_i, M + 1 - \sigma_i)$, a random variable with law $f_{U_{(i)}}(t)$ and the following variance:

$$\mathbb{V}[U_{(i)}] = \frac{\sigma_i(M + 1 - \sigma_i)}{(M + 1)^2(M + 2)}, \quad (24)$$

which tends to zero with the rate M^{-1} . Therefore, we find that $t_i \approx \mathbb{E}[U_{(i)}] = \frac{\sigma_i}{M+1}$.

Thus, if we have enough cells, and know their process time ordering, we can exploit the fact that $f_{U_{(i)}}(t)$ converges to a Dirac delta functional:

$$\mathcal{L}(\Theta; \mathcal{D}_i, \sigma_i) = \int \prod_j P(\mathcal{D}_{ij}; t, \theta_j, \theta_G) f_{U_{(i)}}(t) dt \approx \prod_j P(\mathcal{D}_{ij}; t_i, \theta_j, \theta_G). \quad (25)$$

This amounts to using a plug-in point estimate to compute the likelihood of a cell's data. The same approach can be applied to each cell in turn:

$$\mathcal{L}(\Theta; \mathcal{D}, \sigma) \approx \prod_i \prod_j P(\mathcal{D}_{ij}; t_i, \theta_j, \theta_G) = \prod_j \prod_i P(\mathcal{D}_{ij}; t_i, \theta_j, \theta_G). \quad (26)$$

However, optimizing this quantity, even when conditioning on σ , is impractical because it requires simultaneously optimizing (potentially thousands of) gene-specific parameters along with the (relatively few) global parameters. The interchange of product operations is helpful because it allows us to write down a more tractable loss function for the MLE, which exploits the conditional separability of θ_j :

$$\hat{\Theta}|\sigma = \operatorname{argmax}_{\theta_G} \sum_j \operatorname{argmax}_{\theta_j} \sum_i \ln P(\mathcal{D}_{ij}; t_i, \theta_j, \theta_G). \quad (27)$$

Optimizing this quantity over σ guarantees to return the global MLE:

$$\hat{\Theta} = \operatorname{argmax}_{\sigma} \operatorname{argmax}_{\theta_G} \sum_j \operatorname{argmax}_{\theta_j} \sum_i \ln P(\mathcal{D}_{ij}; t_i, \theta_j, \theta_G). \quad (28)$$

The parameters $\theta_1, \dots, \theta_N, \theta_G$ can be found by standard continuous optimization methods, but estimating σ requires a combinatorial optimization, namely finding an optimal traversal path between cells. In other words, even this approximate approach requires solving the problem of pseudotime inference, which produces a one-dimensional ordering of cells [170]. However, unlike standard pseudotime inference, which describes a set of purely phenomenological relationships informed by proximity between cell expression states, the current theoretical framework endows the solution with a concrete biological interpretation, which is informed by a specific microscopic model of transcription.

The trajectory inference literature treats this class of problems by graph traversal algorithms, generally by constructing a minimum spanning tree or an optimal traversal on the clusters or individual cells [14, 139]. Under fairly severe modeling assumptions, which generally rely on error isotropy, the optimal traversal of cell states reduces to the traveling salesman problem (TSP) via the Hamiltonian path problem with some minor differences [132, 171, 172]. The current approach is considerably more complicated, because the weights between the “nodes”, i.e., the observed cells, cannot be generally written down in closed form, and require optimization for every σ .

Nevertheless, the specific form of the required combinatorial optimization has several useful implications for inference. It is possible to subsample or filter the data to obtain rough estimates of the parameters by sampling a subset of genes. This facilitates the estimation of θ_G , which can be reused for an entire dataset. If only a fraction of genes are systematically modulated across a trajectory, technical noise parameters within θ_G can be estimated from the far more easily tractable fits to the stationary genes. Furthermore, sampling a subset of cells enables the construction of approximate σ and estimation of θ_j . The validity of such approximations can be assessed with relatively simple controls. For example, if a best-fit “trajectory” over cells σ is as good as a random or inverted permutation, the transient model is likely overfit. Finally, existing trajectory inference methods can be exploited to obtain an ordering σ for

the purpose of testing whether it can give results consistent with the stochastic model by calculating the optimal parameters in Eq 27, plugging them into the appropriate CME, and comparing the process occupation measure to the true molecule distributions.

It is plausible that many standard trajectory inference methods can be represented as approximations to the exact solution under particular assumptions about the form of biological model and priors imposed on the trajectory structure. However, a complete discussion of the trajectory inference field is beyond the scope of this paper. Instead, we restrict ourselves to discussing how existing velocity methods, as well as certain clustering algorithms, can be represented as special cases of the exact solution in Eq 23.

Clustering as a special case

First, consider the ergodic case of f on $[0, T]$ with $T \rightarrow \infty$, and suppose there are K cell types at equilibrium. These cell types are distinguished by gene-specific parameter vectors θ_j^k , $k = 1, \dots, K$. Assume only a single RNA species per gene exists, and all genes are independently expressed in a single cell type, without yet imposing a specific biological model. This yields the likelihood

$$\mathcal{L}(\Theta; \mathcal{D}_i) = \int \prod_j P(\mathcal{D}_{ij}; \theta_j) df = \sum_k P(k) \prod_j P(\mathcal{D}_{ij}; \theta_j^k), \quad (29)$$

where $P(\mathcal{D}_{ij}; \theta_j)$ is the probability of observing the data \mathcal{D}_{ij} and $P(k) \in \theta_G$ is the probability of cell i being in cell type k . The process time t is no longer necessary, because all cell types are stationary: the likelihoods are evaluated at the ergodic limit $t = \infty$. Eq 29 amounts to saying that the likelihood of a cell's observation can be represented by using the law of total probability and conditioning on the cell type:

$$P(\mathcal{D}_i; \Theta) = \sum_k P(k) P(\mathcal{D}_i; \theta^k) = \sum_k P(k) \prod_j P(\mathcal{D}_{ij}; \theta_j^k). \quad (30)$$

To optimize this likelihood, we need to specify $P(\mathcal{D}_{ij}; \theta_j)$, which is informed by the biophysics of transcription and mathematical tractability. The log-normal distribution is particularly common: if we treat log-counts $\ln \mathcal{D}$, the law of $P(\ln \mathcal{D}_j; \theta_j)$ is Gaussian. The lognormal distribution can emerge from several hypothesis: through a common, if *ad hoc* approximation to the gamma distribution which emerges from the mesoscopic limit of the CME [147], from the exact solution of a deterministic, macroscopic model with log-normally distributed transcriptional rates [173], and by mere assertion that the negative binomial distribution is similar to the lognormal distribution, without further discussion [172].

The lognormal approximation implies that each “cell type” is essentially a high-dimensional normal distribution in logarithmic state space. This induces a set of gene- and cluster-specific log-means μ_j^k , log-standard deviations s_j^k , and a cluster assignment vector σ , such that $\sigma_i \in \{1, \dots, K\}$. The problem of characterizing cell types, i.e., fitting $P(k)$, μ_j^k , and s_j^k , and providing an optimal point estimate of σ , under this model is equivalent to using the expectation-maximization algorithm to fit a Gaussian mixture model to the logarithmic data [174]. However, some caveats deserve mention. The model choice requires careful consideration. The log-normal heuristic is incoherent with the standard velocity model, which tends to a normal distribution with equal mean and variance in the continuous limit. Furthermore, although the Gaussian mixture model formulation can be justified as an approximation of a particular class of models, it is unlikely that this approximation holds generally.

For completeness, we note that the standard alternative to Gaussian mixture model clustering is community detection based on a graph constructed by defining a neighborhood criterion among cell vectors [37]. However, it has not yet been shown that such an approach can be afforded any well-defined probabilistic meaning. The literature contains numerous assertions that a meaningful Markovian transition probability matrix can be defined on observed cell states [1, 9, 10, 24, 140]. However, the constructed Markov chains have not been demonstrated to possess any particular relationship to an actual biological process (Section C in [S1 Text](#)).

The ‘deterministic’ *velocyto* model as a special case

Strictly speaking, we only need to solve [Eq 23](#) if we want to exploit useful properties of likelihood landscapes and estimators. However, if we are willing to forgo these advantages, we can use a moment-based estimate.

The linearity of the occupation measure can be used to compute summary statistics. For example, we can treat the RNA velocity model defined in [Eq 1](#), with $\mu_u(t)$ and $\mu_s(t)$ giving the instantaneous process averages. The following relations hold at each instant t and over the entire trajectory:

$$\begin{aligned} \frac{d\mu_s(t)}{dt} &= \beta\mu_u(t) - \gamma\mu_s(t), \\ \int \frac{d\mu_s(t)}{dt} df &= \beta \int \mu_u(t) df - \gamma \int \mu_s(t) df. \end{aligned} \quad (31)$$

Each species’ mean occupation measure μ can be related to the instantaneous mean $\mu(t)$:

$$\begin{aligned} \mu &= \sum_{x=0}^{\infty} x P(x) = \sum_{x=0}^{\infty} x \int P(x; t) df \\ &= \sum_{x=0}^{\infty} \int x P(x; t) df = \int \left(\sum_{x=0}^{\infty} x P(x; t) \right) df \\ &= \int \mu(t) df. \end{aligned} \quad (32)$$

This implies the following identity:

$$\int \frac{d\mu_s(t)}{dt} df = \beta\mu_u - \gamma\mu_s, \quad (33)$$

which holds regardless of the transcriptional dynamics encoded in $\alpha(t)$.

The identity formalizes the moment-based approximation to the biological parameters: if the left-hand side (the net velocity of the process) is sufficiently close to zero, the right-hand side gives an estimate for γ/β . Conversely, if this condition is violated, a naïve fit based on the moments (equivalently, a least-squares fit with zero intercept) will be biased by transient contributions, motivating the use of the extrema fitting procedure (as in Fig 2 in SN2 of [1]).

We can investigate the behavior of the net velocity in a simple model system. Suppose $\beta = 1$, $df = T^{-1}dt$, and α is piecewise constant. We define α_k , with $k \in \{1, \dots, K\}$, constant on each interval $I_k \in [\tau_{k-1}, \tau_k]$; the bounds are $\tau_0 := 0$ and $\tau_K := T$. We define the length of an interval

as $\Delta_k = \tau_k - \tau_{k-1}$. The following equations hold for every interval:

$$\begin{aligned} \mu_u(t) &= \alpha_k(1 - e^{-t}) + u_0 e^{-t}, \\ \mu_s(t) &= \frac{e^{-\gamma t}}{\gamma(\gamma - 1)} \left(\alpha_k(\gamma - 1)e^{\gamma t} - \alpha_k \gamma e^{(\gamma-1)t} + \alpha_k + (\gamma - 1)\gamma s_0 + \gamma u_0(e^{(\gamma-1)t} - 1) \right), \\ \int_{I_k} \frac{d\mu_s(t)}{dt} df &= \frac{s_0}{\Delta_k} (e^{-\gamma \Delta_k} - 1) \\ &\quad + \frac{1}{\Delta_k \gamma(\gamma - 1)} (\gamma(\alpha_i[1 - e^{-\Delta_k}] + u_0[e^{-\Delta_k} - e^{-\gamma \Delta_k}]) + \alpha_i(e^{-\gamma \Delta_k} - 1)). \end{aligned} \tag{34}$$

In each interval I_k , the integral approaches zero as Δ_k grows. This result has a qualitative interpretation: as interval duration grows, the process settles into its ergodic equilibrium attractor, and that attractor provides an effective estimator of γ . On the other hand, if the interval is short-lived relative to mRNA lifetime, the integral is dominated by the initial condition, and the system is largely out of equilibrium.

In practice, this means that the degradation rate is identifiable through moments only if the lifetimes of the mRNA species are short relative to the interval lengths, i.e., the net velocity is low enough. On the other hand, if the lifetimes are too short, the transient regimes are sparse and steady states are approached rapidly, giving no information about dynamics (and reducing the problem to the formulation in Eq 29). The quantile fit procedure is simply a heuristic method to winnow the data for near-equilibrium populations under the informal prior that these populations are present in the data. Unfortunately, this approach is subject to the usual pitfall of moment-based methods: if the prior is wrong, there is no easy way to identify its failure.

We have omitted the discussion of normalization and imputation, as they are not amenable to analysis. However, their impact can be evaluated by generating data from the model and processing it with the *velocyto* workflow. The intuition outlined above concords with our simulations: in Fig 9e, we fit the simple model introduced in “Count processing” under “Prospects and solutions” using three different methods. In i., we use simple linear regression on the raw data; in ii. and iii. we apply a quantile fit to normalized and imputed data respectively. In spite of the distortions in the overall phase portrait (Fig 9c), the extrema are stable enough under imputation to generate fairly reliable estimates of γ/β , up to roughly an order of magnitude, whereas the ratio of averages is significantly less precise. This is consistent with the performance reported in Fig 5 (as in the $k = 0$ case, which uses linear regression on the entire dataset).

