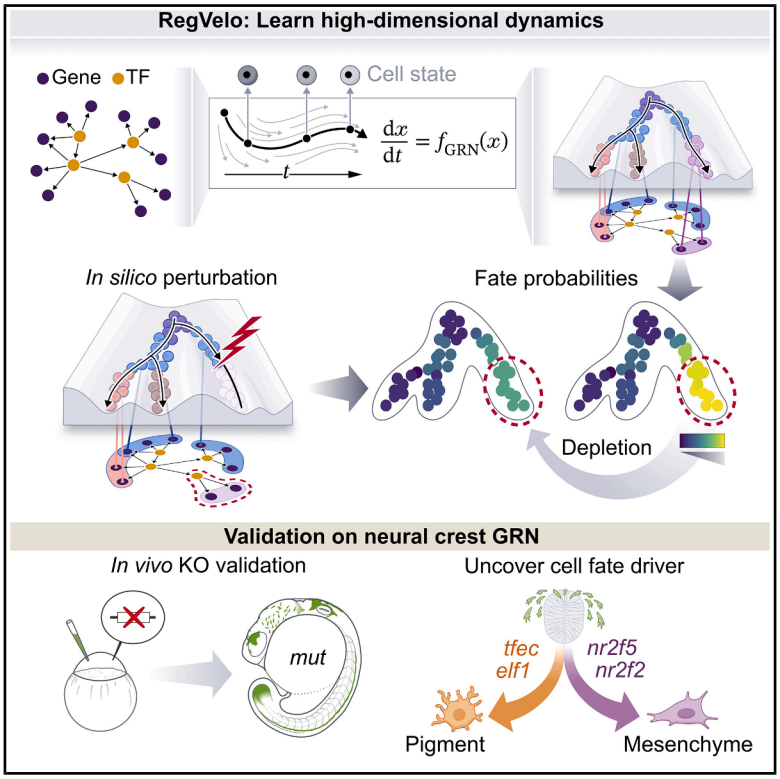
Cell

RegVelo: Gene-regulatory-informed dynamics of single cells

Graphical abstract



Highlights

- RegVelo integrates scRNA-seq with prior GRN knowledge to infer regulatory dynamics
- Transcription is modeled as a regulated process to capture coupled gene dynamics
- Integration with CellRank enables *in silico* regulatory perturbation and fate prediction
- Modeling cranial neural crest development identifies *elf1* as a regulator of pigment cell fate

Authors

Weixu Wang, Zhiyuan Hu, Philipp Weiler, ..., Zhengyuan Xue, Tatjana Sauka-Spengler, Fabian J. Theis

Correspondence

tsauka-spengler@stowers.org (T.S.-S.), fabian.theis@helmholtz-munich.de (F.J.T.)

In brief

RegVelo introduces an end-to-end generative framework that jointly infers gene regulatory networks and developmental dynamics. It serves as an actionable *in silico* "cell," enabling researchers to simulate regulatory perturbations and generate testable hypotheses about the mechanisms underlying cell fate decisions.

Wang et al., 2026, Cell *189*, 3773–3800
June 11, 2026
https://doi.org/10.1016/j.cell.2026.04.022

CellPress

Cell

CellPress
OPEN ACCESS

Article

RegVelo: Gene-regulatory-informed dynamics of single cells

Weixu Wang,[1,2,14] Zhiyuan Hu,[3,4,14] Philipp Weiler,[1,2,5,12,14] Sarah Mayes,[3] Marius Lange,[1,6] Daniel M. Fountain,[3] Julianna O. Haug,[7] Jingye Wang,[8,9] Zhengyuan Xue,[5,10] Tatjana Sauka-Spengler,[3,7,11,13,*] and Fabian J. Theis[1,2,5,13,15,*]

[1]Institute of Computational Biology, Computational Health Center, Helmholtz Munich, Munich, Germany
[2]TUM School of Life Sciences Weihenstephan, Technical University of Munich, Munich, Germany
[3]MRC Weatherall Institute of Molecular Medicine, Radcliffe Department of Medicine, University of Oxford, Oxford, UK
[4]Reproductive Medicine Center, Medical Research Institute, Frontier Science Center for Immunology and Metabolism, Zhongnan Hospital, Wuhan University, Wuhan, China
[5]School of Computation, Information and Technology, Technical University of Munich, Munich, Germany
[6]Department of Biosystems Science and Engineering, ETH Zürich, Basel, Switzerland
[7]Stowers Institute for Medical Research, Kansas City, MO, USA
[8]Institute for Systems Genetics, NYU Grossman School of Medicine, New York, NY, USA
[9]Department of Biology, New York University, New York, NY, USA
[10]Institute of Translational Genomics, Helmholtz Munich, Munich, Germany
[11]MRC WIMM Centre for Computational Biology, MRC Weatherall Institute of Molecular Medicine, University of Oxford, Oxford, UK
[12]Present address: Computational and Systems Biology Program, Memorial Sloan Kettering Cancer Center, New York, NY, USA
[13]These authors contributed equally as senior authors
[14]These authors contributed equally
[15]Lead contact
*Correspondence: tsauka-spengler@stowers.org (T.S.-S.), fabian.theis@helmholtz-munich.de (F.J.T.)
https://doi.org/10.1016/j.cell.2026.04.022

SUMMARY

Cell fate transitions are driven by regulatory circuitry, yet RNA velocity models cellular dynamics without explicitly accounting for gene regulatory interactions, limiting mechanistic insight. Conversely, gene regulatory network (GRN) inference methods largely neglect the dynamic nature of biological systems. To overcome this conceptual disconnect, we present RegVelo, a bottom-up, actionable, and interpretable deep learning framework that jointly models splicing kinetics and gene regulatory interactions. Across diverse biological systems, RegVelo provides reliable predictive power for terminal states, gene interactions, and perturbation simulations. By applying RegVelo to zebrafish neural crest development using full-length Smart-seq3 and shared gene expression and chromatin accessibility measurements, we delineate regulatory programs underlying fate specification. Guided by *in silico* perturbations and validated by CRISPR-Cas9 knockout and single-cell Perturb-seq, we establish *tfec* as an early driver and *elf1* as a regulator of pigment cell fate. RegVelo establishes a quantitative framework for bridging gene regulation and cell fate decisions.

INTRODUCTION

Single-cell assays delineate cell differentiation at a high resolution.[1–3] Waddington's landscape[4] conceptualizes phenotypic states as valleys, with branching ridges shaping cell differentiation. However, constructing models that jointly capture gene regulatory networks (GRNs) and cellular dynamics along this landscape remains a major challenge in computational single-cell biology. Such models are essential for mechanistic insight and for predicting dynamic effects of gene knockouts *in silico* to guide experiments and uncover previously uncharacterized biological findings. Current methods either infer developmental trajectories without modeling gene regulatory relationships[5,6] or they infer GRNs but omit cell dynamics.

Pseudotime[7–11] and RNA velocity[12–14] are the two most widely used concepts for reconstructing cell trajectories[15]: pseudotime orders cells along differentiation paths but requires a predefined root and lacks kinetic information to simulate cellular dynamics; RNA velocity attempts to overcome these limitations via a mechanistic model of spliced mRNA dynamics, enabling estimation of instantaneous state changes and simulation of cell state transitions.[16,17] However, RNA velocity relies on restrictive assumptions, including gene independence and constant transcription rates,[13,14] thereby excluding transcriptional regulation underlying cell differentiation.

Advances in single-cell RNA sequencing (scRNA-seq) have introduced numerous methods for inferring GRNs,[18–23] and orthogonal data views, such as single-cell assay for

transposase-accessible chromatin using sequencing (scATAC-seq), provide epigenetic priors.[24–28] GRNs are commonly used to quantify gene importance,[26,27] refine cell state[18,24] and trajectories,[27] or predict effects of *in silico* gene perturbations.[23,24,26] However, these approaches typically do not model cellular dynamics or connect regulation to dynamics only post hoc.[27] Thus, models that directly link gene regulation to cellular dynamics are needed to elucidate how GRNs orchestrate dynamic cellular processes.

To address this gap, we introduce RegVelo (regulatory velocity), a method that infers transcriptome-wide splicing dynamics coupled through gene regulation. RegVelo leverages deep generative modeling to jointly infer kinetic parameters and latent time via cell-gene information sharing.[14] Building on our earlier velocity variational inference (veloVI)[14] and related models,[13,29–31] RegVelo incorporates a prior GRN-informed neural network that couples gene dynamics by modeling transcription as a regulation- and time-dependent process. The resulting model is a nonlinear, genome-wide dynamic differential equation system, parametrizable and learnable at scale, in contrast to the traditional single-gene closed-form systems. As such, RegVelo provides a continuous velocity vector field and quantifies the uncertainty in cellular state changes along the differentiation path, enabling forward simulations of cell state transitions and regulon-wise *in silico* perturbations, thereby yielding mechanistic predictions of how regulatory changes drive cell fate decisions.

When applied to simulated data and cell cycle datasets, RegVelo's velocity and latent time inference outperforms competing methods that employ less faithful models of transcription dynamics.[29–32] Across datasets of human hematopoiesis and mouse pancreatic endocrinogenesis, RegVelo's velocity inference yielded more consistent fate mapping and identification of putative driver genes than scVelo[13] and veloVI.[14] Integrated with CellRank 2's model-agnostic framework,[17] RegVelo accurately predicts effects of regulatory perturbations on cell fate decisions and reveals gene regulatory circuits underlying lineage-specific events. Applied to zebrafish neural crest development, RegVelo predicts cell fate priming and key lineage drivers, identifying *tfec* as a key early transcriptional driver of pigment cell formation and uncovering *elf1* as a pigment lineage regulator through *in silico* perturbation. These predictions were validated by *in vivo* perturbation experiments, showing strong concordance with experimental observations and significantly improved performance over existing perturbation prediction models.

RESULTS

Learning gene-regulation-informed cellular dynamics

RegVelo is a Bayesian deep generative model of cellular dynamics that takes unspliced and spliced abundances as input and infers gene- and cell-specific latent time, posterior velocity distributions, and a GRN representing transcriptional regulation (STAR Methods). It models dynamics of unspliced and spliced RNA readouts for each gene in each cell using kinetic and neural network parameters governing transcription, splicing, degradation, latent time, and regulatory interactions. By explicitly modeling regulation of target genes by upstream factors,[33] RegVelo models non-constant, cell-specific transcription rates and incorporates gene regulatory mechanisms, an essential aspect of biological processes largely overlooked by existing velocity models. Together with other kinetic parameters, we thereby define the developmental landscape and align cells along it.

RegVelo represents the GRN as a weight matrix, parametrized by a shallow neural network to estimate transcription rates for each gene in each cell. Each entry of the GRN matrix describes the effect of a regulator on the transcription of its targets: positive values describe activation, negative values repression, and zero entries indicate the absence of a regulatory relationship. During model training, we update the GRN parameters to infer the transcriptional dynamics of each gene in a data-driven fashion. At the same time, we constrain the GRN weights by prior knowledge of gene regulation using a penalty function; RegVelo allows prior information on gene regulation curated from single-cell multi-omics experiments[34] or public gene regulation databases.[35] GRN inference often faces two challenges: erroneous a priori GRNs and the need to account for sparsity. Initial GRN estimates are imperfect due to inference from noisy data or incomplete understanding of the regulatory mechanisms, and sparsity is a common assumption in GRN inference[26,36] based on studies of biological systems.[37] To address these challenges, RegVelo refines initial GRN estimates by learning edge weights or additional potential regulatory interactions, and incorporates priors in parameter estimation that impose sparsity constraints either through a sparse prior network or L1-regularizing the Jacobian matrix of the dynamical system to impose smoothness of dynamics (STAR Methods).

RegVelo incorporates cellular dynamic estimates by first encoding unspliced (u) and spliced RNA (s) readouts into posterior parameters of a low-dimensional latent space—the cell representation—with a neural network. An additional neural network takes samples of this cell representation as input to parameterize gene-wise latent time, as in our previous model veloVI. We then model splicing dynamics with ordinary differential equations (ODEs) specified by a neural network bias b representing base transcription and GRN weight matrix W, describing transcription rate α and inferred by a shallow neural network, constant splicing and degradation rate parameters β and γ, respectively, and estimated cell- and gene-specific latent times (Figure 1A; STAR Methods). Importantly, existing methods for inferring RNA velocity consider a set of decoupled one-dimensional ODEs for which analytic solutions exist, but RegVelo relies on a single, high-dimensional ODE,

$$\frac{du_g(t)}{dt} = \alpha_g(t) - \beta_g u_g(t)$$
$$\frac{ds_g(t)}{dt} = \beta_g u_g(t) - \gamma_g s_g(t),$$

that is now coupled through gene regulation-informed transcription

$$\alpha_g = h\left([Ws(t) + b]_g\right),$$

where g indicates the gene and h is a nonlinear activation function (STAR Methods). We predict the gene and cell-specific spliced and unspliced abundances using a parallelizable ODE

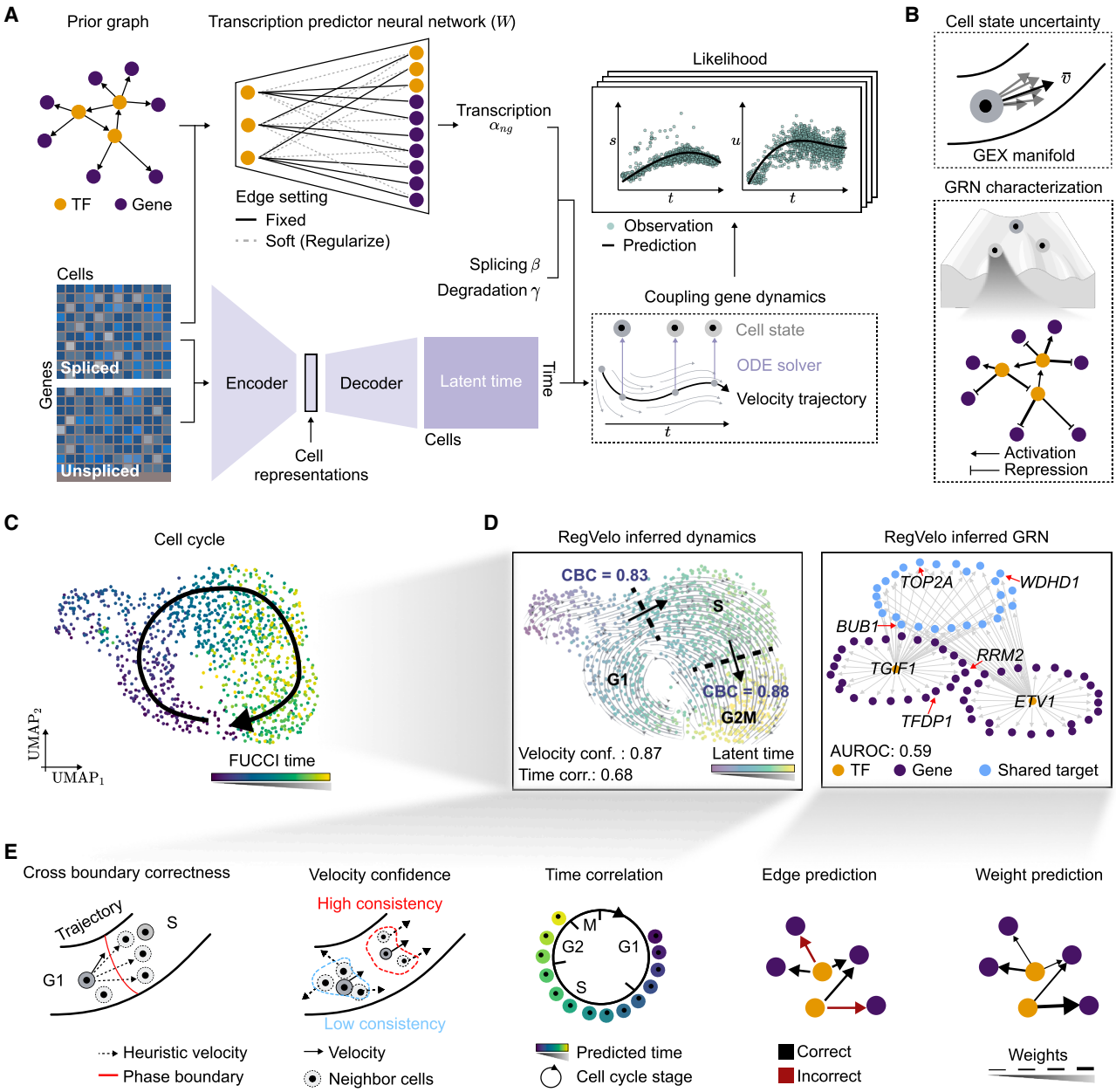


Figure 1. RegVelo models dynamics of cells and their underlying gene regulatory networks jointly

(A) RegVelo encodes the unspliced and spliced abundance of scRNA-seq data into the cell representation through a neural network and feeds the cell representation into a decoder neural network, which outputs a cell-gene-specific latent time. Simultaneously, a prior gene regulatory graph curated from orthogonal data sources, such as ATAC-seq, multiome measurements, or databases, guides a transcription predictor neural network. During training, this network refines and learns gene regulatory relationships, encoding each cell's (n) and gene's (g) transcription rate (α_{ng}). Combined with the additionally estimated splicing and degradation rates β and γ, respectively, the model defines coupled gene dynamics and employs a parallel, high-dimensional ODE solver to estimate and predict dynamic gene expression. The model uses stochastic variational inference[38] to optimize all network and kinetic parameters simultaneously.

(B) As a Bayesian generative model, the variational inference formulation quantifies the uncertainty of velocity estimation[14] (top), and coupling gene dynamics with gene regulation allows us to characterize the underlying gene regulatory network (GRN) of the phenotypic space (bottom).

(C) UMAP representation of a cell cycle dataset of 1,146 U2OS-FUCCI cells,[39] colored by the FUCCI-derived cell-cycle score; the black arrow indicates the ground-truth transition direction.

(D) The UMAP representation colored by the RegVelo-inferred latent time and overlaid with the corresponding velocity stream and inference metrics (left; CBC, cross-boundary correctness; conf, confidence; corr, correlation). We indicate the top two regulators (orange dots) and cell cycle-related genes (red arrows) in the fitted GRN (right).

(legend continued on next page)

solver, as this system does not possess an analytic solution; compared with previous approaches, we solve all gene dynamics at once instead of sequentially for each gene independently of all others. The forward simulation of the ODE solver allows computing the likelihood function encompassing all neural network and kinetic parameters (Figure 1A). We assume that the predicted spliced and unspliced abundances are the expected value of the Gaussian likelihood of the observed dataset and use gradient-based optimization to update all parameters (STAR Methods). After optimization, we define cell-gene-specific velocities as splicing velocities based on the estimated splicing and degradation rates and predicted spliced and unspliced abundance. Overall, RegVelo allows sampling predicted readouts and velocities from the learned posterior distribution.

RegVelo outputs parameter estimates including the inferred GRN, latent time, and the posterior velocity distribution, allowing us to quantify intrinsic and extrinsic cell uncertainty (Figure 1B): intrinsic uncertainty captures cell state change uncertainty along the phenotypic manifold approximated by the first-order direction, while extrinsic uncertainty reflects uncertainty in future cell states.[14] RegVelo generalizes our previous model veloVI by explicitly incorporating a GRN, thereby offering a more realistic model of Waddington's epigenetic landscape (Figure 1B). As such, RegVelo models the dynamics of a system of gene regulatory interactions that impose constraints and drive cell development, rather than modeling gene splicing in isolation[13] or mapping cells across different time points via expression similarity.[40] Moreover, using different gene regulatory systems, RegVelo can generate velocity samples from the respective posterior distributions. By comparing velocity vector fields of perturbed regulatory systems, RegVelo directly links gene regulation to cellular dynamics.

Benchmarking dynamic inference across contexts of varying lineage complexity

Inference of transcriptional dynamics is a challenging task and difficult to evaluate, as ground truth kinetic rates and state changes usually do not exist at scale. We first evaluated model performance on a six-gene simulated GRN comprising three toggle switches in a symmetrical hierarchy, previously used to model cell fate decisions during differentiation (Figure S1A). To benchmark the RegVelo model, veloVI, scVelo's EM model, and other GRN inference methods, we compared velocity, latent time, and GRN inference using correlation between estimates and ground truth, and the area under the receiver operating characteristic (AUROC) (STAR Methods). RegVelo showed improved performance compared with competing methods (two-sided Welch's t test, $p < 0.001$) (Figures S1B–S1D).

To generate a more realistic system, we employed the *dyngen*[41] framework that improves GRN simulation by incorporating protein modalities for a more complete representation of gene expression dynamics.

We first used two synthetic datasets with differing numbers of lineages to examine model behavior and robustness (Figure S1E). RegVelo includes the regularization parameters λ_1 and λ_2 to balance the influence of the prior GRN and sparsity of the learned GRN (STAR Methods). To assess their effect, we evaluated GRN inference across different levels of corrupted prior GRNs and regularization strengths (STAR Methods). For $\lambda_1 < 1$, RegVelo maintained consistent GRN prediction performance based on the AUROC metric, regardless of corruption levels (Figure S1F, right). Conversely, the model became more sensitive to the prior GRN for $\lambda_1 = 10$ (Figure S1F). To balance the influence of the prior and inference robustness, we set $\lambda_1 = 1$ by default. The effect of λ_2 was less pronounced, with GRN inference remaining generally stable for $\lambda_2 < 1$, though optimal values may vary across datasets (Figure S1G). Consequently, we set $\lambda_2 = 0$ by default. Using these defaults, RegVelo maintained consistent identifiability and performance in GRN, velocity, and latent time inference when evaluated against simulated ground truth, indicating robustness across the tested range of prior GRN corruption (Figures S1H and S1I).

We further benchmarked RegVelo for GRN inference using mouse hematopoietic stem and progenitor cell (mHSPC) data.[42,43] RegVelo achieved performance comparable with other methods, with TFvelo and CellOracle maintaining modest advantages in AUROC (by a 6.6% margin) and early precision rate (by a 5.4% margin), respectively (STAR Methods; Figure S1J). To further calibrate GRN inference, we extended our uncertainty measure, originally developed for velocity, to each GRN edge. Examination of these uncertainty estimates on both simulation datasets shows true edges exhibit significantly lower uncertainty values than false edges ($p < 1 \times 1^{-10}$) (Figure S1K). This indicates that the uncertainty estimates may reflect the confidence of the model in its predicted interactions. To assess whether the uncertainty estimates indicate prediction confidence in real data, we binned RegVelo-predicted interactions in the mHSPC dataset according to their uncertainty values and computed prediction accuracy within each bin. As expected, both area under the precision-recall curve (AUPRC) and AUROC decreased with increasing uncertainty, suggesting that the uncertainty estimates are indicative of prediction reliability (Figure S1L).

To test RegVelo's robustness across lineage complexity, we generated datasets with 1, 3, 7, and 9 terminal states (Figure S2A) and evaluated model performance in terms of velocity and GRN prediction. Velocity predictions between RegVelo and competing methods were comparable at low complexity (single lineage), but significantly better for RegVelo at high complexity (multiple lineages; $p < 0.001$). RegVelo generally outperformed scVelo and veloVI for time inference, and achieved comparable or higher AUROC scores than TFvelo and other

(E) Conceptualization of five metrics to evaluate RegVelo's inferred velocities, latent time, and GRN: the CBC quantifies how accurately inferred velocities recapitulate known cell state transitions, velocity consistency evaluates the coherence of velocities within neighboring cells, the latent time analysis assesses how well inferred latent times align with the ground truth, and comparing predicted gene regulatory edges to known relationships evaluates the GRN inference (TF, transcription factors; GEX, gene expression) (STAR Methods).
See also Figures S1, S2, and S3.

methods for GRN inference across complexity levels (Figure S2B).

We further validated RegVelo's inference robustness and consistency using the mouse neural crest dataset, in which we defined four complexity scales (Figure S2C; STAR Methods). As ground-truth velocities and times were unavailable, we used CellRank to quantify how faithfully methods (1) identified the known terminal states via the terminal state identification (TSI) score and (2) recapitulated ground-truth state transitions via cross-boundary correctness (CBC) (STAR Methods). RegVelo identified all terminal states except at the highest complexity (Figure S2D) and significantly outperformed scVelo and veloVI ($p < 0.001$) in TSI across all scales. CBC also improved with complexity relative to other methods (Figure S2E). Cell fate estimation remained highly consistent across scales—the resulting high median correlation (mesenchyme: 0.96 ± 0.02; sensory neuron 1: 0.94 ± 0.02; sensory neuron 2: 0.75 ± 0.15) showed that cell fate mapping was consistent with increasing complexity (Figure S2F). Additionally, RegVelo showed significantly higher and less variable cell fate estimation consistency compared with veloVI (RegVelo: mean = 0.873, std = 0.137; veloVI: mean = 0.697, std = 0.234) (Figure S2F).

Adding regulatory constraints enhances inference of cell-cycle dynamics

Next, we turned to a cell cycle dataset as a real-world setting in which we could robustly approximate the correct differentiation history and direction from fluorescence markers. Fluorescent ubiquitination-based cell-cycle indicator (FUCCI) U2OS cells undergo clear unidirectional transitions during the cell cycle, moving from the G1 to S to G2M phase, and yield a protein-derived cell-cycle score as a proximal measure of ground truth time (Figure 1C).[39] We investigated the inferred GRN next (Figure 1D; STAR Methods). Importantly, we did not provide a prior-informed GRN as an initial estimate to RegVelo but used a fully connected network where each gene interacts with every other gene, instead—a choice that allowed us to quantify RegVelo's ability to infer gene regulation in an unbiased manner. To preprocess nascent and mature mRNA, we followed our previous approaches for RNA velocity inference and provided our generative model with gene expression smoothed by first order over cellular neighborhoods (STAR Methods).

