GraphVelo allows for accurate inference of multimodal velocities and molecular mechanisms for single cells

Yuhao Chen[1,2,a], Yan Zhang[2,a], Jiaqi Gan[2,a], Ke Ni[2], , Ming Chen[1,3,b], Ivet Bahar[4,b], Jianhua Xing[2,5,6,b]

1. *Department of Bioinformatics, College of Life Sciences, Zhejiang University, Hangzhou, 310058, China*
2. *Department of Computational and Systems Biology, School of Medicine, University of Pittsburgh, Pittsburgh, PA, USA*
3. *Zhejiang Key Laboratory of Multi-omics Precision Diagnosis and Treatment of Liver Diseases, Department of General Surgery, Sir Run-Run Shaw Hospital, Zhejiang University School of Medicine, 310016, Hangzhou, China.*
4. *Laufer Center for Physical and Quantitative Biology, and Department of Biochemistry and Cell Biology, Renaissance School of Medicine, Stony Brook University, Stony Brook, NY, USA*
5. *University of Pittsburgh Hillman Cancer Center, Pittsburgh, PA, USA*
6. *Department of Physics and Astronomy, University of Pittsburgh, Pittsburgh, PA, USA*

[a] *Equal contribution*

[b] *Corresponding authors:* xing1@pitt.edu, bahar@laufercenter.org, mchen@zju.edu.cn.

Abstract

RNA velocities and generalizations emerge as powerful approaches for extracting time-resolved information from high-throughput snapshot single-cell data. Yet, several inherent limitations restrict applying the approaches to genes not suitable for RNA velocity inference due to complex transcriptional dynamics, low expression, or lacking splicing dynamics, or data of non-transcriptomic modality. Here, we present GraphVelo, a graph-based machine learning procedure that uses as input the RNA velocities inferred from existing methods and infers velocity vectors lying in the tangent space of the low-dimensional manifold formed by the single cell data. GraphVelo preserves vector magnitude and direction information during transformations across different data representations. Tests on multiple synthetic and experimental scRNA-seq data including viral-host interactome and multi-omics datasets demonstrate that GraphVelo, together with downstream generalized dynamo analyses, extends RNA velocities to multi-modal data and reveals quantitative nonlinear regulation relations between genes, virus and host cells, and different layers of gene regulation.

Introduction

Cells need to constantly detect and adapt to changes in extracellular and intracellular environment. Regulation of their gene transcription is a common mechanism of response. Multiple factors affect the transcriptional activity of eukaryotic genes, including *cis* and *trans* regulatory elements and chromatin structure. High throughput single cell sequencing data provide the landscape of cell genotype. These data lack, however, information on how the genome state changes over time Continuous efforts have been made to extract information about gene regulation and developing methods for connecting the cell states to temporal sequences of events captured by single-cell snapshot data. One group of methods that has received extensive attention is that based on RNA velocity[1] for predicting the changes in RNA expression states in the cell. The original RNA velocity method leverages the ratio between nascent and mature transcripts to estimate the rate of change in gene expression. This seminal study has inspired numerous methods for improved RNA velocity estimation based on information from splicing[2-6], metabolic labeling[7,8], lineage tracing[9], and transcriptional factor binding[10].

The RNA velocity framework has, however, its own inherent limitations. First, none of the RNA velocity estimation methods could be applied to any single cell transcriptomic data without restrictions. For example, the splicing-based method is not applicable prokaryotes or viruses, or organisms without introns. Erroneous inferences of RNA velocities have also been noticed for genes having complex splicing dynamics[11]. Furthermore, it is difficult to estimate the RNA velocities of genes with low expression, which excludes most transcription factors. Second, multi-omics sequencing technologies provide multifaceted information on cellular states alongside the transcriptome modality, and currently there are limited systematic methods to extend such velocity estimations to other modalities[12,13].

The single cell transcriptome state is usually defined by the instantaneous distribution of RNA levels, represented by a multidimensional vector of RNA levels for all (measurable) genes. The usual practice is to map such single cell data onto a reduced space, e.g., transform from a principal component (PC) space to a UMAP representation to facilitate the visualization of the time-evolution of cell state, each cell state being represented by a point in that space. Numerous dimension reduction and manifold learning algorithms have been developed for such representation transformation. In comparison, transforming a velocity vector between

representations is a nontrivial task not rigorously addressed in the single cell field. Even worse, a visually correct vector field does not necessarily imply accurate high-dimensional velocity estimation[14]. La Manno et al. proposed a cosine kernel method to address this challenge, which has been adopted since then in most subsequent studies[1]. Li et al. mathematically proved that the cosine kernel asymptotically gives the correct direction of a velocity vector in the large sampling limit, but the magnitude information is completely lost due to a normalization procedure[15]. This loss of information casts concerns when such quantitative information is needed.

In this study, we tackle the above challenges through a graph-theoretical representation of RNA velocities, called GraphVelo, with dynamical systems underpinnings. GraphVelo takes an ansatz that the measured single cell expression profiles and inferred RNA velocities collectively reflect a dynamical process and are connected through a set of dynamical equations. It exploits such additional constraints that couples high dimensional velocity field and single cell state manifold, and enables the generalization of the approach in the context of multi-modal single cell data. While the combined expression and velocity information has been widely used to infer cell state transition trajectories[2,16], GraphVelo presents the advantage of enabling downstream analyses such as that performed by dynamo[7] to extract quantitative information on causal gene-gene relations that dictate the cell state transitions. Benchmarking of the proposed graph framework against simulated and experimental single cell data lends support to its broad utility.

Results

GraphVelo infers manifold-consistent single cell velocity vectors through tangent space projection and transforms between representations through local linear embedding

Consider that the internal state of a cell can be specified by an N-dimensional state vector $\boldsymbol{x}$, with $N >> 1$ generally. Assume that the temporal evolution of the cell state follows a continuous and smooth curve $\boldsymbol{x}(t)$ (see Methods for more general mathematical formulation). The instant velocity vector $\boldsymbol{v}(\boldsymbol{x},t) = \mathrm{d}\boldsymbol{x}(t)/\mathrm{d}t$ is always tangent to the curve of $\boldsymbol{x}(t)$ (as a function of t) at $\boldsymbol{x}$. One can generalize to the situation that the trajectories of a swarm of cells form a M-dimensional manifold $\mathcal{M}(\boldsymbol{x})$ embedded in the N-dimensional state space with $M << N$ typically, as revealed by high-throughput single cell omics data. Then under the ansatz that a velocity vector $\boldsymbol{v}(\boldsymbol{x},t)$ dictates the evolution of a state vector $\boldsymbol{x}(t)$, $\boldsymbol{v}$ must lie in the tangent space of $\mathcal{M}(\boldsymbol{x})$, denoted as $T_p\,\mathcal{M}$. In

practice, the RNA velocity vectors inferred from existing methods do not automatically satisfy this tangent space requirement (but see[4]).

Taking various inferred single cell RNA velocity vectors, e.g. splicing-based, metabolic labeling-based, or lineage tracing-based, as input, GraphVelo takes advantage of the nature of the low-dimensional cell state manifold to: 1) refine the estimated RNA velocity to satisfy the tangent space requirement; 2) infer the velocities of non-transcriptomic modalities using RNA velocities. GraphVelo thus serves as a plugin that can be seamlessly integrated into existing RNA velocity analysis pipelines, and help process single cell data for downstream cellular dynamics analyses using methods such as dynamo (Fig. 1a).

GraphVelo approximates the tangent space at a cell state $\boldsymbol{x}$ by a k-nearest neighbor (kNN) graph following the local linear embedding algorithm[17], and uses the more reliable data manifold $\mathcal{M}$ to refine the velocity vectors by imposing the constraint that the N-dimensional velocity vector $\boldsymbol{v}$ should lie in the tangent space (Fig. 1b&c). Consider a given point $\boldsymbol{x}_i$ on a manifold corresponding to the expression state i of the single cell. Its infinitesimal neighborhood forms a Euclid space that approximates the tangent space $T_p\mathcal{M}$. With sufficient sampling of the neighboring cell states j in the state space, the incremental displacement vectors between cell state i and its neighboring cell states, $\boldsymbol{\delta}_{ij} = \boldsymbol{x}_j - \boldsymbol{x}_i$, form a set of complete albeit possibly redundant and nonorthogonal/non-normalized basis vectors of the Euclidean space in the local region. Then the projection of the measured velocity vector onto $T_p\mathcal{M}$ can be expressed as a linear combination (see also Supplemental Text I),

$$\boldsymbol{v}_{\parallel}(\boldsymbol{x}_i) = \sum_{j\in\mathcal{N}_i} \phi_{ij}\boldsymbol{\delta}_{ij}, \tag{1}$$

where $\mathcal{N}_i$ is the neighborhood of cell state i, defined by its k nearest neighbors in the feature space determined by sequencing profiles. Direct application of eq. 1 to determine the coefficients ϕ_{ij} is numerically unstable in real data (see Supplemental Text II for detailed discussion). Instead, we performed the projection by optimizing the following tangent space projection (TSP) loss function (Fig. 1c),

$$\mathcal{L}(\boldsymbol{\phi}_i) = \left\|\boldsymbol{v}_i - \boldsymbol{v}_{\parallel i}\right\|^2 - b\cdot\cos(\boldsymbol{\phi}_i, \boldsymbol{\phi}_i^{corr}) + \lambda\|\boldsymbol{\phi}_i\|^2, \tag{2}$$

where $\|\cdot\|$ refers to vector modulus. $\boldsymbol{\phi}_i^{corr}$ is a heuristic "cosine kernel" widely used in the RNA velocity analyses for projecting velocity vectors onto a reduce space, the elements of which are

ϕ_{ij} $(with\ j \in \mathcal{N}_i)$(see also Supplemental Text III); the second term $\cos(\cdot,\cdot)$ denotes the cosine similarity. The first term in the loss function learns the correctly-scaled velocity magnitudes, and the second term retains the reliable direction information based on previous mathematical analyses showing that $\boldsymbol{\phi}_i^{corr}$ asymptotically gives correct direction of the velocity vector[15]. The L-2 regularization is used to bound parameters ϕ_i . b and λ are two hyperparameters for retaining the direction and for regularization, respectively.

With local linear embedding, it is straightforward to transform velocity between different representations. Assuming a mapping function f exists connecting manifold $\mathcal{M}$ and $\aleph$ such that for cell i with state vector $\boldsymbol{x}_i$ in $\mathcal{M}$, the coordinate of the same cell in $\aleph$ is given as $\boldsymbol{y}_i = f(\boldsymbol{x}_i)$. Since a given local patch of continuous manifold is approximated by a Euclidean space, a locally linear transformation connects the patch in the two representations. Consequently, for a vector described by eq. 1 in $\mathcal{M}$, the velocity vector in $\aleph$ is,

$$\boldsymbol{v}_{\parallel}(\boldsymbol{y}_i) = \sum_{j \in \mathcal{N}_i} \phi_{ij} \boldsymbol{\delta}_{ij}^{'}, \tag{3}$$

where $\boldsymbol{\delta}_{ij}' = \boldsymbol{y}_j - \boldsymbol{y}_i$.

Therefore, eq. 1-3 form the mathematical and computational foundation of GraphVelo. With eq. 3 one can extend velocity inference to datasets that velocity inference is not traditionally applicable such as host-virus interactome and multiomics datasets based on the Whitney embedding theorem[18]. Details of the mathematical foundation were given in Methods. With velocity vectors refined with GraphVelo, one can readily perform downstream analyses, as exemplified in Fig.1d-f, which will be further elaborated below in the context of specific applications.

Benchmark studies demonstrate effectiveness of GraphVelo across simulation datasets with diverse topology

To demonstrate the effectiveness of the geometry-constrained projection, we first benchmarked our method on a 3D bifurcation system constrained on a 2D manifold (Methods). We added a random component vertical to the tangent plane to mimic the noise. The resulting velocity vectors inferred by GraphVelo through minimizing the TSP loss were consistent with the ground truth vectors (Fig. 2a). Both GraphVelo and cosine kernel successfully removed the normal components (Fig. 2b i) and maintained the directional information (Fig. 2b ii), but only GraphVelo kept the velocity magnitude information (Fig. 2b iii).

Next, we performed multifaceted evaluations of the ability of GraphVelo to robustly recover the transcriptional dynamics across a range of simulated datasets with different underlining phenotypic structures. We used dyngen[19], a multi-modal scRNA-seq simulation engine, to generate gene-wise dynamics defined by gold-standard transcriptional regulatory networks (Methods). We generated simulated scRNA-seq data for networks with a variety of underlying linear, cyclic, and bifurcating topological structures, and recovered the corresponding vector field using GraphVelo-corrected velocity vectors (Fig. 2c-e). To comprehensively assess the outcome, we used three diverse metrics, cosine similarity, root-mean-square error (RMSE), and accuracy, which evaluate the correctness of velocity direction, magnitude, and sign, respectively. We presented (Fig. 2c-e) the comparative results obtained with the cosine kernel and with GraphVelo (TSP with (i.e., eq. 2 with $b \neq 0$) and without (eq. 2 with $b = 0$) the cosine regularization term). By minimizing TSP loss, GraphVelo preserved both the direction and magnitude of the vector field (Fig. 2f-h). With an increase of noise level by adding Gaussian noise to the ground truth vectors, GraphVelo refined the distorted velocity and outperformed the cosine kernel projection consistently (Extended Data Fig. 1a-c).

Then, we tested whether manifold constraints could preserve the speed of the cell progression across different representations. GraphVelo was able to scale velocity vectors between the original space and the PCA space, showing a high correlation with the ground truth, even as noise levels increased, whereas the cosine kernel failed (Extended Data Fig. 1d). The results on UMAP showed less agreement, which is not surprising. UMAP is a convenient representation for visualizing single cell data but not designed for representing quantitative cell state transition dynamics since UMAP is not a continuous transformation from the original gene space and cannot preserve local distances after projection.

To further explore whether GraphVelo could correct the RNA velocity estimated by the splicing kinetics, we took the velocity inferred using different packages (scVelo[2], dynamo[7] and VeloVI[5]) as input. The output from GraphVelo agreed significantly better with the ground truth compared to the raw input (Extended Data Fig. 1e), highlighting the significant improvement achieved by GraphVelo in evaluating both the direction and magnitude of the velocity vector fields across all datasets.

184 **GraphVelo infers quantitative whole genome RNA velocity from a subset of genes with**
185 **manifold-consistent RNA turnover kinetics**

186 Most RNA velocity methods are based on biophysical models of mRNA turnover dynamics with
187 specific assumptions that may break down in certain cases[11]. For example, the splicing-based RNA
188 velocity may have an erroneous sign for processes under active regulation on mRNA degradation
189 or promotors switching between states with different transcription efficiency (Fig. 3a, Extended
190 Data Fig. 2). GraphVelo first uses the RNA velocities inferred from any method as input, then uses
191 the velocities of high-confidence genes to infer velocities of other genes. One can use several
192 existing approaches on evaluating the confidence scores of inferred RNA velocity values of genes[7].
193 Alternatively, we identified a subset of Manifold-consistent Kinetics (MacK) Genes based on their
194 agreement with prior knowledge or additional information acquired from other methods such as
195 lineage tracing (Fig. 3b),

196 We first applied GraphVelo to a mouse erythroid maturation dataset[20]. This study provided a
197 transcriptional landscape of the erythroid lineage with well-documented differentiation trajectory
198 during mouse gastrulation. Previous analyses have shown that the dataset contains genes with
199 multiple rate kinetics, leading to erroneous prediction of the cell state transition direction[20,21]. We
200 selected the top 200 out of 450 velocity genes as MacK genes, representing those with robustly
201 estimated velocities (see Methods for details). The projected vector field in UMAP showed
202 consistency with prior knowledge in developmental biology (Fig. 3c). We then used the corrected
203 RNA velocities for dynamo velocity field analyses. The vector field-based pseudotime accurately
204 predicted the lineage with scRNA-seq data of temporal mouse embryos (Fig. 3d).

205 Previous studies identified multiple rate kinetics (MURK) genes showing transcription bursts in
206 the middle of erythroid differentiation[21]. For example, *Smim1* and *Hba-x* are two MURK genes
207 showing complex patterns of phase portrait (Fig. 3f i). Consequently, the RNA velocity of *Simi1*
208 inferred with scVelo was negative along a major part of the developmental axis (Fig. 3f ii),
209 contradicting the trend of increasing *Simi1* mRNA levels (Fig. 3f iv). For *Hba-x*, scVelo even
210 failed to infer its RNA velocity. On the other hand, GraphVelo inferred velocities and predicted
211 correct kinetic patterns of these genes (Fig. 3f iii). Similar performances have been observed in
212 other MURK genes (Extended Data Fig. 3a). To examine the overall prediction of cell state
213 transitions from transcription burst genes, we projected the MURK genes velocity inferred from

214 GraphVelo and scVelo to the predefined UMAP. The velocities from GraphVelo but not scVelo
215 correctly captured the directional flow of differentiation using only MURK genes (Extended Data
216 Fig. 3b).

217 Next, we examined the inferred RNA velocities of an entire set of 2,000 highly variable genes.
218 For each gene we calculated the cross-boundary correctness (CBC) score, which quantifies how
219 likely, following its current velocity, a cell can develop to a target cell state[22]. GraphVelo
220 outperformed several existing tools that only used hundreds of 'well-estimated' kinetics genes (Fig.
221 3g). Furthermore, we estimated the speed of cell state transition using the norm of velocity vector
222 in high-dimensional space and identified the transcriptional surge stage (Extended Data Fig. 3c).
223 We hypothesized that the MURK genes, which exhibited a sudden increase in transcription rate
224 during this stage, were responsible for the sharp acceleration in cell state transition speed. Using
225 dynamo, we estimated the acceleration derived from the GraphVelo vector field and found that the
226 acceleration value, as the derivative of the velocity vector, demonstrated its potential as a predictor
227 for transcription burst genes (Extended Data Fig. 3d).

228 With the velocity estimation extended to the whole gene space, we were able to perform
229 comprehensive mechanistic analyses on the entire genome spectrum. First, we calculated the
230 MacK score for each gene using the corrected RNA velocities. We hypothesized that a gene with
231 a higher MacK score indicated a better agreement between its RNA velocity vector and the
232 developmental axis, suggesting that the gene served as a potential lineage-driver gene. We ranked
233 genes based on their scores and performed GO biological process enrichment analyses for the top
234 genes. Indeed, the enriched processes were associated with erythropoiesis, including the heme
235 biosynthetic process and interleukin-12-mediated signaling pathway[23] (Fig. 3e).

236 Next, we applied dynamo to perform differential geometry analyses of the vector field and
237 mechanistically dissected the activation cascade of erythroid marker gene *Klf1* (Extended Data Fig.
238 4a&b). Jacobian analyses based on GraphVelo vector field revealed sequential activation of driver
239 transcription factors (TFs) *Gata2*, *Gata1*, and *Klf1* during erythroid lineage differentiation, with
240 *Gata1* subsequently repressing the expression of *Gata2* (Fig. 3h, Extended Data Fig. 4c)[7]. To
241 further demonstrate the crucial role of transcriptional factor *Gata1* during erythropoiesis, we
242 performed *in silico* genetic perturbation across all cells. Results showed that both inhibiting *Gata1*
243 and upregulating the *Gata1* repressor *Spi1* lead to a reversal of normal developmental flow (Fig.

3i, Extended Data Fig. 4d). The above analyses collectively suggest that activation of *Gata1* in the blood progenitors biased its differentiation to erythropoiesis, agreeing with experimental reports[21].

To further evaluate GraphVelo, we tested the method on another dataset of human bone marrow development[24]. This developmental process has complex progressions from hematopoietic stem cells (HSCs) to three distinct branches: erythroid, monocyte, and common lymphoid progenitor (CLP). Again, we used the top 100 out of 454 velocity genes as MacK genes to predict the RNA velocities of 2,000 highly variable genes. The GraphVelo velocity field accurately recovered the fate of cells on the sophisticated transcriptional landscape in contrast to scVelo (Fig. 3j and k, Extended Data Fig. 5a). By combining the likelihood estimated by scVelo with the MacK score, we identified rapid degradation and transcription burst genes whose dynamics deviated from the RNA velocity assumptions (Extended Data Fig. 5b&c). *ANGPT1* and *RBPMS* are two examples which were overall highly expressed in the progenitors and decreased quickly along the trajectories (Fig. 3l), reminiscent of what was shown in Fig. 3a. These genes misled RNA velocity inference with scVelo assuming a constant degradation rate constant. GraphVelo revealed a cell context-specific transcription rate $\alpha = u + \frac{du}{dt}$ and degradation constant $\gamma = (u - \frac{ds}{dt})/s$, thus a degradation wave along the differentiation path with (Fig. 3l, Extended Data Fig. 5d), consistent with simulation result and reports on regulation of *ANGPT1* mRNA by microRNAs such as *miRNA-153-3p*[25] (Extended Data Fig. 2d).