In light of the formalization, the fitting procedure is contingent rather than stable, necessitating careful study of limitations. Using the average of the full occupation measure is contingent on the net velocity being near zero (Δ_k sufficiently large for all k). Using the average of the extrema is contingent on those extrema having equilibrated (Δ_k sufficiently large for k with largest and smallest α_k). Furthermore, it provides no information about the relative timescales β_j of different genes.

The “stochastic” model, which was introduced in *scVelo*, is practically identical, but exploits additional information from the second moments of the extremal observations. This approach inherits the same issues, such as the reliance on the existence of extrema, and omission of β_j , and introduces new ones, such as the assumption of identical error terms for first and second moments and the inference of error covariance parameters. In principle, further investigation may characterize whether these issues improve or worsen moment-based inference. However,

we suggest that these details are marginal compared to the more fundamental limitations, as well as the discrepancies observed in simulation ([Fig 9diii and 9div](#)).

The ‘dynamical’ scVelo model as a special case

The modeling approach we have presented can be used to contextualize part of the “dynamical” algorithm proposed in scVelo. First, assume that cell type transition times $\tau_{jk} \in \theta_j$, i.e., no global parameters shared by multiple genes exist ($\theta_G = \emptyset$). This reduces [Eq 23](#) to the following form:

$$\mathcal{L}(\Theta; \mathcal{D}) = \prod_i \int \prod_j P(\mathcal{D}_{ij}; t, \theta_j) df(t). \quad (35)$$

Omitting the uncertainty represented in the integral by assigning a time t_i to each cell we obtain:

$$\mathcal{L}(\Theta; \mathcal{D}) \approx \prod_i \prod_j P(\mathcal{D}_{ij}; t_i, \theta_j). \quad (36)$$

Since time assignments t_i are now deterministic, rather than probabilistic, likelihood landscapes become inaccessible. In principle, finding the maximum of [Eq 36](#) can still provide a point estimate of Θ , although this may not be practical: now, f has to be inferred empirically by fitting t_i . Applying the logarithm, we get

$$\ln \mathcal{L}(\Theta; \mathcal{D}) \approx \sum_i \sum_j \ln P(\mathcal{D}_{ij}; t_i, \theta_j) = \sum_j \sum_i \ln P(\mathcal{D}_{ij}; t_i, \theta_j). \quad (37)$$

Using a Gaussian kernel centered on μ_u and μ_s , with standard deviation s_j for both species, as an approximation to the likelihood, we can interpret P as a probability density function:

$$P(\mathcal{D}_{ij}; t_i, \theta_j) = \frac{1}{2\pi s_j^2} \exp\left(-\frac{1}{2s_j^2} \left[(\mathcal{D}_{ij}^u - \mu_u(t_i; \theta_j))^2 + (\mathcal{D}_{ij}^s - \mu_s(t_i; \theta_j))^2 \right]\right). \quad (38)$$

The question of what this kernel means, i.e., what biophysical phenomena it models, is subtler than it appears. On the one hand, there are certain regimes where stochastic spliced and unspliced counts are approximately distributed about the true averages μ_u and μ_s according to normal laws, such as in the high-concentration limit of certain jump processes explored by Van Kampen [[144](#)]. However, this limit yields time-dependent and concentration-dependent s_j , incompatible with [Eq 38](#) and the “stochastic” model described in the scVelo manuscript. Instead, this form presupposes μ_u and μ_s are the true deterministic amounts of mRNA, corrupted by Gaussian isotropic error without an explicitly named source. This model exemplifies the signal processing paradigm, which attempts to identify an underlying “signal” by regressing out incidental Gaussian “noise,” with various levels of biological justification [[175–177](#)].

For a particular gene, the log-likelihood of [Eq 38](#) takes the following form:

$$\begin{aligned} \ln P(\mathcal{D}_{ij}; t_i, \theta_j) &= -\ln 2\pi s_j^2 - \frac{1}{2s_j^2} \left[(\mathcal{D}_{ij}^u - \mu_u(t_i; \theta_j))^2 + (\mathcal{D}_{ij}^s - \mu_s(t_i; \theta_j))^2 \right] \\ &= -\ln 2\pi s_j^2 - \frac{1}{2s_j^2} \| \mathcal{D}_{ij} - \mu(t_i; \theta_j) \|^2, \end{aligned} \quad (39)$$

i.e., we can use a simple two-dimensional Euclidean norm to estimate the log-likelihood of the observation. For M independent observations of gene j with identical parameters θ_j but distinct

times t_i , we find that

$$\begin{aligned} \ln P(\mathcal{D}_j; \theta_j) &= \sum_{i=1}^M \ln P(\mathcal{D}_{ij}, t_i; \theta_j) \\ &= -M \ln 2\pi s_j^2 - \frac{1}{2s_j^2} \sum_{i=1}^M \| \mathcal{D}_{ij} - \mu(t_i; \theta_j) \|^2. \end{aligned} \quad (40)$$

The optimum of this function is invariant under scaling, so we can work with the normalized negative log-likelihood:

$$-\frac{1}{M} \ln P(\mathcal{D}_j; \theta_j) = \ell(\theta_j; \mathcal{D}_j) = \ln 2\pi s_j^2 + \frac{1}{2s_j^2 M} \sum_{i=1}^M \| \mathcal{D}_{ij} - \mu(t_i; \theta_j) \|^2. \quad (41)$$

This contradicts Equations 7–8 of [3], which use a univariate, rather than bivariate, Gaussian error term. On the other hand, the actual implementation appears to use separate s_j for the two species, computed from the extremal points, and combine them in an *ad hoc* fashion.

In principle, solving Eq 37 with the likelihood indicated in 39, i.e., iteratively inferring t_i and θ_j , yields a coherent maximum likelihood estimate of the system parameters. However, the method goes one step further, essentially taking Eq 37 and rewriting it in terms of the negative log-likelihood:

$$-\frac{1}{M} \ln \mathcal{L}(\Theta; \mathcal{D}) = \ell(\Theta; \mathcal{D}) \stackrel{?}{=} \sum_j \ell(\theta_j; \mathcal{D}_j). \quad (42)$$

This approach posits that likelihood optimization over N genes be split into N independent problems, which can be parallelized. However, it is incorrect, as the times t_i are incoherent between different genes j , and the results are uninterpretable. This issue has been tacitly acknowledged, e.g., in a *post hoc* approach adopted to make times “agree” in *scVelo* [3] and a similar proposal by Li et al. [24], although without any rigorous justification.

The actual implementation does not use the raw data \mathcal{D}_{ij} . Rather, it uses a normalized and imputed version of the data. Again, the effect of these transformations is challenging to characterize analytically. However, the simulations shown in Fig 9c and 9d and Fig E in S1 Text suggest that there is no compelling reason to believe the imputed data reliably estimate the underlying averages μ_u and μ_s for a stochastic system out of equilibrium. Furthermore, the noise-corrupted deterministic model used in the likelihood computation is biologically implausible.

Prospects for inferential procedures

In “Occupation measures provide a theoretical framework for scRNA-seq,” we have presented a framework for the description of transient stochastic systems. This framework is versatile enough to describe a range of problems in single-cell sequencing, from clustering to trajectory inference. The methods presented in previous RNA velocity publications are best understood as approximations to exact solutions under fairly strong, informal priors about the process biophysics. The *velocyto* algorithm uses a moment approximation, which assumes the system has effectively equilibrated. The *scVelo* algorithm uses a Gaussian-perturbed ODE model, which assumes the mRNA counts do not have any intrinsic noise, only isotropic measurement noise. The latter yields considerably more information from the data, but imposes considerably stronger assumptions, making the obtained information essentially uninterpretable.

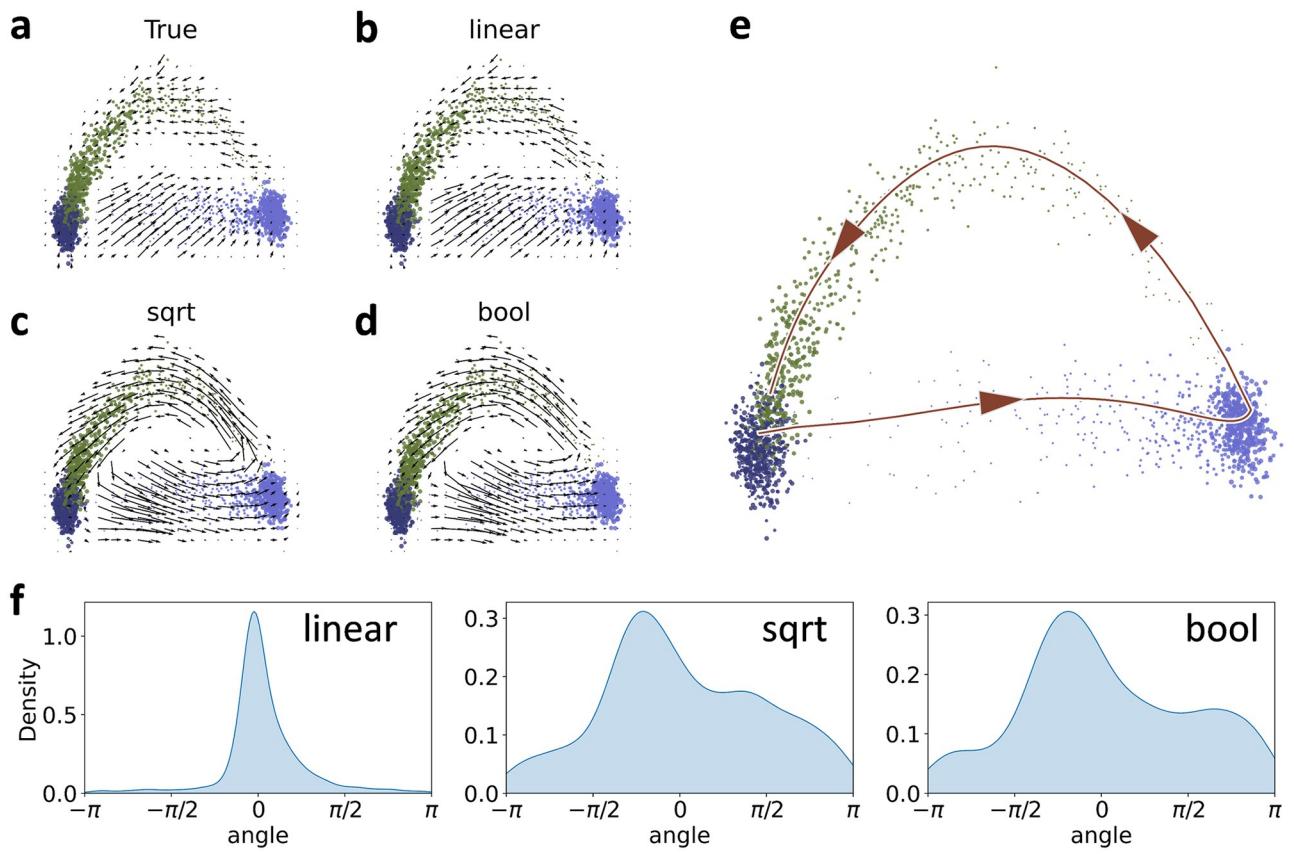


Fig 10. Performance of cell and velocity embeddings on simulated data, compared to ground truth velocity directions. **a.** Linear PCA embedding of ground truth velocities. **b.** Linear PCA embedding of inferred velocities. **c.** Nonlinear PCA embedding of inferred velocities. **d.** Nonlinear, Boolean PCA embedding of inferred velocities. **e.** Embedding of ground truth principal curve; trajectory directions displayed to guide the eye. **f.** Distribution of cell-specific angle deviations relative to ground truth velocity directions.

<https://doi.org/10.1371/journal.pcbi.1010492.g010>

However, our formalization immediately presents options for likelihood-based parameter estimation. We introduced an approximate method that does maintain cell identities and motivated it using the properties of order statistics. This method is challenging because it requires a fairly involved combinatorial optimization. However, it does lend itself to developing further approximations, and provides routes for falsifying hypotheses. We believe that such biophysical models, amenable to approximation and testing, are crucial for the future interpretation of dynamics predictions from sequencing data.