To evaluate whether RegVelo recovers the overall direction of the cell cycle, we compared the extrapolated future states of cells with prior knowledge on state transitions. Additionally, we confirmed that the inferred velocity field was consistent, which is desirable given the unidirectional nature of the underlying process. Specifically, we relied on the CBC score[17,29] that quantifies how faithfully a vector field aligns with known state transitions, *i.e.*, G1 to S, and S to G2M phase in our scenario (Figures 1D, 1E, and S3A). Indeed, RegVelo achieved a comparatively high score (CBC = 0.864 out of 1), high velocity consistency (0.873 out of 1), and its estimated latent time correlated positively with the FUCCI score used as ground truth (Spearman correlation = 0.683; possible values between -1 and 1). Turning to the RegVelo-inferred GRN, we focused on the most central part of the estimated network, identifying the main regulators TGIF1 and ETV1 among the most connected factors, and their top targets included cell cycle progression-related genes, such as *BUB1*,[44] *TFDP1*,[45] and *TOP2A*[46] (Figure 1D; STAR Methods). To assess whether RegVelo-inferred regulons align with experimental evidence, we first curated a list of putative ground truth targets of cell-cycle-related transcription factors (TFs)[47] from the chromatin immunoprecipitation (ChIP)-Atlas database,[48] which integrates large-scale ChIP-seq, ATAC-seq, and Bisulfite-seq experiments, offering epigenetic insight. Our method yielded relatively good agreement (ranking first among the six methods), quantified by edge prediction performance (median AUROC = 0.59) and the correlation between the predicted regulation coefficient and the ChIP-Atlas binding score (Figures 1D and S3G; STAR Methods).

Existing methods for inferring splicing dynamics do not consider regulatory information, and approaches that estimate GRNs ignore cellular dynamics. To assess how combining the two aspects with RegVelo compares to methods focusing on a single facet, we applied our previous inference schemes veloVI and scVelo's EM model, and the GRN inference models CellOracle,[26] GRNBoost,[49] and Pearson correlation. Whereas veloVI and scVelo assume constant rates, recent extensions, such as the non-constant transcription rate version of veloVI,[14] UniTVelo,[29] VeloVAE,[30] TFvelo,[32] and cell2fate,[31] try to model transcription as a time-dependent process. We, thus, also benchmarked against these velocity methods. Overall, RegVelo performed comparably or better than other velocity methods in terms of CBC (+1.5% over the top competitor), alongside general concordance in latent time estimation; among the methods whose inferred direction matched well with the expected direction, RegVelo also had a high velocity consistency score, showing that its velocity vector field aligned more closely with the ground-truth direction of the cell cycle (Figure S3A). We observed similar performance gains when applying the same benchmark to the RPE1 cell line (Figure S3A). Additionally, RegVelo yielded consistent and relatively more accurate estimates of both latent time and velocity compared with earlier non-GRN approaches, even after downsampling observed counts (STAR Methods; Figures S3B–S3E). To further assess whether proximal cells on the phenotypic manifold have similar latent times, we calculated the local consistency of inferred latent time for RegVelo compared with veloVI. RegVelo showed significantly better local latent time consistency than veloVI (two-sided Welch's t test, $p < 1 \times 10^{-10}$) (Figure S3F). In terms of GRN inference, RegVelo outperformed competing approaches on prediction of both edges and regulation coefficients (Figure S3G).

As for the improved performance of RegVelo, we next sought to confirm whether it was mainly driven by incorporating gene regulation rather than differences in global latent time definition, transcription rate estimation, or gene expression smoothing. We systematically ruled out alternative explanations and examined how alterations to the estimated network affected inferred dynamics. First, after re-estimating global cell latent time using graph-based ordering, RegVelo still showed good consistency with the FUCCI cell-cycle indicator on both cell lines (STAR Methods; Figure S3H). Second, RegVelo estimated transcription rates more accurately than methods, such as CellDancer, which

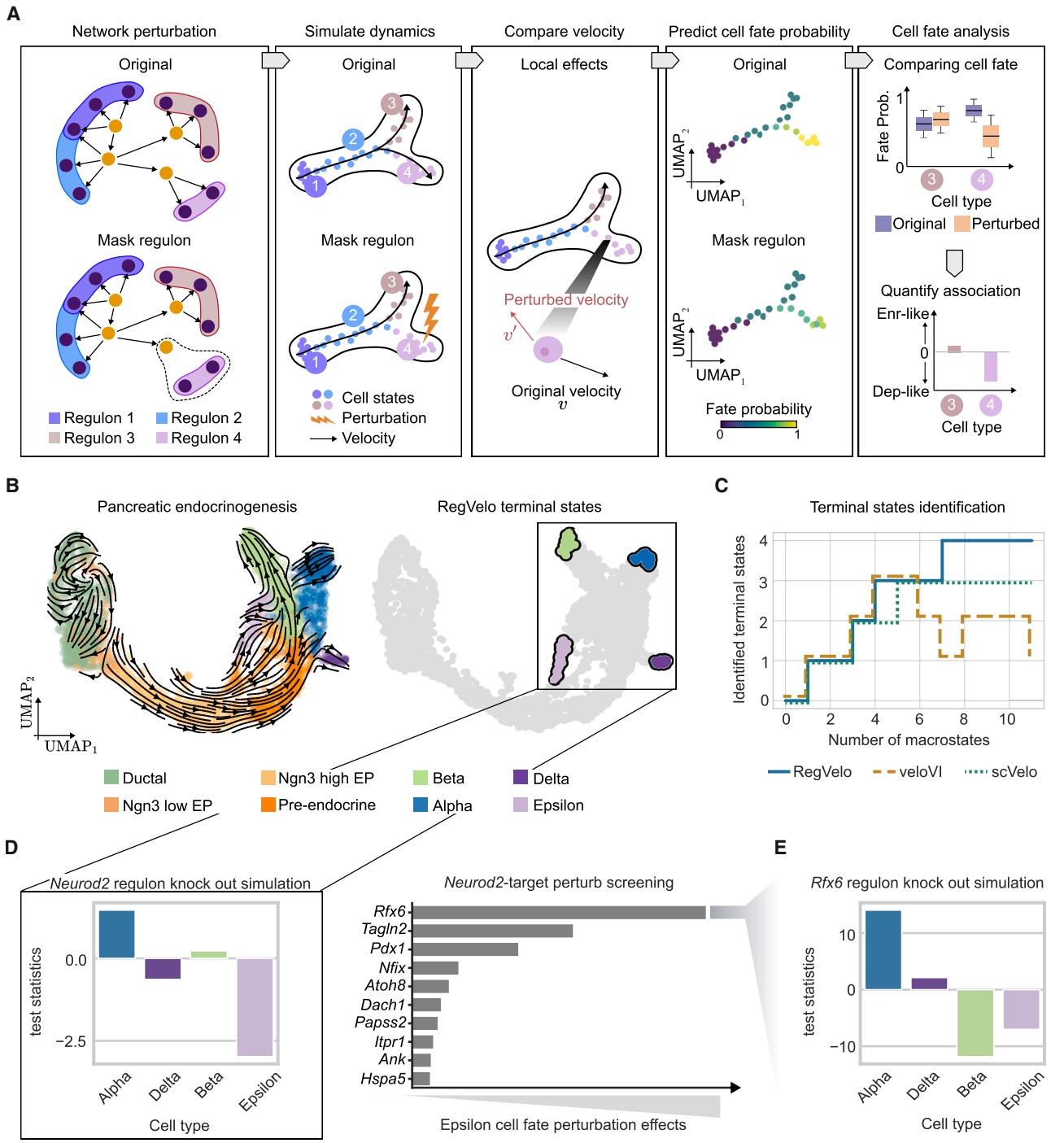


Figure 2. Perturbation simulation quantifies genetic regulation effects on cell fate decisions in pancreatic endocrinogenesis

(A) Perturbations are simulated by masking regulons and recalculating velocities. Resulting velocity shifts and CellRank[16,17]-derived fate changes quantify local and global perturbation effects, respectively.

(B) UMAP of 3,696 pancreatic endocrinogenesis cells colored by annotated cell type and the RegVelo-inferred velocity stream (left), and terminal states predicted by CellRank using RegVelo's velocities (right).

(C) The number of correctly identified terminal states compared with the number of specified macrostates indicates how faithfully a method recovers known biology with CellRank; different colors and line styles denote methods. Macrostates, defined by CellRank, are stable phenotypic regions unlikely to leave.[16]

(legend continued on next page)

ignore regulation (one-sided Welch's *t* test: RegVelo versus gene expression [GEX], $p = 6.972 \times 10^{-13}$; RegVelo versus CellDancer, $p = 4 \times 10^{-13}$) (STAR Methods; Figure S3I). Finally, RegVelo provided velocity estimates that are more consistent than those of veloVI across different neighborhood sizes (STAR Methods; Figure S3J). After excluding these factors, we assessed the influence of the GRN by predicting splicing dynamics without any regulatory relationships present. To mimic spurious gene-gene interactions, we also randomly reconnected genes in RegVelo's network (STAR Methods). As expected, both scenarios led to a lower CBC score and latent time correlation (Figure S3K), highlighting that incorporating gene regulation improves cellular dynamic inference.

These results on both simulation and real cell cycle data demonstrate RegVelo's ability to infer cellular dynamics and gene regulation more faithfully than competing methods.

Regulatory-constrained dynamic inference accurately recovers lineage formation in pancreatic endocrinogenesis

RegVelo is a generative model that couples cellular dynamics with regulatory networks. We can, thus, perform *in silico* counterfactual inference to test the cellular response upon unseen perturbations of a TF: for a trained RegVelo model, we ignore regulatory effects of the TF by removing all its downstream targets from the GRN, *i.e.*, depleting the regulon, and generate the perturbed velocity vector field (Figure 2A; STAR Methods). The dissimilarity between the original and perturbed cell velocities—the perturbation effect score—reflects the local changes on each cell induced by perturbations; we quantify this score with cosine dissimilarity (STAR Methods).

RNA velocity describes a high-dimensional vector field representing cellular change along the phenotypic manifold, but lacks interpretability and quantifiable measures of the long-term cell behavior. We recently proposed CellRank[16,17] to bridge this gap by leveraging gene expression and an estimated vector field to model cell state transitions through Markov chains and infer terminal cell states. For each terminal state, CellRank calculates the probability of a cell transitioning to this state—the fate probability—that allows us to predict the cell's future state. By combining RegVelo's generative model with CellRank, we connect gene regulation with both local cell dynamics and long-term cell fate decisions, and how they change upon *in silico* perturbations. For perturbation analyses, we compare CellRank's prediction of cell fate probabilities for the original and perturbed vector fields to quantify perturbation effects on terminal states (Figure 2A).

To highlight how our model elucidates cellular dynamics by incorporating gene regulation, we applied RegVelo to a pancreatic endocrine development scRNA-seq dataset.[50] To guide RegVelo's inference of the cellular dynamics, we estimated a prior GRN by scanning for motifs within candidate cis-regulatory elements (CREs) and integrating gene expression and peak accessibility with Pando[25] (STAR Methods). For GRN construction, we relied on a multiome dataset of paired ATAC and gene expression of the same system at embryonic days 14.5 and 15.5.[51]

Passing RegVelo-inferred velocity field to CellRank, we delineated the cycling population of ductal cells and predicted the four terminal states, namely alpha, beta, delta, and epsilon cells[50] (Figures 2B and 2C; STAR Methods). We specifically explored how these lineages form, predicting that some epsilon cells become alpha cell progenitors, consistent with previous reports[50–52] (Figure S4A). We identified two states based on fate probabilities that divide the epsilon cell population: state A showed higher expression of alpha cell-specific genes (*Pou6f2*,[53] *Irx1*,[54] and *Smarca1*[55]), and state B expressed *Neurog3*, consistent with Neurogenin3 activity-dependent epsilon cell genesis[56] (Figure S4B; STAR Methods). As previously, deleting or shuffling the learned GRN led to the loss of identified terminal states (Figure S4C, left), reaffirming the value of combining transcriptional dynamics with gene regulation. Moreover, previous methods, veloVI and scVelo, not integrating the GRN component, achieved a lower TSI score (Figure S4C, right; STAR Methods). To estimate minimal data requirements for velocity inference with RegVelo, we evaluated model performance across cell numbers and read depths: our results show that RegVelo recovered all terminal states with moderately high velocity estimation consistency (>0.7) when using >1,000 cells and retaining 75% of the original read counts (Figure S4D).

RegVelo disentangles ductal and epsilon cell dynamics in pancreatic endocrinogenesis through *in silico* perturbations

Following our analysis of cell maturation, we investigated how to interpret the role of gene regulation in the cycling population of ductal cells with RegVelo. By correlating gene expression with a cell cycling phase score inferred from prior knowledge on cycling markers (Figure S4E), we found that *E2f1* was the highest-ranked gene (Figure S4E; Table S1; STAR Methods). To predict the cell set related to E2f1 functions, we removed the E2f1 regulon from the GRN and used cosine dissimilarity to compare the vector fields before and after perturbation. We thereby identified a ductal cell subpopulation exhibiting a large change that overlaps mostly with the S-phase and, thus, the stage likely most affected by E2f1[57] (Fisher's exact test, $p < 1 \times 10^{-10}$) (Figure S4F, left). Importantly, this perturbation effect score does not simply reflect *E2f1* expression, as Ngn3-high EP cells express *E2f1* at a higher level than ductal cells (Figure S4F, right). Moreover, gene set enrichment analysis (GSEA) and gene ontology (GO) enrichment analyses suggest that the predicted E2f1 downstream targets are mainly enriched for E2F target and cell cycle phase transition-related pathway (Figures S4G and S4H), consistent with previous findings that E2f1 acts as a cell cycle regulator.[58] We also identified targets and downstream pathways of other top regulators, consistent with their known functions (Figure S4H; Table S1).[59,60] These findings suggest

(D) *Neurod2* regulon perturbation effects on cell fates (left) and the corresponding ten *Neurod2*-targets affected most in the epsilon lineage (right), assessed by *t* test statistics comparing perturbed and original fate probabilities.
(E) *Rfx6* regulon perturbation effects on cell fates.
See also Figure S4.

that the RegVelo-predicted perturbation effect score provides a useful readout for pinpointing cell states and pathways that are functionally influenced by specific regulators.

We explored the E2f1 regulon network further to investigate whether RegVelo could recover known E2f1 downstream targets that contributed to cell cycle regulation. As a first step, we validated that the inferred GRN was identifiable and robust under different runs of RegVelo (Pearson correlation > 0.95) (Figure S4I, left). We then used the prior GRN as a baseline to compare the performance of RegVelo-inferred ductal cell GRN when predicting E2f1 targets (STAR Methods). Using E2f1 ChIP-seq results from the ChIP-Atlas database as ground truth, RegVelo-inferred GRN showed significant improvement over the prior GRN in predicting E2f1 targets (two-sided Welch's t test, $p < 1 \times 10^{-10}$) (Figure S4I, right). Focusing on the top 20 predicted targets, we recovered eight cycling-related genes, including *Cdt1*,[61] *Uhrf1*,[62] *Stmn1*,[63] *Id2*,[64] *Mcm6*, and *Mcm7*.[65] The Pando-based prior GRN, however, did not include key cell cycling modulators such as *Hells* and *Cenpk* as E2f1 targets, whereas the RegVelo-inferred GRN predicted them among the top downstream effectors (Figure S4J). Recent ChIP experimental results support E2F1 binding to both target promoter regions.[66,67] In summary, RegVelo accurately infers cellular dynamics and gene regulation, enabling the prediction of perturbation effects in this pancreatic endocrine dataset.

To interpret TF perturbation effects on cell fate decisions with RegVelo, we first ensured that RegVelo gives stable cell fate probability predictions and avoids stochastic effects, for example, from random parameter initialization. To verify robustness, we measured the correlation of velocities or cell fate probabilities between pairs of estimates from different model runs (STAR Methods). RegVelo scored a high mean correlation and low variance, implying its high identifiability on cell fate prediction (Figure S4K). Next, we quantified the perturbation effect of each TF regulon and validated its stability over multiple model fits for each lineage (Figure S4L; STAR Methods). We reasoned that TFs with the strongest depletion effects (highest perturbation simulation ranks) represent putative lineage drivers. Focusing on these top-ranked TFs, RegVelo successfully recovered known drivers across four lineages, including *Smarca1*,[55] *Arx*[68] in the alpha, *Pdx1*,[69] *Mnx1*[70] in the beta, and *Hhex*[71] in the delta lineage (Figure S4M), and achieved an overall high ranking for the curated driver, evaluated via the AUROC (AUROC = 0.95) (STAR Methods). While these regulators may also function in other contexts,[72] their ranking aligns with their known activity in the respective lineages.

We further focused on the poorly understood process of epsilon cell differentiation. Our *in silico* screening predicted *Neurod2* as one of the putative drivers of epsilon cell differentiation (Figure S4M). Previous studies have found that pancreatic cells transiently expressed *Neurod2* during pancreatic endocrinogenesis[73] and suggested the role of *Neurod2* in epsilon cell formation.[51] To uncover the potential mechanisms of *Neurod2* in orchestrating epsilon cell differentiation, we employed RegVelo's generic circuit screening feature to perturb each downstream target of Neurod2 and predict the outcomes of these perturbations on epsilon differentiation. According to these results, perturbation of the Neurod2-*Rfx6* regulation contributes the most to the maturation of epsilon cells (Figure 2D), an observation consistent with recent studies highlighting the importance of this regulation.[74] Perturbing the Rfx6 regulon suggests its involvement in beta and epsilon cell differentiation, an observation supported by previous experimental studies[75] (Figure 2E). This is consistent with its known pleiotropic role in endocrine lineage specification, rather than being exclusive to a single fate.

RegVelo recovers regulatory motifs governing cell fate decisions in hematopoietic differentiation

Previous RNA velocity models, which assume successfully captured general dynamics in pancreatic endocrine development, failed on hematopoietic datasets,[13,23] because their assumption of constant transcription rates does not hold during the dynamic regulatory processes of hematopoiesis.[15,17,76] As RegVelo does not model transcription as constant, we hypothesized that it could inform on gene regulatory processes during hematopoiesis. Furthermore, hematopoiesis is a well-studied system, offering an opportunity to further validate our TF perturbation predictions based on the literature.

To apply RegVelo to a hematopoiesis dataset including five distinct lineages[23] (Figure 3A), we first curated a prior GRN, relying on two orthogonal resources to mitigate the effect of incompleteness: as a first estimate, we used a publicly available GRN inferred from single-nucleus ATAC and gene expression measurements of bone marrow mononuclear cells with *Dictys*.[27] We then extended this representation with an additional GRN curated from human hematopoiesis ChIP-seq datasets from the UniBind database[77] to obtain our final prior GRN. After feeding this GRN into our standard RegVelo workflow and passing the estimated vector field to CellRank, our analysis recovered all five terminal states and recapitulated known state transitions faithfully, consistent with the known lineage commitment and hierarchy of hematopoiesis[23] (Figures 3B and S5A; STAR Methods). We also observed a change in RegVelo's intrinsic estimation uncertainty from high in progenitor cells (MEP and GMP) to low in terminal states (Figure 3C), consistent with the fact that mature cells tend to maintain a more stable differentiated phenotype compared with their progenitor cells under homeostatic conditions.[78] Having validated RegVelo's ability to model major cellular changes along the phenotypic manifold in the analyzed dataset, we next analyzed *in silico* perturbation predictions and the recovered GRN.

Comparing cell-specific fate probability changes provides an intuitive way to evaluate perturbation effects. Deletion of the GATA1 regulon, a master regulator of the megakaryocyte-erythroid progenitor (MEP) lineage,[83] was predicted by RegVelo to result in depletion-like effects in erythrocytes and enrichment-like effects in monocytes. Similarly, deletion of SPI1, the master regulator of the granulocyte-monocyte progenitor (GMP) lineage,[84] was predicted to result in depletion-like effects in monocyte and neutrophil branches and enrichment-like effects in erythrocytes (Figure 3D). We developed the "depletion likelihood," a summary statistic ranging from 0 to 1 that quantifies changes in cell fate probabilities before and after perturbation. This metric reflects the global influence of a TF on lineage identity and enables causal ranking of putative drivers based

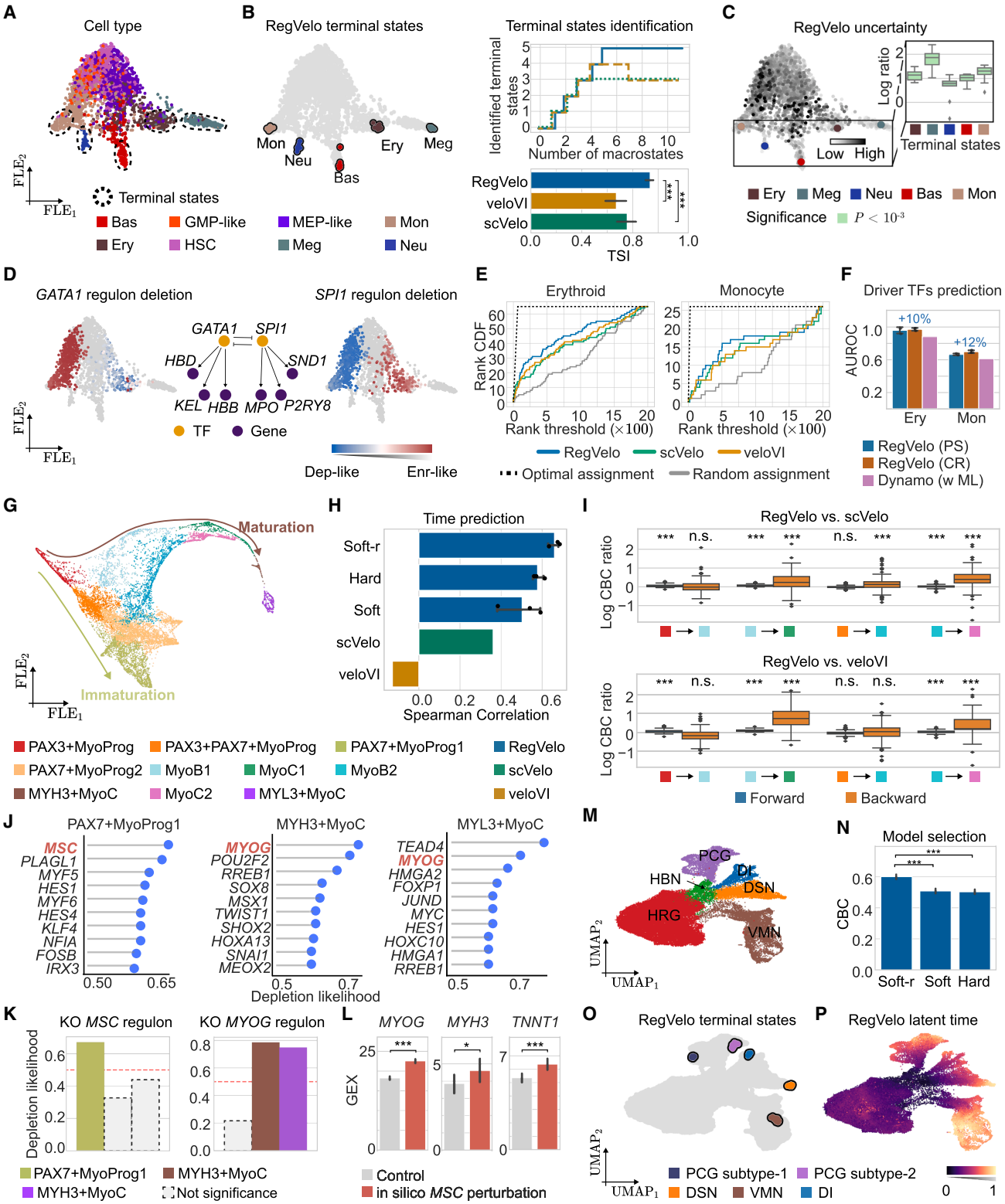


Figure 3. Inferring regulatory dynamics during human hematopoiesis differentiation, embryonic myogenesis, and hindbrain development

(A) Force-directed layout embedding (FLE) of 1,947 human hematopoietic stem and progenitor cells (HSPCs), colored according to original cell type annotation (HSC, hematopoietic stem cell; Meg, megakaryocyte; Ery, erythrocyte; Bas, basophil; Neu, neutrophil; Mon, monocyte; MEP-like, megakaryocyte and

(legend continued on next page)

on their tendency to deplete a given lineage when perturbed. We further evaluated the significance of depletion using a Wilcoxon rank-sum test (Figure S5B; STAR Methods). Beyond *GATA1* and *SPI1*, our analysis identified additional putative drivers with high depletion likelihood across hematopoietic lineages, including *GATA2* in basophils (critical for basophil differentiation[85]), *MEF2C*, *NFIB*, and *GFI1B* in megakaryopoiesis,[86,87] and *ETV6* in monocyte differentiation[88] (Figure S5C; Table S2). RegVelo's GRN recovers the well-known SPI1-GATA1 toggle switch motif (Figure 3D), even when replacing the prior GRN with a full connect GRN (Figure S5D),[89,90] and predicted GATA1 downstream effectors with reported roles in regulating hemoglobin gene expression, consistent with experimental findings[91] (Figure 3D).

To assess RegVelo's ability to capture specific cellular transitions, we benchmarked its inferred cellular dynamics against those from veloVI and scVelo (STAR Methods). Following the CellRank 2 workflow,[17] we used the log2 ratio of CBC scores as a quantitative metric to evaluate velocity inference, and additionally introduced the backward CBC as a complementary measure to the original CBC metric[17] (Figure S5E). We observed that RegVelo achieved higher CBC scores than scVelo in 5 out of 6 cell state transitions. Although veloVI showed an overall comparable CBC score of forward cell state transition to RegVelo, it resulted in lower CBC scores of backward cell state transition, indicating a potential incorrect backflow in the veloVI vector field (Figure S5F; STAR Methods). To assess the effect of model regularization, we compared the results of unregularized and Jacobian-L1 (J-L1) regularized RegVelo estimates. The regularization led to improved cell state transition predictions, indicating that increased model sparsity improves dynamic inference, consistent with previous studies on NeuralODE models[92] (Figure S5F).