GraphVelo reconstructs host–pathogen transcriptome dynamics from infection trajectory

Continuous battle between human immune surveillance and viral immune evasion takes place in the host cell system after viral entry. scRNA-seq data provide a massive and parallel way of assessing the time evolution of both host and viral transcripts, unraveling the delicate inherent dynamic of a virus-host system[26,27]. While existing splicing-based methods can robustly estimate RNA velocity of host coding genes, it is infeasible to infer the dynamics of the viral transcriptome due to lack of intron in the viral genome. GraphVelo enables inference of the velocities of virus RNA abundance based on the kinetics of host transcripts velocities, as illustrated next.

We analyzed a human cytomegalovirus (HCMV) viral infection dataset to learn viral transcriptomic kinetics in monocyte-derived dendritic cells (moDCs)[28]. The result from GraphVelo unraveled how viral infection progressed along the transcriptional space (Fig. 4a). The

273 velocity vectors pointed to directions consistent with an increasing trend of the percentage of viral
274 RNAs in a single cell, which inherently served as an indicator of the infection time course[29].
275 Compared to the trend obtained with the raw RNA velocities from scVelo, the vector field-based
276 pseudotime calculated using GraphVelo-corrected RNA velocities consistently showed higher
277 correlation with the (pseudo)temporal progression of viral infection as reflected by viral RNA
278 percentage (Fig. 4b).

279 Furthermore, the examination of individual genes revealed that the RNA velocities consistently
280 predicted the trend of the mRNA expression level change with increasing virus load (Fig. 4c,
281 Extended Data Fig. 6a&b). Most viral genes started with a fast-increase phase, and the expressions
282 of some genes (e.g. *UL22A*) gradually saturated at high virus load, together with the corresponding
283 RNA velocities approaching zero. One exception is *UL122*, whose expression profile increased
284 first then decreased to a steady state level lower than the peak value. This overshooting is
285 characteristic of a negative feedback network structure[30]. Indeed, a recent study reported that
286 *UL122* negatively regulates its own promotor[31]. Furthermore, comparison of MacK scores across
287 GraphVelo, the CellRank pseudotime kernel, and randomized prediction showed that GraphVelo-
288 computed viral RNA velocities aligned best with the transcriptome gradient of viral load (Fig. 4d).
289 Note that the MacK score also served as a reliable predictor for dynamics-driving factors,
290 specifically viral genes in this case (Extended Data Fig. 6d).

291 We further quantified the velocity norm of all viral factors as infection speed, and observed that
292 the transcription of viral factors was significantly restricted initially, then gradually increased
293 along the trajectory (Extended Data Fig. 6e). Interestingly, most of the genes that exhibited positive
294 correlation with the infection speed were related to viral DNA synthesis, while those negatively
295 correlated to the infection speed were engaged in host viral defense response[32](Fig. 4e, Extended
296 Data Fig. 6f).

297 **GraphVelo identifies host genomic response modules and predicts host-virus gene**
298 **interactions**

299 With GraphVelo-inferred RNA velocities, we probed the time evolution of lytic infection and the
300 complex interplay between host and viral functional genomes. By fitting the GraphVelo velocity
301 trends along the viral load axis, we identified genes with similar kinetic patterns (Fig. 4c). Using
302 the smoothened velocity trends to calculate the distances, we clustered the genes into seven major

303 modules and visualized them on the UMAP space (Fig. 4f). Not surprisingly, viral genes were
304 concentrated in several enclosed regions, indicating that they formed distinct functional genomic
305 modules during the lytic cycle[29]. The genes showed two major dynamical features: acceleration
306 and deceleration along the viral load axis (Fig. 4g).

307 To systematically investigate whether the dichotomy between host kinetics genes and viral genes
308 share similar dynamics, we performed gene functional enrichment analyses of host genes residing
309 close to the viral gene clusters. Genes located in the deceleration part were associated with
310 repressing viral genome replication, which includes negative regulation of viral life cycle and
311 known restriction factors in antiviral responses activated in DCs such as the induction of cytokine
312 and chemokine responses as well as interactions with neutrophils[33,34]. In parallel, the toll-like
313 receptor signaling pathway, required for antiviral defense of the host, was arrested[35]. Neutrophil
314 related processes, which typically cooperate closely with DCs to modulate adaptive immune
315 responses[36], were suppressed. Therefore, the deceleration part also showed how critical set of host
316 factors were silenced by viral entry to achieve immune evasion.

317 The acceleration groups, on the other hand, demonstrated how viruses hijacked the host cell
318 endogenous cellular programs for virus replication. Notably, the pathways related to viral genome
319 replication were triggered, promoting DNA replication and transcription, such as negative
320 regulation of G1/S transition of mitotic cell cycle[37], cellular response to DNA damage stimulus[38]
321 and regulation of transcription from RNA polymerase II promoter[39]. Cells showed a shift towards
322 a transcriptional signature resembling the G1 phase (Extended Data Fig. 6g), agreeing with
323 previous report on HCMV infection[40]. Along the infection process, antiviral interferon (IFN)-γ
324 response of moDC cells was first activated then suppressed. These results highlighted the
325 organized and antagonistic strategies adopted by both host cells and viruses during their tug-of-
326 war for survival and proliferation.

327 To investigate the crosstalk between host and viral factors systematically in depth, we performed
328 dynamo Jacobian analyses. We scanned the entire spectrum of viral genome and delineated how
329 the HCMV factors silence IFN and NFκB signaling (Fig. 4i). A large proportion of the identified
330 viral factors functioned in evading host cell immune responses, a finding supported by several
331 recent studies[41-43]. *In silico* virus-directed knock out experiments revealed altered accumulation
332 patterns of viral transcripts (Fig. 4j). Notably, inhibition of *UL112*, which ranked first with the

333 total viral RNA inhibition in our analyses, led to a qualitatively distinct trajectory. These results
334 highlight the multifunctional *UL112* locus in the viral genome as a potential target for antiviral
335 intervention[29,44]. The analyses demonstrated potential usage of GraphVelo-inferred velocities for
336 understanding the interactions between viral and host factors, assessing the effects of perturbations
337 on infection, and designing potential antiviral interventions[29].

338 **GraphVelo permits multi-omics velocity inference and chromatin dynamics analyses**

339 The molecular anatomy during cell development entails multiple layers, and how different layers
340 coordinate to regulate gene expression is a fundamental problem. For example, the anagen hair
341 follicle features distinct lineages branching from a central population of progenitor cells. Ma et
342 al[45]. used SHARE-seq to capture both the transcriptome and the epigenome data simultaneously
343 for the lineage commitment process from transit-amplifying cells (TACs) to the inner root sheath
344 (IRS), cuticle layer, and medulla. Upon robust selection of estimated genes following dynamo
345 criteria (Methods), we further refined the RNA velocities of these genes through tangent space
346 projection and obtained the chromatin open/close dynamics from the corresponding scATAC data
347 using GraphVelo. The resultant vector field in the combined transcriptome-epigenome space
348 proved to reconstruct the correct multilineages differentiation paths during the anagen phase (Fig.
349 5a).

350 To test the consistency of dynamics across different modalities, we performed CellRank terminate
351 stage analyses[16] from the refined velocity vectors. Using GraphVelo velocities of either the RNA
352 modality or the ATAC modality, we accurately estimated three diverse terminal stages (Fig. 5b).
353 For comparison, we also performed similar analyses using MultiVelo, scVelo with all velocity
354 genes or robustly estimated genes in above GraphVelo studies and pseudotime-based vector field
355 inferred by CellRank. The 2D projection of these vector field functions also exhibited seemingly
356 correct velocity flow direction (Extended Data Fig. 7a). However, none of them captured the cell
357 fate commitment based on coarse-grained transition matrix (Fig. 5b, Extended Data Fig. 7b).
358 Notably, the results from the RNA modality and the ATAC modality of MultiVelo gave
359 inconsistent results. GraphVelo-corrected velocities, on the other hand, helped identify the top-
360 correlating genes towards individual terminal populations which showed agreement with previous
361 study[46] (Fig. 5c, Extended Data Fig. 8).

362 Next, we conducted differential geometry analyses based on the composite GraphVelo vector field.
363 We identified novel root cells, which were also characterized by chromatin potential (Fig. 5d)[45].
364 These novel root cells expressed distinct marker genes compared to the expected root cells using
365 the Wilcoxon test (Fig. 5e). Moreover, we unraveled differentially expressed markers identified
366 by the original study[45], as well as new differentiation-potent genes and validated their initiation
367 properties in another transcriptome dataset (Extended Data Fig. 9a)[46]. To further investigate how
368 these two distinct groups of root cells convert to other cell types, we performed a least action path
369 (LAP) analysis between different cell phenotypes. The expected and novel root cells converted to
370 the IRS terminal state following two distinct least action paths in the vector field (Extended Data
371 Fig. 9b&c). The two paths revealed different temporal change patterns of transcription factor
372 expression profiles (Extended Data Fig. 9d). We calculated the mean-squared displacement (MSD)
373 for every transcription factor to explore the dynamics of TFs along the path from novel root to IRS.
374 The result demonstrated that the fate conversion by novel root was mediated by the Shh-Runx1
375 signaling axis (Fig. 5f, Extended Data Fig. 9d), which has been demonstrated in human embryonic
376 stem cells[47] and is crucial for hair development[48]. In summary, GraphVelo unraveled the multiple
377 molecular mechanisms that orchestrated hair follicle morphogenesis.

378 With available chromatin velocity and RNA velocity, we set to quantify the coupling/decoupling
379 relationships between chromatin structure and gene expression for each gene (see Methods). Here
380 chromatin structure refers to the extent of exposure/accessibility of the gene locus to the
381 environment as indicated by scATACseq data, shortly called open or closed state; and chromatin
382 velocities refer to changes between these states as inferred from scATACseq counts at the
383 examined gene locus. We used dynamic time warping (DTW) distances between the velocities
384 from different omics layers to quantify the similarity between temporal patterns of these two
385 modalities for each gene. A higher DTW value indicates higher similarity. Using the elbow of the
386 ranked distance curve as a cutoff we identified genes that showed decoupled transcription and
387 chromatin structure dynamics. These decoupled genes had an accumulation of cell cycle-
388 dependent (CCD) genes found in previous study[49] (Fig. 5g). This group of genes showed strong
389 involvement in cell cycle-related processes, as indicated by GO enrichment analyses (Fig. 5h).
390 Close examinations indicated that the transcription of cell cycle related genes decreased along the
391 differentiation path, while the chromatin structure at the corresponding loci remained open (Fig.
392 5i, Extended Data Fig. 10). To validate this hypothesis, we further applied GraphVelo to a recently

published 10x Multiome dataset from developing human cortex[50] (Extended Data Fig. 11a). Following the same analyses, we identified decoupled genes and found out that most of these genes were related to cell cycle (Extended Data Fig. 11b-f). which has also been reported in a previous MultiVelo study[13].

We further performed dynamo differential geometry analyses on the composite transcriptome-chromatin vector field. One intriguing phenomenon observed in lineage dynamics is that *Lef1* and *Hoxc13* are the driver TFs correlated with domains of regulatory chromatin (DORCs) of *Wnt3* [45]. Differential geometry analyses on the composite vector field can go beyond correlation analyses and provide an underlying casual mechanism. As a prerequisite for such analyses, GraphVelo-inferred RNA and chromatin velocities of the three genes correctly predicted the trend of change of mRNA and ATAC-seq counts (Extended Data Fig. 12a&b), in contrast to the performance of MultiVelo (Extended Data Fig. 12c). Then, Jacobian analyses on the GraphVelo vector field confirmed that priming activation of *Lef1* subsequently activated the *Hoxc13* TF[51] (Extended Data Fig. 12d&e). Both *Lef1* and *Hoxc13* were found to activate the *Wnt3* target gene, initiating lineage commitment (Extended Data Fig. 12f &g). To quantitatively understand how the two TFs affect *Wnt3* chromatin structure and transcription, we plotted the response heatmap to reflect the distributions of Jacobian elements versus the abundance of mature mRNA for each TF (Fig. 5j). The two terms $\partial f_{Wnt3-\mathrm{chrom}} / \partial x_{Lef1}$ and $\partial f_{Wnt3-\mathrm{chrom}} / \partial x_{Hoxc13}$ started with positive values at low concentrations of TF mRNA copy numbers then decreased to zero, indicating that increasing the level of either TF lead to further opening of the *Wnt3* chromatin region, and the effect saturated at high TF expression. The other two term $\partial f_{Wnt3} / \partial x_{Lef1}$ and $\partial f_{Wnt3} / \partial x_{Hoxc13}$ increased with the TF levels, indicating that these two TFs also activated Wnt3 transcription. Upon integration of the Jacobian elements over regulator expression changes, we obtained the effective dose-response curves obtained (see Methods), which revealed more transparently the TF dose lag between the opening of the target chromatin region and the initialization of transcription (Fig. 5k).

Our results therefore illustrated the sequential events that these two driver TFs, *Lef1 and Hoxc13*, drove as pioneer transcription factors (PTFs) to initiate local chromatin opening and then activated the transcription of *Wnt3*. Notably, computational methods and experimental research confirm that *Lef1* acts as a nucleosome binder and exhibits diverse binding patterns across various cell lines[52].

The *Hox* family of TFs has also been shown to have the capacity to bind their targets in an inaccessible chromatin context and trigger the switch to an accessible state[53], consistent with our analyses that *Hoxc13* revealing that it shared a regulation mechanism similar to that of PTF *Lef1*.

Discussion

In this work, we provided a general framework that extends the framework developed for RNA velocity and related approaches to various data modalities such as proteomics, spatial genomics, 3d genome organization, and imaging data, which were originally beyond reach of this framework. We validated GraphVelo using various *in vivo* cellular kinetics models, confirmed its efficacy and robustness in handling complex and noisy multimodal data. Upon application to various datasets, we unraveled gene regulation relations of an extended list of genes, host-virus gene regulations, and coupling between transcription and local chromatin structures. GraphVelo can be seamlessly integrated with broad downstream analyses, such as dynamo continuous vector field analyses, as well as Markovian analyses using graph dynamo or CellRank.

References

1. La Manno, G. *et al.* RNA velocity of single cells. *Nature* **560**, 494-498 (2018).
2. Bergen, V., Lange, M., Peidli, S., Wolf, F.A. & Theis, F.J. Generalizing RNA velocity to transient cell states through dynamical modeling. *Nat Biotechnol* **38**, 1408-1414 (2020).
3. Li, S. *et al.* A relay velocity model infers cell-dependent RNA velocity. *Nat Biotechnol* **42**, 99-108 (2024).
4. Lederer, A.R. *et al.* Statistical inference with a manifold-constrained RNA velocity model uncovers cell cycle speed modulations. *Nat Methods* (2024).
5. Gayoso, A. *et al.* Deep generative modeling of transcriptional dynamics for RNA velocity analysis in single cells. *Nat Methods* **21**, 50-59 (2024).
6. Gao, C.F., Vaikuntanathan, S. & Riesenfeld, S.J. Dissection and integration of bursty transcriptional dynamics for complex systems. *Proc Natl Acad Sci U S A* **121**, e2306901121 (2024).
7. Qiu, X. *et al.* Mapping transcriptomic vector fields of single cells. *Cell* **185**, 690-711 e45 (2022).
8. Weiler, P., Lange, M., Klein, M., Pe'er, D. & Theis, F. CellRank 2: unified fate mapping in multiview single-cell data. *Nat Methods* **21**, 1196-1205 (2024).
9. Wang, K. *et al.* PhyloVelo enhances transcriptomic velocity field mapping using monotonically expressed genes. *Nat Biotechnol* **42**, 778-789 (2024).
10. Li, J., Pan, X., Yuan, Y. & Shen, H.B. TFvelo: gene regulation inspired RNA velocity estimation. *Nat Commun* **15**, 1387 (2024).
11. Bergen, V., Soldatov, R.A., Kharchenko, P.V. & Theis, F.J. RNA velocity-current challenges and future perspectives. *Mol Syst Biol* **17**, e10282 (2021).
12. Gorin, G., Svensson, V. & Pachter, L. Protein velocity and acceleration from single-cell multiomics experiments. *Genome Biol* **21**, 39 (2020).
13. Li, C., Virgilio, M.C., Collins, K.L. & Welch, J.D. Multi-omic single-cell velocity models epigenome-transcriptome interactions and improves cell fate prediction. *Nat Biotechnol* **41**, 387-398 (2023).
14. Zheng, S.C., Stein-O'Brien, G., Boukas, L., Goff, L.A. & Hansen, K.D. Pumping the brakes on RNA velocity by understanding and interpreting RNA velocity estimates. *Genome Biol* **24**, 246 (2023).
15. Tiejun Li, T.L., Jifan Shi, J.S., Yichong Wu, Y.W. & Peijie Zhou, P.Z. On the Mathematics of RNA Velocity I: Theoretical Analysis. *CSIAM Transactions on Applied Mathematics* **2**, 1-55 (2021).
16. Lange, M. *et al.* CellRank for directed single-cell fate mapping. *Nat Methods* **19**, 159-170 (2022).

17. Roweis, S.T. & Saul, L.K. Nonlinear Dimensionality Reduction by Locally Linear Embedding. *Science* **290**, 2323-2326 (2000).
18. Whitney, H. The Self-Intersections of a Smooth n-Manifold in 2n-Space. *Annals of Mathematics* **45**, 220-246 (1944).
19. Cannoodt, R., Saelens, W., Deconinck, L. & Saeys, Y. Spearheading future omics analyses using dyngen, a multi-modal simulator of single cells. *Nat Commun* **12**, 3942 (2021).
20. Pijuan-Sala, B. *et al.* A single-cell molecular map of mouse gastrulation and early organogenesis. *Nature* **566**, 490-495 (2019).
21. Barile, M. *et al.* Coordinated changes in gene expression kinetics underlie both mouse and human erythroid maturation. *Genome Biol* **22**, 197 (2021).
22. Qiao, C. & Huang, Y. Representation learning of RNA velocity reveals robust cell transitions. *Proc Natl Acad Sci U S A* **118**(2021).
23. Dybedal, I., Larsen, S. & Jacobsen, S.E. IL-12 directly enhances in vitro murine erythropoiesis in combination with IL-4 and stem cell factor. *J Immunol* **154**, 4950-5 (1995).
24. Setty, M. *et al.* Characterization of cell fate probabilities in single-cell data with Palantir. *Nat Biotechnol* **37**, 451-460 (2019).
25. Ning, W. *et al.* Blocking exosomal miRNA-153-3p derived from bone marrow mesenchymal stem cells ameliorates hypoxia-induced myocardial and microvascular damage by targeting the ANGPT1-mediated VEGF/PI3k/Akt/eNOS pathway. *Cell Signal* **77**, 109812 (2021).
26. Bost, P. *et al.* Host-Viral Infection Maps Reveal Signatures of Severe COVID-19 Patients. *Cell* **181**, 1475-1488 e12 (2020).
27. Ratnasiri, K., Wilk, A.J., Lee, M.J., Khatri, P. & Blish, C.A. Single-cell RNA-seq methods to interrogate virus-host interactions. *Semin Immunopathol* **45**, 71-89 (2023).
28. Costa, B. *et al.* Human cytomegalovirus exploits STING signaling and counteracts IFN/ISG induction to facilitate infection of dendritic cells. *Nat Commun* **15**, 1745 (2024).
29. Hein, M.Y. & Weissman, J.S. Functional single-cell genomics of human cytomegalovirus infection. *Nat Biotechnol* **40**, 391-401 (2022).
30. Alon, U. *An introduction to systems biology: design principles of biological circuits*, (Chapman & Hall/CRC, 2007).
31. Ball, C.B. *et al.* Human Cytomegalovirus IE2 Both Activates and Represses Initiation and Modulates Elongation in a Context-Dependent Manner. *mBio* **13**, e00337-22 (2022).
32. Manska, S. & Rossetto, C.C. Identification of cellular proteins associated with human cytomegalovirus (HCMV) DNA replication suggests novel cellular and viral interactions. *Virology* **566**, 26-41 (2022).
33. Gredmark-Russ, S. & Soderberg-Naucler, C. Dendritic cell biology in human cytomegalovirus infection and the clinical consequences for host immunity and pathology. *Virulence* **3**, 621-34 (2012).
34. Paulus, C., Krauss, S. & Nevels, M. A human cytomegalovirus antagonist of type I IFN-dependent signal transducer and activator of transcription signaling. *Proc Natl Acad Sci U S A* **103**, 3840-5 (2006).
35. Park, A. *et al.* HCMV-encoded US7 and US8 act as antagonists of innate immunity by distinctively targeting TLR-signaling pathways. *Nat Commun* **10**, 4670 (2019).
36. Yang, D., de la Rosa, G., Tewary, P. & Oppenheim, J.J. Alarmins link neutrophils and dendritic cells. *Trends Immunol* **30**, 531-7 (2009).
37. Bogdanow, B. *et al.* Human cytomegalovirus tegument protein pp150 acts as a cyclin A2-CDK-dependent sensor of the host cell cycle and differentiation state. *Proc Natl Acad Sci U S A* **110**, 17510-5 (2013).
38. Luftig, M.A. Viruses and the DNA Damage Response: Activation and Antagonism. *Annu Rev Virol* **1**, 605-25 (2014).
39. Ball, C.B. *et al.* Human Cytomegalovirus Infection Elicits Global Changes in Host Transcription by RNA Polymerases I, II, and III. *Viruses* **14**(2022).
40. Bogdanow, B., Phan, Q.V. & Wiebusch, L. Emerging Mechanisms of G(1)/S Cell Cycle Control by Human and Mouse Cytomegaloviruses. *mBio* **12**, e0293421 (2021).
41. Goodwin, C.M., Ciesla, J.H. & Munger, J. Who's Driving? Human Cytomegalovirus, Interferon, and NFkappaB Signaling. *Viruses* **10**(2018).
42. Manandhar, T., Ho, G.T., Pump, W.C., Blasczyk, R. & Bade-Doeding, C. Battle between Host Immune Cellular Responses and HCMV Immune Evasion. *Int J Mol Sci* **20**(2019).
43. Zeng, J. *et al.* Insights into the Transcriptome of Human Cytomegalovirus: A Comprehensive Review. *Viruses* **15**(2023).