Embedding

To complement the internal controls discussed in “Embedding” under “Logic and methodology,” we performed a set of comparisons with data simulated with no unknown sources of noise. We embedded a simulation of the system introduced in “Count processing” under “Prospects and solutions” and illustrated in Fig 9 into a two-dimensional principal component space. The results are shown in Fig 10. Even the ground truth velocity directions (Fig 10a) only retained a small amount of information after the transition from 100 dimensions to two. This experiment provides us with the answer to the second question in “RNA velocity biophysics”: even if we have “true” velocity directions, they only contain a limited amount of highly local information. As expected from the fair performance in Fig 9eiii, the inferred linear embedding

([Fig 10b](#)) was globally and locally ([Fig 10f](#)) faithful: the model precisely matches the assumptions of the parameter inference workflow. However, the estimates were rapidly distorted upon applying the nonlinear embedding procedure ([Fig 10c](#)), rotating many cell-specific directions and suggesting transitions from the green reverting population to the light blue perturbed population, whereas, the true trajectory is from light blue to green. The results of the Boolean procedure were slightly more faithful to the linear projection ([Fig 10f](#)) but otherwise qualitatively similar ([Fig 10d](#)). This is the method's best-case performance.

Even in the PCA projection, the performance of the nonlinear velocity embedding leaves much to be desired: the procedure is biophysically uninterpretable, discards the vast majority of information, and risks failure when model assumptions are violated. For example, it can generate false positive velocity fields when the ground truth is completely static ([Fig F in S1 Text](#), as simulated using the procedure under "Steady-state bursty model"); even directly inspecting the phase plots may be insufficient diagnose to this problem (e.g., compare [Fig G in S1 Text](#) to Extended Data Figs 6c and 7c of [1] and Figs 2c and 3g of [3]).

The nonlinear, non-deterministic embeddings ubiquitous in the analysis of scRNA-seq data degrade the performance further. In [Fig 11](#), we embedded a system with three potential terminal states, generated by the simulation procedure described in "Transient constitutive model: Multipotent differentiation." Cell projection into PCA appeared to conflate two of the branches; the nonlinear embeddings effaced causal relationships altogether. As before, the arrows were broadly coherent whether or not they included quantitative information. Finally, as described under "Background," the embedding procedure has previously demonstrated catastrophic failure to capture known dynamics in biological datasets [5, 9, 10, 21, 29]. Therefore, although embeddings are qualitatively appealing, they are unstable, challenging to validate, and harbor intrinsic global- and local-scale pitfalls that arise even in simple scenarios.

Nevertheless, some human-interpretable visualization is desirable. In light of the frustrating dearth of theory and interpretability for the commonly used embedding procedures, we note that the stochastic formulation we have presented can be used to speculate about more rigorous and stable embedding methods that would use rather than discard quantitative information. Instead of individual cells, which inevitably exhibit noise, we suggest constructing and emphasizing the underlying graph governing parameter values (as in, e.g., [Fig 1F of \[140\]](#)). Alternatively, since the current low-dimensional embeddings are used to support claims about presence of *a priori* human-interpretable features such as equilibrium cell types, limit cycles, and transient differentiation trajectories, it may be better to fit a hierarchical model consisting of those features and reporting the best-fit model. For example, if the goal is to cluster data, then it makes sense to fit [Eq 29](#). On the other hand, if the goal is an elucidation of tree-like differentiation trajectories, it may be better to incrementally grow a trajectory mixture model until its complexity outweighs its likelihood per a statistical information criterion. Formally, this would correspond to optimizing the likelihood of samples from analogs of [Eq 20](#). If a method has succeeded in inferring the underlying topology and dynamics, a meaningful and well-defined principal curve induced by the underlying mechanism, as shown in [Fig 10e](#) and the PCA in [Fig 11](#), could be plotted.

Results & discussion

Summary

The two main steps in RNA velocity, namely the model estimation and embedding, originate from different approaches to data analysis that can be at odds with each other. The count processing and inference steps, which comprise the model estimation procedure, serve to identify parameters for a transcription model under some fairly strong assumptions, such as

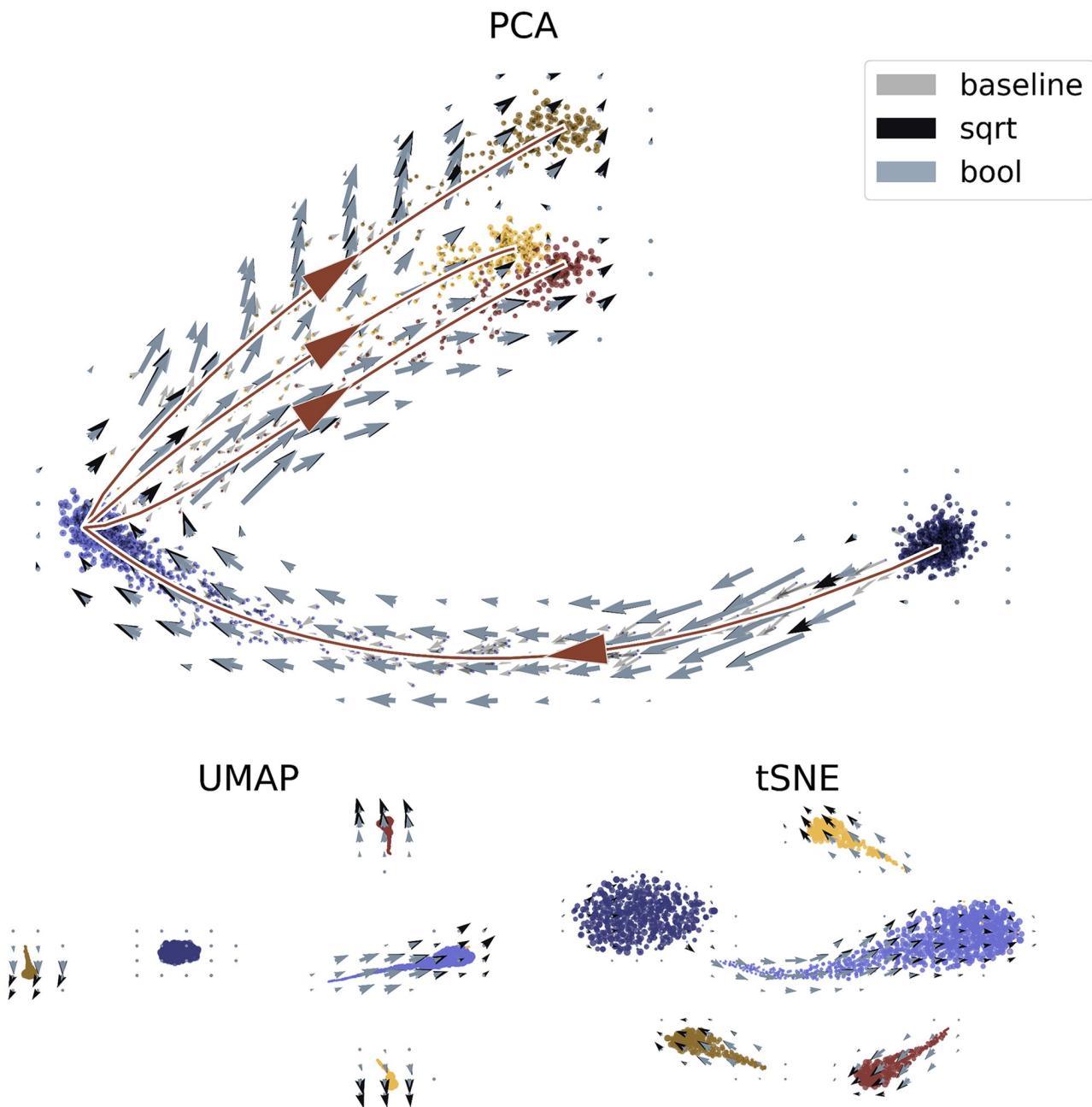


Fig 11. Performance of cell and velocity embeddings on simulated data, compared to ground truth principal curve. Top: PCA embedding with linear baseline and nonlinear aggregated velocity directions, as well as ground truth principal curve; trajectory directions displayed to guide the eye. Bottom: UMAP and *t*-SNE embeddings with nonlinear velocity projections.

<https://doi.org/10.1371/journal.pcbi.1010492.g011>

constitutive production and approximately Gaussian noise. This procedure can be treated as an informal approximation of a method to solve a system implied by the simulation design in the original publication. However, as we have seen, this system abstracts away many aspects of the technical artifacts present in single-cell RNA sequencing, and the transcriptional dynamics that drive the molecular biology of cells. The embedding processes used, which are entirely *ad hoc*, discard nearly all of the quantitative information, and can occasionally fail. Particularly

problematic is that the failures, when they occur, are difficult to identify. Moreover, failure may result from many problems, including overlaps in the embedding and erroneous clustering. Such problems may be mitigated or exacerbated by tuning hyperparameters. These challenges, contradictions, and the assumptions inherent in the many choices that are made for each of the steps, have not been previously characterized in full detail, and they add up to a mixed picture. On the one hand, at least in some simulated cases, the RNA velocity method does work, and the latent signal is strong enough to capture broad trends. On the other hand, catastrophic failure can lurk at any step of the velocity workflow, and there are no theorems to alert users to failure modes, or to diagnose or delimit the extent of failures. Instability and reliance on user-tuned hyperparameters are not grounds for abandoning the method; the same problems crop up with kernel density estimation, *k*-means clustering, histogram binning, time series smoothing, and many other analysis tasks. However, the tendency to compensate for lack of theorems with more *ad hoc* filters and more sophisticated modeling that requires optimization of neighborhood sizes, normalization procedures, and thresholds only exacerbates the problems.

The mathematical foundations of stochastic biophysics have been studied for several decades; they are well-understood, and amenable to generalizations and approximations. The chemical master equation allows for the elucidation of technical noise [130], and the quantitative exploration of transcriptional regulation [147] and splicing [81]. As discussed in the section “Prospects and solutions,” the same modeling framework can be used to describe general classes of differentiation processes. Rather than starting with heuristics and then seeking to unravel their meaning, in this approach one begins by motivating, defining, and solving a general system, and only subsequently deriving approximations and statistical summaries. These can range from simple moment expressions to low-dimensional principal curves as illustrated in Fig 11. Furthermore, with such an approach, one can leverage the machinery of Bayesian inference to directly fit full distributions, with the advantages of interpretability and statistical robustness. This highlights that the primary challenge in RNA velocity is not its extension via additional heuristics, but rather the development of tractable inference procedures.

Proposals

We conclude with a summary of the main steps of an RNA velocity workflow, along with some insights and proposals that emerged from our work:

Pre-processing. As demonstrated in “Pre-processing” under “Logic and methodology,” the specific choice of processing software does produce discrepancies in results, although the small, and largely arbitrary, variations in assignment rules do not provide compelling reasons to select one method over another, pending the development of more detailed splicing network definitions. There are, however, substantial processing time differences [27, 30] that can affect reproducibility of results, leading us to prefer fast pseudoalignment methods.

Filtering. This step is necessary for tractability, since full spliced and unspliced matrices can be challenging to analyze. While caution must be applied in selecting thresholding criteria, we find no reason to deviate from the standards typically applied in RNA velocity analyses.

Model definition. RNA velocity methods have been inspired by stochastic models of transcription. However there has not been a strong link between the models and the implemented methods, which are based on loose analogies and heuristics. We believe that explicit construction and discussion of biophysical models is imperative when developing RNA velocity methods, so that results can be meaningful and interpretable. In particular, we caution against the class of continuous and constitutive models implemented in velocity packages thus far; as discussed above, bursty models are tractable [81] and substantially more plausible according to live-cell data.

Normalization and imputation: The normalization and averaging of data to produce continuous curves is intended to remove cell size effects and to denoise the data. We found several problems with this approach. Firstly, this assertion is not motivated by theory, and our theoretical concerns in Section B in [S1 Text](#) suggest that model-agnostic “correction” is inappropriate. Secondly, as discussed in “Count processing” under “Prospects and solutions” and illustrated in [Fig 9](#), the imputed data do not accurately recapitulate the supposed ground truth even in the simplest case. Finally, imputation prevents the most natural interpretation of counts as discrete random variables. Based on Kim et al. [110] and [130], we advise against normalization; it is more meaningful and accurate to apply parameterized models of extrinsic noise and gene-gene coupling. The interpretability afforded by discrete models outweighs the potential benefits of *ad hoc* normalization. Furthermore, we strongly recommend against imputation more generally: studies such as [68, 110, 178] have revealed distortions, and the approach possesses fundamental instabilities (as in [Fig 5](#)).

Inference. From a probabilistic perspective, current inference procedures are problematic. Instead of currently implemented procedures, it is more appropriate to build and solve mechanistic, fully stochastic models that allow for fitting copy numbers. This can be computationally facilitated by a data selection process coherent with the “marker gene” paradigm: if a gene does not need to be fit to a transient model, one should not try to fit one. Thus, we recommend fitting ergodic distributions to genes that are not meaningfully modulated across the dataset, ergodic mixture distributions to (fewer) genes that vary across disjoint cell types, and occupation measures to (even fewer) genes that exhibit transient behaviors. Although joint inference is relatively challenging, we believe that a formulation that can exploit existing combinatorial optimization frameworks may be a productive avenue for exploration.