Using cell-cell transition probabilities computed with CellRank to infer terminal states, RegVelo was the only method among those benchmarked to recover all five terminal states (TSI = 0.95), and achieved significantly higher TSI score than other methods (one-sided Welch's *t* test: RegVelo versus veloVI, $p = 1.59 \times 10^{-5}$; RegVelo versus scVelo: $p = 2.23 \times 10^{-5}$) (Figures 3A and 3B; STAR Methods). As an extension of veloVI, we specifically compared RegVelo inference results to veloVI's predictions: first, we examined how robust each method's velocity or latent time estimations were under different runs. Overall, RegVelo's estimates were more stable and associated better across multiple runs compared with veloVI (Figure S5G). Compared with RegVelo's low intrinsic estimation uncertainty in mature cell types compared with progenitor cells, veloVI's inference resulted in high intrinsic uncertainty across all cell types (Figure S5H). By assessing lineage-specific rankings of predicted cell fate probabilities against known hematopoietic trajectories,[23,93] RegVelo recovered three of four expected biological lineage orderings (Erythroid, Basophil, and Monocyte), whereas veloVI and scVelo each captured

erythrocyte progenitor-like; and GMP-like, granulocyte and monocyte progenitor-like). Terminal states are defined based on the well-established hematopoietic hierarchy.[23]

(B) RegVelo with CellRank recovers five terminal states (left). The number of correctly identified terminal states compared with the number of specified macrostates highlights that RegVelo performs optimally (right).

(C) FLE of HSPCs colored by RegVelo-inferred uncertainty and terminal states (top; STAR Methods). Log-ratio comparison of progenitor versus terminal uncertainty (bottom; STAR Methods). Box plots indicate the median (center line), interquartile range (hinges), and 1.5× interquartile range (whiskers) ($n = 30$ cells; one-sided Welch's *t* tests).

(D) FLE colored by RegVelo's cell-specific perturbation effect in the erythroid and monocyte lineage after deleting the GATA1 (left) or SPI1 (right) regulon, and the regulatory subnetwork surrounding the predicted toggle switch between GATA1 and SPI1 (middle).

(E) Ranking of known lineage-associated genes for the erythroid and monocyte lineages using different methods with the CellRank workflow (STAR Methods).

(F) AUROC for lineage driver prediction (PS, perturbation simulation; CR, CellRank; w ML, with metabolic labeling; $N = 3$ trained models each). We report the improvement in the average AUROC score of RegVelo (CR) compared with Dynamo (W/ ML) since the performance of RegVelo (PS) and RegVelo (CR) is similar.

(G) FLE of 12,207 human limb cells associated with myogenesis,[79] colored by original cell type annotation[79] (MyoB, myoblast; MyoC, myocyte; MyoProg, myogenic progenitor).

(H) Comparison of three candidate RegVelo models ($n = 3$ trained models), scVelo and veloVI on latent time prediction. Soft-r model was selected based on best agreement with known differentiation stages. Error bars: 95% confidence intervals (CI). (Soft-r, soft constraint with Jacobian regularization; Soft, soft constraint; Hard, hard constraint).

(I) Log-ratio of CBC score comparing RegVelo, scVelo (top), and veloVI (bottom) across defined myogenic transitions (STAR Methods). One-sided Welch's *t* tests were used to assess significance. Box plots display median (center line), interquartile range (hinges), and 1.5× IQR (whiskers). Top (versus scVelo): PAX3+MyoProg → MyoB1 (forward: $n = 121, p < 1 \times 10^{-10}$; backward: $n = 185, p = 0.20$); MyoB1 → MyoC1 (forward: $n = 77, p < 1 \times 10^{-10}$; backward: $n = 89, p = 4.012 \times 10^{-6}$); PAX3+PAX7+MyoProg → MyoB2 (forward: $n = 399, p = 0.98$; backward: $n = 335, p < 1 \times 10^{-10}$); MyoB2 → MyoC2 (forward: $n = 283, p = 6.12 \times 10^{-6}$; backward: $n = 217, p < 1 \times 10^{-10}$). Bottom (versus veloVI): PAX3+MyoProg → MyoB1 (forward: $n = 121, p = 1.09 \times 10^{-8}$; backward: $n = 185, p = 0.99$); MyoB1 → MyoC1 (forward: $n = 77, p < 1 \times 10^{-10}$; backward: $n = 89, p < 1 \times 10^{-10}$); PAX3+PAX7+MyoProg → MyoB2 (forward: $n = 399, p = 0.99$; backward: $n = 335, p = 0.552$); MyoB2 → MyoC2 (forward: $n = 283, p < 1 \times 10^{-10}$; backward: $n = 217, p < 1 \times 10^{-10}$).

(J) Putative driver TFs identified via RegVelo's *in silico* simulation, highlighting known drivers from the original study[79] in red.

(K) RegVelo-predicted depletion likelihood of known drivers (One-sided Wilcoxon rank sum test).

(L) RegVelo predicts upregulation of maturation markers (*MYOG*, *MYH3*, *TNNT1*) following MSC depletion. Error bars: 95% CI. One-sided Welch's *t* test for significance (*MYOG*, $p < 1 \times 10^{-10}$; *MYH3*, $p = 0.039$; *TNNT1*, $p = 7.02 \times 10^{-5}$).

(M) UMAP of 49,469 human hindbrain cells,[80] colored by cell types (HRG, hindbrain radial glia; HBN, Hindbrain neuroblast; PCG, Pontocerebellar glutamatergic; DI, dorsal interneurons; DSN, dorsal sensory neuroblasts; VMN, visceral motor neuron)[24] (STAR Methods).

(N) Comparing three candidate RegVelo models on the CBC score. ($n = 3$ trained models; one-sided Welch's *t* test, Soft-r versus hard: $n = 1{,}275, p < 1 \times 10^{-10}$; Soft-r versus soft: $p < 1 \times 10^{-10}$; left). The Soft-r model showed the best performance.

(O) CellRank predicted terminal states with RegVelo-inferred velocity.

(P) UMAP embeddings colored by inferred latent time from RegVelo.[81,82]

See also Figures S5 and S6.

one of these orderings (Megakaryocyte for veloVI and Basophil for scVelo) (Figure S5I).

In addition to the cell cycle and pancreatic endocrine systems, we performed a GRN ablation study in hematopoietic systems to assess how regulatory priors and the removal of GRN affect dynamic inference. We found that eliminating informative regulatory priors or the GRN impaired TSI (two-sided paired *t* test: $P = 0.0284$ for w prior versus w/o prior, $P = 0.0042$ for w prior versus no GRN) (Figure S6A) and compromised the prediction of cell state transitions (Figure S6B). Building on previous findings that inaccurate velocity inference for genes, such as *PF4*, arises from complex induction dynamics overlooked by conventional models,[23] we examined additional lineage markers, *MPO* for monocytes and *HLA-DPB1* for neutrophils, and observed that only RegVelo, with the GRN incorporated, correctly captured their induction patterns (Figure S6C). Further analysis of *SPI1* and *GATA1* revealed their regulatory influence along the fate continuum of erythroid and monocytic lineages, illustrating their activation dynamics during hematopoietic differentiation (Figure S6D).

Estimating fate probabilities with CellRank based on RegVelo's vector field allows prioritization of lineage-associated genes by correlating fate probabilities with gene expression.[17] We employed this CellRank 2[17] strategy to assess whether incorporating gene regulation into the splicing model improves identification of known lineage-priming genes compared with classical models that omit regulatory information (STAR Methods). To evaluate performance, we first curated a list of known lineage markers for comparison with predictions from each method (STAR Methods). We then ranked genes using different velocity models and assessed how well the ranked genes recovered known markers. We focused on monocyte and erythroid lineages, which represent two distinct branches with well-characterized markers. Compared with the competing methods, using RegVelo's velocity estimates as input for CellRank resulted in higher-ranking of marker genes for both lineages (Figure 3E; Table S2). Notably, regardless of whether ranking was based on perturbation simulations or CellRank with RegVelo-estimated velocities, RegVelo outperformed dynamo in predicting driver TFs for both lineages, even though dynamo had access to metabolic labeling data (Figure 3F; STAR Methods; Table S2).

The analysis of the hematopoietic system with RegVelo highlighted our model's capability to recover lineage hierarchy and regulatory connections.

RegVelo recovers trajectories and canonical drivers of human myogenesis via side-information-guided model selection

Inferring dynamic processes requires careful parameter tuning[5]; however, selecting a model that more accurately reflects biologically relevant dynamics, such as real time, developmental stages, and known cell state transitions, is essential for identifying an appropriate model.[17,94] To address this, we introduced the *ModelComparison* class to systematically evaluate RegVelo candidate models based on their ability to predict key side information (STAR Methods).

To demonstrate ease of use, we applied this framework to human embryonic and fetal limb myogenesis systems, which capture a bifurcation between progenitor maintenance and muscle maturation (Figure 3G).[79] Using the *ModelComparison* class and the correlation between estimated latent time and embryonic stage as guidance, we ranked the RegVelo model with soft constraint with the Jacobian regularization (Soft-r) highest (Figure 3H; STAR Methods). The selected model exhibited comparable or significantly better prediction of state transitions than both scVelo and veloVI, especially for the MyoB1-to-MyoC1 and MyoB2-to-MyoC2 transition, as measured by forward and backward CBC (Figure 3I). We further screened TFs to evaluate their depletion effects on different lineages. RegVelo identified *MSC* as the top regulator of progenitor fate, and *MYOG* as the top driver of maturation fates (Figures 3J and 3K; Table S2). It also predicted upregulation of maturation genes such as *MYOG*, *MYH3*, and *TNNT1* following MSC regulon knockout, consistent with experimental observations[79] (Figure 3L). We further applied *ModelComparison* to the human embryonic hindbrain system[80] (Figure 3M). With cell state transition prediction as guidance (Figure 3N; STAR Methods), the selected RegVelo model recovered terminal states (Figure 3O) and produced latent time estimates that reflected the known differentiation hierarchy from HRG and HBN progenitors to the four specialized neuronal types[95] (Figure 3P). These results demonstrate that the *ModelComparison* class effectively supports RegVelo model selection in human developmental contexts.

Modeling regulatory dynamics of zebrafish neural crest development

Neural crest cells contribute diverse derivative cell types to functionally distinct tissues and organs, including bone, cartilage, and connective components of the craniofacial skeleton, most of the body's pigment cells, and elements of the peripheral nervous system. Deciphering the dynamic programs that control neural crest development may improve understanding of developmental disorders and neural-crest-derived cancers.[96–98] We next used RegVelo to combine ultra-deep single-cell transcriptomic readouts with GRN priors into a dynamic, actionable model of neural crest development. To decode complex regulatory dynamics of the neural crest, we employed Smart-seq3[99] and profiled 1,180 neural crest cells and their derivatives at seven time points, achieving a median coverage of 7,791 genes per cell (Figures 4A, 4B, S7A, and S7B). We then applied RegVelo with a prior SCENIC+-inferred GRN from a stage-matched single-cell multiome dataset[100] (Figure 4A; STAR Methods).

We first evaluated the extent to which the RegVelo model follows the established differentiation hierarchy during neural crest development. RegVelo recovered previously identified terminal states,[100] namely post-otic migratory neural crest (mNC hox34), non-hox facial mesenchymal, second pharyngeal arch (arch 2), and pigment cells (Figures 4C and 4D). Additionally, the RegVelo-inferred latent time positively correlated with actual developmental stages (Figure S7C), and placed the non-hox neural plate border (NPB nohox) cell state near the beginning of the developmental process[101] (Figures 4C and S7C). RegVelo recovered more terminal states than scVelo and VeloVI, supporting improved dynamic inference (Figure 4D).

To systematically identify putative lineage drivers *in silico*, we quantified perturbation effects on cell differentiation toward the

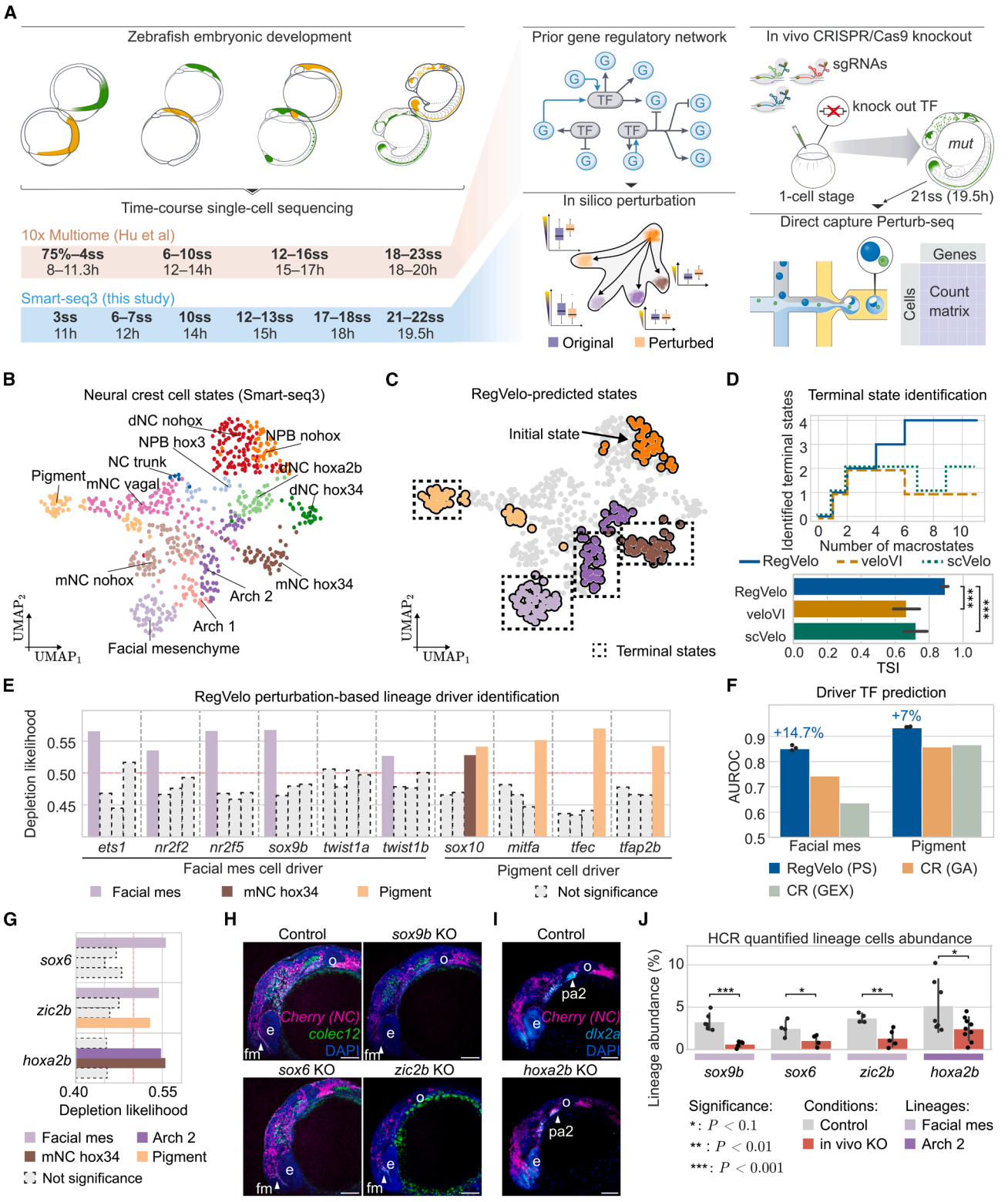


Figure 4. RegVelo uncovers key regulators of cell fate decisions during zebrafish neural crest development

(A) Schematic of the experimental design for time-resolved single-cell gene expression, multimodal assays, and *in vivo* Perturb-seq to capture neural crest cell development (ss, somite stage; h, hour post fertilization; mut, mutant).

(legend continued on next page)

four terminal states. RegVelo linked perturbation effects of known drivers to depletion of the corresponding lineages, including *mitfa*,[102] *sox10*[103] for the pigment lineage, and *sox9b*,[104] *nr2f2,*[105] *twist1b*[106] for the mesenchymal lineages (Figure 4E). While these factors are not exclusively specific to these lineages, their predicted effects are consistent with previously reported lineage roles in neural crest development.

As RegVelo simulates perturbed dynamics *de novo*, we identified lineage drivers expressed at the beginning of a trajectory but not in terminal states, whereas correlation-based methods tended to overlook these drivers (Figure S7D). For instance, NPB and delaminating neural crest cells (early states during NC formation) highly expressed the cranial mesenchymal lineage drivers, *ets1*,[107] *nr2f5*,[105] *sox9b*,[108] and *twist1b*[109] that were significantly downregulated in terminal-state cells ($p < 0.001$) (Figures S7D–S7F). RegVelo predicted these genes as mesenchymal lineage drivers, whereas correlation-based analyses with the classical CellRank workflow and *in silico* perturbation approach by dynamo[23] predicted them as putative drivers of mNC hox34 development (Figures S7G–S7I). Finally, our RegVelo-based workflow ranked these genes consistently higher than the correlation-based approach, even when we replaced gene expression with TF-regulon activity (Figure 4F), and used combinations of GRN inference,[24,25,28] TF-regulon activity quantification,[18,28,110] and trajectory inference approaches[7,13,17,51] (Wilcoxon rank-sum test, $p = 0.0247$, $n = 3$) (Figure S7J).

To explain how RegVelo identifies early-expressed drivers, we assessed its two key inputs: RNA splicing dynamics and a prior GRN. We first showed that only correlating spliced and unspliced RNA with lineage progression did not improve driver ranking over total RNA (Figure S7K). In contrast, early drivers of cranial mesenchymal fates (arch 2 and facial mesenchyme) exhibited higher regulon activity (see STAR Methods), consistent with improved driver ranking when correlating regulon activity, rather than gene expression, with lineage progression (Figures S7L and 4F, left). This suggests that the prior GRN linking TFs to downstream lineages is the contributor to RegVelo's early driver predictions. To directly assess this contribution, we trained RegVelo using a fixed binary or randomized prior GRN. The fixed binary prior GRN produced reasonable driver rankings (Figure S7M) (AUROC = 0.8), whereas randomizing edges substantially reduced performance (AUROC = 0.52 ± 0.13), confirming the importance of the prior's regulatory structure. We then examined the role of RegVelo-learned edge weights in driver ranking: whereas using fixed SCENIC+ weights yielded inferior rankings (AUROC = 0.65), allowing RegVelo to refine edge weights substantially improved its performance (Figure S7M) (AUROC = 0.91 ± 0.02). Furthermore, shuffling these learned weights significantly reduced performance (Figure S7M) (AUROC = 0.38 ± 0.09), underscoring the importance of data-driven weight refinement.

To further illustrate how the learned weighted GRN mechanistically connects early drivers to lineage fate, we perturbed nr2f5 and ets1 regulons using RegVelo's ODE simulator. We first identified the top regulatory links for both TFs, including facial mesenchyme-associated genes involved in skeletal development (*serpinh1b*,[111] *alcama*,[112] and *tgfbr3*[113]) and cranial neural crest cell migration (*tgfbr3*[114] and *nrp2a*[115,116]).

We next perturbed nr2f5 and ets1 regulons and used RegVelo's ODE simulator to predict the changes in target gene expression. With the learnable weighted GRN model, the simulated dynamics closely matched the observed data (median $R^2 = 0.92$) (Figure S7O) and predicted decreased target gene expression, mirroring the mesenchymal depletion observed in the CellRank-based perturbation analysis. In contrast, simulations based on a fixed binary GRN showed poorer dynamic fitting (Figures S7P and S7Q) and reduced perturbation effects, particularly for *ets1* (median $R^2 = 0.54$) (Figure S7P). We further observed that including more stages in the input data improved prediction of facial mesenchymal drivers and enhanced the learned edge weights for both TFs. This suggests that RegVelo leverages temporal dynamics to refine lineage-associated regulatory interactions (Figure S7R). Together, these results indicate that RegVelo's capacity to link early TF drivers to lineage identities depends not only on the regulatory context provided by the prior GRN, but also on its ability to refine these interactions through the learning of edge weights that incorporate temporal information.

To experimentally test the functional requirement of RegVelo-predicted drivers for lineage specification, we performed genetic knockouts of predicted drivers for the facial mesenchymal lineage (*sox6*, *sox9b*, and *zic2b*) and the arch 2 lineage (*hoxa2b*). We analyzed the effects using multiplexed hybridization chain reaction fluorescence *in situ* hybridization (HCR-FISH) (Figures 4E and 4G). By directly quantifying lineage cell abundance in

(B and C) UMAP embedding of 697 neural crest cells from the Smart-seq3 dataset, colored by cell states (B) and known initial and terminal states (C) (NPB, neural plate border; dNC, delaminating neural crest; mNC, migratory neural crest; mes, mesenchyme; arch 2, second pharyngeal arch).

(D) We plot the number of identified terminal states against the number of macrostates specified (top) and the TSI score using RegVelo (blue), veloVI (orange), and scVelo (green) ($n = 20$; two-sided Welch's t test, $p < 0.001$). Error bars: 95% CI. Line styles and colors indicate different methods.

(E) RegVelo-predicted depletion likelihood of known lineage drivers in facial mesenchyme or pigment lineages (one-sided Wilcoxon rank-sum test).

(F) AUROC of lineage driver prediction for facial mesenchymal/pigment lineage specification based on RegVelo's perturbation simulation (PS), or CellRank using gene expression (GEX) or gene activity (GA) of TFs.

(G) RegVelo-predicted depletion likelihood of putative drivers in facial mesenchyme (*sox6* and *zic2b*) or arch 2 (*hoxa2b*) lineages (one-sided Wilcoxon rank-sum test).

(H and I) Representative confocal microscopy images of *colec12* (H; facial mesenchymal marker), *dlx2a* (I; arch 2 marker), and *Cherry* (neural crest marker) expression in control (Ctrl) and *sox9b, sox6, zic2b*, and *hoxa2b* crispant zebrafish embryos at the 21-somite stage. E, eye; fm, facial mesenchymal; o, otic; pa2, arch 2.

(J) Quantification of the abundance of all facial mesenchymes (*sox9b*, *sox6*, and *zic2b*) and arch 2 (*hoxa2b*) from confocal images. (One-sided Welch's t test; sox9b: $N_{Ctrl} = 6, N_{KO} = 5, P = 0.0003$; sox6: $N_{Ctrl} = 4, N_{KO} = 4, p = 0.049$; zic2b: $N_{Ctrl} = 4, N_{KO} = 5, p = 0.0038$; hoxa2b: $N_{Ctrl} = 7, N_{KO} = 9, p = 0.033$; Error bars: 95% CI).

See also Figures S7 and S8.

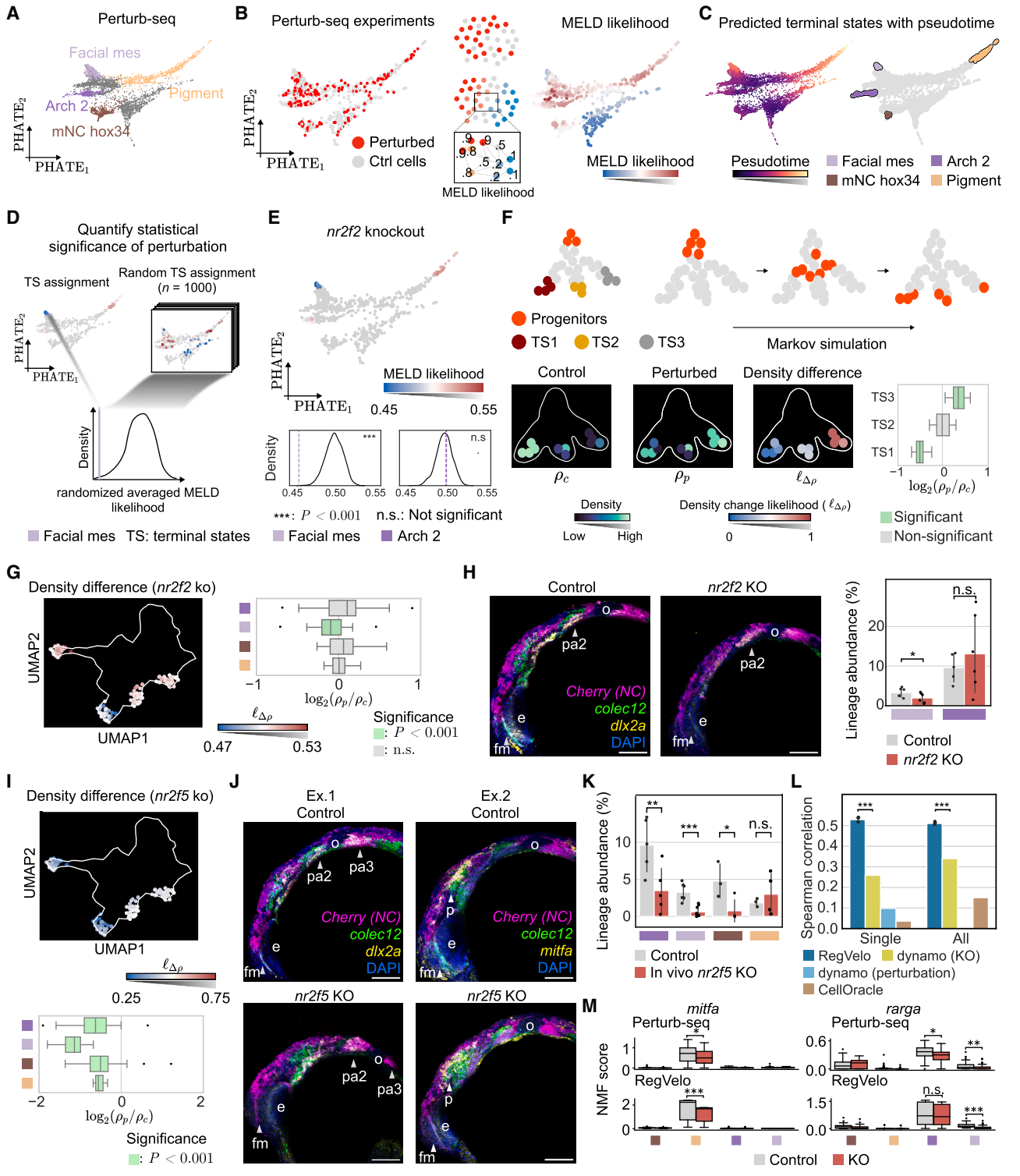


Figure 5. RegVelo predicts perturbation effects of key regulators during zebrafish neural crest development

(A) PHATE embedding of 39,453 neural crest development cells from the combined Perturb-seq datasets, color indicates terminal lineage.