44. Schommartz, T., Tang, J., Brost, R. & Brune, W. Differential Requirement of Human Cytomegalovirus UL112-113 Protein Isoforms for Viral Replication. *J Virol* **91**(2017).
45. Ma, S. *et al.* Chromatin Potential Identified by Shared Single-Cell Profiling of RNA and Chromatin. *Cell* **183**, 1103-1116 e20 (2020).
46. Joost, S. *et al.* The Molecular Anatomy of Mouse Skin during Hair Growth and Rest. *Cell Stem Cell* **26**, 441-457 e7 (2020).
47. Pethe, P., Noel, V.S. & Kale, V. Deterministic role of sonic hedgehog signalling pathway in specification of hemogenic versus endocardiogenic endothelium from differentiated human embryonic stem cells. *Cells Dev* **166**, 203685 (2021).
48. St-Jacques, B. *et al.* Sonic hedgehog signaling is essential for hair development. *Curr Biol* **8**, 1058-68 (1998).
49. Mahdessian, D. *et al.* Spatiotemporal dissection of the cell cycle with single-cell proteogenomics. *Nature* **590**, 649-654 (2021).
50. Trevino, A.E. *et al.* Chromatin and gene-regulatory dynamics of the developing human cerebral cortex at single-cell resolution. *Cell* **184**, 5053-5069 e23 (2021).
51. Wang, S. *et al.* The inconsistent regulation of HOXC13 on different keratins and the regulation mechanism on HOXC13 in cashmere goat (Capra hircus). *BMC Genomics* **19**, 630 (2018).
52. Peng, Y. *et al.* Detection of new pioneer transcription factors as cell-type-specific nucleosome binders. *Elife* **12**(2024).
53. Desanlis, I. *et al.* HOX13-dependent chromatin accessibility underlies the transition towards the digit development program. *Nat Commun* **11**, 2491 (2020).
54. Gillespie, D.T. The chemical Langevin Equation. *J. Chem. Phys.* **113**, 297-306 (2000).
55. Zwanzig, R. Ensemble method in the theory of irreversibility. *J. Chem. Phys.* **33**, 1338-1341 (1960).
56. Mori, H. Transport, collective motion, and Brownian motion. *Prog. Theor, Phys.* **33**, 423-455 (1965).
57. Xing, J. & Kim, K.S. Application of the projection operator formalism to non-Hamiltonian dynamics. *Journal of Chemical Physics* **134**(2011).
58. Hao, Y. *et al.* Integrated analysis of multimodal single-cell data. *Cell* **184**, 3573-3587 e29 (2021).
59. Hastie, T. & Tibshirani, R. *Generalized additive models*, (Chapman and Hall, London ; New York, 1990).
60. Mitra, S. *et al.* Single-cell multi-ome regression models identify functional and disease-associated enhancers and enable chromatin potential analysis. *Nat Genet* **56**, 627-636 (2024).

Methods

Dynamical systems theory formulation of cellular state transitions

Assume that a cell state can be represented by the cell volume V (or cellular compartment size) and the copy number of $L >> 1$ pairs of gene products ($\boldsymbol{n}_m$, $\boldsymbol{n}_p$), where m and p designate mRNA and protein, and the bold fonts indicate vectors. For simplicity here we only consider m and p. It is straightforward to generalize to finer cell state specifications, for example, with distinction of nuclear and cytosol localizations, posttranslational states of proteins, other species such as microRNAs, epigenetic states, etc.

The temporal evolution of the cell state is described by a set of chemical master equations. When the copy numbers of molecular species are not too small, and the chemical reactions are not strongly coupled, Gillespie showed that the chemical master equations can be approximated by a set of chemical Langevin equations[54]. With extrinsic noises also included, we assume the ansatz that the dynamics of cell state is described by a set of generic stochastic differential equations,

$$\frac{d\boldsymbol{x}_m}{dt} = \boldsymbol{F}(\boldsymbol{x}_m, \boldsymbol{x}_p) + \zeta(\boldsymbol{x}_m, \boldsymbol{x}_p, t),$$

$$\frac{d\boldsymbol{x}_p}{dt} = \boldsymbol{G}(\boldsymbol{x}_m, \boldsymbol{x}_p) + \eta(\boldsymbol{x}_m, \boldsymbol{x}_p, t), \tag{4}$$

where the L-dimensional vectors $\boldsymbol{x}_m$ and $\boldsymbol{x}_p$ are cellular concentrations of m and p, and the ζ and η are taken as white noises with zero mean.

The low-dimensional manifold assumption is central to machine learning approaches on data analyses. From dynamical systems theory perspective, after a transient time a multi-dimensional dynamical system often converges to a low-dimensional slow manifold. In practice such property has been exploited with techniques such as quasi-steady-state approximation, quasi-equilibrium approximation. For a rigorous formulation, assume that one can identify a set of variables $(\boldsymbol{z}, \boldsymbol{Z})$ with $(2L\text{-}M)$ dimensional fast variables $\boldsymbol{z} = \boldsymbol{z}(\boldsymbol{x}_m, \boldsymbol{x}_p)$ and M-dimensional slower variables $\boldsymbol{Z} = \boldsymbol{Z}(\boldsymbol{x}_m, \boldsymbol{x}_p)$. Xing and Kim extended the celebrated Zwanzig-Mori projection[55,56] to a general dynamical system described by eq. 4[57]. The projection procedure results in a set of stochastic integral-differential equations of $\boldsymbol{Z}$ with colored noises, which are formally equivalent to eq. 4. Then if one assumes clear time scale separation between $\boldsymbol{z}$ and $\boldsymbol{Z}$, the equations reduce to a set of

Langevin equations with white noises, $\frac{d\boldsymbol{Z}}{dt} = \boldsymbol{A}\big(\boldsymbol{Z}(\boldsymbol{x}_m, \boldsymbol{x}_p)\big) + \eta(\boldsymbol{Z}, t)$, where $(\boldsymbol{Z}, t)$ are white noises with zero mean. Through ensemble averaging over the vicinity of a given point $\boldsymbol{Z}$, one has

$$\boldsymbol{v} \equiv < \frac{d\boldsymbol{Z}}{dt} >_{\boldsymbol{Z}} = \boldsymbol{A}\big(\boldsymbol{Z}(\boldsymbol{x}_m, \boldsymbol{x}_p)\big). \tag{5}$$

The equations define a *M*-dimensional manifold embedded in the $(\boldsymbol{x}_m, \boldsymbol{x}_p)$ space. A scRNA-seq data then measures the corresponding manifold projected to the transcriptomic subspace.

One should note that in practice the reported RNA velocity vector of a cell state *i* is typically obtained through averaging the raw velocity vectors of cell states within its neighborhood $\mathcal{N}_i$ on the manifold as a numerical approximation of the ensemble average, $\boldsymbol{v}_{\boldsymbol{x}i} = < \frac{d\boldsymbol{x}_m}{dt} > \approx \sum_{j \in \mathcal{N}_i} \boldsymbol{v}_{\boldsymbol{x}j}$. In this work, we used the k-nearest neighbor (kNN) algorithm to define the neighborhood in single modality datasets. For multiomics data, the neighborhood of a cell state was defined using weighted nearest neighbors (WNN)[58] in the composite cell state space.

Mathematical foundation on applying GraphVelo to multiomics datasets

Equation 3 in the main text applies to transformation between a manifold embedded in a state space and in a subspace. According to the Whitney Embedding theorem[18], any smooth real *M*-dimensional manifold can be embedded in a *2M*-dimensional real space provided that $M > 0$. Consider a full set of genes versus a subset in a scRNAseq dataset, or a combined scRNAseq/scATACseq multiomics dataset versus the scRNAseq subset. Assume that the full cell state space has a dimensionality *N*, while a single cell data manifold is typically low-dimensional with $M << N$. Then the Whitney Embedding theorem[18] suggests that with proper choice of the subset the manifolds in the full space and the subspace are homeomorphic or at least piece-wise homeomorphic (Fig. 1b, Supplemental Text IV), i.e., a one-to-one mapping exists between the two. Then applying eq. 3 allows one to infer the velocity vectors for the full-space representation from those of the subspace.

Reconstruction of extended dynamo vector field from multiomics data

Dynamo is a general framework of reconstructing dynamical models from scRNAseq data, and it is straightforward to generalize to multi-modal data. The framework is based on specific realizations of eq. 5, $\boldsymbol{v} \equiv < \frac{d\boldsymbol{x}}{dt} >_{\boldsymbol{x}} = \boldsymbol{A}(\boldsymbol{x})$, with state vector $\boldsymbol{x}$ being the transcript concentrations

for scRNAseq data, and combined transcript concentrations and continuous quantification of locus-specific chromatin open-close state for multiomics scRNAseq/scATACseq data. The variables $\boldsymbol{x}$ can be defined in various representations, e.g., the original gene space, principal component subspace, latent space defined by variational autoencoder, etc. With GraphVelo it is straightforward to transform $\boldsymbol{v_x}$ between different representations.

The vector field functions $\boldsymbol{A}(\boldsymbol{x})$ contain quantitative regulation relations between genes that are learned from single cell data points of $(\boldsymbol{x}, \boldsymbol{v_x})$. Various algorithms can be used to learn the *analytical* forms of $\boldsymbol{A}(\boldsymbol{x})$. The original dynamo paper illustrated a Reproducing Kernel Hilbert Space (RKHS) representation method. The method expresses $\boldsymbol{A}(\boldsymbol{x})$ as a linear combination of pre-selected basis functions, $\boldsymbol{v} = \boldsymbol{A}(\boldsymbol{x}) = \sum_\alpha \boldsymbol{c_\alpha} \boldsymbol{\Gamma}_\alpha(\boldsymbol{x})$, similar to the more familiar Taylor expansion that uses a linear combination of polynomial functions to represent a continuous analytical function. It should be noted that the basis functions and so $\boldsymbol{A}(\boldsymbol{x})$ are generally *nonlinear* functions of $\boldsymbol{x}$. Following Qiu et al.[7], we chose Gaussian functions centered at selected reference points $\widetilde{\boldsymbol{x}}_\alpha$, $\boldsymbol{\Gamma}_a(\boldsymbol{x}, \widetilde{\boldsymbol{x}}_\alpha) = e^{-2w\|\boldsymbol{x}-\widetilde{\boldsymbol{x}}_\alpha\|^2}$, with default parameter value of w in the package dynamo. Then we determined the coefficient vectors $\boldsymbol{c_\alpha}$through minimizing the loss function $\Phi(\boldsymbol{c}_1, \boldsymbol{c}_2, \dots, \boldsymbol{c}_\mathrm{m}) = \sum_{i=1}^{n} \|\boldsymbol{v}_i - \sum_\alpha \Gamma(\boldsymbol{x}, \widetilde{\boldsymbol{x}}_\alpha)\boldsymbol{c}_\alpha\|^2 + \frac{\lambda}{2}\sum_{\alpha=1}^{m}\sum_{\beta=1}^{m} \boldsymbol{c}_\alpha^T \Gamma(\widetilde{\boldsymbol{x}}_\alpha, \widetilde{\boldsymbol{x}}_\beta)\boldsymbol{c}_\beta$, where the first sum was over all the data points, and the second term was Tikhonov regularization weighted by λ. The superscript T means matrix transpose. One can also use neural networks, e.g., variational autoencoders, to learn an optimal set of basis functions, and other algorithms such as neural ODE to learn $\boldsymbol{B}(\boldsymbol{x})$. The difference is merely algorithmic under the same framework of dynamical systems theories.

Denoise velocity vectors in the space of principal components (PCs). Learning coefficients $\boldsymbol{\phi}_i$ in gene space directly often fails due to thousands of gene profiles. To avoid the curse of high dimensionality and learn parameters in a compact manifold, we designed a procedure to denoise the velocities in a reduced PCA space. Specifically, we extrapolated the cell state i in the original space using the infinitesimal propagation operator to extrapolate the future state:

$$\boldsymbol{x}_i' = \boldsymbol{x}_i + \boldsymbol{v}_i \cdot dt_i$$

Moreover, we estimated an optimal step size dt based on the local density to guarantee the cell states are bound to the manifold:

$$dt_i = \text{median}\left(\frac{\frac{1}{k}\sum_{j=1}^{k}\|\boldsymbol{\delta}_{ij}\|}{\|\boldsymbol{v}_i\|}\right)$$

After utilizing the cell-dependent t_i to forcing the predictions inhabiting regions of the phenotypical manifold, we applied dimension reduction to project both current and future status from the gene space to the PCA space through linear transformation. Then we obtained the projected velocity vectors as:

$$\boldsymbol{v}_i^{\text{PCA}} = \frac{\boldsymbol{x}_i' Q - \boldsymbol{x}_i Q}{dt_i}$$

where Q is the PC loading matrix estimated using x that serves as the coordinate transformation matrix.

Manifold-consistent kinetic genes. The kinetic assumptions between nascent and mature RNA fail when the underlying parameters shift along the developmental trajectory[11], which leads to transcription burst and rapid degradation in the phase portraits (Fig. 3a). An internal clock exists during cell proliferation and differentiation. Current methods rely on different criteria to select confident estimated velocity genes (see Supplemental Text V for detailed discussion).. Here, we presume that the velocity of robustly estimated genes should be consistent with the (pseudo)time derivative estimated under the manifold assumption. We can utilize any available *t*s inferred from data manifold by either pseudotime, velocity latent time or lineage tracing to approximate the temporal information and use *k* nearest neighbors to define the locally linear plane. After ordering cells within the local Euclidean space, we calculate the MacK score for any gene g as an indicator of whether the sign of estimated velocity agrees with the dynamic cascades within manifold,

$$\text{MacK score} = \frac{1}{n}\sum_{j\in\mathcal{N}(i)} \mathbb{I}\left(sgn\left(\frac{\Delta x_{ij}(g)}{\Delta t_{ij}}\right) = sgn(v_i(g))\right)$$

Where $\mathcal{N}_i$ indicates the neighbor points of cell *i*, $\mathbb{I}$ represents the indicator function and *sgn* returns the sign of the values. $\Delta x_{ij}(g), v_i(g)$ are the difference in abundance of gene g between cell *i* and *j*, and the velocity of gene g in cell *i*, respectively. We parallelize the calculation to scale efficiently with the number of genes, which is important due to the number of highly variable genes.

656 **Dynamo criteria.** Dynamo offers a correction strategy by removing genes with low gene-wise
657 confidence in the phase plane. This allows us to identify genes that appear in incorrect phase
658 portrait positions and contribute to erroneous flow directions (illustrated in Fig3. a). To filter out
659 genes with misleading dynamic patterns, a gene's behavior during cell differentiation from
660 progenitor to terminal states is evaluated based on its expression pattern relative to steady state.

661 **Post-GraphVelo analyses**

662 **Markovian analyses.** Marius et al.[16] have developed a framework named CellRank to study
663 cellular dynamics based on Markov chain formulation. We use CellRank to identify cell state
664 transitions using velocity kernel and identify terminal states within datasets by GPCCA function
665 module. In addition, CellRank pseudotime kernel is used for methods comparison in real datasets.

666 **Functional form vector field and differential geometry analyses.** Dynamo first proposed to
667 learn nonlinear functional form of RNA velocity vector field, providing a physical-informed
668 framework which integrates mechanism modeling and single cell data analyses. We use dynamo
669 to learn continuous vector filed and perform differential geometry analyses such as gene
670 acceleration, vector field-based pseudotime, LAP, Jacobian analyses and in silico perturbation.

671 **Pseudotemporal orderings.** GraphVelo itself does not compute an ordering index of cells as we
672 are seeking for a more quantitative method to infer RNA velocity. With an accurate RNA velocity
673 as input, we can approximate the vector field precisely. Thus, we use the scalar potential estimated
674 from the functional form vector field with Hodge decomposition as a proxy of time, which is
675 implemented by dynamo package.

676 **Approximating smooth velocity trends with generalized addictive model.** The variation of
677 transcription rates contains the high order dynamic information of the cell system. To model the
678 dynamic patterns of RNA velocity along transition path from noisy data, we refine the velocity
679 vectors by local geometry via TSP and further fit GAM to velocity value of each gene that has
680 been refined by GraphVelo. For any gene g, we model the velocity trend for the temporal variable
681 t via

682 $$v_{gi} = \beta_0 + f(t_i)$$

683 Where v_{gi} indicates the velocity of gene g in cell i, f is built using penalized B-splines which allow
684 us to automatically model non-linear mapping while maintaining additivity[59]. To visualize the
685 velocity trends, we select 100 equally spaced testing points along transition path and predict gene
686 expression at each of them using the fitted model. The estimated velocity trends can be treated as
687 smoothed time series for further analyses.

688 **Clustering velocity trends along infection trajectory.** With manifold-constrained velocity
689 estimated by GraphVelo, we are able to cluster genes into different functional modules which are
690 involved in the same regulatory circuit. We recover transcription variation of both host and virus
691 factors along infection trajectory by fitting GAMs in temporal indicator, the percentage of viral
692 RNA. Next, we select 100 equally spaced time points and generate the GAM-smoothed velocity
693 trends. We compute a kNN graph and cluster the kNN graph using the leiden algorithm. We used
694 k = 15 for the velocity-trend kNN graph and the leiden algorithm with resolution parameter set to
695 0.3 to avoid over clustering the trends. For each recovered cluster, we compute its mean and
696 standard deviation (pointwise, for all generated points that were used for smoothing) and visualize
697 the smoothed trends per cluster.

698 **Characterizing decoupling genes based on dynamic time warping distance between multi-**
699 **modality velocities.** We perform the DTW distance calculation by dtaidistance package. To
700 eliminate the influence of scale in different modality, we maximum-normalize the chromatin/RNA
701 velocity to the same range of [0, 1]. Then we fit the velocity trends of both modalities along vector
702 field-based pseudotime to yield the smoothed velocity trends. We calculate the DTW distance
703 between velocity trends per gene. To distinguish the decoupling genes based on their multi-
704 modality velocities, we rank the genes based on the DTW distance and identify elbow point as
705 cutoff. Genes with distance metric larger than the cutoff are characterized as decoupling genes and
706 used for visualization and functional analysis.