Embedding. RNA velocity embedding procedures inherit problems accrued with the steps discussed above. However, even in an idealized situation where an interpretable and well-fit model is used, current embedding practices are counter-productive for interpreting the data, as discussed with reference to controls in subsections named “Embedding” under “Logic and methodology” and “Prospects and solutions.” Despite the implication of causal relationships between cells encoded by cell-cell transition probabilities, embedding procedures are currently *ad hoc*. Moreover, it has been shown that current methods distort local neighborhoods and the global topology in an unpredictable manner [122, 124]. Directed graphs over the cells are attractive, but do not have a coherent interpretation relative to the underlying biophysics (Section C in [S1 Text](#)). Instead of such methods, we recommend directly working with the latent process governing the transcriptional variation. Nevertheless, two-dimensional visuals may be useful for summarizing the raw data; using simulations, we demonstrate an interpretable method for embedding a true principal curve in deterministic principal component space in “Embedding” under “Prospects and solutions.”

Methods and data

Pre-processing concordance

There is no well-defined “ground truth” for mRNA counts in arbitrary datasets. However, to obtain a qualitative understanding of potential pitfalls, we performed controlled experiments to analyze the discrepancies between outputs produced by popular software implementations.

The concordance analysis was heavily inspired by the benchmarking of Soneson et al. [27]; however, our goals and scope differ. First, we sought to analyze the reproducibility of the findings across several datasets; the original analysis only treated a single dataset generated using the 10x v2 chemistry. To this end, we analyzed ten datasets that used the 10x Genomics v2 and v3 protocols. Second, we sought to focus on the processing workflows most relevant to casual

Table 1. The datasets used to compare performance of molecule quantification software.

Dataset ID	Description	Biology	Source
desai_dmso	Embryonic stem cells, control	E14 mouse	[179]
desai_idu	Embryonic stem cells, with IdU disruption	E14 mouse	[179]
pbmc_1k_v3	Peripheral blood mononuclear cells	Healthy human	10x Genomics
pbmc_10k_v3	Peripheral blood mononuclear cells	Healthy human	10x Genomics
heart_1k_v3	Heart cells	E18 mouse	10x Genomics
heart_10k_v3	Heart cells	E18 mouse	10x Genomics
neuron_1k_v3	Brain cells	E18 mouse	10x Genomics
neuron_10k_v3	Brain cells	E18 mouse	10x Genomics
brain_5k_v3	Brain cells	E18 mouse	10x Genomics
brain_nuc_5k_v3	Brain nuclei	E18 mouse	10x Genomics

<https://doi.org/10.1371/journal.pcbi.1010492.t001>

use. The original analysis examined thirteen quantification workflows, whereas we examined three: *velocyto*, *kallisto|bustools*, and *salmon*. These were run with default settings. We have made available all the scripts and *loom* files generated by the workflows (“Data availability”).

We obtained ten datasets generated with the 10x Genomics scRNA-seq platform (Table 1). Two were released as part of a study by Desai et al. and used v2 chemistry [179]. Eight were released by 10x Genomics and used v3 chemistry. The dataset metadata are outlined under “Data availability.”

To implement the *velocyto* workflow, we ran *CellRanger* on the datasets using human and mouse reference genomes, pre-built by 10x Genomics (*GRCh38* and *mm10* 2020-A). We then processed the aligned outputs using the `run10x` command provided in *velocyto*.

To implement the *kallisto|bustools* workflow, we ran the `ref` command on the pre-built genomes to build references, using the standard `--workflow lamanno` option. We then processed the raw data with the `count` command, passing in the generated reference and using the `--workflow lamanno` option.

To implement the *salmon alevin-fry* workflow, we ran the *alevin-fry* velocity workflow documented at <https://combine-lab.github.io/alevin-fry-tutorials/2021/alevin-fry-velocity/>, from the initial reference construction to the final *anndata* output. This output was converted directly to *loom* files for the comparative analysis. We used the same pre-built 10x Genomics reference genomes (*GRCh38* and *mm10* 2020-A) as above.

Simulation

Transient constitutive model: Perturbation and reversion. To generate Fig 9, we simulated data from the constitutive transcription model with the “cell type” structure *ABA*. In this model all cells start out in state *A* at $t = 0$, switch to state *B* at $t = \tau_1$, and revert back to state *A* at $t = \tau_2$. We generated 2000 cells and 100 genes. As shown in Fig 9a and formalized in Eq 21, we defined three time periods corresponding to each cell type. The simulation time horizon was set to $T = 10$, with synchronized transition times $\tau_1 = 3$ and $\tau_2 = 7$. The gene-specific transcription rates α_1 and α_2 were generated from a lognormal distribution with log-mean 0 and log-standard deviation 1. The gene-specific splicing rates β were generated from a lognormal distribution with log-mean 1 and log-standard deviation 0.5. The gene-specific degradation rates γ were generated from a lognormal distribution with log-mean 0.5 and log-standard deviation 0.25, to reflect the intuition that splicing is somewhat faster than degradation. Sampling times were generated from a continuous uniform random variable on the interval $[0, T]$.

The solutions in Fig 9b were computed from an approximation to the generating function. We did not account for the initial condition, which suffices because the transcription rate on

$(0, \tau_1)$ is low. The true values of μ_u and μ_s were computed from the solutions to the governing ordinary differential equation. The true values of σ_u^2 and σ_s^2 were set to μ_u and μ_s , respectively, as the mean of a Poisson distribution is identical to its variance. To generate Fig 10, we simulated data from the same model, with 2000 cells and 100 genes.

Transient constitutive model: Multipotent differentiation. To generate Fig 11, we simulated data from the constitutive transcription model with the “cell type” structure $AB(C/D/E)$. In this model all cells start out in state A at $t = 0$, switch to state B at $t = \tau_1$, and then transition to one of the terminal states C , D , or E at $t = \tau_2$. We generated 2000 cells for 100 genes. The gene parameter and observation time distributions, as well as switching times, were identical to those reported in “Transient constitutive model: Perturbation and reversion.” Each cell fate was assigned randomly, with equal probabilities of 1/3. We used an identical procedure for Fig A in S1 Text.

To generate Fig 8, and demonstrate qualitative differences between process time and expression-based pseudotime, we followed the same simulation as described above for the structure $AB(C/D/E)$. Specifically we limited the multipotent differentiation to a bifurcation ($AB(C/D)$) where cells start in state A at $t = 0$, switch to state B at $t = \tau_1$, and then transition to one of the terminal states C or D at $t = \tau_2$. From the simulated, raw counts, we calculated the ground truth (spliced) average μ_s and generated the plotted expression values for the 2000 cells by adding Gaussian jitter/noise to these deterministic averages.

Steady-state bursty model. To generate Fig F and G in S1 Text, we generated synthetic data assuming that RNA transcription is at an equilibrium, but has heterogeneity due to different cell types. The bursty transcription model was implemented using the PGF schema outlined in its derivation [106]. We specified parameters for 100 genes, with cell-independent burst sizes b and splicing rates β . Burst sizes b were generated from a lognormal distribution with log-mean 0.3 and log-standard deviation 0.8, clipped to stay in the range [0.05, 25]. The splicing rates β were set to 1 with no loss of generality. We simulated 10 cell types distinguished by average burst frequencies α and degradation rates γ , with 300 cells per cell type.

Average gene-specific log-degradation rates $\langle \gamma \rangle$ were generated from a normal distribution with mean -0.3 and log-standard deviation 0.3, to reflect the intuition that splicing is somewhat faster than degradation [1]. Gene- and cell type-specific degradation rates γ were generated from a lognormal distribution with log-mean $\langle \gamma \rangle$ and log-standard deviation 0.1, clipped to stay in the range [0.08, 4], to reflect the intuition that extrinsic noise in degradation rates is low relative to that in transcription rates.

Burst frequencies were generated from a lognormal distribution with log-mean -1 and log-standard deviation 0.5, clipped to stay in the range [0.005, 1], to encode the intuition that transcriptional activity is relatively rare. Analytical means μ_u , μ_s and standard deviations σ_u , σ_s were computed for spliced and unspliced distributions. Histograms were generated up to $\mu + 5\sigma$ in each direction, clipped to be no lower than 10. To keep molecule counts realistically low and the histograms tractable, we rejected and regenerated parameter sets that produced $(\mu_u + 5\sigma_u) \times (\mu_s + 5\sigma_s) > 1.5 \times 10^4$. To generate observations, we sampled directly from the histograms.

The phase plots displayed in Fig G in S1 Text were manually selected after sorting for simulated genes with the highest $\sum_i \Delta s_i$ and $-\sum_i \Delta s_i$.

Filtering

In Figs 5–7, we analyzed the forebrain dataset. To pre-process it, we implemented a procedure largely identical to that used to generate Fig 4a of the original publication [1]. We performed several rounds of filtering:

For the forebrain dataset, we used the following sequence of thresholds:

1. Discarding cells in the 0.5th percentile of total unspliced counts.
2. Discarding genes with fewer than 40 spliced counts, or expressed in fewer than 30 cells.
3. Selecting the top 2000 genes by coefficient of variation vs. mean, as implemented in the *velocyto* function `score_cv_vs_mean`, with maximum expression average of 35 (based on spliced counts).

In all other figures, we analyzed simulated data and omitted filtering, as all genes *a priori* had the correct dynamics.

Normalization

After importing forebrain data, we normalized and log-transformed spliced and unspliced counts using the default schema implemented in the *velocyto* function `normalize`:

$$\begin{aligned} s_{ij} &\leftarrow \frac{1}{M} \frac{\sum_{i,j} s_{ij}}{\sum_j s_{ij}} \\ u_{ij} &\leftarrow \frac{1}{M} \frac{\sum_{i,j} u_{ij}}{\sum_j u_{ij}}, \end{aligned} \quad (43)$$

i.e., the “cell sizes” or total counts of spliced and unspliced molecules were separately normalized so each cell’s total was set to the mean over the dataset.

For the simulated data, we did not use normalization. This approach was inconsistent with previous simulated benchmarks [1], but we had three reasons for omitting it. First, as discussed in “Count processing” under “Prospects and solutions,” normalization purports to “regress out” systematic technical and biological effects. We did not include these phenomena in the model. Second, the ground truth principal curves we constructed for the analyses in the section “Prospects and solutions” (e.g., in Fig 10e) relied on evaluating the true gene-specific $\mu_s(t)$ on a grid over $[0, T]$, then log-transforming and projecting them to the two-dimensional principal component space. This was straightforward to do when the PC space was computed from raw counts, but more challenging otherwise. Finally, our omission made the velocity embedding procedure coherent: with the PC projection based on raw counts, we used the same underlying space to impute counts and extrapolate velocities.

Embedding construction

For the forebrain dataset, we used size- and log-normalized spliced counts to construct the principal component projection. The UMAP and *t*-SNE embeddings were calculated from the top 25 principal components (as illustrated in Fig 6). For the simulated data, we used the log-normalized spliced counts to compute the same embeddings.

Imputation

Prior to fitting the models, data were smoothed by pooling across 50 nearest neighbors in the 25-dimensional principal component space defined in “Embedding construction,” as quantified by Euclidean distance. To implement this step, we used the default parameters of the *velocyto* function `knn_imputation`. The choice of neighborhood space was arbitrary, and we imposed it for consistency. The original report used an adaptive principal component space

based on the fraction of explained variance, whereas *scVelo* uses a default of 30 principal components. We observed no substantial difference in results between the adaptive and fixed schema. For the forebrain dataset, we pooled the normalized counts. In case of the simulations, we pooled the raw counts. In Fig 9 and Fig E in S1 Text, we deviated from this procedure to investigate the impact and suitability of normalizing simulated data. The figures demonstrate the respective effects of pooling the normalized and raw counts.

Inference and extrapolation

By default, the parameter γ/β was fit to the extrema of the imputed dataset: imputed unspliced counts were regressed as a linear function of imputed spliced counts with an offset. The extrema selection procedure used the defaults implemented in the *velocyto* function `fit_gammas`.

In Fig 9e, we deviated from this procedure to investigate the suitability and stability of inference from pooled data. In Fig 9ei, we performed linear regression on the raw counts of the entire dataset, whereas in Fig 9eii, we performed linear regression on the quantiles of the normalized dataset. Finally, in the “Raw” or $k = 0$ cases illustrated in Fig 5, we performed linear regression on the raw counts of the entire dataset to contrast with regression on the extrema.