(B) PHATE embedding of 834 neural crest development cells from the *nr2f2* knockout (KO) and control panel (left). MELD first estimates the condition-specific density across a graph constructed from all cells and samples, yielding one density estimate per condition (middle). These estimates are then normalized to calculate a condition-associated relative likelihood (right).

(legend continued on next page)

knockout versus control embryos, we confirmed that loss of these RegVelo-predicted drivers resulted in disrupted lineage specification (Figures 4H–4J, S7S, and S7T).

We next examined how sequencing depth and related technical parameters influence RegVelo's performance in driver prediction and TSI. Under conditions with sufficient read depth (Figure S8A) (5, 5%–50% subsampling, typically corresponding to >25,000 median reads per cell), TSI remained stable, and driver ranking declined moderately (Figure S8B) (TSI = 0.82, AUROC > 0.8, $N = 10$ trained models). Correlation analysis revealed distinct technical dependencies: TSI was primarily influenced by the number of retained cells, whereas driver ranking depended on the number of retained genes per cell (Figure S8B). Fate-mapping and velocity estimates were also consistent across the 5%–50% sampling range, with noticeable deviations emerging only at 1%. This likely explains the lowered driver ranking at very low depth, as RegVelo relies on cell fate mapping to predict drivers (Figure S8C). Analogous analyses on a sci-RNA-seq mouse neural crest developmental dataset (Figure S8D), characterized by inherently shallow sequencing, confirmed RegVelo's consistent and superior inference of TSI (Figures S8D and S8E). We further conducted *in silico* perturbations of classical lineage drivers—*Mitf* (melanocytes),[117] *Sox10* (myelinating Schwann cells),[118] and *Pou4f1* and *Neurod1* (neuronal)[119,120]—as an orthogonal validation of RegVelo's ability to predict classical lineage regulators during neural crest development (Figure S8F). For instance, *Neurod1* is an early TF expressed in peripheral nervous system neurons (Figure S8G). Whereas RegVelo's perturbed vector fields accurately predicted depletion of enteric and otic sensory neurons,[120,121] a simple Pearson correlation-based analysis[17] failed to do so (Figure S8H).

RegVelo screens TF knockout effects for cell fate decisions in zebrafish

Advances in single-cell Perturb-seq have facilitated systematic evaluation of transcription factor function through targeted knockouts in living organisms.[122] Using this approach, we evaluated RegVelo's predictions of TF knockouts with CRISPR-Cas9 and direct capture Perturb-seq workflows[98] (Figures 4A and 5A; STAR Methods). We designed 2 to 4 chemically modified single guide RNAs[123] per TF (*tfec*, *mitfa*, *bhlhe40*, *tfeb*, *elf1*, *nr2f2*, and *nr2f5*) to maximize knockout efficiency, generated zebrafish F0 crispants[124] by injecting the Cas9-gRNA ribonucleoprotein complex into heterozygous foxd3-Citrine embryos[125] at the 1-cell stage, and profiled single-cell transcriptomics of foxd3-Citrine-expressing cells at the 21-somite stage, in which terminal cell types were identified. We quantified *in vivo* perturbation effects as lineage-specific changes between untargeted control and knockout embryos using manifold enhancement of latent dimensions (MELD) relative likelihood scores.[126] This likelihood is the ratio of density estimates between perturbed and control conditions, indicating how much more likely a cell is to appear under perturbation; lower values indicate greater depletion (Figure 5B; STAR Methods). To systematically evaluate the perturbation effects, we incorporated CellRank to define terminal states in the Perturb-seq data and implemented a permutation test framework to determine the statistical significance of lineage-specific perturbation effects (STAR Methods; Figures 5C and 5D). Experimental results recapitulated known relationships between specific factors and their associated lineages. For instance, the *in vivo* knockout of the top pigment driver, *mitfa*,[102] resulted in pigment lineage depletion (lineage-specific MELD likelihood = 0.3354, permutation test $p = 0.001$), while knocking out the cranial ectomesenchymal driver *nr2f2*[127] depleted mesenchymal cells as predicted by RegVelo (lineage-specific MELD likelihood = 0.47, permutation test $p = 0.001$) (Figure 5E).

We next systematically examined the *in silico* knockout performance using the *in vivo* Perturb-seq data. By analogy to MELD likelihood, we introduced a perturbation metric in RegVelo—density change likelihood—based on Markov simulations, which quantifies density shifts in terminal states and assesses their statistical significance (Figure 5F; STAR Methods). Using this

(C) Estimated pseudotime (see STAR Methods) integrated with *PseudotimeKernel* in CellRank[17] to infer terminal states for all 39,453 neural crest development cells.
(D) Permutation testing framework for evaluating statistical significance of perturbation effects on terminal states (STAR Methods).
(E) MELD-derived likelihood scores for terminal states in the *nr2f2* KO panel (top). Statistical testing (1,000 randomizations) shows a significant depletion in facial mesenchyme but not arch2 (bottom).
(F) RegVelo simulates cell differentiation *in silico* via Markov simulations starting from progenitor populations. After a fixed number of simulation steps, terminal cell visitation densities are computed. A density change likelihood ($\ell_{\Delta\rho}$), analogous to MELD likelihood, is then calculated, and statistical tests are performed to assess perturbation effects on each terminal state.
(G) Density change likelihood following nr2f2 regulon depletion (paired *t* test). Only facial mesenchymal cells showed a significant change ($n = 30, p = 0.008$).
(H) Representative confocal microscopy images (left) and HCR signal quantification (right) of colec12 (facial mesenchyme marker) and dlx2a (arch 2 marker) expression in control and nr2f2 crispant zebrafish embryos at the 21-somite stage. e, eye; o, otic. Error bars indicate 95% confidence intervals. (One-sided Welch's *t* test, facial mesenchymes: $N_{Ctrl} = 6, N_{KO} = 6, p = 0.046$; arch 2: $N_{Ctrl} = 5, N_{KO} = 6, p = 0.76$).
(I) Density change likelihood following nr2f5 regulon depletion (paired *t* test, $n = 30$, arch 2: $p = 4.93 \times 10^{-6}$; facial mesenchyme: $p = 3.47 \times 10^{-8}$; mNC hox34: $p = 4.93 \times 10^{-8}$; Pigment: $p = 6.31 \times 10^{-6}$)
(J) Representative confocal microscopy images of *colec12* (facial mesenchyme, labeled as fm), *dlx2a* (arch 2/pa2, arch 3/pa3), and *mitfa* (pigment, labeled as p) expression in control and *nr2f5* crispant zebrafish embryos at the 21-somite stage (Ex., example).
(K) Quantification of the abundance of all four lineage cell types from HCR confocal images. (One-sided Welch's *t* test; arch 2: $N_{Ctrl} = 5, N_{KO} = 5, p = 0.009$; facial mesenchyme: $N_{Ctrl} = 6, N_{KO} = 9, p = 3.59 \times 10^{-5}$; mNC hox34: $N_{Ctrl} = 3, N_{KO} = 4, p = 0.018$; pigment: $N_{Ctrl} = 3, N_{KO} = 4, p = 0.53$; Error bars: 95% CI).
(L) Benchmark of *in silico* perturbation effect prediction using RegVelo, dynamo, and CellOracle ($n = 3$ trained RegVelo models; one-sided Welch's *t* test; single knockout: $P = 1.78 \times 10^{-6}$; single + multiple knockout: $P = 6.59 \times 10^{-7}$).
(M) Gene program activity changes for *mitfa* (left) and *rarga* (right). Boxplots: NMF scores for Ctrl (gray) versus KO (red) in Perturb-seq (top) and RegVelo predictions (bottom).
See also Figure S9.

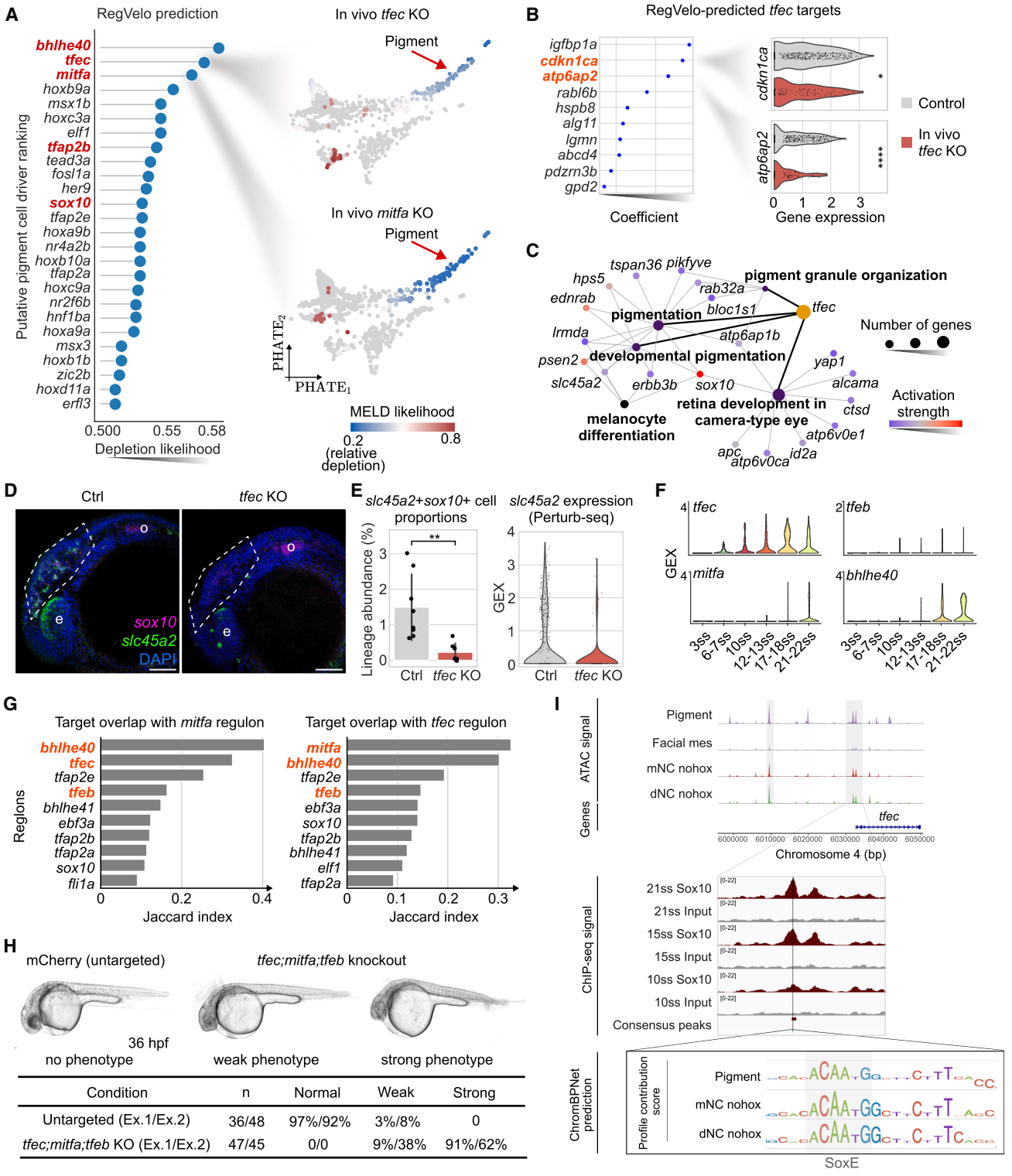


Figure 6. tfec regulates pigment cell development as an early driver

(A) Top putative pigment lineage driver TFs via RegVelo simulation, highlighting known pigment drivers in red (left). The PHATE embeddings visualize the MELD likelihood, indicating the pigment lineage depletion in *tfec* and *mitfa* knockouts (right).

(B) The top 10 positively regulated tfec targets predicted by RegVelo and ranked by inferred weights (left). Violin plot of top target gene expression in *in vivo* Perturb-seq dataset (right; $n = 779$ cells; one-sided Welch's t test, $p = 0.02$ for *cdkn1ca*, $p = 3.44 \times 10^{-8}$ for *atp6ap2*).

(legend continued on next page)

metric, RegVelo predictions, consistent with Perturb-seq results (Figure 5E) (facial mesenchyme: permutation test $p = 0.001$; arch 2: permutation test $p = 0.762$), identified facial mesenchyme as the terminal state showing the strongest reduction in density following *nr2f2* perturbation (Figure 5G). To validate this lineage-specific effect in individual *nr2f2* knockout embryos, we performed multiplexed HCR-FISH staining for lineage markers, followed by cell segmentation and state quantification (STAR Methods). Consistent with both RegVelo predictions and Perturb-seq data, *nr2f2* knockout led to a noticeable reduction in facial mesenchymal but not arch 2 lineage (Figure 5H), despite both deriving from cranial mesenchyme-biased neural crest.[105]

As complementary validation, we predicted perturbation effects for another ectomesenchymal driver, *nr2f5*.[105] Consistent with the perturb-seq results, which showed significant density depletion across all fates in the *nr2f5* knockouts, RegVelo also predicted significant depletion effects, particularly in cranial mesenchymal fates such as arch 2 and facial mesenchyme (Figures 5I and S9A). This strong depletion of mesenchymal fates was further supported by HCR-FISH analysis following *nr2f5* knockout (Figures 5J and 5K). In addition to *nr2f5* and *nr2f2*, RegVelo's *density change likelihood* also showed pronounced depletion effects of pigment lineages for identified pigment drivers, including *tfec*, *mitfa*, *tfeb*, and *bhlhe40*, consistent with MELD likelihood scores obtained in the perturb-seq data (Figure S9A).

To systematically quantify this concordance across the entire perturbation screen and assess RegVelo predictions against the ground truth, we extended our seven-TF Perturb-seq data with an additional complementary Perturb-seq dataset,[98] creating a joint panel of 11 single-TF and 4 multiple-TF knockouts. Only TFs present in the preprocessed Smart-seq3 dataset were retained to ensure comparability (see STAR Methods for preprocessing and filtering). We computed the Spearman correlations between predicted density change likelihoods and experimentally derived MELD likelihood scores, and compared RegVelo predictions to dynamo[23] and CellOracle,[26] which use either dynamic modeling or GRN inference alone (STAR Methods). For single-TF knockouts, RegVelo achieved nearly double the mean Spearman correlation (0.52) compared with the other methods (Spearman correlation < 0.25) (Figures 5L and S9A), with a similar trend for all knockout experiments in the panel (average Spearman correlation of 0.51 versus 0.32; Figure 5L). We further calculated the precision and recall of lineage perturbation prediction (STAR Methods); RegVelo achieved values of 0.6 for both metrics—nearly twice the values for other methods (Figure S9B).

To further investigate the regulatory programs recovered by RegVelo, we compared Perturb-seq data with differentially expressed genes (DEGs) predicted by RegVelo and found they significantly overlapped across most knockout conditions (Figure S9C). Several downregulated gene programs were consistently shared between RegVelo and Perturb-seq genetic knockouts (Figure S9D). Clustering these programs revealed two distinct meta-programs (Figure S9E): Meta-program-1, functionally enriched for mesenchymal cell differentiation, and Meta-program-2, enriched for melanocyte migration and pigmentation processes (Figure S9F). In specific gene perturbations, RegVelo predicted the downregulation of pigment cell-related programs in *mitfa* knockout and the suppression of facial mesenchyme programs in *rarga* knockout, a known regulator of cranial mesenchymal fate[128] (Figure 5M).

Overall, having established that RegVelo correctly and robustly infers the dynamic cellular changes during zebrafish neural crest development, we sought to employ our framework to identify regulatory relationships driving this embryonic developmental process.

tfec regulates pigment cell development as an early driver

Previous studies have established the differentiation programs of pigment progenitor cells into iridophores, melanophores, and xanthophores at later developmental stages.[129,130] However, the initial specification of the pigment lineage within the migratory neural crest remains unresolved. We, thus, turned to the trained neural crest RegVelo model to predict drivers and found bhlhe40, tfec, and mitfa[117] ranked as the top three pigment cell regulons (Figure 6A), and the known pigment lineage[131] markers cdkn1ca and atp6ap2 among the top positively regulated targets of tfec (Figures 6B and 6C). Additionally, other target genes of the tfec regulon, such as *slc45a2*, *lrmda*, and *psen2*, were enriched in pathways of melanocyte differentiation (adjusted $p = 0.004$) (Figure 6C). RegVelo-predicted target effector slc45a2 regulates melanosomal pH in zebrafish,[132] and its ortholog SLC45A2 mediates pigmentation in humans.[133] We observed that slc45a2-expressing NC derivatives were

(C) Gene Ontology network of RegVelo-predicted targets. Activation strength refers to the regulatory weight.
(D) Representative confocal microscopy images of *sox10* and *slc45a2* expression in control (left) and *tfec* knockout (right) zebrafish embryos at 21-ss.
(E) Quantification of the abundance of all sox10 and slc45a2 co-expressing cells from HCR confocal images (left; one-sided Welch's *t* test, $N_{Ctrl} = 8$, $N_{KO} = 7$, $p = 0.002$; Error bars: 95% CI); Expression of slc45a2 in Perturb-seq before and after tfec KO (right).
(F) Expression patterns of representative bHLH factors in the pigment lineage within our Smart-seq3 data along developmental stages. GEX, gene expression.
(G) Top regulons with the highest Jaccard index in RegVelo-inferred GRN, indicating shared target genes, with the *mitfa* and *tfec* enhancer-driven regulons. It implies potential redundancy among *tfec, mitfa, bhlhe40*, and *tfeb* regulons (highlighted in red).
(H) Contrast images illustrate the albino phenotype in F0 crispants of tfec;mitfa;tfeb at 36 h post fertilization (hpf). Representative images for no phenotype in control embryos and weak or strong phenotypes in the KO embryos (top panel). Bottom panel shows the counts and proportions for each phenotype for two experiments (chi-squared test of independence, $p < 1 \times 10^{-10}$ for both repeated experiments). Ex., experiment.
(I) Genomic track view within the *tfec* locus highlights two regions containing ATAC peaks (top panel) overlapping with reproducible Sox10 ChIP-seq peaks ($IDR^{141} < 0.05$). The middle panel shows Sox10 biotin ChIP-seq and input profile with merged biological duplicates per stage, as well as consensus peaks, with zoomed-in views at the right highlighted region, which harbors a motif hit, from the top panel. The bottom panel shows the ChromBPNet-predicted contribution score, showing a Sox10 motif hit call overlapping with the Sox10 ChIP-seq peak at the *tfec* promoter region.[134]
See also Figure S10.

significantly reduced in *tfec* knockout embryos ($p = 0.001$) (Figure 6D and 6E). We, thus, hypothesized that *tfec* could contribute to pigment cell fate priming as an early expressed driver and turned to orthogonal computational and experimental approaches for validation.

Single-cell transcriptomic analysis of both our Smart-seq3 dataset and a published multiome dataset[100] identified tfec as an early basic helix-loop-helix (bHLH) TF expressed in the cranial neural crest, preceding the precursor driver mitfa and others such as tfeb and bhlhe40 (Figures 6F and S10A). Additionally, our Perturb-seq experiment showed significant depletion effects for tfec knockouts compared with mitfa in both early and late pigment lineages (two-sided paired Wilcoxon test, $p < 1 \times 10^{-12}$) (Figures S10B and S10C). Targets of tfec in RegVelo-reweighted GRN and targets of other pigment lineage drivers, mitfa and tfeb, largely overlapped (Figures 6G and S10D). Therefore, we hypothesized that tfec, mitfa, and tfeb had functional overlap in pigmentation lineage specification. To test this, we performed single-TF and triple-TF knockout and counted the number of embryos falling into each of the three categories of embryonic phenotypes: no phenotype (full pigmentation), weak phenotype (reduced pigmentation), and strong phenotype (no pigmentation). The simultaneous knockout of mitfa, tfec, and tfeb led to pigment cell depletion in 36-h-post-fertilization embryos compared with the untargeted control, whereas this pigment depletion phenotype was not observed in single-gene knockouts (Figures 6H and S10E). Our results aligned with the previously reported single-knockout phenotypic observation.[134]

To explore the transcriptional regulation upstream of tfec, we leveraged the multiome-ATAC data[100] and predicted TF binding sites in SCENIC+-inferred CREs within the tfec locus. We used the base-resolution contribution scores derived from the convolutional neural network models[100] trained and interpreted by ChromBPNet,[135] which allowed us to characterize the putative TF binding instances. Our ChromBPNet-based model predicted Sox10 motif hits in two CREs upstream of *tfec* (chr4 : 6031867 – 6031883 and chr4 : 6019773 – 6019789; cosine similarity > 0.83, by Finemo[135]) (Figure 6I). Sox10, a key regulator of NC development, is essential for the pigment lineage specification[103] (Figure S10F). To further test whether Sox10 directly regulated *tfec*, we conducted Sox10 Biotin ChIP-seq[136,137] with biological replicates at three developmental stages (10 ss, 15 ss, and 21 ss). In Biotin ChIP-seq analysis, we called reproducible peaks between replicates for each stage ($q < 0.05$, by irreproducibility discovery rate [IDR][138]) (Figure S10G; STAR Methods). A reproducible ChIP-seq peak overlapping ChromBPNet-predicted Sox10 binding sites (chr4 : 6031499 – 6032013) was identified within the tfec promoter region at 15 ss and 21 ss (Model-based analysis of ChIP-Seq 2 [MACS2] $q < 2.5 \times 10^{-14}, 4.3 \times 10^{-5}$ and IDR = 0.04, 0.0018 for two stages, respectively) (Figure 6I). These results suggested that key neural crest regulator Sox10 could bind this region and likely directly modulate tfec expression in neural crest cell populations.

Overall, our experimental output of Perturb-Seq and HCR quantification indicated that *tfec* contributed to the transition from initial neural crest cell states to pigment cell states, and *tfec* might act as an early driver, which aligned with the RegVelo-predicted role of *tfec*. Our ChIP-seq profiling and motif hit calling results implied that *tfec* was likely a direct downstream effector of Sox10 in neural crest development.

RegVelo discovers the *elf1* pro-pigment role

Elf1, a TF from the erythroblast transformation specific (ETS) family, was predicted as one of the top ten regulators of pigment fate by RegVelo in neural crest development (Figure 6A). Density simulation of RegVelo further predicted that elf1 perturbation significantly reduces terminal-state pigment cell density ($p = 0.0091$) Figure 7A). Comparing the expression of targets ordered by RegVelo-inferred latent time revealed a clear bifurcation between pigment and mesenchymal cell fates, with elf1 targets upregulated in the pigment lineage, and pro-mesenchymal ETS TFs (i.e., fli1a) upregulated in the mesenchymal lineage[100] (Figure 7B).

To test the predicted role of *elf1*, we analyzed *in vivo* Perturb-seq data, which revealed significant depletion in pigment lineage upon *elf1* knockout (Figure 7C). We next confirmed the *elf1* knockout phenotype using orthogonal approaches. HCR RNA fluorescence *in situ* hybridization for *mitfa* (pigment), *Cherry* (neural crest), and *colec12* (mesenchyme) following *elf1* knockout showed reduced pigment cell numbers in the pre-otic cranial region (Figures 7D and 7E). Moreover, both Perturb-seq and HCR results showed that *elf1* knockouts showed enrichment in certain cranial mesenchymal fates, consistent with RegVelo-simulated cell density changes (Figure 7A). At the gene level, we observed significant downregulation of *mitfa* and *sox10* in the *elf1* KO versus the untargeted control of Perturb-seq (Figure 7F). Together, these experimental validations support the pro-pigment, anti-mesenchymal role of *elf1* predicted by RegVelo.

To explore the regulatory context of *elf1*, we examined its upstream regulators and downstream targets in comparison with RegVelo-predicted regulatory interactions (Table S3). We first performed differential expression analyses using Perturb-seq data from both pigment and mesenchymal fate driver knockouts to assess the expression patterns of *elf1* and related genes. According to Perturb-seq analysis, *elf1* was significantly downregulated in *tfec* knockouts and upregulated in *fli1a* knockouts (Figure 7G), aligning with RegVelo-predicted regulatory weights from *tfec* or *fli1a* to *elf1* (Table S3).

To identify direct targets of elf1 in the context of this regulatory network, we performed elf1 CUT&RUN and identified 1,967 reproducible peaks ($q < 0.05$, by IDR) (Figure S10H). Among these reproducible peaks, 714 harbored canonical ETS family motif hits ($p < 0.004$, by FIMO[139]) and were annotated as the putative direct elf1 binding sites. Motif enrichment analysis by SEA[140] confirmed that the ELF1 motif was enriched in these putative direct binding sites ($q = 9.81 \times 10^{-11}$). We identified potential direct elf1 binding at the *glulb*, *cpeb4b*, and *pleca* loci, all of which were predicted as elf1 targets in the RegVelo-reweighted neural crest GRN (Table S3). These included the promoter region of cpeb4b (chr21 : 41369360 – 41369582), a distal region at chr19 : 22202744 – 22202944 associated with *pleca* expression ($p = 0.0015$, by Signac linkage analysis), and a distal region at chr6 : 23843706 – 23844026 with *glulb* annotated as the nearest gene (Figure 7H).