707 **Synthetic datasets**

708 **Generating two genes bifurcation process and mapping it to 3d.** The bifurcation data (n=2,000
709 cells) for the toggle-switch system is simulated using the Gillespie algorithm. We use activation
710 and inhibition Hill functions to model the induction and suppression effects between the two genes:

$$\dot{x} = \frac{a_1 x^n}{S_1^n + x^n} + \frac{b_1 K_1^n}{K_1^n + y^n} - \gamma_1 x,$$

$$\dot{y} = \frac{a_2 y^n}{S_2^n + y^n} + \frac{b_2 K_2^n}{K_2^n + x^n} - \gamma_2 y,$$

We use the simulation backend implemented by dynamo with default parameters except the timescale (reset τ =1) to generate the bifurcating process. We then map the synthetic dataset onto a sphere (radius r=70) and yield the variable z as:

$$z = \sqrt{max(r^2 - x^2 - y^2, 0)}$$

Then we are able to calculate the correctly-scaled 3d vectors by infinitesimal propagation operator with sufficient small step size (dt=1 in our case):

$$\dot{z} = \lim_{\Delta t \to 0} \frac{\sqrt{\max(r^2 - (x + \dot{x}\Delta t)^2 - (y + \dot{y}\Delta t)^2, 0)} - z}{\Delta t}$$

Generating scRNA-seq synthetic data with dyngen. To generate high-dimensional single-cell transcriptomic data in silico, we use a multi-modal simulation engine, dyngen, to account for different developmental topologies. We constructe module networks to represent regulatory cascades and feedback loops driving progressive changes in gene expression and influencing cell fate decisions. We generate three datasets with 1,000 cells and 100 genes using the linear, cyclic, and bifurcating loop backbones provided by dyngen, with all other parameters set to default values. These datasets include simulated nascent and mature mRNA counts along with ground-truth RNA velocities and known manifold structure.

Simulation of rapid degradation and transcription burst events on phase portrait. As for genes with variable degradation rate, we present a minimal regulatory network with linear model in which an external signal both inhibits transcription and promotes microRNA (miRNA). The miRNA exerts a linear influence on the degradation rate of mRNA. We have miRNA's velocity as

$$\frac{dm}{dt} = \alpha_m - \gamma_m m$$

The nascent gene transcription rate and mature mRNA's degradation rate would change to

$$\alpha = \alpha_0 - k_\alpha t$$

$$\gamma = \gamma_0 + k_\gamma m$$

where α_0 and γ_0 represent the constant transcription rate and degradation rate without the effect of miRNA, k_α and k_γ represent the magnitude of influence from miRNA, t indicates the simulation time to mimic the cell-context change along trajectory.

$$\frac{du}{dt} = \alpha - \beta u \ = \ \alpha_0 - k_\alpha m \ - \beta u$$
$$\frac{ds}{dt} = \beta u - \gamma s = \beta u - (\gamma_0 + k_\gamma m)s$$

We set the initial condition to a steady state with $u_0 = \alpha_0/\beta$ and $s_0 = \alpha_0/\gamma_0$, while the miRNA abundance $m_0 = 0$. We simulate u_t and s_t as the microRNA signal m_t gradually increases. The aim is to evaluate whether the estimated RNA velocity consistently aligns in sign with the ground truth RNA velocity.

To generate genes with transcription burst phase portrait, we set the initial condition to $u_0 = 0$, $s_0 = 0$, together with γ as constant and α promotes to $\alpha' = 3\alpha$ when simulation reaches specific time.

Processing of sequencing data

All sequencing data in this study are downloaded publicly (see details in the 'Data availability' section).

Analyses of scRNA-seq datasets. For the erythroid lineage of the mouse gastrulation, we follow the standard data pre-processing procedures implemented by scVelo and select 9,815 cells and 2,000 highly variable genes to construct the k-nearest neighbor graph using 30 nearest neighbors for downstream calculation.

For the human bone marrow dataset, we follow the standard data pre-processing procedures implemented by scVelo and selected 5,780 cells and 2,000 highly variable genes to construct the *k*-nearest neighbor graph using 30 nearest neighbors for downstream calculation. To estimate the variation of degradation rate γ along the differentiation lineages, we divide the cells into five discrete time bins based on precomputed Palantir pseudotime. We then estimate cell-specific degradation rates and visualized their distribution shifts along the hematopoiesis trajectory.

761 **Analyses of HCMV dataset.** We sample the cells from donor 1 to eliminate sample-specific
762 variation and further filter out cells lacking immediate early *UL123* gene expression to focus on
763 the viral infection trajectory. We adapt monocle preprocessing recipe implemented by dynamo and
764 yield 1,454 cells and 2,000 highly variable genes for further analyses. The number of nearest
765 neighbors were set to 30 as default.

766 We collect the pathway-related genes from MSigDB and perform the Jacobian analyses
767 implemented by dynamo, using viral genes as regulators and host genes as effectors. We rank the
768 regulation relationships based on the collections of Jacobian elements. We pick top 50 inhibited
769 effectors of viral genome and select the common set between pathway genes and all the effectors
770 for visualization.

771 In silico knock out experiments are performed via dynamo. We suppressed every single virus
772 factors per time using *dynamo.pd.perturbation* function and calculated the change of total viral
773 gene expression after perturbation.

774 **Analyses of multi-modality datasets.**

775 **RNA velocity estimation for multi-modality datasets.** In traditional scRNA-seq datasets, RNA
776 velocity methods use smoothed spliced and unspliced RNA counts through nearest-neighbor
777 pooling, based on the PCA space computed from transcripts alone. However, this approach is not
778 suitable for multimodal scenarios, as it overlooks hidden variables by relying on a single modality.
779 To construct a consistent manifold combining information from multi-modal genomics data, we
780 utilized WNN as implemented in MultiVelo[13]. The WNN algorithm combines low-dimensional
781 representations from RNA and ATAC omics data. Specifically, we use PCA results from scRNA-
782 seq data and latent semantic indexing (LSI) from scATAC-seq as inputs. The nearest neighbors
783 identified by WNN were then used to calculate the first moment, reducing noise in separate
784 modalities and approximating a unified manifold for GraphVelo.

785 **Mouse skin dataset preprocessing.** The preprocessed SHARE-seq mouse skin dataset[45] is
786 adopted directly from MultiVelo data resources. All the procedures are consistent with MultiVelo
787 except we get the LSI representation processed by SCARlink[60]. We construct the WNN graph
788 using 50 nearest neighbors for downstream calculation. We run scVelo with 'stochastic' mode to
789 estimate the RNA velocity based on the WNN graph as we discussed above.

Human cortex dataset preprocessing. The preprocessed human cerebral cortex data is adopted directly from MultiVelo data resources. All the procedures are consistent with MultiVelo except we get the LSI representation processed by SCARlink. We construct the WNN graph using 50 nearest neighbors for downstream calculation. We run scVelo with 'stochastic' mode to estimate the RNA velocity based on the WNN graph.

Data availability

All the sequencing raw data are publicly accessible. The mouse gastrulation subset to erythroid lineage can be extracted using scVelo's CLI: *scvelo.datasets.gastrulation_erythroid()* or from the original work under accession number E-MTAB-6967 of ArrayExpress. The human bone marrow can be extracted using scVelo's CLI: *scvelo.datasets. bonemarrow()* or through the Human Cell Atlas data portal at https://data.humancellatlas.org/explore/projects/091cf39b-01bc-42e5-9437-f419a66c8a45. The HCMV infected moDC data can be accessed via Zenodo https://doi.org/10.5281/zenodo. 10404879 (https://zenodo.org/records/10404879). The preprocessed mouse skin development dataset can be accessed via https://figshare.com/articles/dataset/Mouse_Hair_Follicle_RNA_Data/22575307 and https://figshare.com/articles/dataset/Mouse_hair_follicle_ATAC_data/22575313. The preprocessed human cortex dataset can be downloaded from https://figshare.com/articles/dataset/Developing_Human_Cortex_RNA_Data/22575376 and https://figshare.com/articles/dataset/Developing_Human_Cortex_ATAC_Data/22575370.

Code availability

Python package GraphVelo can be accessed from https://github.com/xing-lab-pitt/GraphVelo. Reproducibility and tutorials can be found in https://github.com/xing-lab-pitt/GraphVelo/notebook.

References

54. Gillespie, D.T. The chemical Langevin Equation. *J. Chem. Phys.* **113**, 297-306 (2000).
55. Zwanzig, R. Ensemble method in the theory of irreversibility. *J. Chem. Phys.* **33**, 1338-1341 (1960).
56. Mori, H. Transport, collective motion, and Brownian motion. *Prog. Theor, Phys.* **33**, 423-455 (1965).

57. Xing, J. & Kim, K.S. Application of the projection operator formalism to non-Hamiltonian dynamics. *Journal of Chemical Physics* **134**(2011).
58. Hao, Y. *et al.* Integrated analysis of multimodal single-cell data. *Cell* **184**, 3573-3587 e29 (2021).
59. Hastie, T. & Tibshirani, R. *Generalized additive models*, (Chapman and Hall, London ; New York, 1990).
60. Mitra, S. *et al.* Single-cell multi-ome regression models identify functional and disease-associated enhancers and enable chromatin potential analysis. *Nat Genet* **56**, 627-636 (2024).

Author contributions

JX conceived and formulated the theoretical framework, YC, and JG performed tests on simulated data, YC and JX analyzed scRNA-seq datasets, YC, YZ, JG, and KN developed the package GraphVelo, YC and JX wrote the draft, and all authors edited the manuscript.

Acknowledgments

This work was partially supported by National Institute of General Medical Sciences (1R01GM148525 to JX and 1R01GM139297 to IB), and National Science Foundation (2325149) to JX.

This work was supported by the National Key Research and Development Program of China (2023YFE0112300); National Natural Sciences Foundation of China (32261133526; 32270709; 32070677); the 151 talent project of Zhejiang Province (first level), the Science and Technology Innovation Leading Scientist (2022R52035)

Figures:

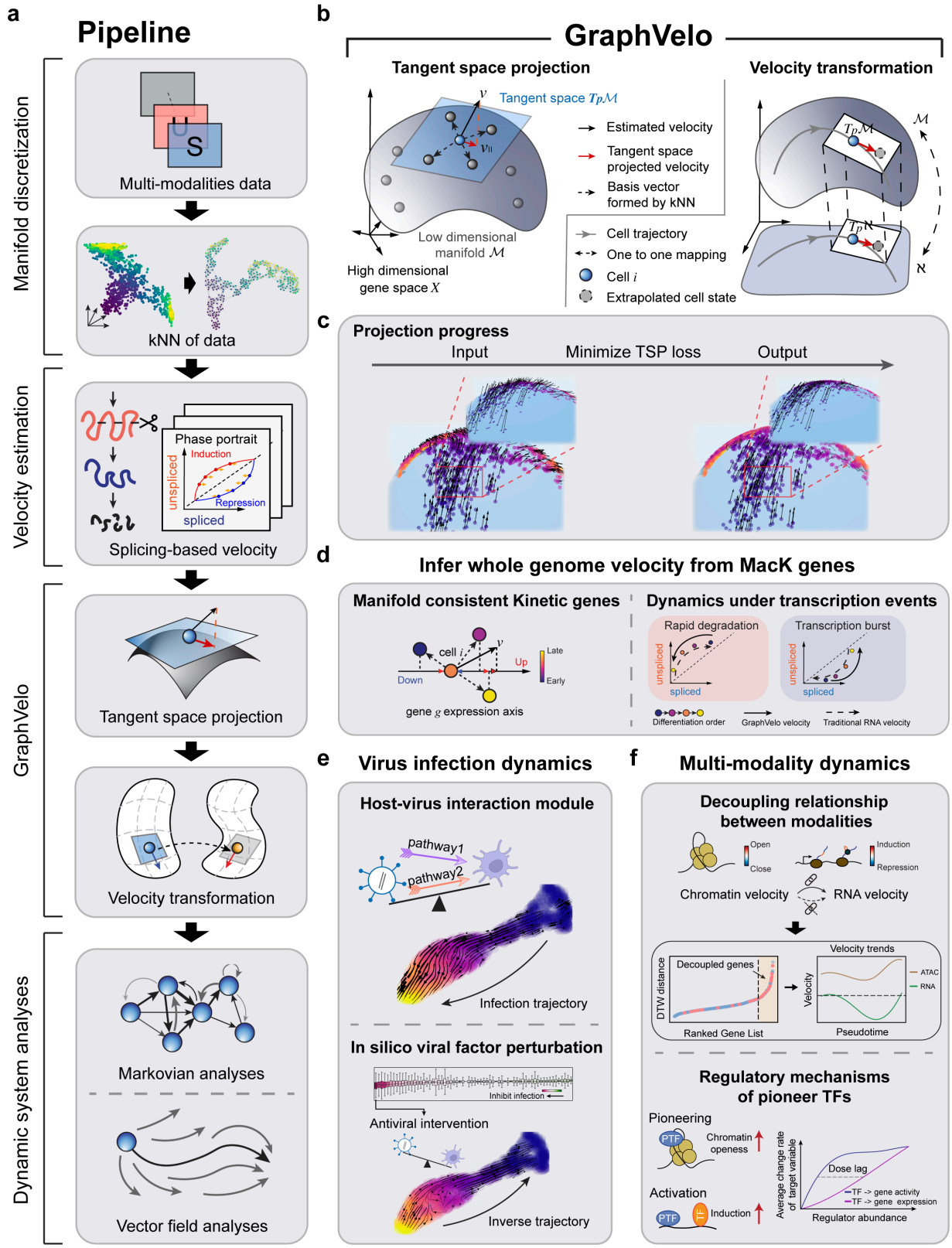

a
Pipeline
Manifold discretization
Multi-modalities data
kNN of data
Velocity estimation
Phase portrait
Induction
Repression
unspliced
spliced
Splicing-based velocity
GraphVelo
Tangent space projection
Velocity transformation
Dynamic system analyses
Markovian analyses
Vector field analyses
b
GraphVelo
Tangent space projection
Tangent space $T_p\mathcal{M}$
Estimated velocity
Tangent space projected velocity
Basis vector formed by kNN
Cell trajectory
One to one mapping
Cell i
Extrapolated cell state
Low dimensional manifold $\mathcal{M}$
High dimensional gene space X
Velocity transformation
c
Projection progress
Input
Minimize TSP loss
Output
d
Infer whole genome velocity from MacK genes
Manifold consistent Kinetic genes
cell i
Up
Down
Late
Early
gene g expression axis
Dynamics under transcription events
Rapid degradation
Transcription burst
Differentiation order
GraphVelo velocity
Traditional RNA velocity
e
Virus infection dynamics
Host-virus interaction module
pathway1
pathway2
Infection trajectory
In silico viral factor perturbation
Inhibit infection
Antiviral intervention
Inverse trajectory
f
Multi-modality dynamics
Decoupling relationship between modalities
Open
Close
Induction
Repression
Chromatin velocity
RNA velocity
Velocity trends
Decoupled genes
DTW distance
Ranked Gene List
Velocity
Pseudotime
ATAC
RNA
Regulatory mechanisms of pioneer TFs
Pioneering
PTF
Chromatin openess
Activation
TF
Induction
Average change rate of target variable
Dose lag
TF -> gene activity
TF -> gene expression
Regulator abundance

Figure 1. Refining RNA velocity by tangent space projection and transforming between representations using GraphVelo.

(a) Workflow of RNA velocity-based analyses incoporating GraphVelo. Note GraphVelo takes any form of RNA velocity (i.e., not just splicing-based velocity) as input, and the kNN neighborhood is defined in the full state space (e.g., by both scRNAseq and scATACseq in multiomics data).

(b) Schematic of tangent space projection and velocity transformation between homeomophic manifolds. *Left*: RNA velocity vectors are projected onto the tangent space defined by the discretized local manifold of neighborhood cell samples. *Right*: GraphVelo allows for transformation of velocity vectors from a manifold embeded in a higher dimensional space ($\mathcal{M}$) to that in a lower dimensional space ($\aleph$), and *vice versa*.

(c) The process of minimizing the loss function of tangent space projection. Noisy velocity vectors (*left*) generated by adding random components orthogonal to those sampled from an analytical 2D manifold were projected back onto the 2D manifold, resulting in smooth velocity vectors that lie in the tangent space (*right*).

(d) GraphVelo allows whole genome velocity inference based on the robustly estimated MacK genes (see also Figure 3). Velocities of genes undergoing variable kinetic rates, such as rapid degradation or transcription burst, are difficult to be correctly inferred by other methods, but can be inferred robustly with GraphVelo.

(e) Virus infection dynamics and underlying host-virus interaction mechanisms uncovered by GraphVelo (see also Figure 4). *Upper*: pathways involved in host-virus interactions were identified using GraphVelo. *Lower*: GraphVelo predicted reversed trajectory of viral infection in response to *in silico* perturbations of viral factors.

(f) GraphVelo provides a consistent view of epigenetic and transcription dynamics (see also Figure 5). *Upper*: GraphVelo analyses on multiomics data revealed that most cell-cycle dependent genes showed decoupling between transcription dynamics and chromatin accessibility change dynamics. *Lower*: Effective dose-response curves reconstructed from multiomics data revealed pioneer transcription factors increased chromatin accessibility then transcription of targe genes.

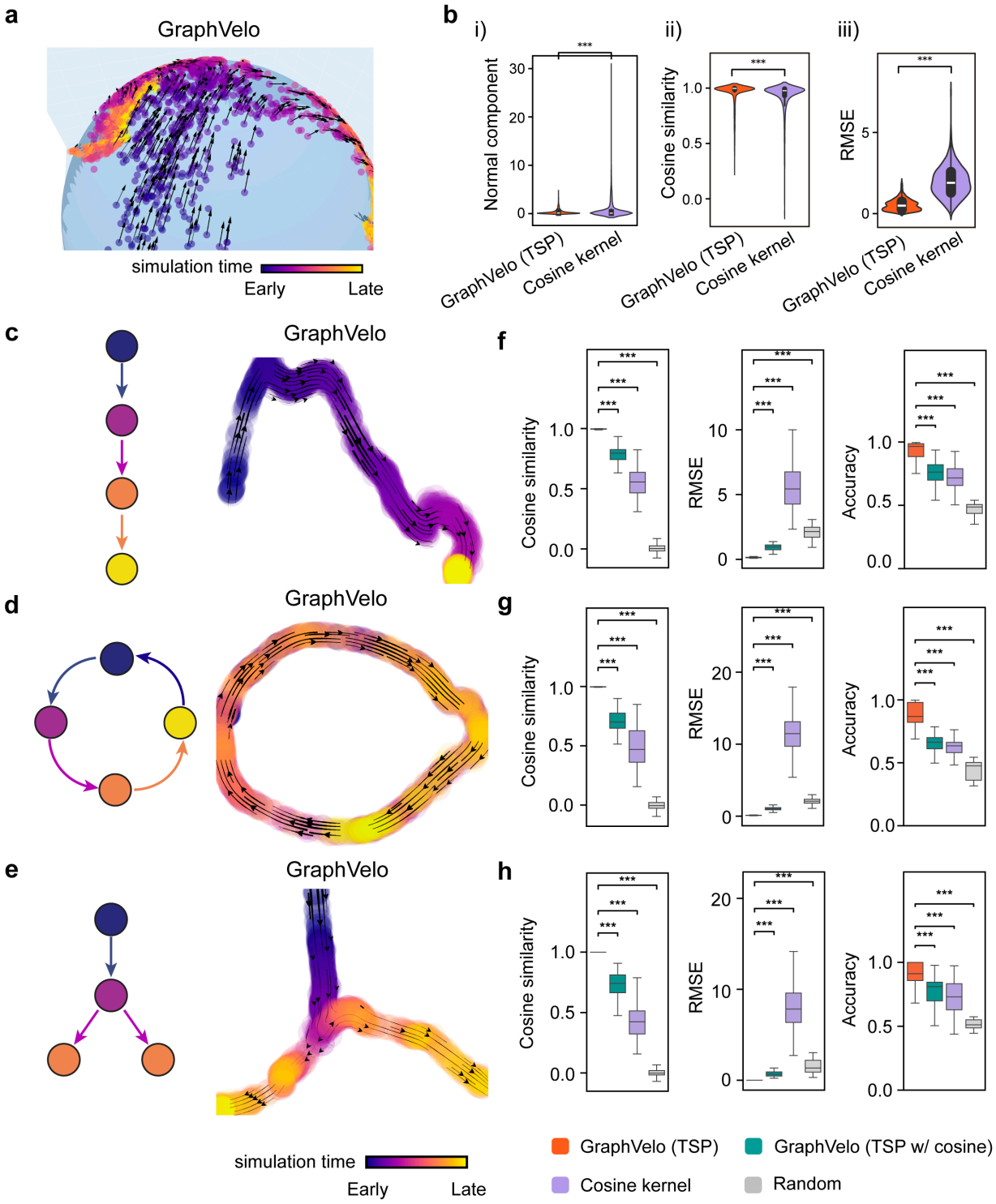


Figure 2. Testing graphVelo on simulated datasets.

(a) Velocity vectors of an analytical three variables bifurcating vector field constrained to a spherical surface. The data points were colored by simulation time.