The standard inference procedure produced two parameters per gene j : the slope, a putative estimate of γ/β , and the intercept, which we denote as q . To compute the velocity of gene j in cell i for the nonlinear velocity embedding, we used the following formula:

$$v_{ij} = u_{ij} - \left(\frac{\gamma_j}{\beta_j} s_{ij} + q_j \right), \quad (44)$$

where u and s denote imputed quantities. To extrapolate the velocity and predict the spliced abundance after a time interval Δt , we calculated $\Delta s_{ij} = v_{ij}\Delta t$. This time interval was set to 1, for consistency with the *velocyto* implementation. The extrapolated value $s_{ij} + \Delta s_{ij}$ does not appear to be used in *velocyto*, as Δs_{ij} contains the directional information used in the nonlinear embedding.

The schema described above is consistent with that of *velocyto*, but cannot be used to compute linear velocity embeddings (as given in Eq 3). The initial condition (u_{ij}, s_{ij}) used for extrapolation must be in the space used to build the PCA representation. Therefore, for linear velocity embeddings, we used Eq 44 in the corresponding space: normalized counts for the forebrain dataset and raw counts for simulated data.

However, if $\Delta s_{ij} < 0$, the naïve extrapolation $s_{ij} + \Delta s_{ij}$ was not guaranteed to give a non-negative value that could be log- and PCA-transformed. In principle, we could have used an arbitrary Δt , and clip any $s_{ij} + \Delta s_{ij} < 0$ to zero. However, this approach, though closer in spirit to *velocyto*, would have risked extrapolating beyond the physical regime and could have introduced biases.

Instead, we chose an extrapolation time Δt^* guaranteed to stay in the physical regime:

$$\begin{aligned} \Delta t^* &= \operatorname{argmax}_{\Delta t} \bigcap_{i,j} [s_{ij} + v_{ij}\Delta t > 0] \\ &= \min_{i,j} \frac{s_{ij}}{|v_{ij}|}, \end{aligned} \quad (45)$$

where we filtered for $v_{ij} < -10^{-6}$. Finally, we set Δs_{ij} to zero whenever $s_{ij} < 10^{-6}$ and $v_{ij} < -10^{-6}$, to avoid extrapolation into the negative regime: this fairly rare case occurs due to nonzero q_j .

Velocity embedding

For the nonlinear velocity embeddings, we used 150 embedding neighbors and the square-root transformation by default. We deviated from this procedure in Fig D in [S1 Text](#) to investigate the impact of transformation and neighborhood choices. We used the default hyperparameter $\sigma = 0.05$ to calculate the softmax over directions to embedding neighbors (as described on pp. 7–8 in SN1 of [1]). To implement the embeddings, we called the *velocyto* functions `estimate_transition_prob` and `calculate_embedding_shift`, which automatically correct for cell density. We did not use the neighborhood downsampling, randomization, or expression scaling options for the figures in this report; we observed no substantial difference in results between these schema and our standard procedure. The “high-dimensional space,” used to evaluate the displacements $s_q - s_i$ in [Eq 4](#), was the matrix of imputed spliced counts. The extrapolations Δs_i were obtained from the procedure in “Inference and extrapolation.”

For the linear velocity embeddings, we log- and PCA-transformed the matrix $s_{ij} + v_{ij}\Delta t^*$, with the timescale obtained by the procedure in [Eq 45](#).

The “Boolean” schema for velocity embedding is qualitatively similar to the schema proposed in the original publication; we previously proposed it to bypass the unit inconsistency between different genes’ β_j (and thus vectors v_j) in the context of the *protaccel* package [4]. Instead of computing a correlation coefficient, we simply calculated the fraction of concordant signs between the velocity and the displacements to neighbors. In the parlance of [Eq 4](#):

$$\begin{aligned} r(s_q - s_i, \Delta s_i) &= \frac{1}{N} \sum_{j=1}^N \left(\delta\left(\text{sign}(s_{q,j} - s_{i,j}), \text{sign}(\Delta s_{i,j})\right) \right) \\ w(s_q - s_i, \Delta s_i) &= \frac{\exp(r(s_q - s_i, \Delta s_i)/\sigma)}{\sum_{q=1}^k \exp(r(s_q - s_i, \Delta s_i)/\sigma)}, \end{aligned} \quad (46)$$

where δ is the Kronecker delta operating on inputs in $\{-1, 0, 1\}$.

We plotted cell-specific arrows for the linear baseline and aggregated the nonlinear velocity arrows using a 20×20 grid. We used this convention to distinguish the embedding methods, which are conceptually and quantitatively different, in plots that showed several velocity fields at once (e.g., the PCA plot in [Fig 7](#)). The grid directions were computed using the *velocyto* function `calculate_grid_arrows`, which applies a Gaussian kernel to average over the cell-specific embedded velocities nearest the grid point. We used the default parameters for the kernel, with 100 neighbors and a smoothing parameter of 0.5. We aggregated linear velocity projections in [Fig 10a and 10b](#). As the arrow scale did not appear to have a quantitative interpretation, we set it manually to match the plot proportions.

Supporting information

S1 Text. Supplementary derivations and figures.
(PDF)

Acknowledgments

G.G. thanks Dr. John J. Vastola for fruitful discussions about landscape representations of biophysical systems.

Author Contributions

Conceptualization: Gennady Gorin, Lior Pachter.

Data curation: Gennady Gorin, Meichen Fang, Tara Chari.

Formal analysis: Gennady Gorin, Meichen Fang, Tara Chari.

Funding acquisition: Lior Pachter.

Investigation: Gennady Gorin, Meichen Fang, Tara Chari.

Methodology: Gennady Gorin.

Project administration: Gennady Gorin, Lior Pachter.

Resources: Lior Pachter.

Software: Gennady Gorin, Meichen Fang, Tara Chari.

Supervision: Lior Pachter.

Validation: Gennady Gorin, Meichen Fang, Tara Chari.

Visualization: Gennady Gorin, Meichen Fang, Tara Chari.

Writing – original draft: Gennady Gorin, Meichen Fang, Tara Chari, Lior Pachter.

Writing – review & editing: Gennady Gorin, Meichen Fang, Tara Chari, Lior Pachter.

References

1. La Manno G, Soldatov R, Zeisel A, Braun E, Hochgerner H, Petukhov V, et al. RNA velocity of single cells. *Nature*. 2018; 560(7719):494–498. <https://doi.org/10.1038/s41586-018-0414-6> PMID: 30089906
2. Zeisel A, Kostler WJ, Molotski N, Tsai JM, Krauthgamer R, Jacob-Hirsch J, et al. Coupled pre-mRNA and mRNA dynamics unveil operational strategies underlying transcriptional responses to stimuli. *Molecular Systems Biology*. 2011; 7(1):529–529. <https://doi.org/10.1038/msb.2011.62> PMID: 21915116
3. Bergen V, Lange M, Peidli S, Wolf FA, Theis FJ. Generalizing RNA velocity to transient cell states through dynamical modeling. *Nature Biotechnology*. 2020. <https://doi.org/10.1038/s41587-020-0591-3> PMID: 32747759
4. Gorin G, Svensson V, Pachter L. Protein velocity and acceleration from single-cell multiomics experiments. *Genome Biology*. 2020; 21:39. <https://doi.org/10.1186/s13059-020-1945-3> PMID: 32070398
5. Li C, Virgilio M, Collins KL, Welch JD. Single-cell multi-omic velocity infers dynamic and decoupled gene regulation. *bioRxiv*: 2021.12.13.472472; 2021.
6. Tedesco M, Giannese F, Lazarević D, Giansanti V, Rosano D, Monzani S, et al. Chromatin Velocity reveals epigenetic dynamics by single-cell profiling of heterochromatin and euchromatin. *Nature Biotechnology*. 2021. PMID: 34635836
7. Weng G, Kim J, Won KJ. VeTra: a tool for trajectory inference based on RNA velocity. *Bioinformatics*. 2021; p. btab364. <https://doi.org/10.1093/bioinformatics/btab364> PMID: 33974009
8. Zhang Z, Zhang X. Inference of high-resolution trajectories in single cell RNA-Seq data from RNA velocity. *bioRxiv*: 2020.09.30.321125; 2020.
9. Gupta R, Cerletti D, Gut G, Oxenius A, Claassen M. Cytopath: Simulation based inference of differentiation trajectories from RNA velocity fields. *bioRxiv*: 2020.12.21.423801; 2020.
10. Lange M, Bergen V, Klein M, Setty M, Reuter B, Bakhti M, et al. CellRank for directed single-cell fate mapping. *Nature Methods*. 2022. <https://doi.org/10.1038/s41592-021-01346-6> PMID: 35027767
11. Schwabe D, Formichetti S, Junker JP, Falcke M, Rajewsky N. The transcriptome dynamics of single cells during the cell cycle. *Molecular Systems Biology*. 2020; 16(11). <https://doi.org/10.15252/msb.20209946> PMID: 33205894
12. Harmanci AS, Harmanci AO, Zhou X, Deneen B, Rao G, Klisch T, et al. scRegulocity: Detection of local RNA velocity patterns in embeddings of single cell RNA-Seq data. *bioRxiv*: 2021.06.01.446674; 2021.

13. Wang X. Velo-Predictor: an ensemble learning pipeline for RNA velocity prediction. *BMC Bioinformatics*. 2021; p. 12. <https://doi.org/10.1186/s12859-021-04330-1> PMID: 34479487
14. Cannoodt R, Saelens W, Deconinck L, Saeys Y. Spearheading future omics analyses using dyngen, a multi-modal simulator of single cells. *Nature Communications*. 2021; 12(1):3942. <https://doi.org/10.1038/s41467-021-24152-2> PMID: 34168133
15. Zhang Z, Zhang X. VeloSim: Simulating single cell gene-expression and RNA velocity. *bioRxiv*: 2021.01.11.426277; 2021.
16. Atta L, Fan J. VeloViz: RNA-velocity informed 2D embeddings for visualizing cellular trajectories. *bioRxiv*: 2021.01.28.425293; 2021. <https://doi.org/10.1101/bioinformatics/btab653> PMID: 34500455
17. Hie BL, Yang KK, Kim PS. Evolutionary velocity with protein language models. *bioRxiv*: 2021.06.07.447389; 2021.
18. Svensson V, Pachter L. RNA Velocity: Molecular Kinetics from Single-Cell RNA-Seq. *Molecular Cell*. 2018; 72(1):7–9. <https://doi.org/10.1016/j.molcel.2018.09.026> PMID: 30290149
19. Charrouf M, Reinders MJT, Mahfouz A. Untangling biological factors influencing trajectory inference from single cell data. *bioRxiv*: 2020.02.11.942102; 2020.
20. Tritschler S, Büttner M, Fischer DS, Lange M, Bergen V, Lickert H, et al. Concepts and limitations for learning developmental trajectories from single cell genomics. *Development*. 2019; 146(12): dev170506. <https://doi.org/10.1242/dev.170506> PMID: 31249007
21. Bergen V, Soldatov RA, Kharchenko PV, Theis FJ. RNA velocity—current challenges and future perspectives. *Molecular Systems Biology*. 2021; 17(8). <https://doi.org/10.1525/msb.202110282> PMID: 34435732
22. Lavenant H, Zhang S, Kim YH, Schiebinger G. Towards a mathematical theory of trajectory inference. *arXiv*: 2102.09204; 2021.
23. Zhang S, Afanassiev A, Greenstreet L, Matsumoto T, Schiebinger G. Optimal transport analysis reveals trajectories in steady-state systems. *PLOS Computational Biology*. 2021; 17(12):e1009466. <https://doi.org/10.1371/journal.pcbi.1009466> PMID: 34860824
24. Li T. On the Mathematics of RNA Velocity I: Theoretical Analysis. *CSIAM Transactions on Applied Mathematics*. 2021; 2(1):1–55. <https://doi.org/10.4208/csiam-am.SO-2020-0001>
25. Gorin G, Pachter L. Special function methods for bursty models of transcription. *Physical Review E*. 2020; 102(2):022409. <https://doi.org/10.1103/PhysRevE.102.022409> PMID: 32942485
26. Haque A, Engel J, Teichmann SA, Lönnberg T. A practical guide to single-cell RNA-sequencing for biomedical research and clinical applications. *Genome Medicine*. 2017; 9(1):75. <https://doi.org/10.1186/s13073-017-0467-4> PMID: 28821273
27. Soneson C, Srivastava A, Patro R, Stadler MB. Preprocessing choices affect RNA velocity results for droplet scRNA-seq data. *PLOS Computational Biology*. 2021; 17(1):e1008585. <https://doi.org/10.1371/journal.pcbi.1008585> PMID: 33428615
28. Zheng SC, Stein-O'Brien G, Boukas L, Goff LA, Hansen KD. Pumping the brakes on RNA velocity—understanding and interpreting RNA velocity estimates. *bioRxiv*: 2022.06.19.494717; 2022.
29. Qiu X, Zhang Y, Martin-Rufino JD, Weng C, Hosseinzadeh S, Yang D, et al. Mapping transcriptomic vector fields of single cells. *Cell*. 2022; p. S0092867421015774. <https://doi.org/10.1016/j.cell.2021.12.045> PMID: 35108499
30. Melsted P, Booeshaghi AS, Liu L, Gao F, Lu L, Min KH, et al. Modular, efficient and constant-memory single-cell RNA-seq preprocessing. *Nature Biotechnology*. 2021; 39(7):813–818. <https://doi.org/10.1038/s41587-021-00870-2> PMID: 33795888
31. Moses L, Pachter L. Museum of Spatial Transcriptomics. *bioRxiv*: 2021.05.11.443152; 2021.
32. George L, Indig FE, Abdelmohsen K, Gorospe M. Intracellular RNA-tracking methods. *Open Biology*. 2018; 8(10):180104. <https://doi.org/10.1098/rsob.180104> PMID: 30282659
33. Specht EA, Braselmann E, Palmer AE. A Critical and Comparative Review of Fluorescent Tools for Live-Cell Imaging. *Annual Review of Physiology*. 2017; 79(1):93–117. <https://doi.org/10.1146/annurev-physiol-022516-034055> PMID: 27860833
34. Golding I, Paulsson J, Zawilski SM, Cox EC. Real-Time Kinetics of Gene Activity in Individual Bacteria. *Cell*. 2005; 123(6):1025–1036. <https://doi.org/10.1016/j.cell.2005.09.031> PMID: 16360033
35. Munsky B, Fox Z, Neuert G. Integrating single-molecule experiments and discrete stochastic models to understand heterogeneous gene transcription dynamics. *Methods*. 2015; 85:12–21. <https://doi.org/10.1016/j.ymeth.2015.06.009> PMID: 26079925
36. Wold B, Myers RM. Sequence census methods for functional genomics. *Nature Methods*. 2008; 5(1):19–21. <https://doi.org/10.1038/nmeth1157> PMID: 18165803