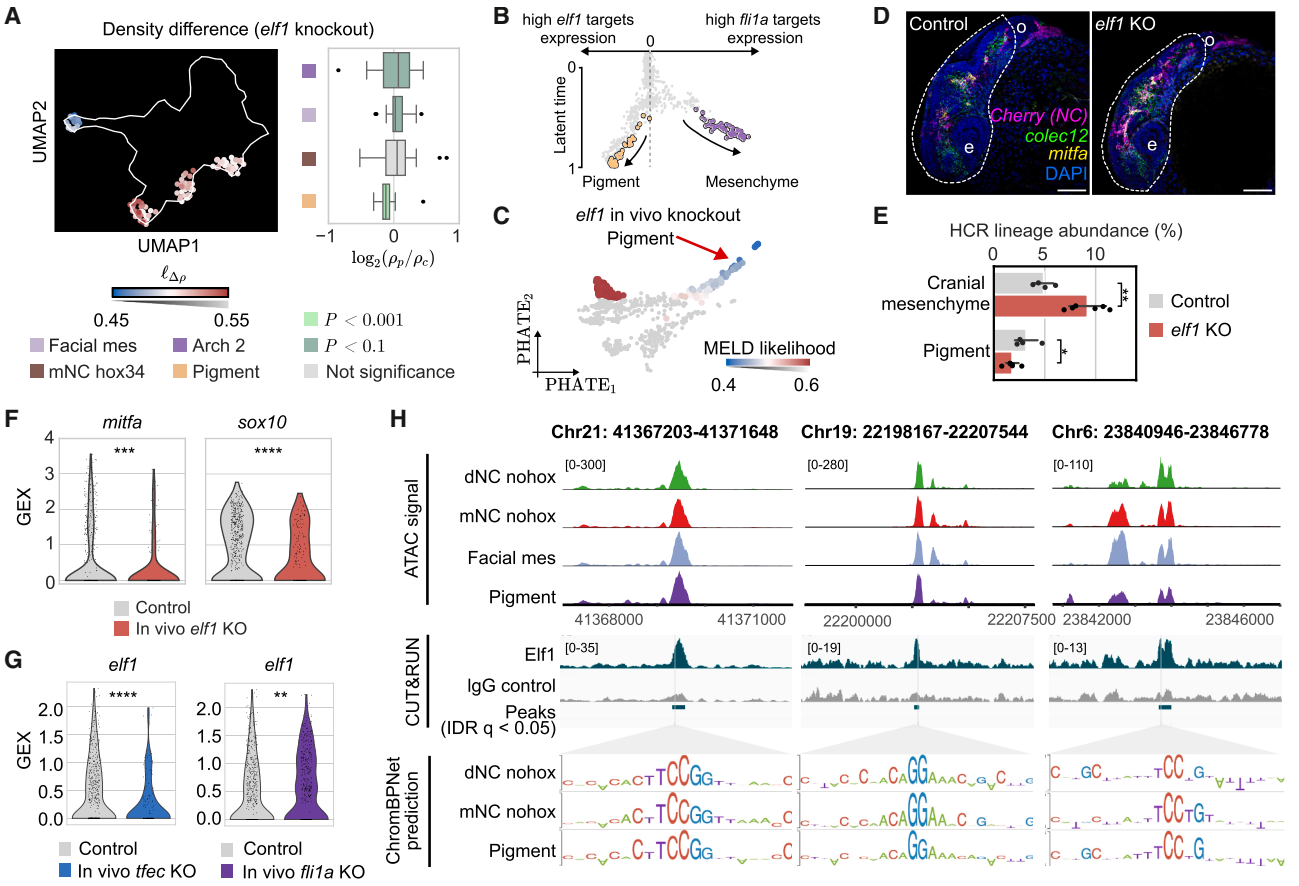


Figure 7. RegVelo identifies an early pigment lineage driver, elf1, together with its regulatory context

(A) Density change likelihood ($\ell_{\Delta\rho}$) following elf1 regulon depletion (left; paired *t* tests, $n = 30$, arch 2: $p = 0.03$; facial mesenchyme: $p = 0.033$; mNC hox34: $p = 0.93$; pigment: $p = 0.009$).

(B) Visualizing the $\log_2$-fold change of the average expression levels of all fli1a targets versus all elf1 targets in each cell (*x* axis) along RegVelo latent time (*y* axis).

(C) PHATE embedding colored by *in vivo* elf1 perturbation scores quantified by MELD (top); Pigment depletion and facial mesenchyme enrichment in terminal states confirmed by permutation test ($p = 0.001$, 1,000 randomizations).

(D) Representative confocal microscopy images of *colec12* and *mitfa* expression in control (left) and *elf1* crispant (right) zebrafish embryos at 21-ss. Dashed line indicates regions of interest.

(E) Quantification of the abundance of cranial mesenchyme and pigment cells from confocal images after HCR (One-sided Welch's *t* test; facial mesenchymes: $N_{Ctrl} = 4, N_{KO} = 6, p = 0.001$; pigment: $N_{Ctrl} = 4, N_{KO} = 6, p = 0.018$). Error bars: 95% CI.

(F) mitfa and sox10 are significantly downregulated in elf1-knockout neural crest cells based on Perturb-seq data (one-sided Welch's *t* test, mitfa: $p = 0.0003$; sox10: $p = 2.68 \times 10^{-5}$).

(G) elf1 expression levels are significantly decreased in tfec knockout and significantly increased in fli1a knockout (one-sided Welch's *t* test, tfec knockout: $p = 8.61 \times 10^{-8}$; fli1a knockout: $p = 0.0011$).

(H) Genomic track view of ATAC signals of cluster-specific pseudobulks (top panel) overlapping with reproducible Elf1 CUT&RUN peaks. The middle panel shows Elf1 CUT&RUN and IgG profile with merged biological duplicates per group, as well as reproducible peaks, with light gray lines indicating ETS family motif hits. The bottom panel shows the ChromBPNet-predicted contribution score, showing ETS motif hits overlapping with the Elf1 CUT&RUN peaks. The putative binding regions are located at chr21 : 41369360 – 41369582 (left), chr19 : 22202744 – 22202944 (middle), and chr6 : 23843706 – 23844026 (right), while the subtitles indicate the genomic coordinate ranges of each panel.

See also Figure S10.

DISCUSSION

We have presented RegVelo, a framework that jointly learns RNA velocity and underlying gene regulation from single-cell gene expression data using variational inference. The step from decoupled single-gene models to a fully integrated dynamic model by explicitly incorporating gene regulation allows our method to compare favorably to previous approaches on dynamic inferences. Notably, forward simulation of the learned gene regulatory dynamics provides a physically grounded *virtual cell*[141] style simulation for modeling regulatory mechanisms in cell fate decision behavior.

We recovered murine pancreatic endocrinogenesis trajectories and regulatory mechanisms, faithfully predicted drivers

and regulatory effects in ductal cell cycle progression, and lineage conformation. In human hematopoiesis, where previous RNA velocity methods underperformed even with access to metabolic labeling information, RegVelo recovered the toggle-switch network motif between GATA1 and SPI1 (PU.1) and, through RegVelo's perturbation scheme, illustrated their roles in erythroid and monocyte-fate decisions.

While most ETS factors have pro-mesenchymal bias in zebrafish cranial neural crest,[100] RegVelo led us to report *elf1* as an exception that promotes pigment fates. Our finding is consistent with the DNA-binding anti-cooperativity of ELF1 caused by an electrostatic repulsion mechanism, in contrast to FLI1 and ETS1.[142] Our newly found driver for the pigment lineage, *elf1*, has also been reported as the intermediate-state marker in melanoma,[143] where the tumor cell states intermediate or neural crest-like, mesenchyme-like, and melanoblast-like were reported to mirror normal cell states in the neural crest cellular trajectories.[144] Future studies can investigate whether *elf1* is involved in a shared regulatory circuitry between neural crest fate decisions and melanoma. Overall, our systematic *in silico* and *in vivo* TF knockout datasets serve as valuable resources for studying cranial neural crest development and relevant diseases, as well as for benchmarking the accuracy of knockout simulation algorithms.

As an end-to-end dynamic, interpretable, and actionable model, RegVelo distinguishes itself from other studies that focus on interpreting cell dynamics through gene regulation[23,145,146] in three aspects. (1) These approaches depend on predefined cellular velocity estimates[23,145,146] from models ignoring regulatory events.[13,14] In contrast, RegVelo simultaneously learns gene regulation and velocity. (2) RegVelo integrates prior GRN knowledge from multiple modalities into velocity estimation. (3) Previous models predict perturbation effects by manually activating or suppressing gene expression,[23,145] where the output represents the immediate perturbed direction of each cell. RegVelo, however, encodes perturbation by directly changing the underlying regulatory circuits and simulating gene expression dynamics under the perturbed regulatory system. This forward simulation allows for the prediction of nonlinear perturbation effects over latent time rather than focusing only on the immediate first-order direction, which misses long-term consequences. We demonstrated this by recovering early lineage drivers in zebrafish neural crest and validating TF perturbation predictions across four lineages *in vivo*. Our results suggested GRN-infused hybrid models may hold promise to be more robust on out-of-distribution data than purely data-based generative models.[147]

Built upon veloVI, RegVelo is a flexible model that can easily integrate additional modalities to overcome current limitations in model formulation and assumptions. For example, incorporating metabolic labeling data or *in vivo* time-resolved scRNA-seq data, such as Zman-seq,[148] enables the development of differential equations to describe transcriptional regulation and infer cell-specific kinetic rates, such as degradation.[23] Similarly, CLIP-seq and DART-seq have provided insights into RNA-binding protein (RBP)-RNA interactions at the single-molecule level,[149,150] and combining such post-transcriptional gene-gene interactions can improve the regulatory modeling among different molecules and estimate cell-specific splicing rates. Moreover, we and other existing frameworks typically assume that gene expression depends on the abundance of upstream RNA regulators rather than protein levels—integrating ribosome profiling data enables modeling the translation process and quantifying the protein activity of regulators, which enables a more nuanced estimation of transcription rates.[151] Finally, extending the biophysical model of splicing kinetics beyond mRNA toward chromatin and protein dynamics will lead to a more complete and accurate description of the underlying process, though integration remains challenging due to modality differences and limited shared measurements.

Overall, RegVelo innovates transcriptome-wide dynamic modeling from pure trajectory inference to uncovering causal mechanisms involved during cell fate decisions. With this flexible model as a prototype, we expect future extensions by incorporating additional regulatory layers at different molecular levels, ultimately moving toward a more comprehensive and interpretable model capable of simulating and predicting cell behavior and responses even more accurately.

Limitations of the study

While RegVelo demonstrates promising performance, several limitations highlight opportunities for future improvement. First, as in our previous work,[13,14] RegVelo assumes that the global latent time, obtained by averaging gene-specific latent times, may fail to capture all gene-specific temporal dynamics and ignores the geometric structure of the cellular temporal space. Therefore, this limitation can be addressed by replacing gene-specific latent times with a single, shared temporal coordinate system, but estimating the initial expression value for each gene in each cell requires additional regularization techniques. Second, the current regulatory model is restricted to simple linear interactions and does not yet incorporate TF activity inferred from sequence motifs or multimodal data such as chromatin accessibility. Finally, our bootstrap-based method for estimating regulatory uncertainty requires multiple model runs, incurring substantial computational cost. Future work will explore the integration of uncertainty estimation within a Bayesian framework, such as Bayesian neural networks,[152] to produce more efficient, better-calibrated probabilistic estimates of GRN edges and improve downstream tasks, including perturbation prediction.

RESOURCE AVAILABILITY

Lead contact

Requests for further information and resources should be directed to and will be fulfilled by the lead contact, Fabian J. Theis (fabian.theis@helmholtz-munich.de).

Materials availability

Materials generated in this study will be made available upon reasonable request from the lead contact.

Data and code availability

- The cell cycle, human hematopoiesis, and pancreatic endocrinogenesis data presented and used in this study are publicly available via the original publications; we provide additional access to each of these datasets and the zebrafish data via a figshare collection at https://figshare.com/projects/RegVelo_reproducibility_datasets/226860. The zebrafish data is available at https://tracks.stowers.org/compbio/ucsc_cellbrowser/.

- The following datasets generated in this study are publicly available: zebrafish Smart-seq3 data at GEO: GSE256009, zebrafish Perturb-seq data at GEO: GSE256008, Sox10 biotin ChIP-seq and Elf1 CUT&RUN data at GEO: GSE303928, and human head multiome data at BioProject: PRJNA1214067.
- RegVelo is released under the BSD-3-Clause license, with code available at https://github.com/theislab/regvelo. Code to reproduce the results in the paper can be found at https://github.com/theislab/regvelo_reproducibility.
- Additional information required to reanalyze the data reported in this document is available from the lead contact upon request.

ACKNOWLEDGMENTS

We thank all members of the Theis and Sauka-Spengler laboratories for helpful discussions. W.W. thanks Dr. Xiaojie Qiu for his feedback on RegVelo concerning implementation and benchmarking aspects; A. Palma and M. Lienen for helpful discussions about parallel ODE solvers and neural ODE models; D. Klein for insights into pancreatic endocrine development; L. Kummerle for discussing RegVelo perturbation benchmark results; Dr. Peng He, Dr. Bao Zhang, and Dr. J. Patrick Pett for providing processed limb myogenesis and GRN datasets; Dr. Louis Faure for providing BAM files of mouse neural crest development datasets; and Yifan Chen, Ding Huang, and Zekai Li for refactoring the code repository. We thank the MRC WIMM Flow Cytometry Facility, MRC WIMM Advanced Single Cell OMICS Facility, MRC WIMM Centre for Computational Biology, and MRC WIMM Wolfson Imaging Centre for their help and technical support in the zebrafish component of the study and Stowers Institute Sequencing, Imaging and Aquatics Technology Centers, and C. Scott for help with zebrafish injections. Z.H. thanks the Core Facility of Medical Research Institute at Wuhan University for the HPC and technical support. Scientific illustrations shown in Figure 4A were created by Uta Mackensen. This work was co-funded by the European Union (ERC, DeepCell-101054957) and the Wellcome Leap ΔTissue Program (9E8E84F7-8991-4D4A-A9EC) and received funding from the European Union's Horizon 2022 Horizon Research and innovation programme under grant agreement no. 101057775. F.J.T. acknowledges support from the German Federal Ministry of Education and Research (BMBF) through the HOPARL project (grant number 031L0289A). M.L. further acknowledges financial support by the DFG through the Graduate School of QBM (GSC 1006), by the Joachim Herz Foundation when employed by the Technical University of Munich and Helmholtz Munich, and through an EMBO Postdoctoral Fellowship. This work was also supported by the Wellcome Trust award # 215615/Z/19/Z (Z.H., S.M., D.M.F., and T.S.-S.) and Stowers Institute for Medical Research institutional support to T.S.-S. Z.H. acknowledges the Fundamental Research Funds for the Central Universities (2042025kf0022 and 2042022dx0003) and the National Natural Science Foundation of China (32500725). For all support via EU funding, the views and opinions expressed are those of the authors only and do not necessarily reflect those of the European Union or the European Research Council. Neither the European Union nor the granting authority can be held responsible for them.

AUTHOR CONTRIBUTIONS

W.W., Z.H., and P.W. contributed equally. W.W., P.W., M.L., and F.J.T. conceptualized the study. Z.H. and T.S.-S. conceptualized the zebrafish study design. W.W. designed and implemented RegVelo with contributions from P.W. and M.L. Z.H., S.M., J.O.H., and D.M.F. performed wet-lab experiments. W.W., Z.H., and P.W. designed and implemented analysis methods with contributions from J.W. and Z.X. T.S.-S. and F.J.T. supervised the work. P.W., W.W., Z.H., T.S.-S., and F.J.T. wrote the manuscript. All authors read and approved the final paper.

DECLARATION OF INTERESTS

F.J.T. consults for Immunai Inc., CytoReason Ltd, Cellarity, and BioTuring Inc. and has an ownership interest in Dermagnostix GmbH and Cellarity.

DECLARATION OF GENERATIVE AI AND AI-ASSISTED TECHNOLOGIES IN THE WRITING PROCESS

During the preparation of this work, the author(s) used ChatGPT in order to improve language. After using this tool or service, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

STAR★METHODS

Detailed methods are provided in the online version of this paper and include the following:

- KEY RESOURCES TABLE
- EXPERIMENTAL MODEL AND STUDY PARTICIPANT DETAILS
  - Fish husbandry and Smart-seq3 single-cell RNA-Seq
  - In vivo Perturb-seq of zebrafish neural crest development
  - Whole-mount hybridization chain reaction (HCR) and phenotypic check
  - Sox10 biotin ChIP-seq
  - CUT&RUN
  - Hindbrain scRNA-seq and single nuclei multiome sequencing
- METHOD DETAILS
  - RegVelo model specification
  - RegVelo generative process
  - RegVelo inference
  - Related work
  - scKINETICS
  - TFvelo
  - Velorama
  - Relation to RegVelo
  - Downstream tasks
  - Benchmarking metrics
  - Hyperparameters sensitivity analysis
  - Selecting appropriate model using side information
  - Evaluating the impact of the GRN on dynamics inference
  - Evaluating the impact of the prior GRN and learned edge weights on driver prediction
  - Quantifying perturbation effects
  - Data simulation
- QUANTIFICATION AND STATISTICAL ANALYSIS
  - General processing
  - Trajectory inference benchmark
  - Driver gene analysis
  - Perturbation screening
  - Benchmarking related methods
  - Simulated dataset – Dyngen simulation
  - Mouse hematopoietic stem and progenitor cell differentiation
  - Mouse neural crest development – Smart-seq2
  - Mouse neural crest development – sci-RNA-seq3
  - Cell cycling dataset – U2OS cell line
  - Cell cycling dataset – RPE1 cell line
  - Pancreatic endocrinogenesis
  - Human hematopoiesis
  - Zebrafish neural crest development
  - Human limb development
  - Human embryonic hindbrain development

SUPPLEMENTAL INFORMATION

Supplemental information can be found online at https://doi.org/10.1016/j.cell.2026.04.022.

Received: December 6, 2024
Revised: December 16, 2025
Accepted: April 10, 2026
Published: May 12, 2026

REFERENCES

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

2. Vento-Tormo, R., Efremova, M., Botting, R.A., Turco, M.Y., Vento-Tormo, M., Meyer, K.B., Park, J.-E., Stephenson, E., Polański, K., Goncalves, A., et al. (2018). Single-cell reconstruction of the early maternal-fetal interface in humans. Nature *563*, 347–353. https://doi.org/10.1038/s41586-018-0698-6.

3. Litviňuková, M., Talavera-López, C., Maatz, H., Reichart, D., Worth, C.L., Lindberg, E.L., Kanda, M., Polanski, K., Heinig, M., Lee, M., et al. (2020). Cells of the adult human heart. Nature *588*, 466–472. https://doi.org/10.1038/s41586-020-2797-4.

4. Waddington, C.H. (2014). The Strategy of the Genes, First Edition (Routledge). https://doi.org/10.4324/9781315765471.

5. Saelens, W., Cannoodt, R., Todorov, H., and Saeys, Y. (2019). A comparison of single-cell trajectory inference methods. Nat. Biotechnol. *37*, 547–554. https://doi.org/10.1038/s41587-019-0071-9.

6. Wolf, F.A., Hamey, F.K., Plass, M., Solana, J., Dahlin, J.S., Göttgens, B., Rajewsky, N., Simon, L., and Theis, F.J. (2019). PAGA: graph abstraction reconciles clustering with trajectory inference through a topology preserving map of single cells. Genome Biol. *20*, 59. https://doi.org/10.1186/s13059-019-1663-x.

7. Haghverdi, L., Büttner, M., Wolf, F.A., Buettner, F., and Theis, F.J. (2016). Diffusion pseudotime robustly reconstructs lineage branching. Nat. Methods *13*, 845–848. https://doi.org/10.1038/nmeth.3971.

8. Bendall, S.C., Davis, K.L., Amir, E.-A.D., Tadmor, M.D., Simonds, E.F., Chen, T.J., Shenfeld, D.K., Nolan, G.P., and Pe'er, D. (2014). Single-cell trajectory detection uncovers progression and regulatory coordination in human B cell development. Cell *157*, 714–725. https://doi.org/10.1016/j.cell.2014.04.005.

9. Qiu, X., Mao, Q., Tang, Y., Wang, L., Chawla, R., Pliner, H.A., and Trapnell, C. (2017). Reversed graph embedding resolves complex single-cell trajectories. Nat. Methods *14*, 979–982. https://doi.org/10.1038/nmeth.4402.

10. Setty, M., Kiseliovas, V., Levine, J., Gayoso, A., Mazutis, L., and Pe'er, D. (2019). Characterization of cell fate probabilities in single-cell data with Palantir. Nat. Biotechnol. *37*, 451–460. https://doi.org/10.1038/s41587-019-0068-4.

11. Street, K., Risso, D., Fletcher, R.B., Das, D., Ngai, J., Yosef, N., Purdom, E., and Dudoit, S. (2018). Slingshot: cell lineage and pseudotime inference for single-cell transcriptomics. BMC Genomics *19*, 477. https://doi.org/10.1186/s12864-018-4772-0.

12. La Manno, G., Soldatov, R., Zeisel, A., Braun, E., Hochgerner, H., Petukhov, V., Lidschreiber, K., Kastriti, M.E., Lönnerberg, P., Furlan, A., et al. (2018). RNA velocity of single cells. Nature *560*, 494–498. https://doi.org/10.1038/s41586-018-0414-6.

13. Bergen, V., Lange, M., Peidli, S., Wolf, F.A., and Theis, F.J. (2020). Generalizing RNA velocity to transient cell states through dynamical modeling. Nat. Biotechnol. *38*, 1408–1414. https://doi.org/10.1038/s41587-020-0591-3.

14. Gayoso, A., Weiler, P., Lotfollahi, M., Klein, D., Hong, J., Streets, A., Theis, F.J., and Yosef, N. (2024). Deep generative modeling of transcriptional dynamics for RNA velocity analysis in single cells. Nat. Methods *21*, 50–59. https://doi.org/10.1038/s41592-023-01994-w.

15. Weiler, P., Van den Berge, K., Street, K., and Tiberi, S. (2023). A Guide to Trajectory Inference and RNA Velocity. Methods Mol. Biol. *2584*, 269–292. https://doi.org/10.1007/978-1-0716-2756-3_14.

16. Lange, M., Bergen, V., Klein, M., Setty, M., Reuter, B., Bakhti, M., Lickert, H., Ansari, M., Schniering, J., Schiller, H.B., et al. (2022). CellRank for directed single-cell fate mapping. Nat. Methods *19*, 159–170. https://doi.org/10.1038/s41592-021-01346-6.

17. Weiler, P., Lange, M., Klein, M., Pe'er, D., and Theis, F. (2024). CellRank 2: unified fate mapping in multiview single-cell data. Nat. Methods *21*, 1196–1205. https://doi.org/10.1038/s41592-024-02303-9.

18. Aibar, S., González-Blas, C.B., Moerman, T., Huynh-Thu, V.A., Imrichova, H., Hulselmans, G., Rambow, F., Marine, J.-C., Geurts, P., Aerts, J., et al. (2017). SCENIC: single-cell regulatory network inference and clustering. Nat. Methods *14*, 1083–1086. https://doi.org/10.1038/nmeth.4463.

19. Matsumoto, H., Kiryu, H., Furusawa, C., Ko, M.S.H., Ko, S.B.H., Gouda, N., Hayashi, T., and Nikaido, I. (2017). SCODE: an efficient regulatory network inference algorithm from single-cell RNA-Seq during differentiation. Bioinformatics *33*, 2314–2321. https://doi.org/10.1093/bioinformatics/btx194.

20. Specht, A.T., and Li, J. (2017). LEAP: constructing gene co-expression networks for single-cell RNA-sequencing data using pseudotime ordering. Bioinformatics *33*, 764–766. https://doi.org/10.1093/bioinformatics/btw729.

21. Chan, T.E., Stumpf, M.P.H., and Babtie, A.C. (2017). Gene Regulatory Network Inference from Single-Cell Data Using Multivariate Information Measures. Cell Syst. *5*, 251–267.e3. https://doi.org/10.1016/j.cels.2017.08.014.

22. Qiu, X., Rahimzamani, A., Wang, L., Ren, B., Mao, Q., Durham, T., McFaline-Figueroa, J.L., Saunders, L., Trapnell, C., and Kannan, S. (2020). Inferring Causal Gene Regulatory Networks from Coupled Single-Cell Expression Dynamics Using Scribe. Cell Syst. *10*, 265–274.e11. https://doi.org/10.1016/j.cels.2020.02.003.

23. Qiu, X., Zhang, Y., Martin-Rufino, J.D., Weng, C., Hosseinzadeh, S., Yang, D., Pogson, A.N., Hein, M.Y., Hoi Joseph Min, K., Wang, L., et al. (2022). Mapping transcriptomic vector fields of single cells. Cell *185*, 690–711.e45. https://doi.org/10.1016/j.cell.2021.12.045.

24. Bravo González-Blas, C., De Winter, S., Hulselmans, G., Hecker, N., Matetovici, I., Christiaens, V., Poovathingal, S., Wouters, J., Aibar, S., and Aerts, S. (2023). SCENIC+: single-cell multiomic inference of enhancers and gene regulatory networks. Nat. Methods *20*, 1355–1367. https://doi.org/10.1038/s41592-023-01938-4.

25. Fleck, J.S., Jansen, S.M.J., Wollny, D., Zenk, F., Seimiya, M., Jain, A., Okamoto, R., Santel, M., He, Z., Camp, J.G., et al. (2023). Inferring and perturbing cell fate regulomes in human brain organoids. Nature *621*, 365–372. https://doi.org/10.1038/s41586-022-05279-8.

26. Kamimoto, K., Stringa, B., Hoffmann, C.M., Jindal, K., Solnica-Krezel, L., and Morris, S.A. (2023). Dissecting cell identity via network inference and in silico gene perturbation. Nature *614*, 742–751. https://doi.org/10.1038/s41586-022-05688-9.

27. Wang, L., Trasanidis, N., Wu, T., Dong, G., Hu, M., Bauer, D.E., and Pinello, L. (2023). Dictys: dynamic gene regulatory network dissects developmental continuum with single-cell multiomics. Nat. Methods *20*, 1368–1378. https://doi.org/10.1038/s41592-023-01971-3.

28. Yuan, Q., and Duren, Z. (2024). Inferring gene regulatory networks from single-cell multiome data using atlas-scale external data. Nat. Biotechnol. *43*, 247–257. https://doi.org/10.1038/s41587-024-02182-7.

29. Gao, M., Qiao, C., and Huang, Y. (2022). UniTVelo: temporally unified RNA velocity reinforces single-cell trajectory inference. Nat. Commun. *13*, 6586. https://doi.org/10.1038/s41467-022-34188-7.

30. Gu, Y., Blaauw, D., and Welch, J.D. (2022). Bayesian inference of RNA velocity from multi-lineage single-cell data. Preprint at bioRxiv. https://doi.org/10.1101/2022.07.08.499381.

31. Aivazidis, A., Memi, F., Kleshchevnikov, V., Er, S., Clarke, B., Stegle, O., and Bayraktar, O.A. (2025). Cell2fate infers RNA velocity modules to improve cell fate prediction. Nat. Methods *22*, 698–707. https://doi.org/10.1038/s41592-025-02608-3.