871 **(b)** Violinplots of: i) normal component of velocity vectors, ii) cosine similarity and iii) root mean
872 square error (RMSE) between ground truth and velocity vectors projected by GraphVelo and
873 cosine kernel, respectively.

874 **(c-e)** Simulation of scRNA-seq data using dyngen under linear, cycling, and bifurcating
875 differentiation models (*left*), and velocity fields (*right*) reconstructed using GraphVelo-corrected
876 velocities, respectively. Each simulation consists of 1,000 cell states and 100 genes. The cells in
877 different states were colored by their simulation time along trajectory.

878 **(f-h)** Boxplots of cosine similarity, , and accuracy between the ground truth velocity vectors and
879 dyngen simulated velocities after projection using GraphVelo TSP loss without cosine
880 regularization, GraphVelo TSP loss with cosine regularization, cosine kernel, and random
881 predictor, respectively.

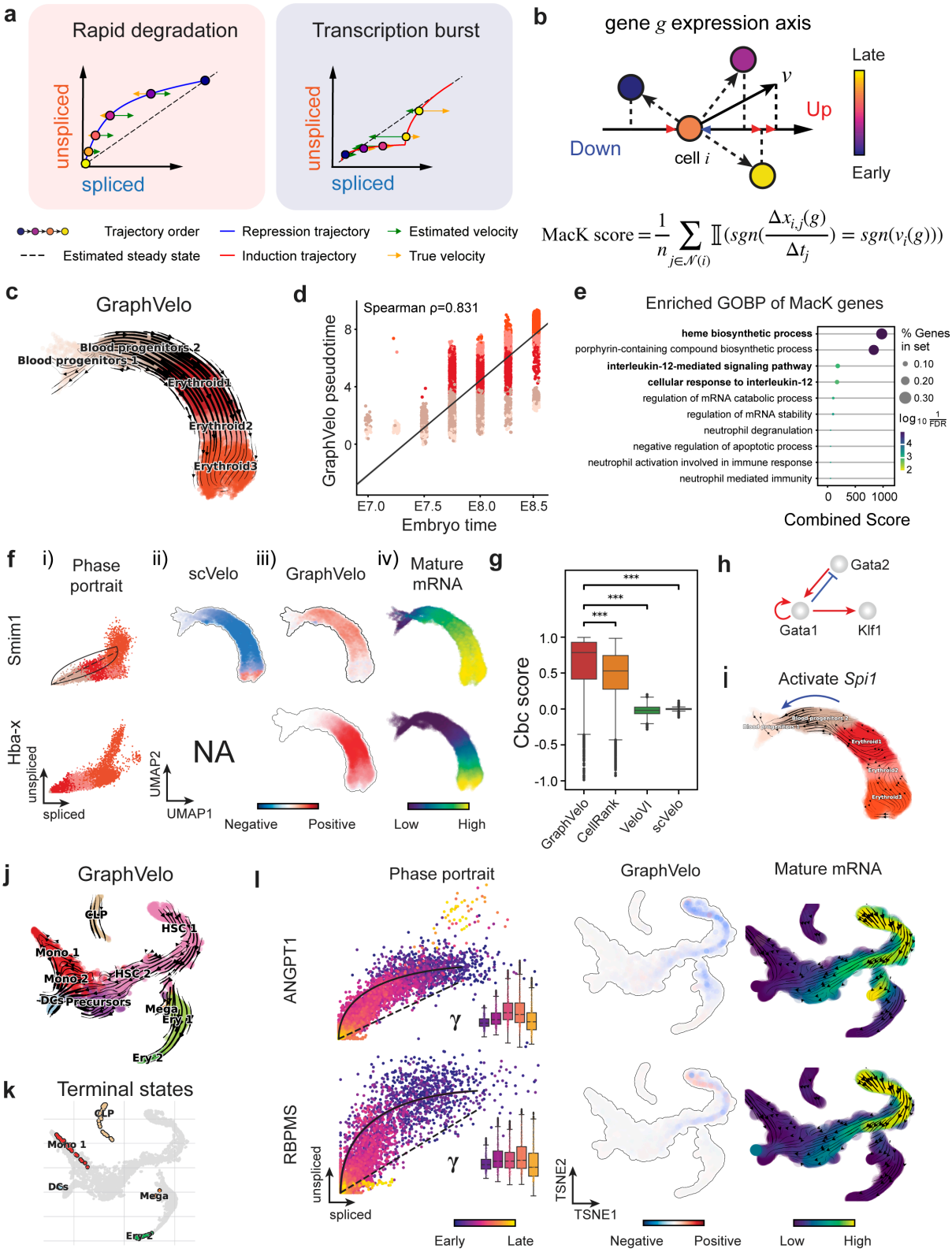


Figure 3. Delineating transcriptome-wise progression with manifold-consistent kinetic genes using GraphVelo

885 **(a)** Schematitc of transcritional events mislead RNA velocity estimation in the phase portrait by
886 standard approaches. *Left*: for genes exhibiting rapid degradation, the cells appear above the steady
887 state line on the phase portrait, whereas the true velocity is negative. *Right*: For genes exhibiting
888 transcription burst, the transcription rate abruptly increases at intermediate states, leading to a
889 steady state line whose slope is overestimated.

890 **(b)** Schematic of manifold-consistent score calculation for robustly estimated velocity genes.

891 **(c)** The projected velocity field from GraphVelo are consistent with the erythroid differentiation
892 by using all highly variable genes.

893 **(d)** The correlation between GraphVelo vector field-based pseudotime and embryo time for
894 erythroid lineage cells. Spearman correlation coefficients are shown.

895 **(e)** GO enrichment analyses of top ranked MacK genes.

896 **(f)** Scatter plots of: i) phase portrait, ii) velocities estimated by scVelo, iii) refined velocities by
897 GraphVelo, and iv) mature mRNA expression of transcription burst genes (*Smim1*, *Hba-x*). Cells
898 were colored by cell type, corresponding velocity, and mature mRNA abundance, respectively,
899 and visualized on the phase portrait and UMAP representation.

900 **(g)** Cross boundary correctness calculated by the velocity input from GraphVelo, CellRank
901 pseudotime kernel, VeloVI and scVelo, respectively.

902 **(h)** Gene regulatory cascade unraveled by GraphVelo-based vector field analyses that drives cell
903 lineage commitment.

904 **(i)** Activation of *Gata1* inhibitor TF *Spi1* lead to reversed velocity flows in gastrulation erythroid
905 maturation investigated through in silico perturbation analyses on GraphVelo-based vector field.

906 **(j)** Velocities derived from GraphVelo for the branching lineage in the hematopoiesis development
907 and projected onto a pre-defined TSNE embedding. Directions of the projected cell velocities on
908 TSNE are in agreement with the reported differentiation directions.

909 **(k)** Terminal states identified by CellRank based on Markov chain formulation derived from
910 GraphVelo velocities.

911 **(l)** Phase portrait, velocity estimated by scVelo, refined velocity by GraphVelo, and gene
912 expression of mature mRNA of identified rapid degradation genes (*NPR3, ANGPT1*). The cells
913 were colored by the palantir pseudotime[24] in the phase portrait. The box plots showed cell-specific
914 $\boldsymbol{\gamma}$ for cells divided into bins according to pseudotime ordering in the phase protrait.

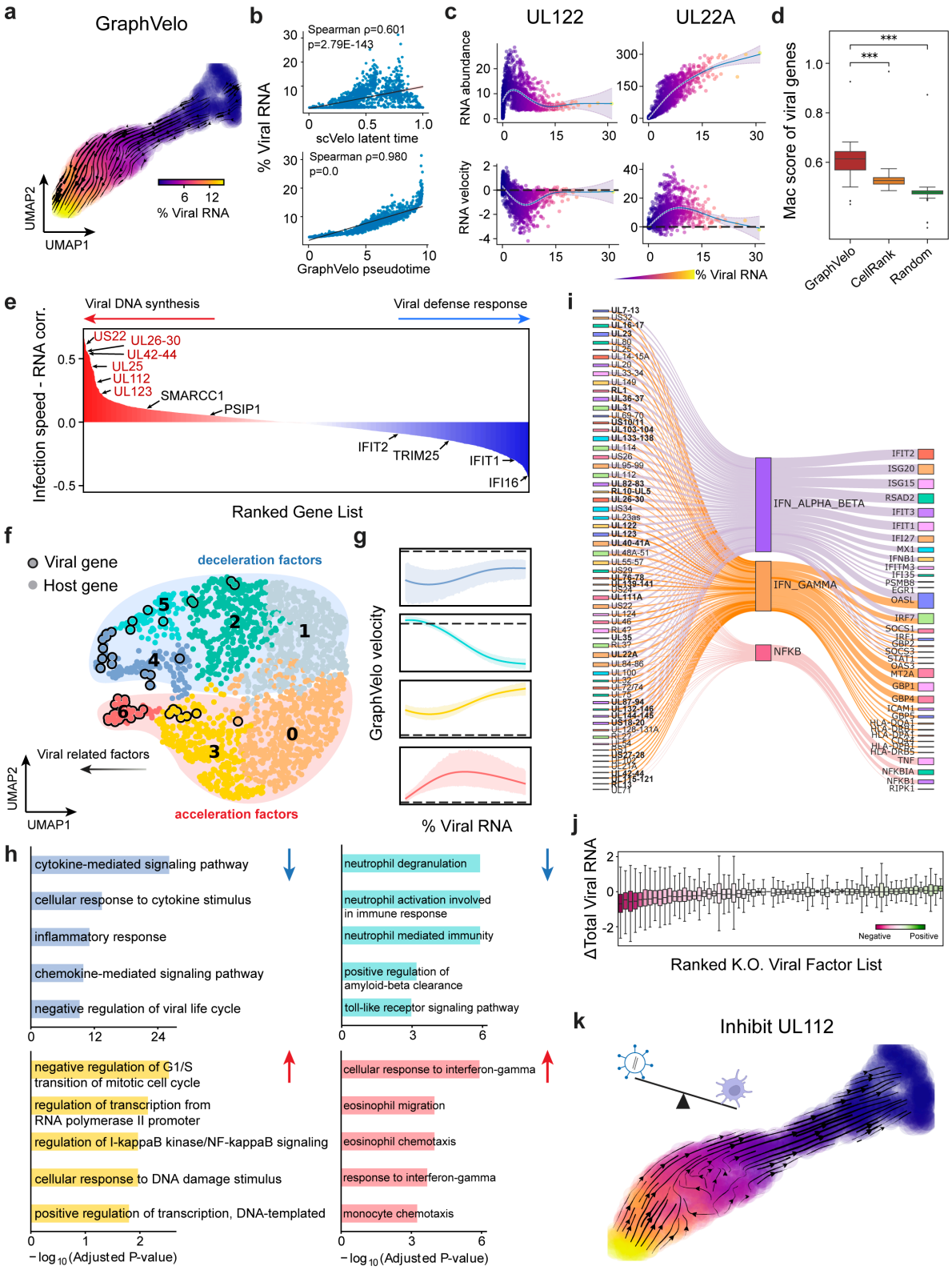


Figure 4. Using GraphVelo velocities to infer host-virus infection trajectory and identify host-pathogen interactions.

918 **(a)** Viral infection captured by the GraphVelo velocity field. Cells were colored by the percentage
919 of viral RNA within a single cell.

920 **(b)** Correlation between viral RNA percentage and pseudotime inferred by scVelo or GraphVelo.
921 Spearman correlation coefficients and *P* values were shown.

922 **(c)** Viral RNA velocities infered by GraphVelo along the viral RNA percentage axis. The black
923 dot line highlights the zero velocity.

924 **(d)** Boxplot summarizing the MacK scores of all viral genes calculated by GraphVelo, CellRank
925 pseudotime kernel and random predictor.

926 **(e)** Correlation between viral infection speed and RNA abundance. Genes were ranked by
927 Spearman correlatioin coefficients. Host and viral genes that contribute to viral DNA synthesis
928 were marked in the left side and those contribute to viral defense response were marked in the right
929 side. Viral genes were highlighted in red.

930 **(f)** UMAP representation of host and viral genes with distances defined by their dynamic
931 expression patterns along the viral RNA percentage axis.

932 **(g)** Example dynamic expression patterns within specific clusters (Leiden4, 5, 3, 6 from top to
933 bottom) along the viral RNA percentage axis. Zero velocity was highlighted by black dot line.

934 **(h)** GO enrichment of each cluster in (g).

935 **(i)** Top host genes inhibited by each viral factor based on dynamo Jacobian analyses. Host effectors
936 were organized by their involved pathways.

937 **(j)** Dynamo prediction of total viral RNA change in response to in silico viral factor knockout.
938 Viral factors were ranked by the mean of total viral RNA changes.

939 **(k)** Vector field change resultant from infinitesmal inhibition of *UL122* during the viral infection
940 process.

941

942

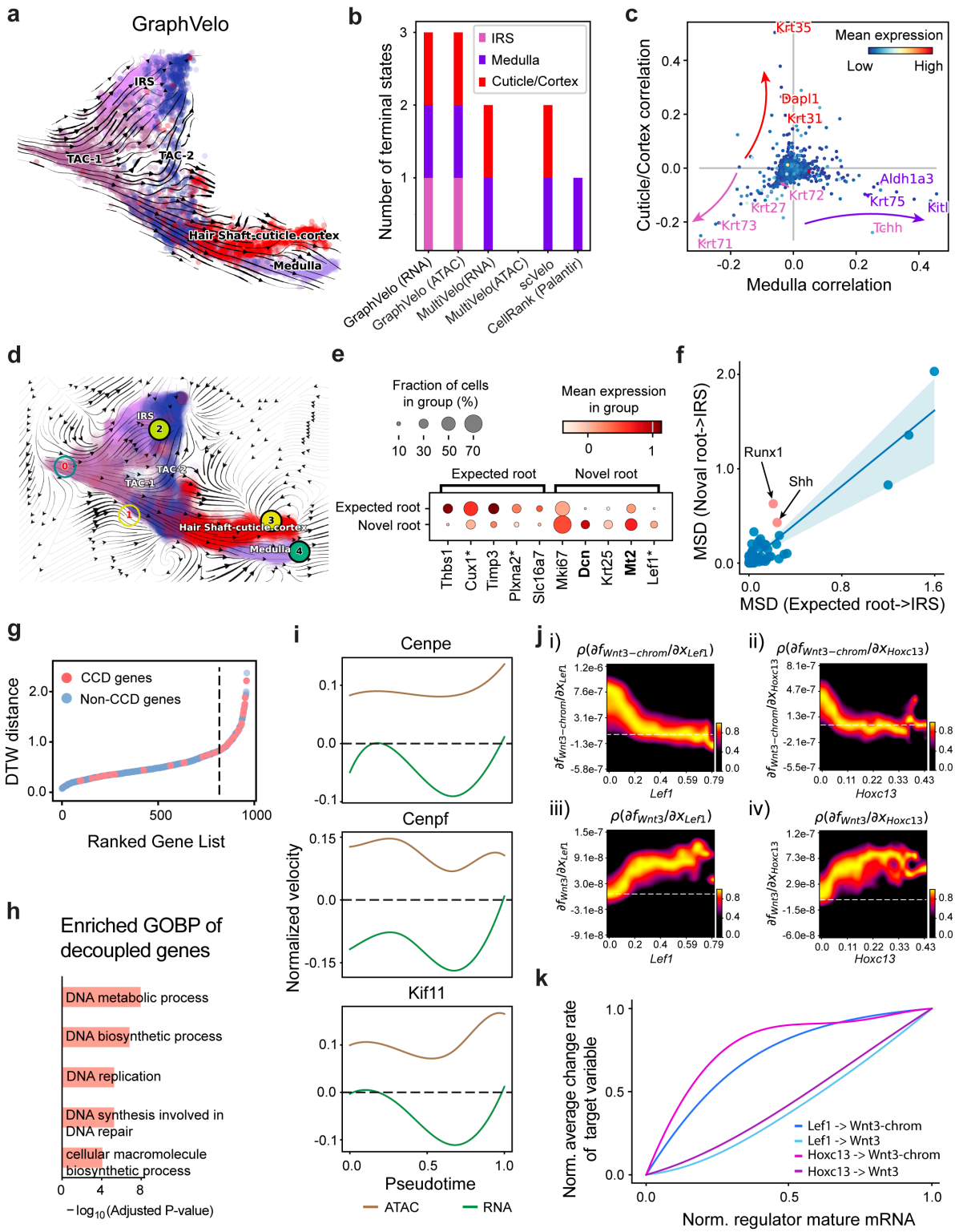


Figure 5. Inferring epigenome and transcriptome consistent dynamics in mouse hair follicle development using GraphVelo multiomics velocities.

946 **(a)** GraphVelo velocity fields of mouse hair follicle development. Cells were colored by cell
947 macrostates.

948 **(b)** Number of terminal states predicted by CellRank using velocities inferred with different
949 methods.

950 **(c)** Driver genes along multiple lineages identified through CellRank.

951 **(d)** Topological analyses of GraphVelo vector field identified novel root cells and attractors
952 residing in three terminal states(IRS, hair shaft-cuticle cortex, and medulla).

953 **(e)** Expression levels of marker genes in novel root cells and expected root cells. Markers identified
954 by Ma et al.[45] were highlighted with stars and newly identified markers were highlighted in bold.

955 **(f)** Regression results of MSD values along the transition path from the expected root or novel root
956 to IRS. Two genes *Runx1* and *Shh* genes with large MSD originating from the novel root were
957 highlighted.

958 **(g)** DTW distance between RNA velocity and chromatin velocity of individual genes. CCD genes
959 were colored in red. The dotted line indicates the elbow point separating the decoupled genes from
960 the rest.

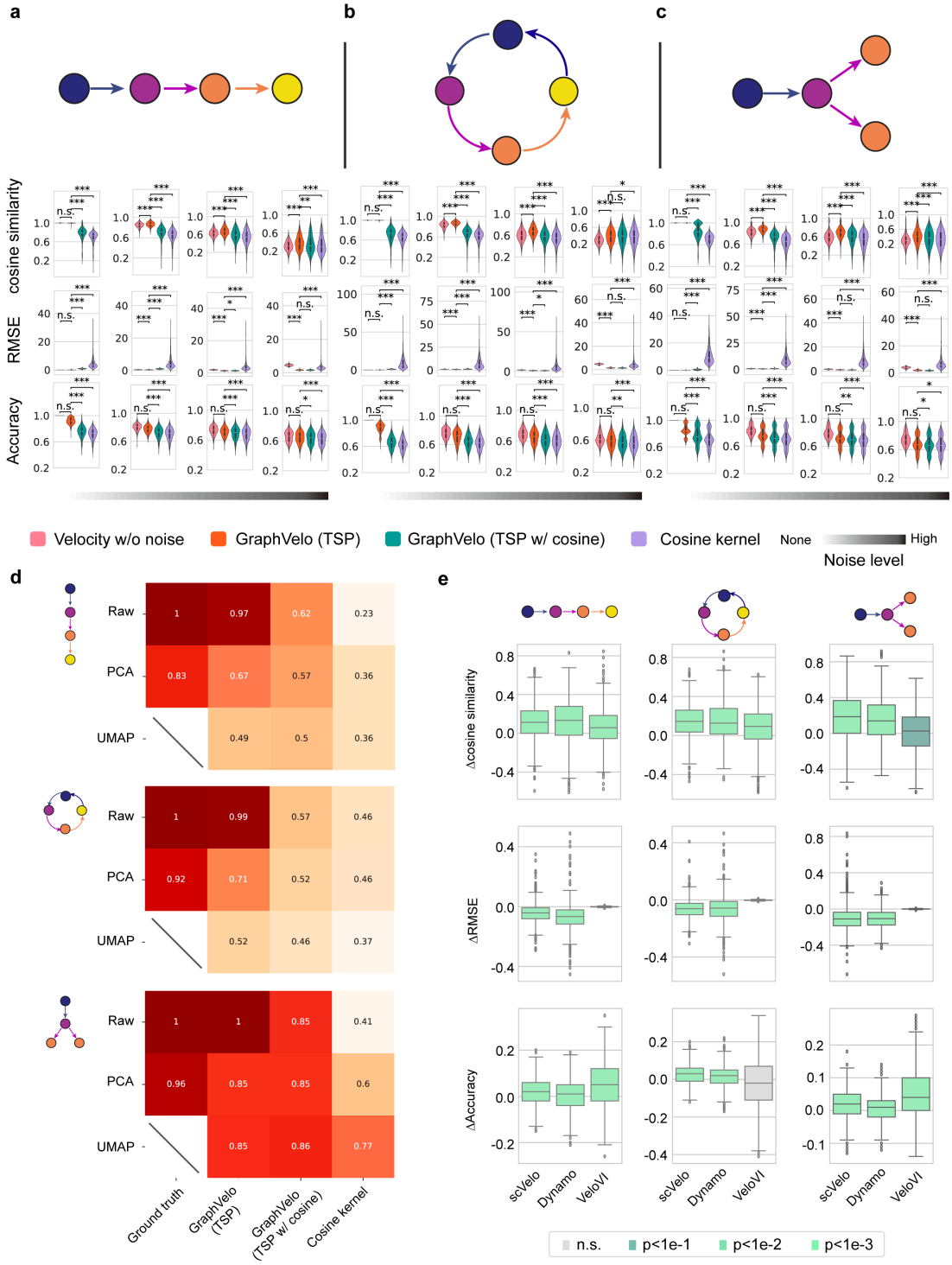
961 **(h)** GO enrichment of decoupled genes in (g).