37. Luecken MD, Theis FJ. Current best practices in single-cell RNA-seq analysis: a tutorial. *Molecular Systems Biology*. 2019; 15(6):e8746. <https://doi.org/10.1525/msb.20188746> PMID: 31217225
38. Guo J, Grow EJ, Micochova H, Maher GJ, Lindskog C, Nie X, et al. The adult human testis transcriptional cell atlas. *Cell Research*. 2018; 28(12):1141–1157. <https://doi.org/10.1038/s41422-018-0099-2> PMID: 30315278
39. Soldatov R, Kaucka M, Kastriti ME, Petersen J, Chontorotzea T, Englmaier L, et al. Spatiotemporal structure of cell fate decisions in murine neural crest. *Science*. 2019; 364(6444):eaas9536. <https://doi.org/10.1126/science.aas9536> PMID: 31171666
40. Xiong H, Luo Y, Yue Y, Zhang J, Ai S, Li X, et al. Single-Cell Transcriptomics Reveals Chemotaxis-Mediated Intraorgan Crosstalk During Cardiogenesis. *Circulation Research*. 2019; 125(4):398–410. <https://doi.org/10.1161/CIRCRESAHA.119.315243> PMID: 31221018
41. Litviňuková M, Talavera-López C, Maatz H, Reichart D, Worth CL, Lindberg EL, et al. Cells of the adult human heart. *Nature*. 2020; 588(7838):466–472. <https://doi.org/10.1038/s41586-020-2797-4> PMID: 32971526
42. Han X, Zhou Z, Fei L, Sun H, Wang R, Chen Y, et al. Construction of a human cell landscape at single-cell level. *Nature*. 2020; 581(7808):303–309. <https://doi.org/10.1038/s41586-020-2157-4> PMID: 32214235
43. Yu L, Wei Y, Duan J, Schmitz DA, Sakurai M, Wang L, et al. Blastocyst-like structures generated from human pluripotent stem cells. *Nature*. 2021; 591(7851):620–626. <https://doi.org/10.1038/s41586-021-03356-y> PMID: 33731924
44. Bassez A, Vos H, Van Dyck L, Floris G, Arijs I, Desmedt C, et al. A single-cell map of intratumoral changes during anti-PD1 treatment of patients with breast cancer. *Nature Medicine*. 2021; 27(5):820–832. <https://doi.org/10.1038/s41591-021-01323-8> PMID: 33958794
45. Jansky S, Sharma AK, Körber V, Quintero A, Toprak UH, Wecht EM, et al. Single-cell transcriptomic analyses provide insights into the developmental origins of neuroblastoma. *Nature Genetics*. 2021; 53(5):683–693. <https://doi.org/10.1038/s41588-021-00806-1> PMID: 33767450
46. Couturier CP, Ayyadhyury S, Le PU, Nadaf J, Monlong J, Riva G, et al. Single-cell RNA-seq reveals that glioblastoma recapitulates a normal neurodevelopmental hierarchy. *Nature Communications*. 2020; 11(1):3406. <https://doi.org/10.1038/s41467-020-17186-5> PMID: 32641768
47. Shah S, Takei Y, Zhou W, Lubeck E, Yun J, Eng CHL, et al. Dynamics and Spatial Genomics of the Nascent Transcriptome by Intron seqFISH. *Cell*. 2018; 174(2):363–376.e16. <https://doi.org/10.1016/j.cell.2018.05.035> PMID: 29887381
48. Shah S, Lubeck E, Zhou W, Cai L. In Situ Transcription Profiling of Single Cells Reveals Spatial Organization of Cells in the Mouse Hippocampus. *Neuron*. 2016; 92(2):342–357. <https://doi.org/10.1016/j.neuron.2016.10.001> PMID: 27764670
49. Park J, Choi W, Tiesmeyer S, Long B, Borm LE, Garren E, et al. Cell segmentation-free inference of cell types from *in situ* transcriptomics data. *Nature Communications*. 2021; 12(1):3545. <https://doi.org/10.1038/s41467-021-23807-4> PMID: 34112806
50. Samacoits A, Chouaib R, Safieddine A, Traboulsi AM, Ouyang W, Zimmer C, et al. A computational framework to study sub-cellular RNA localization. *Nature Communications*. 2018; 9(1):4584. <https://doi.org/10.1038/s41467-018-06868-w> PMID: 30389932
51. Kim JK, Kolodziejczyk AA, Ilicic T, Teichmann SA, Marioni JC. Characterizing noise structure in single-cell RNA-seq distinguishes genuine from technical stochastic allelic expression. *Nature Communications*. 2015; 6(1):8687. <https://doi.org/10.1038/ncomms9687> PMID: 26489834
52. Kim J, Marioni JC. Inferring the kinetics of stochastic gene expression from single-cell RNA-sequencing data. *Genome Biology*. 2013; 14:R7. <https://doi.org/10.1186/gb-2013-14-1-r7> PMID: 23360624
53. Jiang R, Sun T, Song D, Li JJ. Statistics or biology: the zero-inflation controversy about scRNA-seq data. *Genome Biology*. 2022; 23:31. <https://doi.org/10.1186/s13059-022-02601-5> PMID: 35063006
54. Delmans M, Hemberg M. Discrete distributional differential expression (D3E)—a tool for gene expression analysis of single-cell RNA-seq data. *BMC Bioinformatics*. 2016; 17:110. <https://doi.org/10.1186/s12859-016-0944-6> PMID: 26927822
55. Ham L, Brackston RD, Stumpf MPH. Extrinsic Noise and Heavy-Tailed Laws in Gene Expression. *Physical Review Letters*. 2020; 124(10):108101. <https://doi.org/10.1103/PhysRevLett.124.108101> PMID: 32216388
56. Svensson V. Droplet scRNA-seq is not zero-inflated. *Nature Biotechnology*. 2020; 38(2):147–150. <https://doi.org/10.1038/s41587-019-0379-5> PMID: 31937974
57. Lopez R, Regier J, Cole MB, Jordan MI, Yosef N. Deep generative modeling for single-cell transcriptomics. *Nature Methods*. 2018; 15(12):1053–1058. <https://doi.org/10.1038/s41592-018-0229-2> PMID: 30504886

58. Amrhein L, Harsha K, Fuchs C. A mechanistic model for the negative binomial distribution of single-cell mRNA counts. *bioRxiv*: 657619; 2019.
59. Bonnaffoux A, Herbach U, Richard A, Guillemin A, Gonin-Giraud S, Gros PA, et al. WASABI: a dynamic iterative framework for gene regulatory network inference. *BMC bioinformatics*. 2019; 20(1):1–19. <https://doi.org/10.1186/s12859-019-2798-1> PMID: 31046682
60. Swain PS, Elowitz MB, Siggia ED. Intrinsic and extrinsic contributions to stochasticity in gene expression. *Proceedings of the National Academy of Sciences*. 2002; 99(20):12795–12800. <https://doi.org/10.1073/pnas.162041399> PMID: 12237400
61. Xu H, Skinner SO, Sokac AM, Golding I. Stochastic Kinetics of Nascent RNA. *Physical Review Letters*. 2016; 117(12):128101. <https://doi.org/10.1103/PhysRevLett.117.128101> PMID: 27667861
62. Raj A, Peskin CS, Tranchina D, Vargas DY, Tyagi S. Stochastic mRNA Synthesis in Mammalian Cells. *PLoS Biology*. 2006; 4(10):e309. <https://doi.org/10.1371/journal.pbio.0040309> PMID: 17048983
63. Zheng GXY, Terry JM, Belgrader P, Ryvkin P, Bent ZW, Wilson R, et al. Massively parallel digital transcriptional profiling of single cells. *Nature Communications*. 2017; 8(1):14049. <https://doi.org/10.1038/ncomms14049> PMID: 28091601
64. Bray NL, Pimentel H, Melsted P, Pachter L. Near-optimal probabilistic RNA-seq quantification. *Nature Biotechnology*. 2016; 34(5):525–527. <https://doi.org/10.1038/nbt.3519> PMID: 27043002
65. Du Y, Huang Q, Arisdakessian C, Garmire LX. Evaluation of STAR and Kallisto on Single Cell RNA-Seq Data Alignment. *G3: Genes, Genomes, Genetics*. 2020; 10(5):1775–1783. <https://doi.org/10.1534/g3.120.401160> PMID: 32220951
66. Petukhov V, Guo J, Baryawno N, Severe N, Scadden DT, Samsonova MG, et al. dropEst: pipeline for accurate estimation of molecular counts in droplet-based single-cell RNA-seq experiments. *Genome Biology*. 2018; 19(1):78. <https://doi.org/10.1186/s13059-018-1449-6> PMID: 29921301
67. Li WV, Li JJ. An accurate and robust imputation method sclmpipe for single-cell RNA-seq data. *Nature Communications*. 2018; 9(1):997. <https://doi.org/10.1038/s41467-018-03405-7> PMID: 29520097
68. Andrews T, Hemberg M. False signals induced by single-cell imputation. *F1000Research*. 2019; 7:1740. <https://doi.org/10.12688/f1000research.16613.1>
69. Lynch M, Conery JS. The Origins of Genome Complexity. *Science*. 2003; 302(5649):1401–1404. <https://doi.org/10.1126/science.1089370> PMID: 14631042
70. Wang ET, Sandberg R, Luo S, Khrebtukova I, Zhang L, Mayr C, et al. Alternative isoform regulation in human tissue transcriptomes. *Nature*. 2008; 456(7221):470–476. <https://doi.org/10.1038/nature07509> PMID: 18978772
71. Conze T, Göransson J, Razzaghian HR, Ericsson O, Öberg D, Akusjärvi G, et al. Single molecule analysis of combinatorial splicing. *Nucleic Acids Research*. 2010; 38(16):e163–e163. <https://doi.org/10.1093/nar/gkq581> PMID: 20587504
72. Tian L, Jabbari JS, Thijssen R, Gouil Q, Amarasinghe SL, Kariyawasam H, et al. Comprehensive characterization of single cell full-length isoforms in human and mouse with long-read sequencing. *bioRxiv*: 2020.08.10.243543; 2020.
73. Booshaghi AS, Yao Z, van Velthoven C, Smith K, Tasic B, Zeng H, et al. Isoform cell-type specificity in the mouse primary motor cortex. *Nature*. 2021; 598(7879):195–199. <https://doi.org/10.1038/s41586-021-03969-3> PMID: 34616073
74. Liu X, Andrews MV, Skinner JP, Johanson TM, Chong MMW. A comparison of alternative mRNA splicing in the CD4 and CD8 T cell lineages. *Molecular Immunology*. 2021; 133:53–62. <https://doi.org/10.1016/j.molimm.2021.02.009> PMID: 33631555
75. Pimentel H, Parra M, Gee SL, Mohandas N, Pachter L, Conboy JG. A dynamic intron retention program enriched in RNA processing genes regulates gene expression during terminal erythropoiesis. *Nucleic Acids Research*. 2016; 44(2):838–851. <https://doi.org/10.1093/nar/gkv1168> PMID: 26531823
76. Dvinge H, Bradley RK. Widespread intron retention diversifies most cancer transcriptomes. *Genome Medicine*. 2015; 7(1):45. <https://doi.org/10.1186/s13073-015-0168-9> PMID: 26113877
77. Wong JL, Au AYM, Ritchie W, Rasko JEJ. Intron retention in mRNA: No longer nonsense: Known and putative roles of intron retention in normal and disease biology. *BioEssays*. 2016; 38(1):41–49. <https://doi.org/10.1002/bies.201500117> PMID: 26612485
78. Galante PAF, Sakabe NJ, Kirschbaum-Slager N, De Souza SJ. Detection and evaluation of intron retention events in the human transcriptome. *RNA*. 2004; 10(5):757–765. <https://doi.org/10.1261/rna.5123504> PMID: 15100430
79. Singh J, Padgett RA. Rates of *in situ* transcription and splicing in large human genes. *Nature Structural & Molecular Biology*. 2009; 16(11):1128–1133. <https://doi.org/10.1038/nsmb.1666> PMID: 19820712