32. Li, J., Pan, X., Yuan, Y., and Shen, H.-B. (2024). TFvelo: gene regulation inspired RNA velocity estimation. Nat. Commun. *15*, 1387. https://doi.org/10.1038/s41467-024-45661-w.

33. Lambert, S.A., Jolma, A., Campitelli, L.F., Das, P.K., Yin, Y., Albu, M., Chen, X., Taipale, J., Hughes, T.R., and Weirauch, M.T. (2018). The Human Transcription Factors. Cell *172*, 650–665. https://doi.org/10.1016/j.cell.2018.01.029.

34. Ma, S., Zhang, B., LaFave, L.M., Earl, A.S., Chiang, Z., Hu, Y., Ding, J., Brack, A., Kartha, V.K., Tay, T., et al. (2020). Chromatin Potential Identified by Shared Single-Cell Profiling of RNA and Chromatin. Cell *183*, 1103–1116.e20. https://doi.org/10.1016/j.cell.2020.09.056.

35. Liu, T., Ortiz, J.A., Taing, L., Meyer, C.A., Lee, B., Zhang, Y., Shin, H., Wong, S.S., Ma, J., Lei, Y., et al. (2011). Cistrome: an integrative platform for transcriptional regulation studies. Genome Biol. *12*, R83. https://doi.org/10.1186/gb-2011-12-8-r83.

36. Zhang, S., Pyne, S., Pietrzak, S., Halberg, S., McCalla, S.G., Siahpirani, A.F., Sridharan, R., and Roy, S. (2023). Inference of cell type-specific gene regulatory networks on cell lineages from single cell omic datasets. Nat. Commun. *14*, 3064. https://doi.org/10.1038/s41467-023-38637-9.

37. Chen, C.-H., Zheng, R., Tokheim, C., Dong, X., Fan, J., Wan, C., Tang, Q., Brown, M., Liu, J.S., Meyer, C.A., et al. (2020). Determinants of transcription factor regulatory range. Nat. Commun. *11*, 2472. https://doi.org/10.1038/s41467-020-16106-x.

38. Kingma, D.P., and Welling, M. (2013). Auto-encoding Variational Bayes. Preprint at arXiv. https://doi.org/10.48550/ARXIV.1312.6114.

39. Mahdessian, D., Cesnik, A.J., Gnann, C., Danielsson, F., Stenström, L., Arif, M., Zhang, C., Le, T., Johansson, F., Schutten, R., et al. (2021). Spatiotemporal dissection of the cell cycle with single-cell proteogenomics. Nature *590*, 649–654. https://doi.org/10.1038/s41586-021-03232-9.

40. Schiebinger, G., Shu, J., Tabaka, M., Cleary, B., Subramanian, V., Solomon, A., Gould, J., Liu, S., Lin, S., Berube, P., et al. (2019). Optimal-Transport Analysis of Single-Cell Gene Expression Identifies Developmental Trajectories in Reprogramming. Cell *176*, 928–943. e22.

41. Cannoodt, R., Saelens, W., Deconinck, L., and Saeys, Y. (2021). Spearheading future omics analyses using dyngen, a multi-modal simulator of single cells. Nat. Commun. *12*, 3942. https://doi.org/10.1038/s41467-021-24152-2.

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

43. Nestorowa, S., Hamey, F.K., Pijuan Sala, B., Diamanti, E., Shepherd, M., Laurenti, E., Wilson, N.K., Kent, D.G., and Göttgens, B. (2016). A single-cell resolution map of mouse hematopoietic stem and progenitor cell differentiation. Blood *128*, e20–e31. https://doi.org/10.1182/blood-2016-05-716480.

44. Taylor, S.S., and McKeon, F. (1997). Kinetochore localization of murine Bub1 is required for normal mitotic timing and checkpoint response to spindle damage. Cell *89*, 727–735. https://doi.org/10.1016/S0092-8674(00)80255-X.

45. Bandara, L.R., Lam, E.W., Sørensen, T.S., Zamanian, M., Girling, R., and La Thangue, N.B. (1994). DP-1: a cell cycle-regulated and phosphorylated component of transcription factor DRTF1/E2F which is functionally important for recognition by pRb and the adenovirus E4 orf 6/7 protein. EMBO J. *13*, 3104–3114. https://doi.org/10.1002/j.1460-2075.1994.tb06609.x.

46. Nielsen, C.F., Zhang, T., Barisic, M., Kalitsis, P., and Hudson, D.F. (2020). Topoisomerase IIα is essential for maintenance of mitotic chromosome structure. Proc. Natl. Acad. Sci. USA *117*, 12131–12142. https://doi.org/10.1073/pnas.2001760117.

47. Ogata, H., Goto, S., Sato, K., Fujibuchi, W., Bono, H., and Kanehisa, M. (1999). KEGG: Kyoto Encyclopedia of Genes and Genomes. Nucleic Acids Res. *27*, 29–34. https://doi.org/10.1093/nar/27.1.29.

48. Zou, Z., Ohta, T., Miura, F., and Oki, S. (2022). ChIP-Atlas 2021 update: a data-mining suite for exploring epigenomic landscapes by fully integrating ChIP-seq, ATAC-seq and Bisulfite-seq data. Nucleic Acids Res. *50*, W175–W182. https://doi.org/10.1093/nar/gkac199.

49. Moerman, T., Aibar Santos, S., Bravo González-Blas, C., Simm, J., Moreau, Y., Aerts, J., and Aerts, S. (2019). GRNBoost2 and Arboreto: efficient and scalable inference of gene regulatory networks. Bioinformatics *35*, 2159–2161. https://doi.org/10.1093/bioinformatics/bty916.

50. Bastidas-Ponce, A., Tritschler, S., Dony, L., Scheibner, K., Tarquis-Medina, M., Salinno, C., Schirge, S., Burtscher, I., Böttcher, A., Theis, F.J., et al. (2019). Comprehensive single cell mRNA profiling reveals a detailed roadmap for pancreatic endocrinogenesis. Development *146*, dev173849. https://doi.org/10.1242/dev.173849.

51. Klein, D., Palla, G., Lange, M., Klein, M., Piran, Z., Gander, M., Meng-Papaxanthos, L., Sterr, M., Saber, L., Jing, C., Bastidas-Ponce, A., Cota, P., Tarquis-Medina, M., Parikh, S., Gold, I., Lickert, H., Bakhti, M., Nitzan, M., Cuturi, M., and Theis, F.J. (2025). Mapping cells through time and space with moscot. Nature *638*, 1065–1075.

52. Yu, X.-X., Qiu, W.-L., Yang, L., Wang, Y.-C., He, M.-Y., Wang, D., Zhang, Y., Li, L.-C., Zhang, J., Wang, Y., et al. (2021). Sequential progenitor states mark the generation of pancreatic endocrine lineages in mice and humans. Cell Res. *31*, 886–903. https://doi.org/10.1038/s41422-021-00486-w.

53. Bohuslavova, R., Fabriciova, V., Lebrón-Mora, L., Malfatti, J., Smolik, O., Valihrach, L., Benesova, S., Zucha, D., Berkova, Z., Saudek, F., et al. (2023). ISL1 controls pancreatic alpha cell fate and beta cell maturation. Cell Biosci. *13*, 53. https://doi.org/10.1186/s13578-023-01003-9.

54. Baron, M., Veres, A., Wolock, S.L., Faust, A.L., Gaujoux, R., Vetere, A., Ryu, J.H., Wagner, B.K., Shen-Orr, S.S., Klein, A.M., et al. (2016). A Single-Cell Transcriptomic Map of the Human and Mouse Pancreas Reveals Inter- and Intra-cell Population Structure. Cell Syst. *3*, 346–360.e4. https://doi.org/10.1016/j.cels.2016.08.011.

55. Duvall, E., Benitez, C.M., Tellez, K., Enge, M., Pauerstein, P.T., Li, L., Baek, S., Quake, S.R., Smith, J.P., Sheffield, N.C., et al. (2022). Single-cell transcriptome and accessible chromatin dynamics during endocrine pancreas development. Proc. Natl. Acad. Sci. USA *119*, e2201267119. https://doi.org/10.1073/pnas.2201267119.

56. Heller, R.S., Jenny, M., Collombat, P., Mansouri, A., Tomasetto, C., Madsen, O.D., Mellitzer, G., Gradwohl, G., and Serup, P. (2005). Genetic determinants of pancreatic epsilon-cell development. Dev. Biol. *286*, 217–224. https://doi.org/10.1016/j.ydbio.2005.06.041.

57. Shan, B., and Lee, W.H. (1994). Deregulated expression of E2F-1 induces S-phase entry and leads to apoptosis. Mol. Cell. Biol. *14*, 8166–8173. https://doi.org/10.1128/mcb.14.12.8166-8173.1994.

58. Zhu, W., Giangrande, P.H., and Nevins, J.R. (2004). E2Fs link the control of G1/S and G2/M transcription. EMBO J. *23*, 4615–4626. https://doi.org/10.1038/sj.emboj.7600459.

59. Zheng, D., Zhao, K., and Mehler, M.F. (2009). Profiling RE1/REST-mediated histone modifications in the human genome. Genome Biol. *10*, R9. https://doi.org/10.1186/gb-2009-10-1-r9.

60. Grossi, E., Nguyen, C.B., Carcamo, S., Kirigin Callaú, V., Moran, S., Filipescu, D., Tagore, S., Firestone, T.M., Keogh, M.-C., Sun, L., et al. (2025). The SWI/SNF PBAF complex facilitates REST occupancy at repressive chromatin. Mol. Cell *85*, 1714–1729.e7. https://doi.org/10.1016/j.molcel.2025.03.026.

61. Zhang, Y., Xing, Y., Zhang, L., Mei, Y., Yamamoto, K., Mak, T.W., and You, H. (2012). Regulation of cell cycle progression by forkhead transcription factor FOXO3 through its binding partner DNA replication factor Cdt1. Proc. Natl. Acad. Sci. USA *109*, 5717–5722. https://doi.org/10.1073/pnas.1203210109.

62. Tien, A.L., Senbanerjee, S., Kulkarni, A., Mudbhary, R., Goudreau, B., Ganesan, S., Sadler, K.C., and Ukomadu, C. (2011). UHRF1 depletion causes a G2/M arrest, activation of DNA damage response and

apoptosis. Biochem. J. *435*, 175–185. https://doi.org/10.1042/BJ20100840.

63. Rubin, C.I., and Atweh, G.F. (2004). The role of stathmin in the regulation of the cell cycle. J. Cell. Biochem. *93*, 242–250. https://doi.org/10.1002/jcb.20187.

64. Lasorella, A., Iavarone, A., and Israel, M.A. (1996). Id2 specifically alters regulation of the cell cycle by tumor suppressor proteins. Mol. Cell. Biol. *16*, 2570–2578. https://doi.org/10.1128/MCB.16.6.2570.

65. Liang, C., and Stillman, B. (1997). Persistent initiation of DNA replication and chromatin-bound MCM proteins during the cell cycle in cdc6 mutants. Genes Dev. *11*, 3375–3386. https://doi.org/10.1101/gad.11.24.3375.

66. Wu, S.C., and Benavente, C.A. (2018). Chromatin remodeling protein HELLS is upregulated by inactivation of the RB-E2F pathway and is nonessential for osteosarcoma tumorigenesis. Oncotarget *9*, 32580–32592. https://doi.org/10.18632/oncotarget.25953.

67. Li, G., Shen, J., Cheng, W., Wang, X., Wang, D., Song, Y., Chen, Y., Li, X., Zhang, M., Ding, Y., et al. (2024). CENPK orchestrates ovarian cancer progression via GOLPH3-Mediated activation of mTOR signaling. Mol. Cell. Endocrinol. *589*, 112253. https://doi.org/10.1016/j.mce.2024.112253.

68. Collombat, P., Hecksher-Sørensen, J., Krull, J., Berger, J., Riedel, D., Herrera, P.L., Serup, P., and Mansouri, A. (2007). Embryonic endocrine pancreas and mature beta cells acquire alpha and PP cell phenotypes upon Arx misexpression. J. Clin. Investig. *117*, 961–970. https://doi.org/10.1172/JCI29115.

69. Gao, T., McKenna, B., Li, C., Reichert, M., Nguyen, J., Singh, T., Yang, C., Pannikar, A., Doliba, N., Zhang, T., et al. (2014). Pdx1 maintains β cell identity and function by repressing an α cell program. Cell Metab. *19*, 259–271. https://doi.org/10.1016/j.cmet.2013.12.002.

70. Pan, F.C., Brissova, M., Powers, A.C., Pfaff, S., and Wright, C.V.E. (2015). Inactivating the permanent neonatal diabetes gene Mnx1 switches insulin-producing β-cells to a δ-like fate and reveals a facultative proliferative capacity in aged β-cells. Development *142*, 3637–3648. https://doi.org/10.1242/dev.126011.

71. Zhang, J., McKenna, L.B., Bogue, C.W., and Kaestner, K.H. (2014). The diabetes gene Hhex maintains δ-cell differentiation and islet function. Genes Dev. *28*, 829–834. https://doi.org/10.1101/gad.235499.113.

72. Spaeth, J.M., Liu, J.-H., Peters, D., Guo, M., Osipovich, A.B., Mohammadi, F., Roy, N., Bhushan, A., Magnuson, M.A., Hebrok, M., et al. (2019). The Pdx1-Bound Swi/Snf Chromatin Remodeling Complex Regulates Pancreatic Progenitor Cell Proliferation and Mature Islet β-Cell Function. Diabetes *68*, 1806–1818. https://doi.org/10.2337/db19-0349.

73. Cota, P., Saber, L., Taskin, D., Jing, C., Bastidas-Ponce, A., Vanheusden, M., Shahryari, A., Sterr, M., Burtscher, I., Bakhti, M., et al. (2023). NEUROD2 function is dispensable for human pancreatic β cell specification. Front. Endocrinol. *14*, 1286590. https://doi.org/10.3389/fendo.2023.1286590.

74. Ma, Z., Zhang, X., Zhong, W., Yi, H., Chen, X., Zhao, Y., Ma, Y., Song, E., and Xu, T. (2023). Deciphering early human pancreas development at the single-cell level. Nat. Commun. *14*, 5354. https://doi.org/10.1038/s41467-023-40893-8.

75. Soyer, J., Flasse, L., Raffelsberger, W., Beucher, A., Orvain, C., Peers, B., Ravassard, P., Vermot, J., Voz, M.L., Mellitzer, G., et al. (2010). Rfx6 is an Ngn3-dependent winged helix transcription factor required for pancreatic islet cell development. Development *137*, 203–212. https://doi.org/10.1242/dev.041673.

76. Barile, M., Imaz-Rosshandler, I., Inzani, I., Ghazanfar, S., Nichols, J., Marioni, J.C., Guibentif, C., and Göttgens, B. (2021). Coordinated changes in gene expression kinetics underlie both mouse and human erythroid maturation. Genome Biol. *22*, 197. https://doi.org/10.1186/s13059-021-02414-y.

77. Puig, R.R., Boddie, P., Khan, A., Castro-Mondragon, J.A., and Mathelier, A. (2021). UniBind: maps of high-confidence direct TF-DNA interactions across nine species. BMC Genomics *22*, 482. https://doi.org/10.1186/s12864-021-07760-6.

78. Bonfanti, P., Barrandon, Y., and Cossu, G. (2012). "Hearts and bones": the ups and downs of "plasticity" in stem cell biology "Hearts and Bones". EMBO Mol. Med. *4*, 353–361. https://doi.org/10.1002/emmm.201200220.

79. Zhang, B., He, P., Lawrence, J.E.G., Wang, S., Tuck, E., Williams, B.A., Roberts, K., Kleshchevnikov, V., Mamanova, L., Bolt, L., et al. (2024). A human embryonic limb cell atlas resolved in space and time. Nature *635*, 668–678. https://doi.org/10.1038/s41586-023-06806-x.

80. Braun, E., Danan-Gotthold, M., Borm, L.E., Lee, K.W., Vinsland, E., Lönnerberg, P., Hu, L., Li, X., He, X., Andrusivová, Ž., et al. (2023). Comprehensive cell atlas of the first-trimester developing human brain. Science *382*, eadf1226. https://doi.org/10.1126/science.adf1226.

81. Wang, V.Y., and Zoghbi, H.Y. (2001). Genetic regulation of cerebellar development. Nat. Rev. Neurosci. *2*, 484–491. https://doi.org/10.1038/35081558.

82. Yu, X., and Zecevic, N. (2011). Dorsal radial glial cells have the potential to generate cortical interneurons in human but not in mouse brain. J. Neurosci. *31*, 2413–2420. https://doi.org/10.1523/JNEUROSCI.5249-10.2011.

83. Iwasaki, H., Mizuno, S.-I., Wells, R.A., Cantor, A.B., Watanabe, S., and Akashi, K. (2003). GATA-1 converts lymphoid and myelomonocytic progenitors into the megakaryocyte/erythrocyte lineages. Immunity *19*, 451–462. https://doi.org/10.1016/S1074-7613(03)00242-5.

84. Chen, H.M., Zhang, P., Voso, M.T., Hohaus, S., Gonzalez, D.A., Glass, C.K., Zhang, D.E., and Tenen, D.G. (1995). Neutrophils and monocytes express high levels of PU.1 (Spi-1) but not Spi-B. Blood *85*, 2918–2928. https://doi.org/10.1182/blood.V85.10.2918.bloodjournal85102918.

85. Wanet, A., Bassal, M.A., Patel, S.B., Marchi, F., Mariani, S.A., Ahmed, N., Zhang, H., Borchiellini, M., Chen, S., Zhang, J., et al. (2021). E-cadherin is regulated by GATA-2 and marks the early commitment of mouse hematopoietic progenitors to the basophil and mast cell fates. Sci. Immunol. *6*, eaba0178. https://doi.org/10.1126/sciimmunol.aba0178.

86. Gekas, C., Rhodes, K.E., Gereige, L.M., Helgadottir, H., Ferrari, R., Kurdistani, S.K., Montecino-Rodriguez, E., Bassel-Duby, R., Olson, E., Krivtsov, A.V., et al. (2009). Mef2C is a lineage-restricted target of Scl/Tal1 and regulates megakaryopoiesis and B-cell homeostasis. Blood *113*, 3461–3471. https://doi.org/10.1182/blood-2008-07-167577.

87. Chen, L., Kostadima, M., Martens, J.H.A., Canu, G., Garcia, S.P., Turro, E., Downes, K., Macaulay, I.C., Bielczyk-Maczynska, E., Coe, S., et al. (2014). Transcriptional diversity during lineage commitment of human blood progenitors. Science *345*, 1251033. https://doi.org/10.1126/science.1251033.

88. Villar, J., Cros, A., De Juan, A., Alaoui, L., Bonte, P.-E., Lau, C.M., Tiniakou, I., Reizis, B., and Segura, E. (2023). ETV3 and ETV6 enable monocyte differentiation into dendritic cells by repressing macrophage fate commitment. Nat. Immunol. *24*, 84–95. https://doi.org/10.1038/s41590-022-01374-0.

89. Zhang, P., Zhang, X., Iwama, A., Yu, C., Smith, K.A., Mueller, B.U., Narravula, S., Torbett, B.E., Orkin, S.H., and Tenen, D.G. (2000). PU.1 inhibits GATA-1 function and erythroid differentiation by blocking GATA-1 DNA binding. Blood *96*, 2641–2648.

90. Nerlov, C., Querfurth, E., Kulessa, H., and Graf, T. (2000). GATA-1 interacts with the myeloid PU.1 transcription factor and represses PU.1-dependent transcription. Blood *95*, 2543–2551.

91. Horak, C.E., Mahajan, M.C., Luscombe, N.M., Gerstein, M., Weissman, S.M., and Snyder, M. (2002). GATA-1 binding sites mapped in the beta-globin locus by using mammalian chIp-chip analysis. Proc. Natl. Acad. Sci. USA *99*, 2924–2929. https://doi.org/10.1073/pnas.052706999.

92. Aliee, H., Richter, T., Solonin, M., Ibarra, I., Theis, F., and Kilbertus, N. (2022). Sparsity in continuous-depth neural networks. Preprint at arXiv. https://doi.org/10.48550/ARXIV.2210.14672.

93. Drissen, R., Buza-Vidas, N., Woll, P., Thongjuea, S., Gambardella, A., Giustacchini, A., Mancini, E., Zriwil, A., Lutteropp, M., Grover, A., et al. (2016). Distinct myeloid progenitor-differentiation pathways identified through single-cell RNA sequencing. Nat. Immunol. *17*, 666–676. https://doi.org/10.1038/ni.3412.

94. Chen, Y., Hu, Z., Ni, K., Feng, S., Chen, W., and Huang, H. (2024). Inferring single-cell RNA kinetics from various biological priors. Preprint at bioRxiv. https://doi.org/10.1101/2024.05.21.595179.

95. Kriegstein, A., and Alvarez-Buylla, A. (2009). The glial nature of embryonic and adult neural stem cells. Annu. Rev. Neurosci. *32*, 149–184. https://doi.org/10.1146/annurev.neuro.051508.135600.

96. Edens, B.M., Stundl, J., Urrutia, H.A., and Bronner, M.E. (2024). Neural crest origin of sympathetic neurons at the dawn of vertebrates. Nature *629*, 121–126. https://doi.org/10.1038/s41586-024-07297-0.

97. Kim, S., Morgunova, E., Naqvi, S., Goovaerts, S., Bader, M., Koska, M., Popov, A., Luong, C., Pogson, A., Swigut, T., et al. (2024). DNA-guided transcription factor cooperativity shapes face and limb mesenchyme. Cell *187*, 692–711.e26. https://doi.org/10.1016/j.cell.2023.12.032.

98. Hu, Z., and Sauka-Spengler, T. (2022). Cellular plasticity in the neural crest and cancer. Curr. Opin. Genet. Dev. *75*, 101928. https://doi.org/10.1016/j.gde.2022.101928.

99. Hagemann-Jensen, M., Ziegenhain, C., Chen, P., Ramsköld, D., Hendriks, G.-J., Larsson, A.J.M., Faridani, O.R., and Sandberg, R. (2020). Single-cell RNA counting at allele and isoform resolution using Smart-seq3. Nat. Biotechnol. *38*, 708–714. https://doi.org/10.1038/s41587-020-0497-0.

100. Hu, Z., Mayes, S., Wang, W., Santos-Pereira, J.M., Theis, F., and Sauka-Spengler, T. (2024). Single-cell multi-omics, spatial transcriptomics and systematic perturbation decode circuitry of neural crest fate decisions. Preprint at bioRxiv. https://doi.org/10.1101/2024.09.17.613303.

101. Rocha, M., Singh, N., Ahsan, K., Beiriger, A., and Prince, V.E. (2020). Neural crest development: insights from the zebrafish. Dev. Dyn. *249*, 88–111. https://doi.org/10.1002/dvdy.122.

102. Opdecamp, K., Nakayama, A., Nguyen, M.T., Hodgkinson, C.A., Pavan, W.J., and Arnheiter, H. (1997). Melanocyte development in vivo and in neural crest cell cultures: crucial dependence on the Mitf basic-helix-loop-helix-zipper transcription factor. Development *124*, 2377–2386. https://doi.org/10.1242/dev.124.12.2377.

103. Aoki, Y., Saint-Germain, N., Gyda, M., Magner-Fink, E., Lee, Y.-H., Credidio, C., and Saint-Jeannet, J.-P. (2003). Sox10 regulates the development of neural crest-derived melanocytes in Xenopus. Dev. Biol. *259*, 19–33. https://doi.org/10.1016/S0012-1606(03)00161-1.

104. Spokony, R.F., Aoki, Y., Saint-Germain, N., Magner-Fink, E., and Saint-Jeannet, J.-P. (2002). The transcription factor Sox9 is required for cranial neural crest development in Xenopus. Development *129*, 421–432. https://doi.org/10.1242/dev.129.2.421.

105. Okeke, C., Paulding, D., Riedel, A., Paudel, S., Phelan, C., Teng, C.S., and Barske, L. (2022). Control of cranial ectomesenchyme fate by Nr2f nuclear receptors. Development *149*, dev201133. https://doi.org/10.1242/dev.201133.

106. Soo, K., O'Rourke, M.P., Khoo, P.-L., Steiner, K.A., Wong, N., Behringer, R.R., and Tam, P.P.L. (2002). Twist function is required for the morphogenesis of the cephalic neural tube and the differentiation of the cranial neural crest cells in the mouse embryo. Dev. Biol. *247*, 251–270. https://doi.org/10.1006/dbio.2002.0699.

107. Wang, C., Kam, R.K.T., Shi, W., Xia, Y., Chen, X., Cao, Y., Sun, J., Du, Y., Lu, G., Chen, Z., et al. (2015). The proto-oncogene transcription factor Ets1 regulates neural crest development through histone deacetylase 1 to mediate output of bone morphogenetic protein signaling. J. Biol. Chem. *290*, 21925–21938. https://doi.org/10.1074/jbc.M115.644864.

108. Yan, Y.-L., Willoughby, J., Liu, D., Crump, J.G., Wilson, C., Miller, C.T., Singer, A., Kimmel, C., Westerfield, M., and Postlethwait, J.H. (2005). A pair of Sox: distinct and overlapping functions of zebrafish sox9 co-orthologs in craniofacial and pectoral fin development. Development *132*, 1069–1083. https://doi.org/10.1242/dev.01674.

109. Das, A., and Crump, J.G. (2012). Bmps and id2a act upstream of Twist1 to restrict ectomesenchyme potential of the cranial neural crest. PLOS Genet. *8*, e1002710. https://doi.org/10.1371/journal.pgen.1002710.

110. Badia-I-Mompel, P., Vélez Santiago, J., Braunger, J., Geiss, C., Dimitrov, D., Müller-Dott, S., Taus, P., Dugourd, A., Holland, C.H., Ramirez Flores, R.O., et al. (2022). decoupleR: ensemble of computational methods to infer biological activities from omics data. Bioinform. Adv. *2*, vbac016. https://doi.org/10.1093/bioadv/vbac016.

111. Bhadra, J., and Iovine, M.K. (2015). Hsp47 mediates Cx43-dependent skeletal growth and patterning in the regenerating fin. Mech. Dev. *138*, 364–374. https://doi.org/10.1016/j.mod.2015.06.004.