962 **(i)** Line plot of nomarlized RNA and chromatin velocity along pseudotime for genes predicted by
963 GraphVelo to have notable decoupling patterns. Chromatin velocity trends were colored as brown
964 and RNA velocity trends were colored as green.

965 **(j)** Heatmaps of Jacobian element distribution along the axis of regulator RNA abundance of four
966 regulator effector circuits: i) *Lef1* versus *Wnt3* chromatin accessibilities. ii) *Hoxc13* versus *Wnt3*
967 chromatin accessibilities. iii) *Lef1* versus *Wnt3* transcription. iv) *Hoxc13* versus *Wnt3*
968 transcription.

969 **(k)** Effective dose-response curves obtained from integrating the averaged Jacobian elements over
970 the corresponding normalized regulator mature mRNA regulator level in (j).

Extended Data Figures:

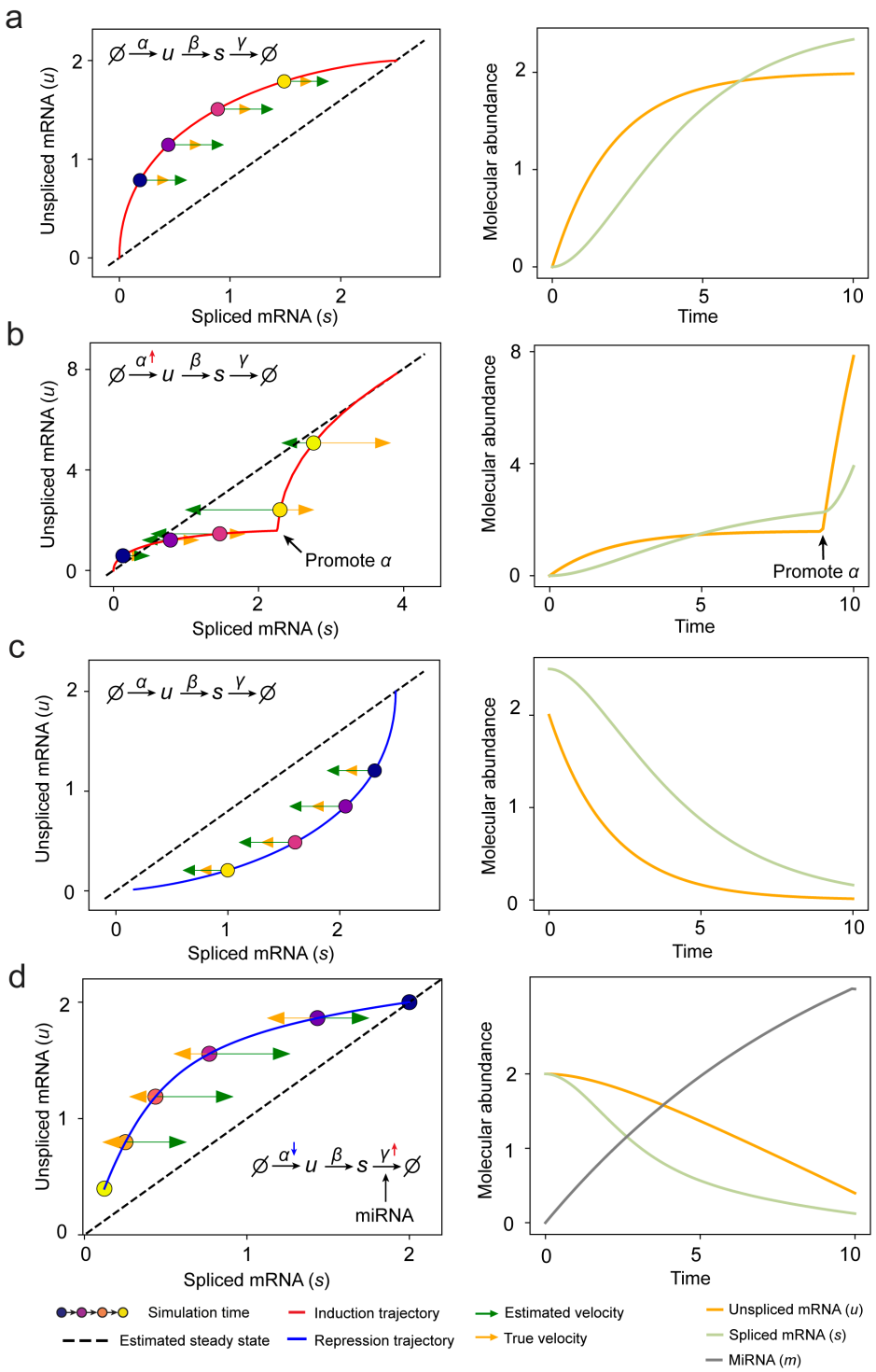


Extended Data Fig. 1. Refining RNA velocity from noisy simulation data or tradition splicing-based methods.

975 **(a-c)** Evaluation of on simulated scRNA-seq data under linear, cycling and bifurcating
976 differentiation models with an increasing noise level in velocity vectors.

977 **(d)** Heatmap of the correlation of cell speed calculated as the norm of velocity vector with respect
978 to the full-dimensional RNA velocity and the norm of velocity vectors projected to PCA or UMAP
979 space using GraphVelo, GraphVelo with cosine regularization, cosine kernel.

980 **(e)** Boxplots of metric evaluations on GraphVelo correction to the original velocities estimated by
981 scVelo, dynamo, and VeloVI. Box colors indicate the statistical test results for improvement
982 relative to zero.



983

984 **Extended Data Fig. 2. RNA velocity inference on simulation results of splicing kinetics with**
985 **constant or changed rates.**

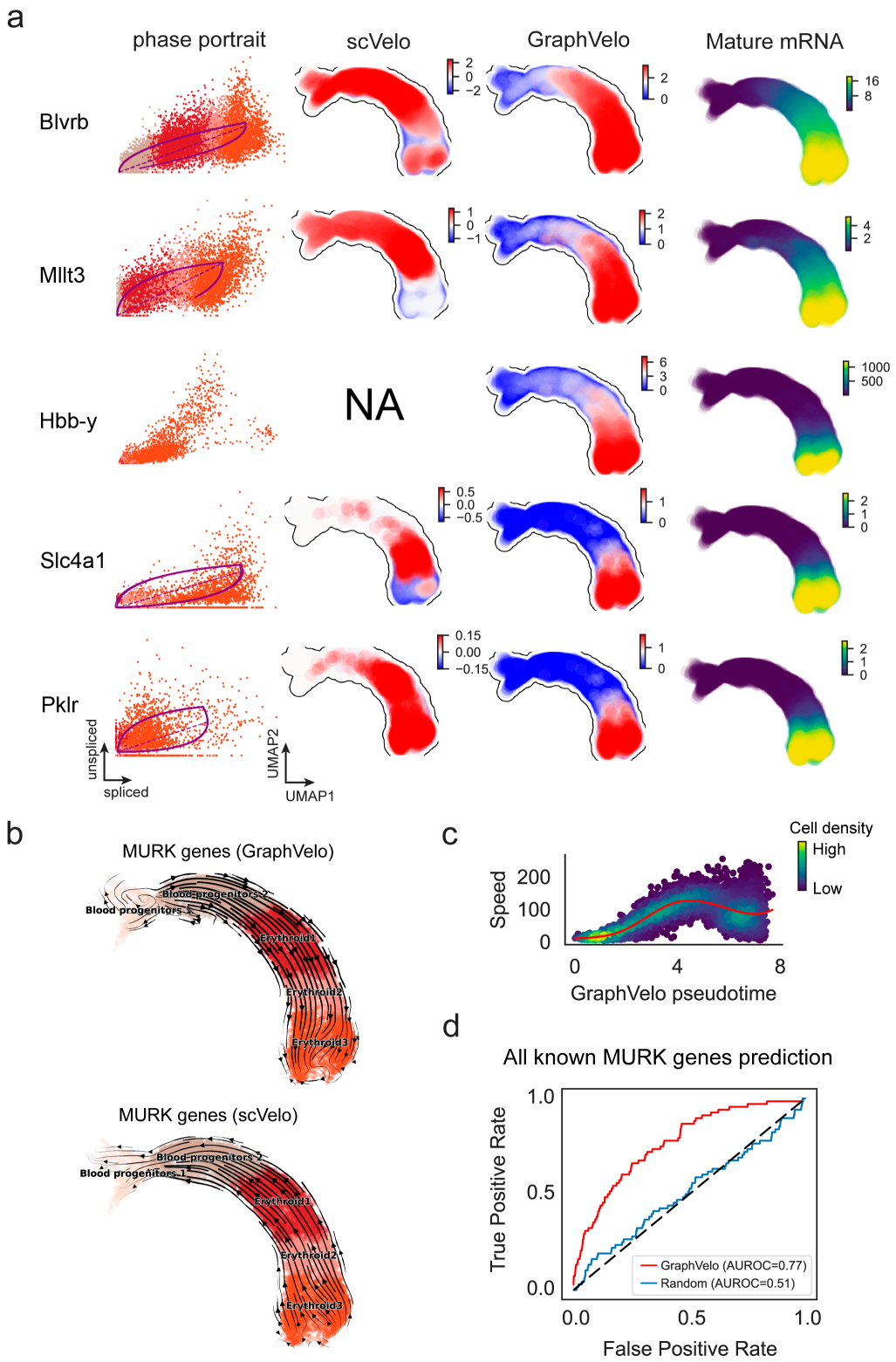
986 **(a)** Standard splicing kinetics along an induction trajectory in phase portrait and the reactions
987 define how abundance levels of molecules change along simulation time. Right panel shows

988 corresponding trajectories over time (same below).

989 **(b)** Transcription burst along an induction trajectory in phase portrait. The transcription rate
990 constant $\boldsymbol{\alpha}$ was promoted at specific time point.

991 **(c)** Standard splicing kinetics along a repression trajectory.

992 **(d)** Rapid time-varying degradation kinetics along repression trajectory. External signal promotes
993 synthesis of microRNA, which enhances degradation of target mRNA resulting in a microRNA -
994 dependent varying degradation rate "constant" $\boldsymbol{\gamma}$, and inhibits the target gene via a decreasing $\boldsymbol{\alpha}$.



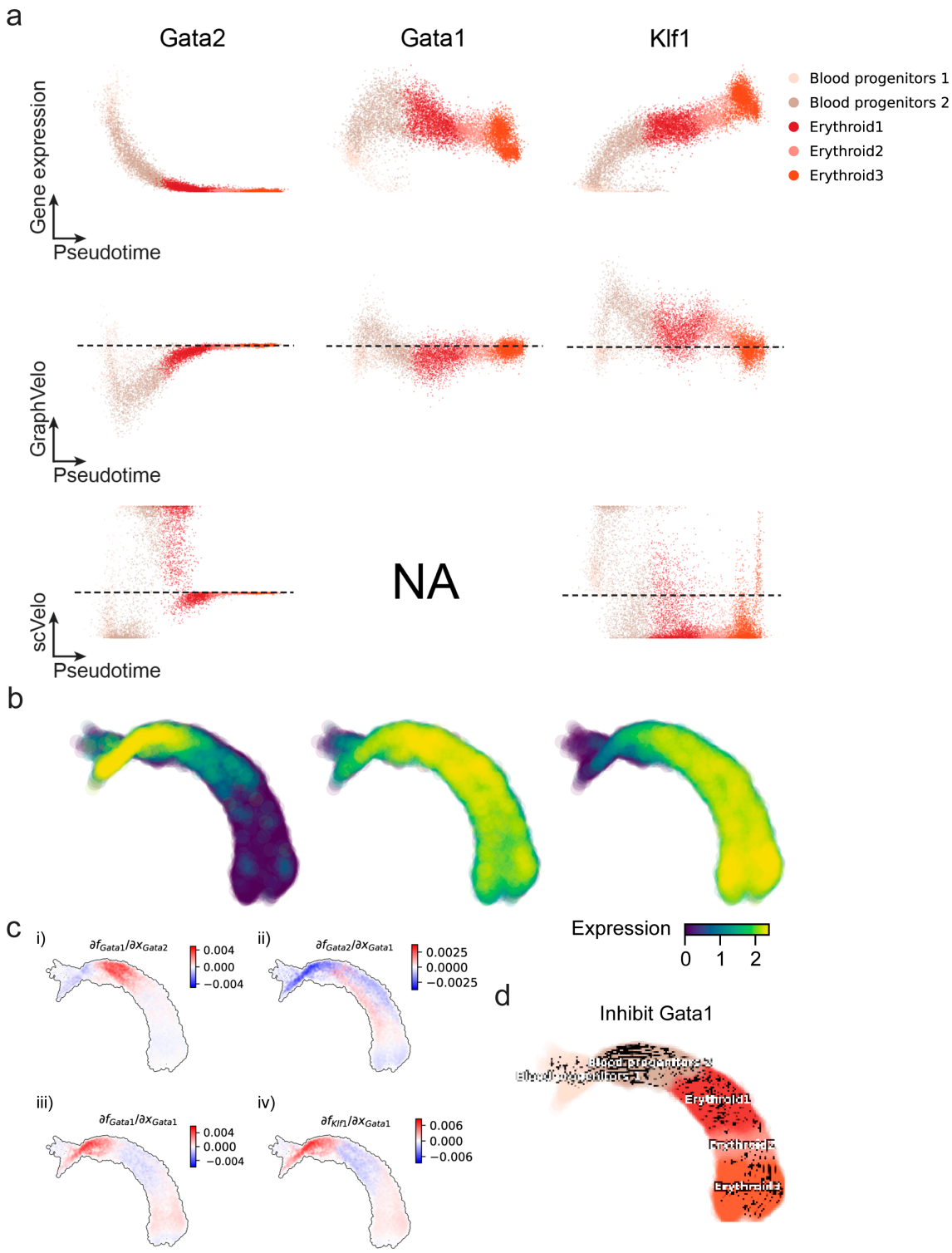
Extended Data Fig. 3. GraphVelo extrapolation of RNA velocities of an extended list of gene set in mouse erythroid maturation dataset.

998 **(a)** Phase portrait, velocity estimated by scVelo, refined velocity by GraphVelo, and gene
999 expression of mature mRNA of a selected set of genes. Cells were colored by cell type,
1000 corresponding velocity, and mature mRNA abundance, respectively, and visualized on the phase
1001 portrait and UMAP, respectively.

1002 **(b)** Velocities of MURK genes derived from GraphVelo and scVelo for gastrulation erythroid
1003 maturation cells projected to a predefined UMAP representation. Note that RNA velocity
1004 estimated by scVelo was in a reverse flow, possibly influenced by transcription burst events.

1005 **(c)** Cell speed distribution along the vector field pseudotime axis. Cells were colored by local
1006 density and red line indicates the fitted curve.

1007 **(d)** Receiver operating curve analyses of MURK gene prediction based on dynamo accelerations
1008 analysis using GraphVelo as input, in contrast to a random predictor. AUC, area under curve.



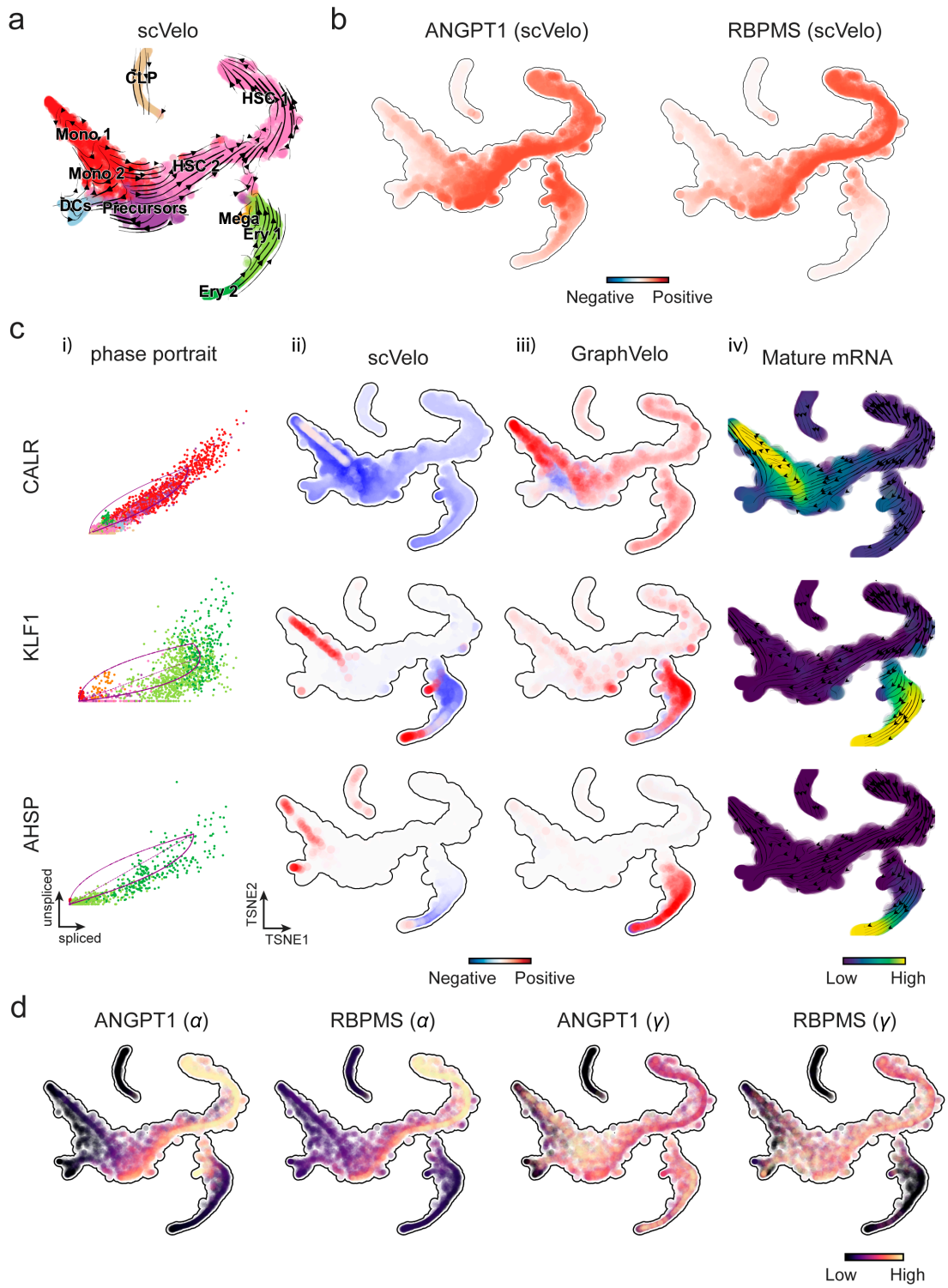
Extended Data Fig. 4. Regulatory cascades of driver TFs in mouse erythroid data.

(a) Expression and corresponding velocities of *Gata2*, *Gata1*, *Klf1* along the vector field-based pseudotime in mouse erythroid data.

1013 **(b)** Gene expression of *Gata2*, *Gata1*, *Klf1* on the UMAP space.

1014 **(c)** Molecular mechanisms of driver TFs underlying erythroid lineage commitment based on
1015 Jacobian analyses. (i) *Gata2* activates *Gata1*. (ii) Repression of *Gata2* by *Gata1*. (iii) Self-
1016 activation of *Gata1*. (iv) *Gata1* activates *Klf1*.

1017 **(d)** In silico perturbation analyses on GraphVelo-based vector field to examine the role of *Gata1*
1018 in gastrulation erythroid maturation.



Extended Data Fig. 5. RNA velocity estimated by scVelo and refined by GraphVelo from a hematopoiesis dataset.