80. Wan Y, Anastasakiis DG, Rodriguez J, Palangat M, Gudla P, Zaki G, et al. Dynamic imaging of nascent RNA reveals general principles of transcription dynamics and stochastic splice site selection. *Cell*. 2021; 184(11):2878–2895.e20. <https://doi.org/10.1016/j.cell.2021.04.012> PMID: 33979654
81. Gorin G, Pachter L. Modeling bursty transcription and splicing with the chemical master equation. *Biochemical Journal*. 2022; 121(6):1056–1069. <https://doi.org/10.1016/j.bj.2022.02.004> PMID: 35143775
82. Jahnke T, Huisenga W. Solving the chemical master equation for monomolecular reaction systems analytically. *Journal of Mathematical Biology*. 2006; 54:1–26. <https://doi.org/10.1007/s00285-006-0034-x> PMID: 16953443
83. Dattani J. Exact solutions of master equations for the analysis of gene transcription models [PhD Dissertation]. Imperial College London; 2015.
84. Dattani J, Barahona M. Stochastic models of gene transcription with upstream drives: exact solution and sample path characterization. *Journal of The Royal Society Interface*. 2017; 14(126):20160833. <https://doi.org/10.1098/rsif.2016.0833> PMID: 28053113
85. Milo R, Phillips R. *Cell Biology by the Numbers*. Garland Science; 2015.
86. Ullah M, Wolkenhauer O. *Stochastic approaches for systems biology*. New York: Springer; 2011.
87. Battich N, Stoeger T, Pelkmans L. Control of Transcript Variability in Single Mammalian Cells. *Cell*. 2015; 163(7):1596–1610. <https://doi.org/10.1016/j.cell.2015.11.018> PMID: 26687353
88. Munsky B, Li G, Fox ZR, Shepherd DP, Neuert G. Distribution shapes govern the discovery of predictive models for gene regulation. *Proceedings of the National Academy of Sciences*. 2018; 115(29):7533–7538. <https://doi.org/10.1073/pnas.1804060115> PMID: 29959206
89. Munsky B, Trinh B, Khammash M. Listening to the noise: random fluctuations reveal gene network parameters. *Molecular Systems Biology*. 2009; 5:318. <https://doi.org/10.1038/msb.2009.75> PMID: 19888213
90. Munsky B, Neuert G, van Oudenaarden A. Using Gene Expression Noise to Understand Gene Regulation. *Science*. 2012; 336(6078):183–187. <https://doi.org/10.1126/science.1216379> PMID: 22499939
91. Zenklusen D, Larson DR, Singer RH. Single-RNA counting reveals alternative modes of gene expression in yeast. *Nature Structural & Molecular Biology*. 2008; 15(12):1263–1271. <https://doi.org/10.1038/nsmb.1514> PMID: 19011635
92. Hilfinger A, Norman T, Paulsson J. Exploiting Natural Fluctuations to Identify Kinetic Mechanisms in Sparsely Characterized Systems. *Cell Systems*. 2016; 2(4):251–259. <https://doi.org/10.1016/j.cels.2016.04.002> PMID: 27135537
93. Skinner SO, Xu H, Nagarkar-Jaiswal S, Freire PR, Zwaka TP, Golding I. Single-cell analysis of transcription kinetics across the cell cycle. *eLife*. 2016; 5:e12175. <https://doi.org/10.7554/eLife.12175> PMID: 26824388
94. Limi S, Senecal A, Coleman R, Lopez-Jones M, Guo P, Polumbo C, et al. Transcriptional burst fraction and size dynamics during lens fiber cell differentiation and detailed insights into the denucleation process. *Journal of Biological Chemistry*. 2018; 293(34):13176–13190. <https://doi.org/10.1074/jbc.RA118.001927> PMID: 29959226
95. Singer ZS, Yong J, Tischler J, Hackett JA, Altinok A, Surani MA, et al. Dynamic Heterogeneity and DNA Methylation in Embryonic Stem Cells. *Molecular Cell*. 2014; 55(2):319–331. <https://doi.org/10.1016/j.molcel.2014.06.029> PMID: 25038413
96. Stumpf PS, Smith RCG, Lenz M, Schuppert A, Müller FJ, Babtie A, et al. Stem Cell Differentiation as a Non-Markov Stochastic Process. *Cell Systems*. 2017; 5(3):268–282.e7. <https://doi.org/10.1016/j.cels.2017.08.009> PMID: 28957659
97. Dar RD, Razooky BS, Singh A, Trimeloni TV, McCollum JM, Cox CD, et al. Transcriptional burst frequency and burst size are equally modulated across the human genome. *Proceedings of the National Academy of Sciences*. 2012; 109(43):17454–17459. <https://doi.org/10.1073/pnas.1213530109> PMID: 23064634
98. Sanchez A, Golding I. Genetic Determinants and Cellular Constraints in Noisy Gene Expression. *Science*. 2013; 342(6163):1188–1193. <https://doi.org/10.1126/science.1242975> PMID: 24311680
99. Bahar Halpern K, Tanami S, Landen S, Chapal M, Szlak L, Hutzler A, et al. Bursty Gene Expression in the Intact Mammalian Liver. *Molecular Cell*. 2015; 58(1):147–156. <https://doi.org/10.1016/j.molcel.2015.01.027> PMID: 25728770
100. Larsson AJM, Johnsson P, Hagemann-Jensen M, Hartmanis L, Faridani OR, Reinius B, et al. Genomic encoding of transcriptional burst kinetics. *Nature*. 2019; 565(7738):251–254. <https://doi.org/10.1038/s41586-018-0836-1> PMID: 30602787
101. Nicolas D, Phillips NE, Naef F. What shapes eukaryotic transcriptional bursting? *Molecular BioSystems*. 2017; 13(7):1280–1290. PMID: 28573295

102. Rodriguez J, Larson DR. Transcription in Living Cells: Molecular Mechanisms of Bursting. *Annual Review of Biochemistry*. 2020; 89(1):189–212. <https://doi.org/10.1146/annurev-biochem-011520-105250> PMID: 32208766
103. Klindziuk A, Kolomeisky AB. Understanding the molecular mechanisms of transcriptional bursting. *Physical Chemistry Chemical Physics*. 2021; p. 10.1039.D1CP03665C.
104. Paulsson J. Models of stochastic gene expression. *Physics of Life Reviews*. 2005; 2(2):157–175. <https://doi.org/10.1016/j.plrev.2005.03.003>
105. Peccoud J, Ycard B. Markovian Modeling of Gene Product Synthesis. *Theoretical Population Biology*. 1995; 48(2):222–234. <https://doi.org/10.1006/tpb.1995.1027>
106. Singh A, Bokes P. Consequences of mRNA Transport on Stochastic Variability in Protein Levels. *Biochemical Journal*. 2012; 103(5):1087–1096. <https://doi.org/10.1016/j.bj.2012.07.015> PMID: 23009859
107. Ahlmann-Eltze C, Huber W. Transformation and Preprocessing of Single-Cell RNA-Seq Data. *bioRxiv*: 2021.06.24.449781; 2021.
108. Rostom R, Svensson V, Teichmann SA, Kar G. Computational approaches for interpreting scRNA-seq data. *FEBS Letters*. 2017; 591(15):2213–2225. <https://doi.org/10.1002/1873-3468.12684> PMID: 28524227
109. McGee WA, Pimentel H, Pachter L, Wu JY. Compositional Data Analysis is necessary for simulating and analyzing RNA-Seq data. *bioRxiv*: 564955; 2019.
110. Kim TH, Zhou X, Chen M. Demystifying “drop-outs” in single-cell UMI data. *Genome Biology*. 2020; 21:196. <https://doi.org/10.1186/s13059-020-02096-y> PMID: 32762710
111. Townes FW, Hicks SC, Aryee MJ, Irizarry RA. Feature selection and dimension reduction for single-cell RNA-Seq based on a multinomial model. *Genome Biology*. 2019; 20(1):295. <https://doi.org/10.1186/s13059-019-1861-6> PMID: 31870412
112. Hafemeister C, Satija R. Normalization and variance stabilization of single-cell RNA-seq data using regularized negative binomial regression. *Genome Biology*. 2019; 20:296. <https://doi.org/10.1186/s13059-019-1874-1> PMID: 31870423
113. Cole MB, Risso D, Wagner A, DeTomaso D, Ngai J, Purdom E, et al. Performance Assessment and Selection of Normalization Procedures for Single-Cell RNA-Seq. *Cell Systems*. 2019; 8(4):315–328. e8. <https://doi.org/10.1016/j.cels.2019.03.010> PMID: 31022373
114. Pickands J. Statistical Inference Using Extreme Order Statistics. *The Annals of Statistics*. 1975; 3(1):119–131.
115. Blanchet J, He F, Murthy KRA. On distributionally robust extreme value analysis. *arXiv*: 1601.06858; 2020.
116. Chernozhukov V. Extremal quantile regression. *The Annals of Statistics*. 2005; 33(2). <https://doi.org/10.1214/009053604000001165>
117. Chernozhukov V, Fernández-Val I, Kaji T. Extremal Quantile Regression: An Overview. *arXiv*: 1612.06850; 2017.
118. Khatri CG. Distributions of order statistics for discrete case. *Annals of the Institute of Statistical Mathematics*. 1962; 14(1):167–171. <https://doi.org/10.1007/BF02868637>
119. Nagaraja HN. Order Statistics from Discrete Distributions. *Statistics*. 1992; 23(3):189–216. <https://doi.org/10.1080/02331889208802365>
120. Arnold BC, Balakrishnan N, Nagaraja HN. *A First Course in Order Statistics*. Classics in Applied Mathematics. Philadelphia: Society for Industrial and Applied Mathematics; 2008.
121. Briggs KM, Song L, Prellberg T. A note on the distribution of the maximum of a set of Poisson random variables. *arXiv*: 0903.4373; 2009.
122. Chari T, Banerjee J, Pachter L. The Specious Art of Single-Cell Genomics. *bioRxiv*: 2021.08.25.457696; 2021.
123. Johnson WB, Lindenstrauss J. Extensions of Lipschitz mappings into a Hilbert space. In: Beals R, Beck A, Bellow A, Hajian A, editors. *Contemporary Mathematics*. vol. 26. Providence, Rhode Island: American Mathematical Society; 1984. p. 189–206.
124. Cooley SM, Hamilton T, Ray JCJ, Deeds EJ. A novel metric reveals previously unrecognized distortion in dimensionality reduction of scRNA-Seq data. *bioRxiv*: 689851; 2020.
125. Kobak D, Berens P. The art of using t-SNE for single-cell transcriptomics. *Nature Communications*. 2019; 10(1):5416. <https://doi.org/10.1038/s41467-019-13056-x> PMID: 31780648
126. Kessler O, Jiang Y, Chasin LA. Order of intron removal during splicing of endogenous adenine phosphoribosyltransferase and dihydrofolate reductase pre-mRNA. *Molecular and Cellular Biology*. 1993; 13(10):6211–6222. <https://doi.org/10.1128/mcb.13.10.6211> PMID: 8413221