112. Choudhry, P., Joshi, D., Funke, B., and Trede, N. (2011). Alcama mediates Edn1 signaling during zebrafish cartilage morphogenesis. Dev. Biol. *349*, 483–493. https://doi.org/10.1016/j.ydbio.2010.11.006.

113. Hill, C.R., Jacobs, B.H., Brown, C.B., Barnett, J.V., and Goudy, S.L. (2015). Type III transforming growth factor beta receptor regulates vascular and osteoblast development during palatogenesis. Dev. Dyn. *244*, 122–133. https://doi.org/10.1002/dvdy.24225.

114. Zhang, H., Shen, S., Qian, Y., Zhang, L., Pu, J., Zhou, X., Yang, N., and Wang, W. (2025). Role of TGF-β3 in Regulating Neural Crest Cell Fate and Craniofacial Development: Insights From Zebrafish Models. Int. Dent. J. *75*, 103874. https://doi.org/10.1016/j.identj.2025.103874.

115. Yu, H.-H., and Moens, C.B. (2005). Semaphorin signaling guides cranial neural crest cell migration in zebrafish. Dev. Biol. *280*, 373–385. https://doi.org/10.1016/j.ydbio.2005.01.029.

116. Schwarz, Q., Vieira, J.M., Howard, B., Eickholt, B.J., and Ruhrberg, C. (2008). Neuropilin 1 and 2 control cranial gangliogenesis and axon guidance through neural crest cells. Development *135*, 1605–1613. https://doi.org/10.1242/dev.015412.

117. Johnson, S.L., Nguyen, A.N., and Lister, J.A. (2011). mitfa is required at multiple stages of melanocyte differentiation but not to establish the melanocyte stem cell. Dev. Biol. *350*, 405–413. https://doi.org/10.1016/j.ydbio.2010.12.004.

118. Finzsch, M., Schreiner, S., Kichko, T., Reeh, P., Tamm, E.R., Bösl, M.R., Meijer, D., and Wegner, M. (2010). Sox10 is required for Schwann cell identity and progression beyond the immature Schwann cell stage. J. Cell Biol. *189*, 701–712. https://doi.org/10.1083/jcb.200912142.

119. Zou, M., Li, S., Klein, W.H., and Xiang, M. (2012). Brn3a/Pou4f1 regulates dorsal root ganglion sensory neuron specification and axonal projection into the spinal cord. Dev. Biol. *364*, 114–127. https://doi.org/10.1016/j.ydbio.2012.01.021.

120. Liu, M., Pereira, F.A., Price, S.D., Chu, M.J., Shope, C., Himes, D., Eatock, R.A., Brownell, W.E., Lysakowski, A., and Tsai, M.J. (2000). Essential role of BETA2/NeuroD1 in development of the vestibular and auditory systems. Genes Dev. *14*, 2839–2854. https://doi.org/10.1101/gad.840500.

121. Filova, I., Bohuslavova, R., Tavakoli, M., Yamoah, E.N., Fritzsch, B., and Pavlinkova, G. (2022). Early Deletion of Neurod1 Alters Neuronal Lineage Potential and Diminishes Neurogenesis in the Inner Ear. Front. Cell Dev. Biol. *10*, 845461. https://doi.org/10.3389/fcell.2022.845461.

122. Replogle, J.M., Norman, T.M., Xu, A., Hussmann, J.A., Chen, J., Cogan, J.Z., Meer, E.J., Terry, J.M., Riordan, D.P., Srinivas, N., et al. (2020). Combinatorial single-cell CRISPR screens by direct guide RNA capture and targeted sequencing. Nat. Biotechnol. *38*, 954–961. https://doi.org/10.1038/s41587-020-0470-y.

123. Pelea, O., Mayes, S., Ferry, Q.R.V., Fulga, T.A., and Sauka-Spengler, T. (2024). Specific Modulation of CRISPR Transcriptional Activators

through RNA-Sensing Guide RNAs in Mammalian Cells and Zebrafish Embryos. eLife *12*. https://doi.org/10.7554/eLife.87722.2.

124. Saunders, L.M., Srivatsan, S.R., Duran, M., Dorrity, M.W., Ewing, B., Linbo, T.H., Shendure, J., Raible, D.W., Moens, C.B., Kimelman, D., et al. (2023). Embryo-scale reverse genetics at single-cell resolution. Nature *623*, 782–791. https://doi.org/10.1038/s41586-023-06720-2.

125. Hochgreb-Hägele, T., and Bronner, M.E. (2013). A novel FoxD3 gene trap line reveals neural crest precursor movement and a role for FoxD3 in their specification. Dev. Biol. *374*, 1–11. https://doi.org/10.1016/j.ydbio.2012.11.035.

126. Burkhardt, D.B., Stanley, J.S., 3rd, Tong, A., Perdigoto, A.L., Gigante, S.A., Herold, K.C., Wolf, G., Giraldez, A.J., van Dijk, D., and Krishnaswamy, S. (2021). Quantifying the effect of experimental perturbations at single-cell resolution. Nat. Biotechnol. *39*, 619–629. https://doi.org/10.1038/s41587-020-00803-5.

127. Barske, L., Rataud, P., Behizad, K., Del Rio, L., Cox, S.G., and Crump, J.G. (2018). Essential role of Nr2f nuclear receptors in patterning the vertebrate upper jaw. Dev. Cell *44*, 337–347.e5. https://doi.org/10.1016/j.devcel.2017.12.022.

128. Linville, A., Radtke, K., Waxman, J.S., Yelon, D., and Schilling, T.F. (2009). Combinatorial roles for zebrafish retinoic acid receptors in the hindbrain, limbs and pharyngeal arches. Dev. Biol. *325*, 60–70. https://doi.org/10.1016/j.ydbio.2008.09.022.

129. Saunders, L.M., Mishra, A.K., Aman, A.J., Lewis, V.M., Toomey, M.B., Packer, J.S., Qiu, X., McFaline-Figueroa, J.L., Corbo, J.C., Trapnell, C., et al. (2019). Thyroid hormone regulates distinct paths to maturation in pigment cell lineages. eLife *8*, e45181. https://doi.org/10.7554/eLife.45181.

130. Howard, A.G.A., IV., Baker, P.A., Ibarra-García-Padilla, R., Moore, J.A., Rivas, L.J., Tallman, J.J., Singleton, E.W., Westheimer, J.L., Corteguera, J.A., Uribe, R.A., et al. (2021). An atlas of neural crest lineages along the posterior developing zebrafish at single-cell resolution. eLife *10*, e60005. https://doi.org/10.7554/eLife.60005.

131. Tatarakis, D., Cang, Z., Wu, X., Sharma, P.P., Karikomi, M., MacLean, A.L., Nie, Q., and Schilling, T.F. (2021). Single-cell transcriptomic analysis of zebrafish cranial neural crest reveals spatiotemporal regulation of lineage decisions during development. Cell Rep. *37*, 110140. https://doi.org/10.1016/j.celrep.2021.110140.

132. Dooley, C.M., Schwarz, H., Mueller, K.P., Mongera, A., Konantz, M., Neuhauss, S.C.F., Nüsslein-Volhard, C., and Geisler, R. (2013). Slc45a2 and V-ATPase are regulators of melanosomal pH homeostasis in zebrafish, providing a mechanism for human pigment evolution and disease. Pigment Cell Melanoma Res. *26*, 205–217. https://doi.org/10.1111/pcmr.12053.

133. Bajpai, V.K., Swigut, T., Mohammed, J., Naqvi, S., Arreola, M., Tycko, J., Kim, T.C., Pritchard, J.K., Bassik, M.C., and Wysocka, J. (2023). A genome-wide genetic screen uncovers determinants of human pigmentation. Science *381*, eade6289. https://doi.org/10.1126/science.ade6289.

134. Petratou, K. (2016) Investigating Gene Regulatory Networks Underlying Zebrafish Pigment Cell Development.

135. Pampari, A., Shcherbina, A., Kvon, E.Z., Kosicki, M., Nair, S., Kundu, S., Kathiria, A.S., Risca, V.I., Kuningas, K., Alasoo, K., et al. (2025). ChromBPNet: bias factorized, base-resolution deep learning models of chromatin accessibility reveal cis-regulatory sequence syntax, transcription factor footprints and regulatory variants. Preprint at bioRxiv, 2024.12.25.630221. https://doi.org/10.1101/2024.12.25.630221.

136. Lukoseviciute, M., Ling, I.T.C., Senanayake, U., Candido-Ferreira, I., Taylor, G., Williams, R.M., and Sauka-Spengler, T. (2020). Tissue-specific in vivo biotin chromatin immunoprecipitation with sequencing in zebrafish and chicken. Star Protoc. *1*, 100066. https://doi.org/10.1016/j.xpro.2020.100066.

137. Lukoseviciute, M., Gavriouchkina, D., Williams, R.M., Hochgreb-Hagele, T., Senanayake, U., Chong-Morrison, V., Thongjuea, S., Repapi, E., Mead, A., and Sauka-Spengler, T. (2018). From Pioneer to Repressor: Bimodal foxd3 Activity Dynamically Remodels Neural Crest Regulatory Landscape In Vivo. Dev. Cell *47*, 608–628.e6. https://doi.org/10.1016/j.devcel.2018.11.009.

138. Li, Q., Brown, J.B., Huang, H., and Bickel, P.J. (2011). Measuring reproducibility of high-throughput experiments. Ann. Appl. Stat. *5*, 1752–1779. https://doi.org/10.1214/11-AOAS466.

139. Grant, C.E., Bailey, T.L., and Noble, W.S. (2011). FIMO: scanning for occurrences of a given motif. Bioinformatics *27*, 1017–1018. https://doi.org/10.1093/bioinformatics/btr064.

140. Ebrahimpoor, M., Spitali, P., Hettne, K., Tsonaka, R., and Goeman, J. (2020). Simultaneous enrichment analysis of all possible gene-sets: Unifying self-contained and competitive methods. Brief. Bioinform. *21*, 1302–1312. https://doi.org/10.1093/bib/bbz074.

141. Bunne, C., Roohani, Y., Rosen, Y., Gupta, A., Zhang, X., Roed, M., Alexandrov, T., AlQuraishi, M., Brennan, P., Burkhardt, D.B., et al. (2024). How to build the virtual cell with artificial intelligence: Priorities and opportunities. Cell *187*, 7045–7063.

142. Madison, B.J., Clark, K.A., Bhachech, N., Hollenhorst, P.C., Graves, B.J., and Currie, S.L. (2018). Electrostatic repulsion causes anticooperative DNA binding between tumor suppressor ETS transcription factors and JUN-FOS at composite DNA sites. J. Biol. Chem. *293*, 18624–18635. https://doi.org/10.1074/jbc.RA118.003352.

143. Aiello-Couzo, N.M., and Kang, Y. (2020). A bridge between melanoma cell states. Nat. Cell Biol. *22*, 913–914. https://doi.org/10.1038/s41556-020-0556-2.

144. Kulesa, P.M., Kasemeier-Kulesa, J.C., Teddy, J.M., Margaryan, N.V., Seftor, E.A., Seftor, R.E.B., and Hendrix, M.J.C. (2006). Reprogramming metastatic melanoma cells to assume a neural crest cell-like phenotype in an embryonic microenvironment. Proc. Natl. Acad. Sci. USA *103*, 3752–3757. https://doi.org/10.1073/pnas.0506977103.

145. Chen, Z., King, W.C., Hwang, A., Gerstein, M., and Zhang, J. (2022). DeepVelo: Single-cell transcriptomic deep velocity field learning with neural ordinary differential equations. Sci. Adv. *8*, eabq3745. https://doi.org/10.1126/sciadv.abq3745.

146. Singh, R., Wu, A.P., Mudide, A., and Berger, B. (2024). Causal gene regulatory analysis with RNA velocity reveals an interplay between slow and fast transcription factors. Cell Syst. *15*, 462–474.e5. https://doi.org/10.1016/j.cels.2024.04.005.

147. Lotfollahi, M., Wolf, F.A., and Theis, F.J. (2019). scGen predicts single-cell perturbation responses. Nat. Methods *16*, 715–721. https://doi.org/10.1038/s41592-019-0494-8.

148. Kirschenbaum, D., Xie, K., Ingelfinger, F., Katzenelenbogen, Y., Abadie, K., Look, T., Sheban, F., Phan, T.S., Li, B., Zwicky, P., et al. (2024). Time-resolved single-cell transcriptomics defines immune trajectories in glioblastoma. Cell *187*, 149–165.e23. https://doi.org/10.1016/j.cell.2023.11.032.

149. Ramanathan, M., Porter, D.F., and Khavari, P.A. (2019). Methods to study RNA-protein interactions. Nat. Methods *16*, 225–234. https://doi.org/10.1038/s41592-019-0330-1.

150. Brannan, K.W., Chaim, I.A., Marina, R.J., Yee, B.A., Kofman, E.R., Lorenz, D.A., Jagannatha, P., Dong, K.D., Madrigal, A.A., Underwood, J.G., et al. (2021). Robust single-cell discovery of RNA targets of RNA-binding proteins and ribosomes. Nat. Methods *18*, 507–519. https://doi.org/10.1038/s41592-021-01128-0.

151. Ozadam, H., Tonn, T., Han, C.M., Segura, A., Hoskins, I., Rao, S., Ghatpande, V., Tran, D., Catoe, D., Salit, M., et al. (2023). Single-cell quantification of ribosome occupancy in early mouse development. Nature *618*, 1057–1064. https://doi.org/10.1038/s41586-023-06228-9.

152. Kwon, Y., Won, J.-H., Kim, B.J., and Paik, M.C. (2020). Uncertainty quantification using Bayesian neural networks in classification: Application to biomedical image segmentation. Comput. Stat. Data Anal. *142*, 106816. https://doi.org/10.1016/j.csda.2019.106816.

153. Williams, R.M., Taylor, G., Ling, I.T.C., Candido-Ferreira, I., Fountain, D.M., Mayes, S., Ateş-Kalkan, P.S., Haug, J.O., Price, A.J., McKinney, S.A., et al. (2024). Chromatin remodeller Chd7 is developmentally regulated in the neural crest by tissue-specific transcription factors. PLOS Biol. *22*, e3002786. https://doi.org/10.1371/journal.pbio.3002786.

154. Trinh, L.A., Chong-Morrison, V., Gavriouchkina, D., Hochgreb-Hägele, T., Senanayake, U., Fraser, S.E., and Sauka-Spengler, T. (2017). Biotagging of specific cell populations in zebrafish reveals gene regulatory logic encoded in the nuclear transcriptome. Cell Rep. *19*, 425–440. https://doi.org/10.1016/j.celrep.2017.03.045.

155. Bocci, F., Zhou, P., and Nie, Q. (2022). spliceJAC: transition genes and state-specific gene regulation from single-cell transcriptome data. Mol. Syst. Biol. *18*, e11176. https://doi.org/10.15252/msb.202211176.

156. Li, S., Zhang, P., Chen, W., Ye, L., Brannan, K.W., Le, N.-T., Abe, J.-I., Cooke, J.P., and Wang, G. (2024). A relay velocity model infers cell-dependent RNA velocity. Nat. Biotechnol. *42*, 99–108. https://doi.org/10.1038/s41587-023-01728-5.

157. Van de Sande, B., Flerin, C., Davie, K., De Waegeneer, M., Hulselmans, G., Aibar, S., Seurinck, R., Saelens, W., Cannoodt, R., Rouchon, Q., et al. (2020). A scalable SCENIC workflow for single-cell gene regulatory network analysis. Nat. Protoc. *15*, 2247–2276. https://doi.org/10.1038/s41596-020-0336-2.

158. Wolf, F.A., Angerer, P., and Theis, F.J. (2018). SCANPY: large-scale single-cell gene expression data analysis. Genome Biol. *19*, 15. https://doi.org/10.1186/s13059-017-1382-0.

159. Gayoso, A., Lopez, R., Xing, G., Boyeau, P., Valiollah Pour Amiri, V., Hong, J., Wu, K., Jayasuriya, M., Mehlman, E., Langevin, M., et al. (2022). A Python library for probabilistic analysis of single-cell omics data. Nat Biotechnol *40*, 163–166.

160. Parekh, S., Ziegenhain, C., Vieth, B., Enard, W., and Hellmann, I. (2018). zUMIs - A fast and flexible pipeline to process RNA sequencing data with UMIs. GigaScience *7*, giy059. https://doi.org/10.1093/gigascience/giy059.

161. Langmead, B., and Salzberg, S.L. (2012). Fast gapped-read alignment with Bowtie 2. Nat. Methods *9*, 357–359. https://doi.org/10.1038/nmeth.1923.

162. Gaspar, J.M. (2018). Improved peak-calling with MACS2. Preprint at bioRxiv. https://doi.org/10.1101/496521.

163. Zhang, Y., Liu, T., Meyer, C.A., Eeckhoute, J., Johnson, D.S., Bernstein, B.E., Nusbaum, C., Myers, R.M., Brown, M., Li, W., et al. (2008). Model-based analysis of ChIP-Seq (MACS). Genome Biol. *9*, R137. https://doi.org/10.1186/gb-2008-9-9-r137.

164. Pachitariu, M., and Stringer, C. (2022). Cellpose 2.0: how to train your own model. Nat. Methods *19*, 1634–1641. https://doi.org/10.1038/s41592-022-01663-4.

165. Westerfield, M. (2000). The Zebrafish Book: A Guide for the Laboratory Use of Zebrafish (Danio rerio) (University of Oregon Press).

166. Kimmel, C.B., Ballard, W.W., Kimmel, S.R., Ullmann, B., and Schilling, T.F. (1995). Stages of embryonic development of the zebrafish. Dev. Dyn. *203*, 253–310. https://doi.org/10.1002/aja.1002030302.

167. Choi, H.M.T., Schwarzkopf, M., Fornace, M.E., Acharya, A., Artavanis, G., Stegmaier, J., Cunha, A., and Pierce, N.A. (2018). Third-generation in situ hybridization chain reaction: multiplexed, quantitative, sensitive, versatile, robust. Development *145*, dev165753. https://doi.org/10.1242/dev.165753.

168. Stringer, C., Wang, T., Michaelos, M., and Pachitariu, M. (2021). Cellpose: a generalist algorithm for cellular segmentation. Nat. Methods *18*, 100–106. https://doi.org/10.1038/s41592-020-01018-x.

169. Malicki, J., Schier, A.F., Solnica-Krezel, L., Stemple, D.L., Neuhauss, S.C., Stainier, D.Y., Abdelilah, S., Rangini, Z., Zwartkruis, F., and Driever, W. (1996). Mutations affecting development of the zebrafish ear. Development *123*, 275–283. https://doi.org/10.1242/dev.123.1.275.

170. Gutierrez-Triana, J.A., Tavhelidse, T., Thumberger, T., Thomas, I., Wittbrodt, B., Kellner, T., Anlas, K., Tsingos, E., and Wittbrodt, J. (2018). Efficient single-copy HDR by 5′ modified long dsDNA donors. eLife *7*, e39468. https://doi.org/10.7554/eLife.39468.

171. Zeisel, A., Köstler, W.J., Molotski, N., Tsai, J.M., Krauthgamer, R., Jacob-Hirsch, J., Rechavi, G., Soen, Y., Jung, S., Yarden, Y., et al. (2011). Coupled pre-mRNA and mRNA dynamics unveil operational strategies underlying transcriptional responses to stimuli. Mol. Syst. Biol. *7*, 529. https://doi.org/10.1038/msb.2011.62.

172. Yu, L., Wei, Y., Duan, J., Schmitz, D.A., Sakurai, M., Wang, L., Wang, K., Zhao, S., Hon, G.C., and Wu, J. (2021). Blastocyst-like structures generated from human pluripotent stem cells. Nature *591*, 620–626. https://doi.org/10.1038/s41586-021-03356-y.

173. Lienen, M., and Günnemann, S. (2022). Torchode: A Parallel ODE Solver for PyTorch. Preprint at arXiv. https://doi.org/10.48550/ARXIV.2210.12375.

174. Finlay, C., Jacobsen, J.-H., Nurbekyan, L., and Oberman, A.M. (2020). How to Train Your Neural ODE: the World of Jacobian and Kinetic Regularization. Preprint at arXiv. https://doi.org/10.48550/ARXIV.2002.02798.

175. Loshchilov, I., and Hutter, F. (2017). Decoupled Weight Decay Regularization. Preprint at arXiv. https://doi.org/10.48550/ARXIV.1711.05101.

176. Burdziak, C., Zhao, C.J., Haviv, D., Alonso-Curbelo, D., Lowe, S.W., and Pe'er, D. (2023). scKINETICS: inference of regulatory velocity with single-cell transcriptomics data. Bioinformatics *39*, i394–i403. https://doi.org/10.1093/bioinformatics/btad267.

177. Bonnaffé, W., and Coulson, T. (2023). Fast fitting of neural ordinary differential equations by Bayesian neural gradient matching to infer ecological interactions from time-series data. Methods Ecol. Evol. *14*, 1543–1563. https://doi.org/10.1111/2041-210X.14121.

178. Shapiro, A. (2003). Monte Carlo Sampling Methods. In Handbooks in Operations Research and Management Science Handbooks in Operations Research and Management Science (Elsevier), pp. 353–425. https://doi.org/10.1016/S0927-0507(03)10006-0.

179. Skok Gibbs, C., Mahmood, O., Bonneau, R., and Cho, K. (2024). PMF-GRN: a variational inference approach to single-cell gene regulatory network inference using probabilistic matrix factorization. Genome Biol. *25*, 88. https://doi.org/10.1186/s13059-024-03226-6.

180. Willard Gibbs, J. (2014). Elementary Principles in Statistical Mechanics (Courier Corporation).

181. Likharev, K.K. (2019). Statistical Mechanics: Lecture Notes. Lecture Notes7 (Institute of Physics Publishing).

182. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Müller, A., Nothman, J., Louppe, G., et al. (2012) Scikit-Learn: Machine Learning in Python.

183. Blei, D.M., Kucukelbir, A., and McAuliffe, J.D. (2017). Variational inference: A review for statisticians. J. Am. Stat. Assoc. *112*, 859–877. https://doi.org/10.1080/01621459.2017.1285773.

184. Ocone, A., Haghverdi, L., Mueller, N.S., and Theis, F.J. (2015). Reconstructing gene regulatory dynamics from high-dimensional single-cell snapshot data. Bioinformatics *31*, i89–i96. https://doi.org/10.1093/bioinformatics/btv257.

185. Li, X., Wong, T.-K.L., Chen, R.T.Q., and Duvenaud, D. (2020). Scalable Gradients for Stochastic Differential Equations. Preprint at arXiv. https://doi.org/10.48550/ARXIV.2001.01328.

186. Kloeden, P.E., and Platen, E. (2011). Numerical Solution of Stochastic Differential Equations (Springer Science & Business Media).

187. Mouse Genome Sequencing Consortium, Waterston, R.H., Lindblad-Toh, K., Birney, E., Rogers, J., Abril, J.F., Agarwal, P., Agarwala, R., Ainscough, R., Alexandersson, M., and et al.. (2002). Initial sequencing and comparative analysis of the mouse genome. Nature *420*, 520–562. https://doi.org/10.1038/nature01262.

188. Shen, W.-K., Chen, S.-Y., Gan, Z.-Q., Zhang, Y.-Z., Yue, T., Chen, M.-M., Xue, Y., Hu, H., and Guo, A.-Y. (2023). AnimalTFDB 4.0: a comprehensive animal transcription factor database updated with variation and expression annotations. Nucleic Acids Res. *51*, D39–D45. https://doi.org/10.1093/nar/gkac907.

189. Reuter, B., Fackeldey, K., and Weber, M. (2019). Generalized Markov modeling of nonreversible molecular kinetics. J. Chem. Phys. *150*, 174103. https://doi.org/10.1063/1.5064530.

190. Yang, T., and Li, W.V. (2025). Generalized Probabilistic Canonical Correlation Analysis for Multi-modal Data Integration with Full or Partial Observations. Preprint at ArXiv.

191. ENCODE Project, Leja, D.L., and Birney, E. (2012) The ENCODE Project Encyclopedia of DNA Elements.

192. Zou, Z., Ohta, T., and Oki, S. (2024). ChIP-Atlas 3.0: a data-mining suite to explore chromosome architecture together with large-scale regulome data. Nucleic Acids Res. *52*, W45–W53. https://doi.org/10.1093/nar/gkae358.

193. Xu, H., Baroukh, C., Dannenfelser, R., Chen, E.Y., Tan, C.M., Kou, Y., Kim, Y.E., Lemischka, I.R., and Ma'ayan, A. (2013). ESCAPE: database for integrating high-content published data collected from human and mouse embryonic stem cellsDatabase. Database (Oxford) *2013*, bat045. https://doi.org/10.1093/database/bat045.

194. Kastriti, M.E., Faure, L., Von Ahsen, D., Bouderlique, T.G., Boström, J., Solovieva, T., Jackson, C., Bronner, M., Meijer, D., Hadjab, S., et al. (2022). Schwann cell precursors represent a neural crest-like state with biased multipotency. EMBO J. *41*, e108780. https://doi.org/10.15252/embj.2021108780.

195. Qiu, C., Martin, B.K., Welsh, I.C., Daza, R.M., Le, T.-M., Huang, X., Nichols, E.K., Taylor, M.L., Fulton, O., O'Day, D.R., et al. (2024). A single-cell time-lapse of mouse prenatal development from gastrula to birth. Nature *626*, 1084–1093. https://doi.org/10.1038/s41586-024-07069-w.

196. Li, H., Handsaker, B., Wysoker, A., Fennell, T., Ruan, J., Homer, N., Marth, G., Abecasis, G., and Durbin, R.; 1000 Genome Project Data Processing Subgroup (2009). The Sequence Alignment/Map format and SAMtools. Bioinformatics *25*, 2078–2079. https://doi.org/10.1093/bioinformatics/btp352.

197. Oki, S., Ohta, T., Shioi, G., Hatanaka, H., Ogasawara, O., Okuda, Y., Kawaji, H., Nakaki, R., Sese, J., and Meno, C. (2018). ChIP-Atlas: a data-mining suite powered by full integration of public ChIP-seq data. EMBO Rep. *19*, e46255. https://doi.org/10.15252/embr.201846255.