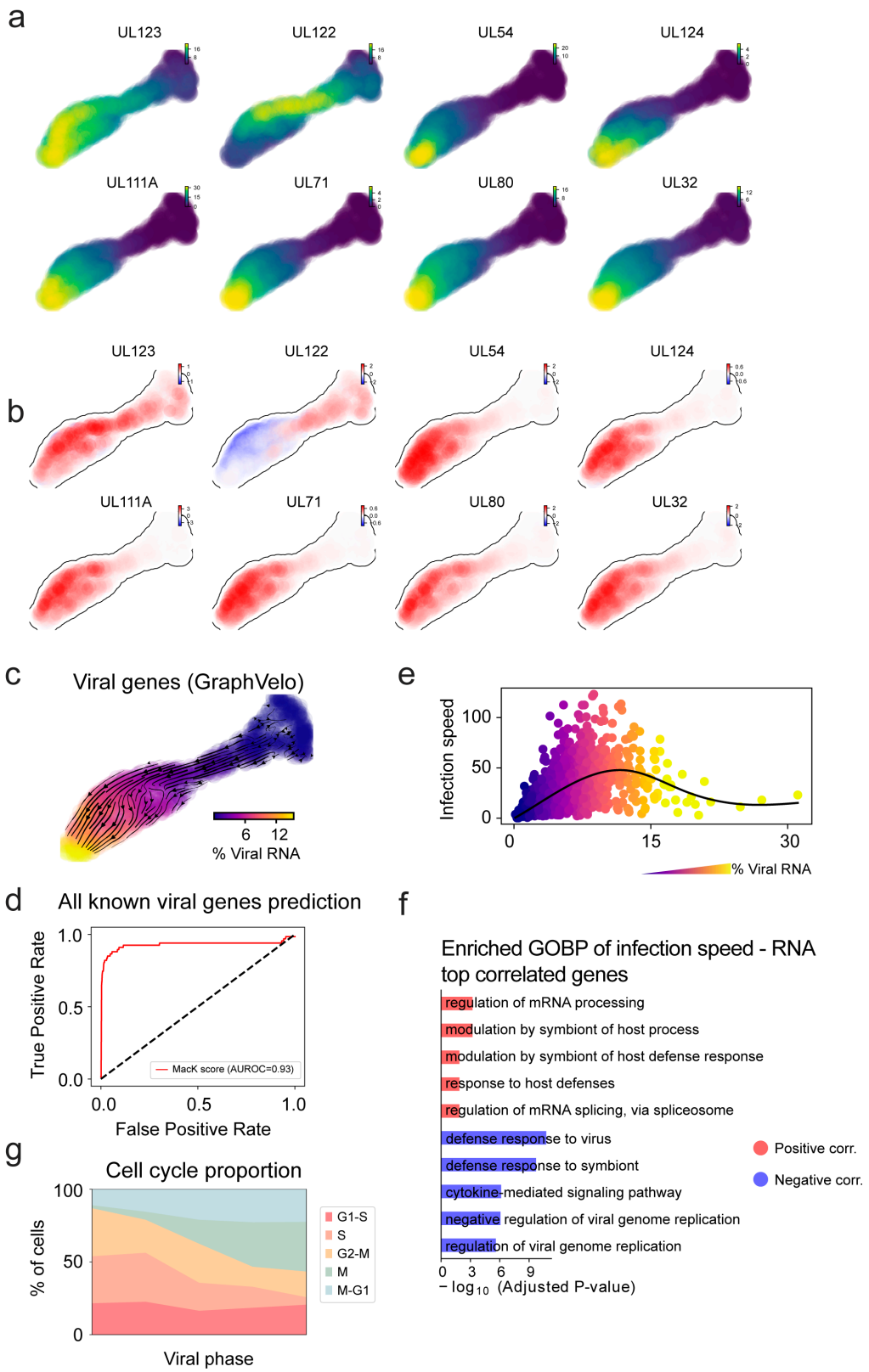
(a) Velocities derived from scVelo in the hematopoiesis development and projected to a pre-

1023 defined TSNE embedding.

1024 **(b)** TSNE visualization of the RNA velocity estimated by scVelo of *ANGPT1* and *RBPMS* genes.

1025 **(c)** Scatter plots of: i) phase portrait, ii) velocities estimated by scVelo, iii) refined velocities by
1026 GraphVelo, and iv) mature mRNA expression of transcription burst genes (e.g. *CALR*, *KLF1* and
1027 *AHSP*). Cells were colored by cell type, corresponding velocity, and mature mRNA abundance,
1028 respectively, and visualized on the phase portrait and TSNE, respectively.

1029 **(d)** Cell-specific transcription rate constant $\boldsymbol{\alpha}$ and degradation rate constant $\boldsymbol{\gamma}$ of rapid degradation
1030 genes *ANGPT1* and *RBPMS* visualized on the TSNE space, which is consistent with simulation
1031 result in Extended Data Fig. 2d.



Extended Data Fig. 6. Dynamic patterns of gene trajectories during HCMV infection.

1034 **(a)** Gene expression of selected viral factors on the UMAP space.

1035 **(b)** Corresponding velocities inferred by GraphVelo for viral factors in (a).

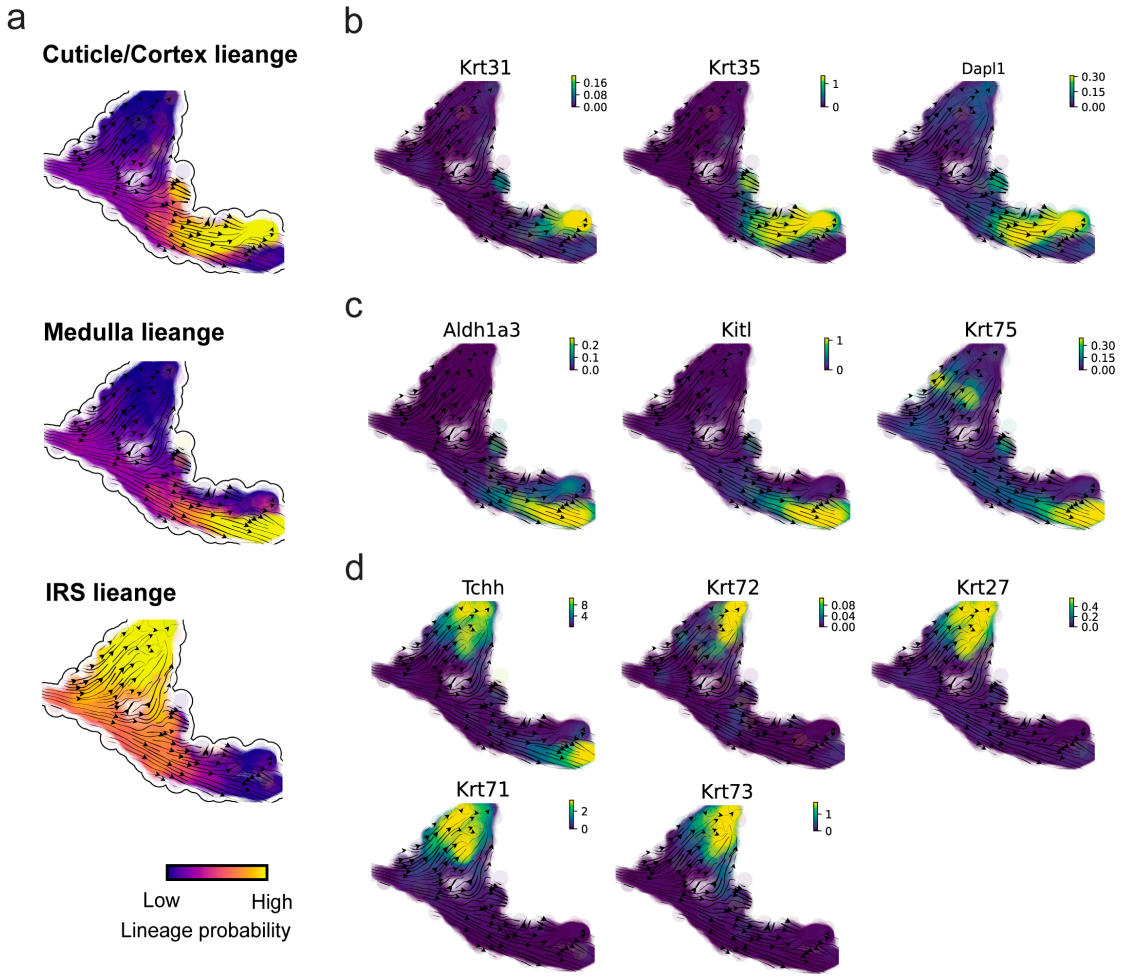
1036 **(c)** Viral RNA velocities derived from GraphVelo for infected cells, projected to the UMAP
1037 representation only using viral genes.

1038 **(d)** Receiver operating curve (ROC) analyses of MacK scores when using all detected viral genes
1039 as the gold standard.

1040 **(e)** Viral infection speed versus viral load. Cells were colored by the percentage of viral RNA.

1041 **(f)** GO enrichment analyses of top host genes correlated with infection speed.

1042 **(g)** Shifts in cell proportions across different cell cycle phases along the viral phase trajectory.
1043 Cells were divided into bins according to viral RNA percentage. The result indicates a decreasing
1044 S phase population corresponding to infection-driven transcriptional alterations of the cell cycle.
1045

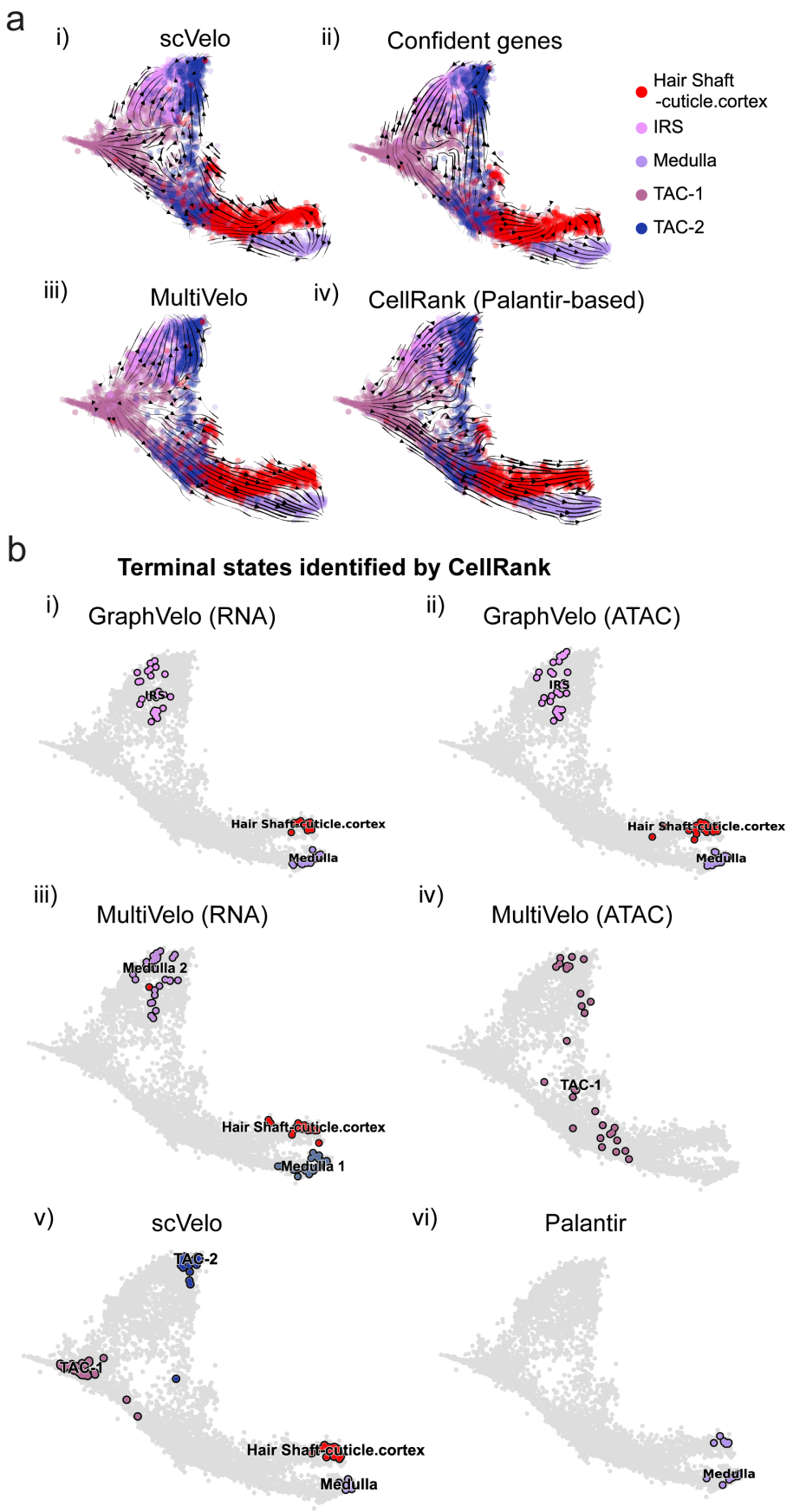


Extended Data Fig. 7. GraphVelo estimation of modality-consistent vector field based on multi-omics velocities.

(a) Velocity field projected to the pre-defined UMAP representation with different methods or gene sets: i) scVelo RNA velocity based on all velocity genes; ii) scVelo RNA velocity based on confident velocity genes filterred by dynamo criteria; iii) MultiVelo RNA velocity based on all velocity genes; iv) CellRank pseudotime kernel based on Palantir pseudotime.

(b) Terminal states identified by CellRank using different representation and corresponding velocity: i) GraphVelo-corrected RNA velocity; ii) GraphVelo-computed chromatin velocity; iii) MultiVelo-computed RNA velocity; iv) MultiVelo-computed chromatin velocity; v) scVelo-computed RNA velocity; vi) Palantir-based pseudotime kernel.

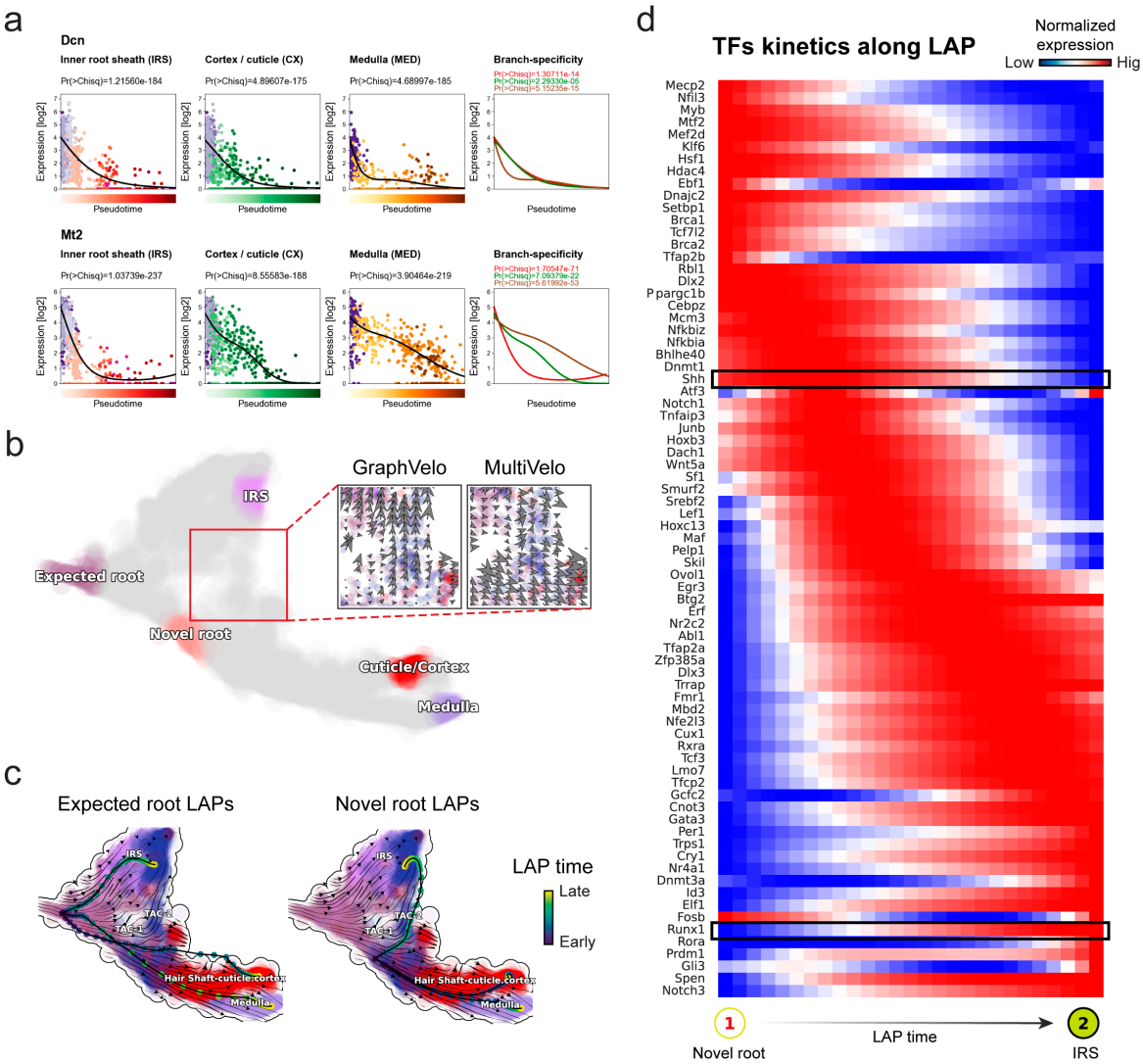
1058


a
i)
scVelo
ii)
Confident genes
Hair Shaft -cuticle.cortex
IRS
Medulla
TAC-1
TAC-2
iii)
MultiVelo
iv)
CellRank (Palantir-based)
b
Terminal states identified by CellRank
i)
GraphVelo (RNA)
IRS
Hair Shaft-cuticle.cortex
Medulla
ii)
GraphVelo (ATAC)
IRS
Hair Shaft-cuticle.cortex
Medulla
iii)
MultiVelo (RNA)
Medulla 2
Hair Shaft-cuticle.cortex
Medulla 1
iv)
MultiVelo (ATAC)
TAC-1
v)
scVelo
TAC-2
TAC-1
Hair Shaft-cuticle.cortex
Medulla
vi)
Palantir
Medulla

Extended Data Fig. 8. Reconstructed lineage commitment during mouse hair follicle differentiation using GraphVelo velocities.

(a) Lineage commitment probablity of each terminal cell type on UMAP vector field.

(b-d) Gene expression distribution of markers for cuticle/cortex, medulla and IRS lineages on the UMAP vector field.



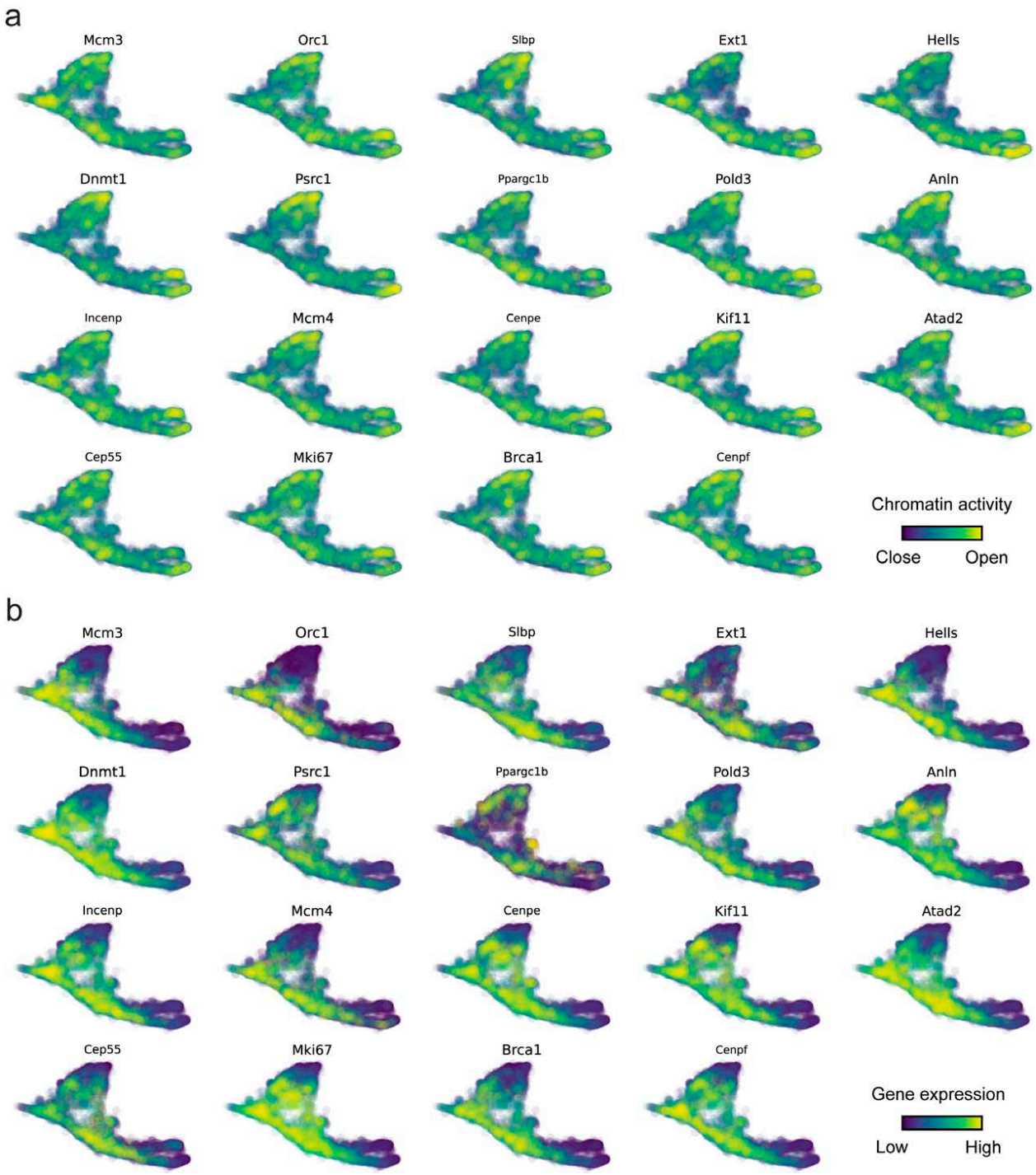
Extended Data Fig. 9. Novel root identified with multi-omic vector field from GraphVelo velocities.