127. de la Mata M, Lafaille C, Kornblith AR. First come, first served revisited: Factors affecting the same alternative splicing event have different effects on the relative rates of intron removal. *RNA*. 2010; 16(5):904–912. <https://doi.org/10.1261/rna.1993510> PMID: 20357345
128. Heber S, Alekseyev M, Sze SH, Tang H, Pevzner PA. Splicing graphs and EST assembly problem. *Bioinformatics*. 2002; 18(suppl_1):S181–S188. https://doi.org/10.1093/bioinformatics/18.suppl_1.S181 PMID: 12169546
129. Hagemann-Jensen M, Ziegenhain C, Chen P, Ramsköld D, Hendriks GJ, Larsson AJM, et al. Single-cell RNA counting at allele and isoform resolution using Smart-seq3. *Nature Biotechnology*. 2020; 38(6):708–714. <https://doi.org/10.1038/s41587-020-0497-0> PMID: 32518404
130. Gorin G, Pachter L. Length Biases in Single-Cell RNA Sequencing of pre-mRNA. *bioRxiv*. 2021.07.30.454514; 2021.
131. Schiebinger G, Shu J, Tabaka M, Cleary B, Subramanian V, Solomon A, et al. Reconstruction of developmental landscapes by optimal-transport analysis of single-cell gene expression sheds light on cellular reprogramming. *bioRxiv*: 191056; 2017.
132. Trapnell C, Cacchiarelli D, Grimsby J, Pokharel P, Li S, Morse M, et al. The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering of single cells. *Nature Biotechnology*. 2014; 32(4):381–386. <https://doi.org/10.1038/nbt.2859> PMID: 24658644
133. Wolf FA, Hamey FK, Plass M, Solana J, Dahlin JS, Göttgens B, et al. PAGA: graph abstraction reconciles clustering with trajectory inference through a topology preserving map of single cells. *Genome Biology*. 2019; 20(1):59. <https://doi.org/10.1186/s13059-019-1663-x> PMID: 30890159
134. Lönnberg T, Svensson V, James KR, Fernandez-Ruiz D, Sebina I, Montandon R, et al. Single-cell RNA-seq and computational analysis using temporal mixture modeling resolves Th1/Tfh fate bifurcation in malaria. *Science Immunology*. 2017; 2(9):eaal2192. <https://doi.org/10.1126/scimmunol.aal2192> PMID: 28345074
135. Qiu X, Hill A, Packer J, Lin D, Ma YA, Trapnell C. Single-cell mRNA quantification and differential analysis with Census. *Nature Methods*. 2017; 14(3):309–315. <https://doi.org/10.1038/nmeth.4150> PMID: 28114287
136. Campbell KR, Yau C. A descriptive marker gene approach to single-cell pseudotime inference. *Bioinformatics*. 2019; 35(1):28–35. <https://doi.org/10.1093/bioinformatics/bty498> PMID: 29939207
137. Haghverdi L, Buettner F, Theis FJ. Diffusion maps for high-dimensional single-cell analysis of differentiation data. *Bioinformatics*. 2015; 31(18):2989–2998. <https://doi.org/10.1093/bioinformatics/btv325> PMID: 26002886
138. Weinreb C, Wolock S, Tusi BK, Socolovsky M, Klein AM. Fundamental limits on dynamic inference from single-cell snapshots. *Proceedings of the National Academy of Sciences*. 2018; 115(10):E2467–E2476. <https://doi.org/10.1073/pnas.1714723115> PMID: 29463712
139. Deconinck L, Cannoodt R, Saelens W, Deplancke B, Saeyns Y. Recent advances in trajectory inference from single-cell omics data. *Current Opinion in Systems Biology*. 2021; 27:100344. <https://doi.org/10.1016/j.coisb.2021.05.005>
140. Zhang J, Nie Q, Zhou T. Revealing Dynamic Mechanisms of Cell Fate Decisions From Single-Cell Transcriptomic Data. *Frontiers in Genetics*. 2019; 10:1280. <https://doi.org/10.3389/fgene.2019.01280> PMID: 31921315
141. Pitman JW. Occupation Measures for Markov Chains. *Advances in Applied Probability*. 1977; 9(1):69–86. <https://doi.org/10.2307/1425817>
142. Yang Y, Nurbekyan L, Negrini E, Martin R, Pasha M. Optimal Transport for Parameter Identification of Chaotic Dynamics via Invariant Measures. *arXiv*: 2104.15138; 2021.
143. Kuntz J, Thomas P, Stan GB, Barahona M. The Exit Time Finite State Projection Scheme: Bounding Exit Distributions and Occupation Measures of Continuous-Time Markov Chains. *SIAM Journal on Scientific Computing*. 2019; 41(2):A748–A769. <https://doi.org/10.1137/18M1168261>
144. Van Kampen NG. Stochastic Processes in Physics and Chemistry. 3rd ed. Elsevier; 2007.
145. Abramowitz M, Stegun I, editors. *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*. 9th ed. United States National Bureau of Standards; 1970.
146. Vastola JJ. In search of a coherent theoretical framework for stochastic gene regulation. Vanderbilt; 2021.
147. Gorin G, Vastola JJ, Fang M, Pachter L. Interpretable and tractable models of transcriptional noise for the rational design of single-molecule quantification experiments. *bioRxiv*. 2021.09.06.459173; 2021.
148. Gorin G, Pachter L. *Monod*: mechanistic analysis of single-cell RNA sequencing count data. *bioRxiv*. 2022.06.11.495771; 2022.

149. Davis ME, Davis RJ. Fundamentals of chemical reaction engineering. International ed ed. McGraw-Hill chemical engineering series. Boston: McGraw-Hill; 2003.
150. Gillespie DT. A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. *Journal of Computational Physics*. 1976; 22(4):403–434. [https://doi.org/10.1016/0021-9991\(76\)90041-3](https://doi.org/10.1016/0021-9991(76)90041-3)
151. Munsky B, Khammash M. The finite state projection algorithm for the solution of the chemical master equation. *The Journal of Chemical Physics*. 2006; 124(4):044104. <https://doi.org/10.1063/1.2145882> PMID: 16460146
152. Wilkinson DJ. Stochastic modelling for systems biology. Third edition ed. Chapman & Hall/CRC mathematical and computational biology. Boca Raton: CRC Press, Taylor & Francis Group; 2019.
153. Phillips R, Kondev J, Theriot J, Garcia HG. Physical biology of the cell. 2nd ed. New York, NY: Garland Science; 2013.
154. Breda J, Zavolan M, van Nimwegen E. Bayesian inference of gene expression states from single-cell RNA-seq data. *Nature Biotechnology*. 2021; 39(8):1008–1016. <https://doi.org/10.1038/s41587-021-00875-x> PMID: 33927416
155. Gayoso A, Lopez R, Xing G, Boyeau P, Valiollah Pour Amiri V, Hong J, et al. A Python library for probabilistic analysis of single-cell omics data. *Nature Biotechnology*. 2022. <https://doi.org/10.1038/s41587-021-01206-w> PMID: 35132262
156. Jia C, Grima R. Accuracy and limitations of extrinsic noise models to describe gene expression in growing cells. *bioRxiv*: 2022.06.15.496247; 2022.
157. Grima R, Schmidt DR, Newman TJ. Steady-state fluctuations of a genetic feedback loop: An exact solution. *The Journal of Chemical Physics*. 2012; 137(3):035104. <https://doi.org/10.1063/1.4736721> PMID: 22830733
158. Cao Z, Grima R. Analytical distributions for detailed models of stochastic gene expression in eukaryotic cells. *Proceedings of the National Academy of Sciences*. 2020; 117(9):4682–4692. <https://doi.org/10.1073/pnas.1910888117> PMID: 32071224
159. Veloso A, Kirkconnell KS, Magnuson B, Biewen B, Paulsen MT, Wilson TE, et al. Rate of elongation by RNA polymerase II is associated with specific gene features and epigenetic modifications. *Genome Research*. 2014; 24(6):896–905. <https://doi.org/10.1101/gr.171405.113> PMID: 24714810
160. Zhang X, Jin H, Yang Z, Lei J, 1 School of Mathematics and Systems Science, Beihang University, Beijing 100191, 2 Zhou Pei-Yuan Center for Applied Mathematics, Tsinghua University, Beijing 100084, et al. Effects of elongation delay in transcription dynamics. *Mathematical Biosciences and Engineering*. 2014; 11(6):1431–1448. <https://doi.org/10.3934/mbe.2014.11.1431> PMID: 25365608
161. Coté A, Coté C, Bayatpour S, Drexler HL, Alexander KA, Chen F, et al. pre-mRNA spatial distributions suggest that splicing can occur post-transcriptionally. *bioRxiv*: 2020.04.06.028092; 2021.
162. Choubey S. Nascent RNA kinetics: Transient and steady state behavior of models of transcription. *Physical Review E*. 2018; 97(2):022402. <https://doi.org/10.1103/PhysRevE.97.022402> PMID: 29548128
163. Gorin G, Wang M, Golding I, Xu H. Stochastic simulation and statistical inference platform for visualization and estimation of transcriptional kinetics. *PLOS ONE*. 2020; 15(3):e0230736. <https://doi.org/10.1371/journal.pone.0230736> PMID: 32214380
164. Melsted P, Ntranos V, Pachter L. The barcode, UMI, set format and BUStools. *Bioinformatics*. 2019; p. btz279. PMID: 31073610
165. Thorne BC, Bailey AM, Peirce SM. Combining experiments with multi-cell agent-based modeling to study biological tissue patterning. *Briefings in Bioinformatics*. 2007; 8(4):245–257. <https://doi.org/10.1093/bib/bbm024> PMID: 17584763
166. Reichl LE. A Modern Course in Statistical Physics. 4th ed. Wiley-VCH Verlag GmbH & Co. KGaA; 2016.
167. Das B, Mitra P. High-Performance Whole-Cell Simulation Exploiting Modular Cell Biology Principles. *Journal of Chemical Information and Modeling*. 2021; 61(3):1481–1492. <https://doi.org/10.1021/acs.jcim.0c01282> PMID: 33683902
168. Thornburg ZR, Bianchi DM, Brier TA, Gilbert BR, Earnest TM, Melo MCR, et al. Fundamental behaviors emerge from simulations of a living minimal cell. *Cell*. 2022; 185(2):345–360.e28. <https://doi.org/10.1016/j.cell.2021.12.025> PMID: 35063075
169. Johnson NL, Kotz S, Balakrishnan N. Continuous univariate distributions, Vol. 1. 2nd ed. Wiley series in probability and mathematical statistics. New York: Wiley; 1994.
170. Saelens W, Cannoodt R, Todorov H, Saeys Y. A comparison of single-cell trajectory inference methods. *Nature Biotechnology*. 2019; 37(5):547–554. <https://doi.org/10.1038/s41587-019-0071-9> PMID: 30936559

171. Gutin G, Punnen AP, editors. *The Traveling Salesman Problem and Its Variations*. 1st ed. No. 12 in Combinatorial Optimization. Springer-Verlag; 2007.
172. Liu Z, Lou H, Xie K, Wang H, Chen N, Aparicio OM, et al. Reconstructing cell cycle pseudo time-series via single-cell transcriptome data. *Nature Communications*. 2017; 8(1):22. <https://doi.org/10.1038/s41467-017-00039-z> PMID: 28630425
173. Keizer EM, Bastian B, Smith RW, Grima R, Fleck C. Extending the linear-noise approximation to biochemical systems influenced by intrinsic noise and slow lognormally distributed extrinsic noise. *Physical Review E*. 2019; 99(5):052417. <https://doi.org/10.1103/PhysRevE.99.052417> PMID: 31212540
174. Lindsay BG. Mixture Models: Theory, Geometry and Applications. NSF-CBMS Regional Conference Series in Probability and Statistics. 1995; 5:i–163.
175. Campbell KR, Yau C. Order Under Uncertainty: Robust Differential Expression Analysis Using Probabilistic Models for Pseudotime Inference. *PLOS Computational Biology*. 2016; 12(11):e1005212. <https://doi.org/10.1371/journal.pcbi.1005212> PMID: 27870852
176. Folia MM, Rattray M. Trajectory inference and parameter estimation in stochastic models with temporally aggregated data. *Statistics and Computing*. 2018; 28(5):1053–1072. <https://doi.org/10.1007/s11222-017-9779-x> PMID: 30147250
177. Ahmed S, Rattray M, Boukouvalas A. GrandPrix: scaling up the Bayesian GPLVM for single-cell data. *Bioinformatics*. 2019; 35(1):47–54. <https://doi.org/10.1093/bioinformatics/bty533> PMID: 30561544
178. Qiu P. Embracing the dropouts in single-cell RNA-seq analysis. *Nature Communications*. 2020; 11(1):1169. <https://doi.org/10.1038/s41467-020-14976-9> PMID: 32127540
179. Desai RV, Chen X, Martin B, Chaturvedi S, Hwang DW, Li W, et al. A DNA repair pathway can regulate transcriptional noise to promote cell fate transitions. *Science*. 2021; 373(6557):eabc6506. <https://doi.org/10.1126/science.abc6506> PMID: 34301855