(a) *Dcn*, *Mt2* expression dynamics during anagen hair follicle keratinocytes[46].

1069 **(b)** Regions identified by topological analysis. Insert are ummarized cell-state transition vectors
1070 calculated by GraphVelo and MultiVelo along the path from the novel root to IRS and projected
1071 onto the UMAP representation.

1072 **(c)** Predicted developmental LAPs from expected root or novel root to to each of the terminal cell
1073 types in the UMAP embedding. Color of the node along the paths indicates the LAP transition
1074 time.

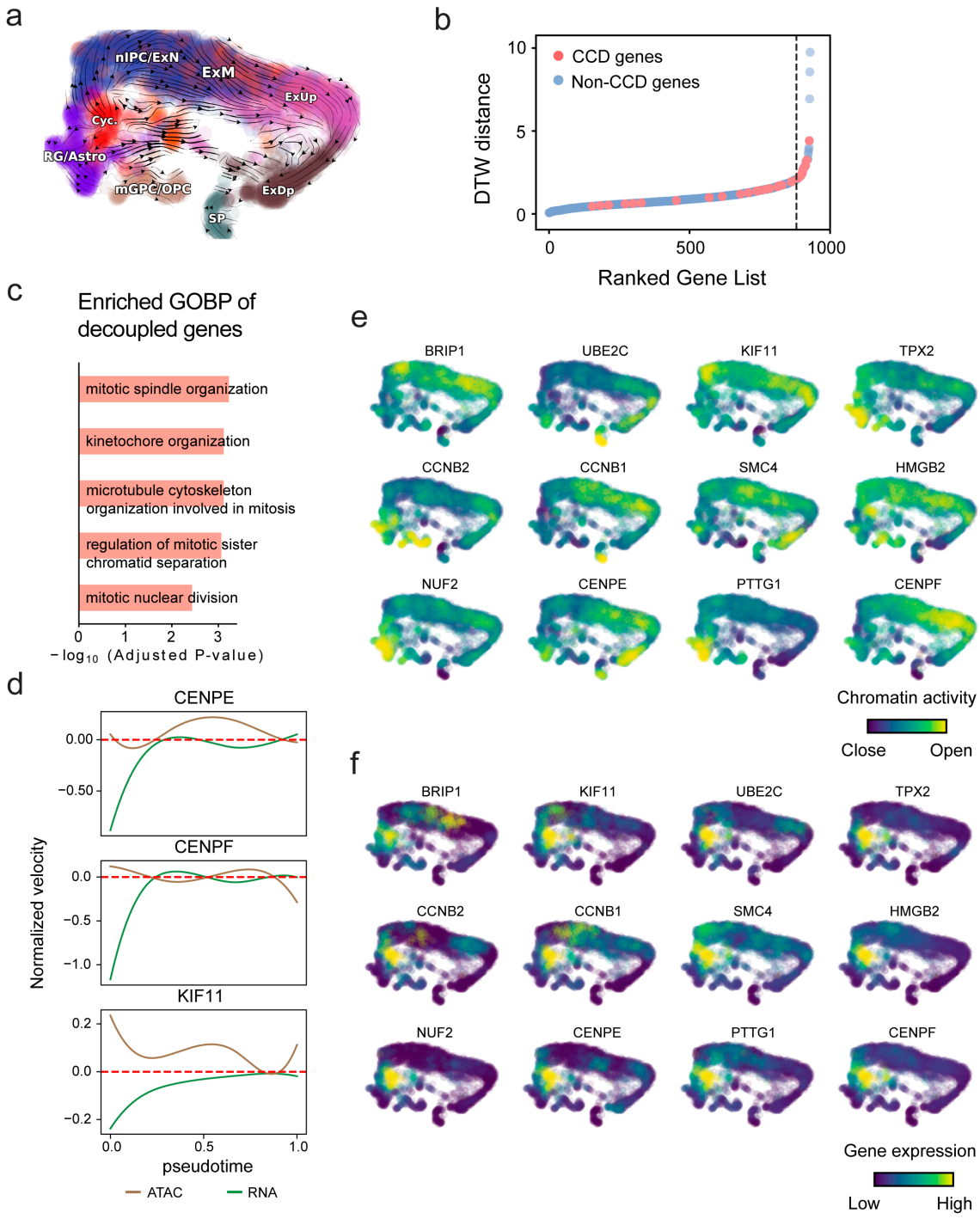
1075 **(d)** TF expression profiles along the LAP from novel root to IRS. *Shh* decays, alongside the
1076 induction of the *Runx1* gene.



Extended Data Fig. 10. Genomic patterns of decoupled CCD genes.

(a) Chromatin activity of decoupled CCD genes.

(b) Gene expression of decoupled CCD genes.



Extended Data Fig. 11. GraphVelo inferrence on epigenome and transcriptome decoupling dynamics in fetal human brain.

(a) GraphVelo velocity field colored by cell macrostates.

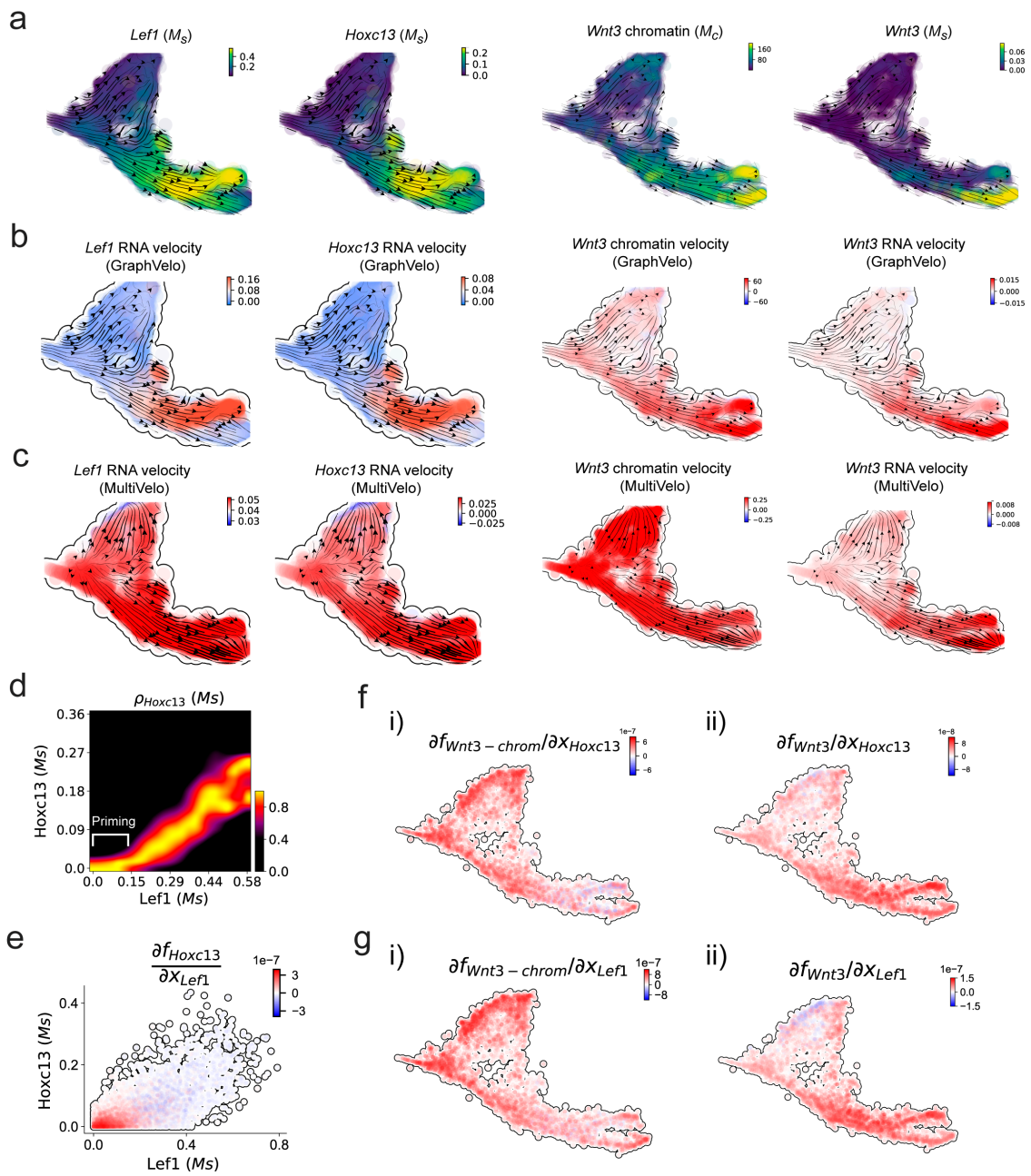
1085 **(b)** DTW distance between RNA velocity and chromatin velocity of individual genes as a measure
1086 of the coupling/decoupling status. CCD genes were colored in red. The dotted line indicates the
1087 elbow point, with the decoupled genes on its right.

1088 **(c)** GO enrichment of decoupled genes in (c).

1089 **(d)** Line plot of nomarlized RNA and chromatin velocity along dynamo vector field pseudotime
1090 for genes predicted by GraphVelo to have notable decoupling patterns. Chromatin velocity trends
1091 were colored in brown and RNA velocity trends were colored in green.

1092 **(e)** Chromatin activity of decoupled CCD genes.

1093 **(f)** Gene expression of decoupled CCD genes.



Extended Data Fig. 12. GraphVelo prediction on the regulatory mechanisms of *Lef1-Hoxc13-Wnt3* circuit during mouse skin development.

(a) Distributions of *Lef1* expression, *Hoxc13* expression, *Wnt3* chromatin openess/accessibility and *Wnt3* expression, respectively, on the projected UMAP vector field.

(b) Velocities of *Lef1* RNA, *Hoxc13* RNA, *Wnt3* chromatin and *Wnt3* RNA inferred by GraphVelo, visualized on the projected UMAP vector field.

1101 **(c)** Velocities of *Lef1* RNA, *Hoxc13* RNA, *Wnt3* chromatin and *Wnt3* RNA inferred by MultiVelo,
1102 visualized on the projected UMAP vector field.

1103 **(d)** Expression of *Hoxc13* gene versus *Lef1* expression. *Lef1* expression primes the activation of
1104 *Hoxc13*.

1105 **(e)** Cell-wise Jacobian analyses of *Lef1-Hoxc13* activation cascade.

1106 **(f)** Jacobian analyses of regulatory interactions between potential PTF *Hoxc13* and i) *Wnt3*
1107 chromatin accessibility or ii) transcription..

1108 **(g)** Jacobian analyses of regulatory interactions between PTF *Lef1* and i) *Wnt3* chromatin
1109 accessibility or ii) transcription.

Supplemental text

I. Connection of the projection to local linear embedding (LLE)

In this work, we use the neighborhood of a data point to approximate the corresponding tangent space of the data manifold. Such approximation has been used in other applications such as the local linear embedding (LLE)[1]. Here we provide a derivation of representing velocity vectors based on LLE.

Consider a data point $\mathbf{x}_i$ and its neighbors. Applying LLE one has $\sum_{j\neq i} w_{ij}(x_j - x_i) = \sum_{j\neq i} w_{ij}\delta_{ij} = 0$, with $\sum_j w_{ij} = 1$. The LLE algorithm gives $w_{ij} = \frac{\sum_k C_{jk}^{-1}}{\sum_{lm} C_{lm}^{-1}}$, with $C_{ij} = (x_i - x_j) \cdot (x_i - x_k)$. Now consider after $\delta t \to 0$ $\mathbf{x}_i$ moves to $\mathbf{x}_i' = \mathbf{x}_i + \boldsymbol{v}_{pi}\delta t$, which should be also within the local linear embedding space. Again with LLE one has $\sum_{j\neq i} w_{ij}'(x_j - x_i') = 0$, with $\sum_j w_{ij}' = 1$. Then one has $\boldsymbol{v}_{pi}$=$\sum_{j\neq i} \frac{(w_{ij}' - w_{ij})}{\delta t}\delta_{ij} = \sum_{j\neq i} \phi_{ij}\delta_{ij}$, with a form similar to that in eq. 1 of the main text.

The above derivation suggests an iterative trajectory propagation algorithm for obtaining the velocities in the new representation:

1) Use a trial $\boldsymbol{v}_{pi}$ to propagate the state and obtain $(\mathbf{x}_i, \mathbf{x}'_i)$. One can propagate backward and forward by $\delta t/2$ and use central difference to estimate the velocity vectors.
2) Apply LLE to get w_{ij} and w'_{ij}. I
3) Obtain a new $\boldsymbol{v}_{pi}$
4) Go back to step 1 until the result converges below a threshold.

We numerically tested the above algorithm, the one with the loss function given in eq. 2 of the main text with the parameter $b = 0$ and $b \neq 0$. All the three algorithms worked well on synthetic data. The first two actually outperformed the third one (used in the main text). However, the first two sometime showed numerical instability once applied to real single cell data. Therefore, in subsequent applications, we used the general loss function form in eq. 2, using the direction information from the second term (with $b \neq 0$) to further regularize the projection.

II. Numerical issues of using eq. 1 for tangent space projection

The relation in eq. 1 provides an algorithm for projecting a measured $\mathbf{v}$ onto $\mathscr{M}$ by minimizing the following loss function,

$$\mathcal{L}(\boldsymbol{\phi}_i) = \left\|\boldsymbol{v}_i - \boldsymbol{v}_{\|i}\right\|^2 + \lambda\|\boldsymbol{\phi}_i\|^2,$$

which unfortunately is numerically unstable. The redundancy of the basis vectors leads to coefficients that are not uniquely determined, and failure of the projection. To see the latter, notice that in real data the subspace formed by the displacement vectors only approximates the tangent space $T\mathscr{M}$ locally, and it likely contains small components in the orthogonal space. Then the projection procedure tries to express both $\mathbf{v}_{\|}(\mathbf{x})$ and all or part of the remaining $(\mathbf{v}_{\boldsymbol{i}} - \mathbf{v}_{\|})$ as a linear combination of $\boldsymbol{\delta}_{\mathrm{ij}}$ with some $|\phi_{\mathrm{ij}}| \gg 0$, which can only be counterbalanced by a very large value of λ. The latter would impose large weight on the regularization. While further data preprocessing may be developed in the future to alleviate the problem, in this work we provided a strategy of adding an additional cosine kernel term for the direction information.

We also experimented with the idea of removing the redundancy through dimension reduction with PCA. The numerical results were not satisfactory, since it converged slowly with the sampling size, which becomes impractical.

III. Reformulation of Cosine kernel used in the literature in the context of tangent space projection

In the original RNA velocity study, a Cosine kernel has been proposed to transform the RNA velocity vectors between different representations[2]. We re-casted the cosine kernel in the context of the tangent space projection with a form similar to that of eq. 1 in the main text:

$$\mathbf{v}_{\parallel}^{corr}(\mathbf{x}_i) = \sum_{j \in \mathcal{N}_i} \phi_{ij}^{\mathrm{corr}} \boldsymbol{\delta}_{ij},$$

where $\phi_{ij}^{\mathrm{corr}} = P_{ij} - \frac{1}{k}$, and $P_{ij} = \frac{\exp[\cos(\mathbf{v}_i, \boldsymbol{\delta}_{ij})/\sigma]}{\sum_m \exp[\cos(\mathbf{v}_i, \boldsymbol{\delta}_{im})/\sigma]}$ defined in the form of softmax functions, with $\cos(\cdot,\cdot)$ denoting the cosine similarity between two input vectors, σ an arbitrary bandwidth parameter, and k the number of neighbors for each cell. Here P_{ij} gives a heuristic transition probability from cell i to j. The term $(-1/k)$, called the "density correction", is designed to correct the potential sampling bias where the embedded velocity vectors tend to point towards the direction of regions with high cell density. Li et al. showed that mathematically the cosine kernel asymptotically converges to the correct direction of a velocity vector[3]. However, the correlation kernel loses information about the magnitude of the velocity vectors $\mathbf{v}_i$ (i.e., the *speed*), due to the normalization in the correlation functions. Intuitively, the correlation kernel is qualitatively guided by the physical intuition that a cell has a high tendency to move along the direction of its velocity vector. Here, we used the direction information of $\phi_{ij}^{\mathrm{corr}}$ to help on constraining $\boldsymbol{\phi}_i$.

IV. Further discussions on applications of GraphVelo to manifolds formed by multi-modal single cell data.

GraphVelo is based on the following mathematical assumption: the manifolds of a given system embedded in two different spaces are homeomorphic so one can establish a one-to-one mapping between the two. Here we use one simple example of scRNAseq/scATACseq multiomic data to illustrate that one can still apply GraphVelo if this assumption is violated. Consider that a cellular system switches between two discrete epigenetic states, and for each epigenetic state there is a corresponding (quasi)continuous transcriptomic manifold. Label the two disjoint manifolds as $\mathcal{M}_1(\mathbf{x})$ and $\mathcal{M}_2(\mathbf{x})$, where $\mathbf{x}$ represents the transcriptomic state. Consider two cells having the same $\mathbf{x}$ but different chromatin state, c_1 on $\mathcal{M}_1$ and c_2 on $\mathcal{M}_2$. Multiomic data allows distinction between two cells. That is, the neighborhood of c_1 is composed of cells on $\mathcal{M}_1$, and the neighborhood of c_2 is composed of cells on $\mathcal{M}_2$. Therefore, GraphVelo analyses treat cells on $\mathcal{M}_1$ and $\mathcal{M}_2$ separately.

V. Feature selection for robustly estimated velocity genes.

In datasets featuring non-differentiating cell types organized in a hierarchical manner, application of RNA velocity may show different kinetic regimes. Analyzing cell fate transition using unreliable velocity genes can lead to emergence of spurious cell state transitions. Different packages subset the well-estimated kinetic genes for further analyses in different scenarios. In the implementation of scVelo, one first regresses out the uncertain genes are based on the r-squared coefficient in the steady-state model and further selects trustful velocity genes according to the log-likelihood in the dynamical mode. PhyloVelo relies on monotonically expressed genes to infer transcriptomic velocity, which is a more stringent requirement. However, the monotonic assumptions are counter intuitive and oversimplify the gene expression kinetics along differentiation or disease trajectories due to the non-sequential nature of these cascades. Dynamo

provides a biological prior-guided method and filters out the unreliable genes based on the knowledge of lineage information.

Reference

1. Roweis, S.T. & Saul, L.K. Nonlinear Dimensionality Reduction by Locally Linear Embedding. *Science* **290**, 2323-2326 (2000).
2. La Manno, G. *et al.* RNA velocity of single cells. *Nature* (2018).
3. Li, T., Shi, J., Wu, Y. & Zhou, P. On the Mathematics of RNA Velocity I: Theoretical Analysis. *CSIAM Transactions on Applied Mathematics* **2**, 1-55 (2021).