198. Battich, N., Beumer, J., de Barbanson, B., Krenning, L., Baron, C.S., Tanenbaum, M.E., Clevers, H., and van Oudenaarden, A. (2020). Sequencing metabolically labeled transcripts in single cells reveals mRNA turnover strategies. Science *367*, 1151–1156. https://doi.org/10.1126/science.aax3072.

199. Schep, A.N., Wu, B., Buenrostro, J.D., and Greenleaf, W.J. (2017). chromVAR: inferring transcription-factor-associated accessibility from single-cell epigenomic data. Nat. Methods *14*, 975–978. https://doi.org/10.1038/nmeth.4401.

200. Chen, T., and Guestrin, C. (2016). XGBoost. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (ACM), pp. 785–794. https://doi.org/10.1145/2939672.2939785.

201. Stuart, T., Srivastava, A., Madad, S., Lareau, C.A., and Satija, R. (2021). Single-cell chromatin state analysis with Signac. Nat. Methods *18*, 1333–1341. https://doi.org/10.1038/s41592-021-01282-5.

202. Yu, G., Wang, L.-G., Han, Y., and He, Q.-Y. (2012). clusterProfiler: an R package for comparing biological themes among gene clusters. Omics *16*, 284–287. https://doi.org/10.1089/omi.2011.0118.

203. Korotkevich, G., Sukhov, V., Budin, N., Shpak, B., Artyomov, M.N., and Sergushichev, A. (2016). Fast gene set enrichment analysis. Preprint at bioRxiv. https://doi.org/10.1101/060012.

204. Hu, C., Li, T., Xu, Y., Zhang, X., Li, F., Bai, J., Chen, J., Jiang, W., Yang, K., Ou, Q., et al. (2023). CellMarker 2.0: an updated database of manually curated cell markers in human/mouse and web tools based on scRNA-seq data. Nucleic Acids Res. *51*, D870–D876. https://doi.org/10.1093/nar/gkac947.

205. Merryweather-Clarke, A.T., Atzberger, A., Soneji, S., Gray, N., Clark, K., Waugh, C., McGowan, S.J., Taylor, S., Nandi, A.K., Wood, W.G., et al. (2011). Global gene expression analysis of human erythroid progenitors. Blood *117*, e96–e108. https://doi.org/10.1182/blood-2010-07-290825.

206. Dobin, A., Davis, C.A., Schlesinger, F., Drenkow, J., Zaleski, C., Jha, S., Batut, P., Chaisson, M., and Gingeras, T.R. (2012). STAR: ultrafast universal RNA-seq aligner. Bioinformatics *29*, 15–21. https://doi.org/10.1093/bioinformatics/bts635.

207. Hao, Y., Hao, S., Andersen-Nissen, E., Mauck, W.M., Zheng, S., Butler, A., Lee, M.J., Wilk, A.J., Darby, C., Zager, M., et al. (2021). Integrated analysis of multimodal single-cell data. Cell *184*, 3573–3587.e29. https://doi.org/10.1016/j.cell.2021.04.048.

208. Moon, K.R., van Dijk, D., Wang, Z., Gigante, S., Burkhardt, D.B., Chen, W.S., Yim, K., van den Elzen, A.V.D., Hirn, M.J., Coifman, R.R., et al. (2019). Visualizing structure and transitions in high-dimensional biological data. Nat. Biotechnol. *37*, 1482–1492. https://doi.org/10.1038/s41587-019-0336-3.

209. Nair, S., Ameen, M., Sundaram, L., Pampari, A., Schreiber, J., Balsubramani, A., Wang, Y.X., Burns, D., Blau, H.M., Karakikes, I., et al. (2023). Transcription Factor Stoichiometry, Motif Affinity and Syntax Regulate Single-Cell Chromatin Dynamics during Fibroblast Reprogramming to Pluripotency. Preprint at bioRxiv, 2023.10.04.560808. https://doi.org/10.1101/2023.10.04.560808.

210. Baranasic, D., Hörtenhuber, M., Balwierz, P.J., Zehnder, T., Mukarram, A.K., Nepal, C., Várnai, C., Hadzhiev, Y., Jimenez-Gonzalez, A., Li, N., et al. (2022). Multiomic atlas with functional stratification and developmental dynamics of zebrafish cis-regulatory elements. Nat. Genet. *54*, 1037–1050. https://doi.org/10.1038/s41588-022-01089-w.

211. Gupta, S., Stamatoyannopoulos, J.A., Bailey, T.L., and Noble, W.S. (2007). Quantifying similarity between motifs. Genome Biol. *8*, R24. https://doi.org/10.1186/gb-2007-8-2-r24.

212. Li, D., Hsu, S., Purushotham, D., Sears, R.L., and Wang, T. (2019). WashU Epigenome Browser update 2019. Nucleic Acids Res. *47*, W158–W165. https://doi.org/10.1093/nar/gkz348.

213. Faust, G.G., and Hall, I.M. (2014). SAMBLASTER: fast duplicate marking and structural variant read extraction. Bioinformatics *30*, 2503–2505. https://doi.org/10.1093/bioinformatics/btu314.

214. Robinson, J.T., Thorvaldsdóttir, H., Winckler, W., Guttman, M., Lander, E.S., Getz, G., and Mesirov, J.P. (2011). Integrative genomics viewer. Nat. Biotechnol. *29*, 24–26. https://doi.org/10.1038/nbt.1754.

215. Götz, M., and Barde, Y.-A. (2005). Radial glial cells defined and major intermediates between embryonic stem cells and CNS neurons. Neuron *46*, 369–372. https://doi.org/10.1016/j.neuron.2005.04.012.

STAR★METHODS

KEY RESOURCES TABLE

REAGENT or RESOURCE	SOURCE	IDENTIFIER
Biological samples		
Human fetal heads dissected and cryopreserved	Collected at University of Cambridge and Cambridge University Hospitals NHS Foundation Trust under REC 96/085, transferred and processed at the University of Oxford under REC[22]/PR/0630 UK regulatory approvals	CAM00006; CAM00008
Human fetal heads embedded in OCT or flash frozen	Collected at University of Cambridge and Cambridge University Hospitals NHS Foundation Trust under REC 96/085, transferred and processed at the University of Oxford under REC[22]/PR/0630 UK regulatory approvals	CAM00010; CAM00011; CAM00012; CAM00013; CAM00014; CAM00015; CAM00016; CAM00017; CAM00018
Chemicals, peptides, and recombinant proteins		
Liberase TM	Roche	Cat# 5401127001
Maxima H Minus Reverse Transcriptase	Thermo Fisher	Cat# EP0751
CryoStor CS10	Thermo Fisher	Cat# C2874
HiFi Hotstart PCR kit	KAPA	Cat# KK2502
Phusion High-Fidelity DNA Polymerase	Thermo Fisher	Cat# F530S
Dynabeads Protein A	Life Technologies	Cat# 10006D
Dynabeads Streptavidin M-280	Invitrogen	Cat# 11206D
EnGen® Spy Cas9 NLS	NEB	Cat# MO646T
Critical commercial assays		
Nextera XT library preparation kit	Illumina	Cat# FC-131-1024
Chromium Next GEM Single Cell 5′ Kit v2	10x Genomics	Cat# PN-1000265
5′ CRISPR Kit	10x Genomics	Cat# PN-1000451
MicroPlex Library Preparation Kit v3 x48 rxns	Diagenode	Cat# C05010001
24 Dual indexes for MicroPlex Kit v3 /48 rxns	Diagenode	Cat# C05010003
Deposited data		
Smart-seq3 data	This paper	GEO: GSE256009
Perturb-seq data	This paper and Hu et al.[100]	GEO: GSE256008
Sox10 biotin ChIP-seq data and Elf1 CUT&RUN data	This paper	GEO: GSE303928
Human head multiome data	This paper, Williams et al.,[153] and D.M.F. et al. (unpublished data)	BioProject: PRJNA1214067
Developing human brain scRNA-seq data	Braun et al.[80]	https://github.com/linnarsson-lab/developing-human-brain
Experimental models: Organisms/strains		
Zebrafish Gt(foxd3-citrine)^{ct110}	Sauka-Spengler laboratory	ct110
Zebrafish Gt(foxd3-mCherry)^{ct110R}	Sauka-Spengler laboratory	ct110R
Tg(Sox10-Avi-2A-Citrine)	This study	N/A
Oligonucleotides		
OligodT30VN: 5′-Biotin-ACGAGCATCA GCAGCATACGAT$_{30}$VN-3′	Hagemann-Jensen et al.[99]	N/A

(*Continued on next page*)

Continued

REAGENT or RESOURCE	SOURCE	IDENTIFIER
Template Switching Oligo (TSO): 5′-Biotin-AGAGACAGATTGCGCAATGNN NNNNNNrGrGrG-3′	Hagemann-Jensen et al.[99]	N/A
Smart-seq3 forward PCR primer: 5′-TCGTCG GCAGCGTCAGATGTGTATAAGAGACAG ATTGCGCAA*T*G-3′ (* phosphorothioate bonds)	Hagemann-Jensen et al.[99]	N/A
Smart-seq3 reverse PCR primer: 5′-ACGAG CATCAGCAGCATAC*G*A-3′	Hagemann-Jensen et al.[99]	N/A
Chemically synthesized Alt-R-modified or unmodified crRNAs, tracrRNAs, or sgRNAs; See Table S4	IDT	N/A
Alt-R CRISPR-Cas9 tracrRNA	IDT	Cat# 1072532
NLS-BirA	Lukoseviciute et al.[136]	N/A
AVI-tag: GGCCTGAATGACATCTTTGAGG CCCAGAAGATCGAGTGGCATGAGGAC	Lukoseviciute et al.[136]	N/A[137]
Recombinant DNA		
pMT-sox10-NLS-BirA-2A-mCherry_Ras	Addgene[154]	Plasmid #80063
Software and algorithms		
RegVelo	This paper	https://github.com/theislab/regvelo
CellRank 2.0 (v2.0.6)	Weiler et al.[17]	https://github.com/theislab/cellrank
MELD (v1.0)	Burkhardt et al.[126]	https://github.com/KrishnaswamyLab/MELD
Dynamo (v1.3.3)	Qiu et al.[23]	https://github.com/aristoteleo/dynamo-release
CellOracle (v0.20.0)	Kamimoto et al.[26]	https://github.com/morris-lab/CellOracle
Dyngen (v1.0.5)	Cannoodt et al.[41]	https://github.com/dynverse/dyngen
scVelo (v0.3.2)	Bergen et al.[13]	https://github.com/theislab/scvelo
veloVI (v0.3.1)	Gayoso et al.[14]	https://github.com/YosefLab/velovi
TFvelo (v1.0)	Li et al.[32]	https://github.com/xiaoyeye/TFvelo
cell2fate (v1.0.0)	Aivazidis et al.[31]	https://github.com/BayraktarLab/cell2fate
UniTVelo (v0.2.5.2)	Gao et al.[29]	https://github.com/StatBiomed/UniTVelo
VeloVAE (v0.1.2)	Gu et al.[30]	https://github.com/welch-lab/VeloVAE
arboreto (v0.1.6)	Moerman et al.[49]	https://github.com/aertslab/arboreto
spliceJAC (v0.0.1)	Bocci et al.[155]	https://github.com/federicobocci/spliceJAC
cellDancer (v1.1.7)	Li et al.[156]	https://github.com/GuangyuWangLab2021/cellDancer
pySCENIC (v0.12.1)	Van de Sande et al.[157]	https://github.com/aertslab/pySCENIC
SCENIC+ (v1.0a2)	Bravo González-Blas et al.[24]	https://github.com/aertslab/scenicplus
Pando (v1.1.1)	Fleck et al.[25]	https://github.com/quadbio/Pando
Dictys (v1.1.0)	Wang et al.[27]	https://github.com/pinellolab/dictys
decoupler (v2.10.0)	Badia-i-Mompel et al.[110]	https://github.com/scverse/decoupler
LINGER (v1.0.0)	Yuan et al.[28]	https://github.com/Durenlab/LINGER
Scanpy (v1.10.3)	Wolf et al.[158]	https://github.com/scverse/scanpy
scVI (v1.2.0)	Gayoso et al.[159]	https://github.com/scverse/scvi-tools
zUMIs (v2.9.7)	Parekh et al.[160]	https://github.com/sdparekh/zUMIs
Bowtie2 (v2.5.2)	Langmead et al.[161]	https://bowtie-bio.sourceforge.net/bowtie2/index.shtml
MACS2 (v2.1.0)	Gaspar and Zhang et al.[162,163]	https://github.com/taoliu/MACS
ChromBPNet (v0.1.7)	Pampari et al.[135]	https://github.com/kundajelab/chrombpnet
Cellpose (v3.1.0)	Pachitariu et al.[164]	https://www.cellpose.org/

EXPERIMENTAL MODEL AND STUDY PARTICIPANT DETAILS

Fish husbandry and Smart-seq3 single-cell RNA-Seq

Animal experiments complied with UK Home Office protocols under the Animals (Scientific Procedures) Act 1986, and followed the Guide for the Care and Use of Laboratory Animals.[165] All vertebrate animal research was conducted at Oxford University Biomedical Services. Adult fish care followed established protocols, and embryos were staged using standard methods under an Olympus dissecting stereomicroscope.[166]

We generated Smart-seq3 data to complement the 10x multiome data used to reconstruct the cranial NC GRN.[100] Embryos were harvested by crossing zebrafish lines Gt(foxd3-mCherry)^{ct110R} and Gt(foxd3-Citrine)ct110,[125] incubating to the desired stages, and dissociating with Liberase TM (Roche) for 25 minutes at 32°C. Cell suspension was filtered through 100-μm strainers (PluriSelect), and stained with 1–3 μg/ml DAPI. Live Citrine-positive (Citrine+Cherry-) and double-positive (Citrine+Cherry+) single cells were isolated using fluorescence-activated cell sorting (FACS) on a Sony MA900 instrument.

To generate full-transcript-length, deep scRNA-Seq data, we prepared the Smart-seq3 libraries following the protocol.[99] During FACS, cells were sorted into individual wells containing 3-μl lysis buffer dispensed by the MANTIS liquid handler (FORMULATRIX). The lysis buffer consisted of 5% polyethylene glycol 8000 (Sigma), 0.1% Triton X-100 (Sigma), 0.5 U/μl RNAse inhibitor (TaKaRa), 0.5 μM OligoT_{30}VN (IDT), 0.5 mM/each dNTPs (Thermo Fisher) and nuclease-free water (Thermo Fisher). Note that these concentrations reflect their concentrations in the 4-μl reverse transcription reaction. The plates were briefly centrifuged, quickly frozen on dry ice and stored at -80°C. In brief, the plates of sorted cells were transferred on dry ice from -80°C, incubated at 72°C for 10 min, and cooled down to 4°C. Quickly, 1 μl/well reverse transcription master mix containing 25 mM Tris-HCl pH8 (Invitrogen), 30 mM NaCl (Invitrogen), 2.5 mM $MgCl_2$ (Invitrogen), 1 mM GTP (Thermo Fisher), 8 mM DTT (Thermo Fisher), 0.5 U/μl RNAse inhibitor (TaKaRa), 2 μM TSO (IDT), 2 U/μl Maxima H-minus (Life Technologies) was added. The reverse transcription program was 42°C 90 min, 10x (50°C 2 min, 42°C 2 min), 85°C 5 min and 4°C hold. MANTIS added 6 μl/well PCR mix containing 1x KAPA HiFi HotStart buffer, 0.3 mM/each dNTPs, 0.5 mM $MgCl_2$, 0.5 μM forward primer, 0.1 μM reverse primer and 1 U/μl polymerase (Roche). The PCR program was 98°C 3 min, 23x (98°C 20 s, 65°C 30 s, 72°C 4 min), 85°C 5 min and 4°C hold. These cDNA samples were cleaned up by 0.6:1 AMPure XP beads and relocated into 384-well plates using FxP (Beckman Coulter), quantified by Quant-iT PicoGreen (Invitrogen) on CLARIOstar (BMG Labtech), quality checked by TapeStation 4200 and High Sensitivity D5000 assays (Agilent) and normalized to 0.3 ng/μl using MANTIS. Next, 150 pg/μl cDNA was tagmented by adding 0.5 μl Smart-seq3 in-house Tagmentation buffer (40 mM Tris-HCl pH7.5, 20 mM $MgCl_2$, 20% dimethylformamide and UltraPure water in 4x buffer), 0.08 μl Amplicon Tagmentation Mix (Illumina) and UltraPure water for a total reaction volume of 2 μl and incubating at 55°C for 10 min. The tagmentation was stopped by adding 0.5 μl of 0.2 % SDS and incubated at room temperature for 5 min. Then 1.5 μl Nextera index primer pairs and 3 μl PCR master mix (1x Phusion HF buffer, 0.2 mM/each dNTPs, 2 U/μl Phusion HF) were added to each well of the tagmentation product and the PCR program was 72°C 3 min, 98°C 3 min, 12x (98°C 10 s, 55°C 30 s, 72°C 30 s), 72°C 5 min and 4°C hold. The libraries were pooled, cleaned up by 0.6:1 AMPure XP beads and sequenced on NovaSeq for 150-bp paired end reads (Illumina).

In vivo Perturb-seq of zebrafish neural crest development

To systematically validate RegVelo's predicted perturbation effects, we implemented CRISPR-Cas9 knockouts and Perturb-seq as previously described.[100] Our Perturb-seq dataset on zebrafish neural crest development included knockouts of seven regulons: *tfec, mitfa, bhlhe40, tfeb, elf1, nr2f2*, and *nr2f5*. For benchmarking, we combined this dataset with an additional one,[100] resulting in 22 conditions: 14 single-gene, 5 two-gene, and 3 multi-gene knockouts. We generated F_0 knockout crispants by injecting the mixture of 250 ng/μl single guide RNAs (sgRNAs), 3.36 μM EnGen Cas9, and 0.05% phenol red into 1-cell-stage embryos from crosses of *Gt(foxd3-Citrine)*^{ct110} and wild-type lines. At the 21-somite stage, foxd3-Citrine-positive cells were isolated by flow cytometry. Gene expression and CRISPR libraries were prepared according to the CG000510 Chromium NextGEM Single Cell 5′ v2 CRISPR User Guide (Rev B, 10x Genomics) and sequenced on NextSeq and NovaSeq platforms (Illumina).

Whole-mount hybridization chain reaction (HCR) and phenotypic check

To visualize the mRNA in situ, we applied HCR[167] targeting certain mRNA in zebrafish embryos by following the 'In situ HCR v3.0 protocol' for whole-mount zebrafish larvae from Molecular Instruments using the commercial hairpins, buffers, probes (Molecular Instruments), or in-house probes (Sangon).[167] To quantify the lineage-specific knockout effects for each embryo, we generated heterozygous *foxd3-mCherry*-expression F0 crispants by CRISPR-Cas9, and fixed the embryos at approximately 21 ss. We performed the HCR of marker genes (e.g., *mCherry* for the NC lineage, *colec12* for the mesenchymal lineage, *dlx2a* for the pharyngeal arch lineage, and *mitfa* for the pigment lineage) in the uninjected and knockout embryos in parallel. Hybridized embryos were embedded in 1% low-melting point agarose and imaged with the z-stack scanning on confocal microscopes (Zeiss).

Next, we segmented the cells for each z stack using Cellpose[168] with the cyto2 or in-house trained models. To specifically quantify the lineage, four types of regions of interest (ROI) were drawn using the Napari Viewer for the facial mesenchyme (ROI1), pharyngeal arch 2 (ROI2), the whole cranial pre-otic region for pigment lineage quantification (ROI3), and mNC_hox34 (ROI4). Images or ROIs were excluded based on these criteria: 1) the embryo was not properly positioned and/or the ROI was not well captured by the field; 2) the channel quality was compromised by photobleaching, fluorophore degradation, or strong autofluorescence; 3) the embryo morphology was not intact. After filtering, images were corrected for uneven illumination by subtracting the background with the

radius of 20 to 50 pixel using OpenCV (https://docs.opencv.org/4.x/d9/d61/tutorial_py_morphological_ops.html). We quantified the mCherry (NC lineage)$^+$colec12$^+$ cells in the ROI1 for the facial mesenchymal lineage, mCherry$^+$dlx2a$^+$ cells in the ROI2 for the pharyngeal arch 2 lineage, mCherry$^+$mitfa$^+$ cells in the ROI3 for the pigment lineage, and mCherry$^+$dlx2a$^+$ cells in the ROI4 for the mNC_hox34 lineage.

To check the phenotypes of wild-type and knockout embryos, we performed live imaging with MVX10 stereomicroscope (Olympus). We tested the reliability of our phenotyping assay by knocking out *sox10*, whose homozygous loss of function can lead to the colorless phenotype.[169] In the F0 crispants, the pigmentation was reduced in most embryos between 36 and 48 hpf (Figure S10F).

Sox10 biotin ChIP-seq

To identify the Sox10 binding sites in zebrafish, we performed the biotin ChIP-seq experiment as previously described.[136] In brief, the Tg(Sox10-Avi-2A-Citrine) zebrafish line were generated by injecting 3 nL mixture of 4 μM crRNA:tracrRNA duplex, 2 μM Cas9 protein (NEB), 300 mM KCl, and 50 ng/μL biotinylated dsDNA donor[170] into each 1-cell-stage embryo. Founders were screened by checking Citrine fluorescence in F1 embryos under the MVX10 stereomicroscope, where we observed a strong Citrine expression pattern resembling the endogenous sox10 expression. This suggests that the Avi tag and Citrine were correctly inserted in the C-terminal of the Sox10 protein. BirA mRNA was transcribed from the plasmid pMT-sox10-NLS-BirA-2A-mCherry_Ras[154] using mMESSAGE mMACHINE SP6 High Yield Kit (Ambient) and purified using RNA Clean & Concentrator (Zymo).

To collect samples for biotin ChIP-seq, Tg(Sox10-Avi-2A-Citrine) and wildtype fish was crossed, and embryos (N = 50–100) injected with 160 pg BirA mRNA were homogenized at 10 ss, 14–16 ss, and 20–22 ss accordingly,[136] followed by fixation, sonication by 4 cycles 30s ON/30s OFF on the Bioruptor Pico (Diagenode). Two biological replicates were processed for each developmental stage group. The immunoprecipitation was conducted using the Dynabeads streptavidin M-280 (Invitrogen) after the pre-binding of Dynabeads Protein A (Life Technologies), while 1/20 of the sample was kept aside as the input control for each immunoprecipitation reaction. Following the intensive wash as previously described,[136] the sheared and immunoprecipitated DNA was purified, prepared into sequencing libraries with 12 to 16-cycle amplification (12–13 cycles for the input control) using MicroPlex Library Preparation Kit v3 (Diagenode), and sequenced on Novaseq (Illumina) with 150-bp paired end reads.

CUT&RUN

Cut&Run was performed as previously described by Skene and Henikoff, 2017, with minor modifications. A total of 150 embryos were collected at 16 somite stage. The chorion was removed for all embryos prior to dissociation in 500 μL Accumax for 25 min. Cells were pelleted at 600x g for 3 min at room temperature, washed twice in wash buffer (20 mM HEPES pH 7.5, 150 mM NaCl, 0.5 mM spermidine, 1× protease inhibitor tablet), and resuspended in 800 μL of wash buffer. Cells were bound to 80 μL of Concavalin-A beads for 10 min with rotation and distributed into 100 μL aliquots for the antibody condition. Two replicates per condition were included for ELF1 and IgG controls. The samples were collected on a magnetic stand and incubated with 100 μL of antibody buffer (wash buffer with 0.05% digitonin and 2 mM EDTA) containing primary antibodies (1:50 dilution; ELF1 bs-7042R, Bioss) for 12 h at 4°C on a thermomixer with 350 rpm.

The samples were spun quickly 100×g, beads were collected using the magnetic stand, washed twice with 200 μL cell permeabilization buffer (wash buffer with 0.05% digitonin), resuspended in 50 μL of cell permeabilization buffer containing 2.5 μL of pAG-MNase fusion protein (700 ng/mL) and incubated 1 hour at 4°C on thermomixer with 350rpm. Samples were washed twice more, resuspended in 100 μL of same buffer, chilled on ice to 0°C, and digestion was initiated by adding 2 μL 100 mM $CaCl_2$ for 1 h at 1°C on a thermomixer with 350 rpm. The reaction was stopped with 50 μL Stop Buffer (340 mM NaCl, 20 mM EDTA, 4 mM EGTA, 0.05% digitonin, 100 μg/ml RNase A, 50 μg/ml glycogen, 1:100 E. coli DNA spike-in) and incubated at 37°C for 10 min to release CUT&RUN fragments from insoluble chromatin.

To collect the fragmented antibody-bound DNA, beads were collected by placing samples on the magnet and the supernatant was transferred to new low-binding tubes. Chromatin proteins were digested with 1 μL 20% SDS and 2.5 μL proteinase K (20 mg/uL) for 10 min at 70°C. DNA was purified by 300 μL phenol:chloroform extraction, followed by 300 μL chloroform cleanup. The aqueous (upper) layer was transferred to fresh tubes containing GlycoBlue (2 mg/mL). To precipitate DNA, 750 μL of 100% ethanol was added and incubated for 12 h at -80°C. Samples were centrifuged by 20 min 16,000×g and resuspended in 15 μL nuclease-Free Water. DNA concentrations were measured with the Qubit fluorometer, and sequencing libraries were prepared.

Hindbrain scRNA-seq and single nuclei multiome sequencing

The curation of the multiome hindbrain dataset is as previously described[153] with expansion of data used in this study including scRNA-seq libraries. Samples were obtained from donors in accordance with ethically approved study REC 96/085 (University of Cambridge and Cambridge University Hospitals NHS Foundation Trust). Samples were transferred under a Material Transfer Agreement and stored and processed in accordance with ethically approved study REC 22/PR/0630 (University of Oxford) on HTA-licensed premises (license number 12433).

Cells and nuclei from 12 embryos across 6 Carnegie Stages (CS14-19) were analysed using either the chromium 10X Next GEM Single Cell Gene Expression v3.1 kit or the chromium 10X Next GEM Single Cell Multiome ATAC + Gene Expression kit. Libraries were sequenced in accordance with 10X recommendations and processed with Cellranger v7.1.0. One library was rejected due to poor median genes per nucleus.

Pooled libraries were demultiplexed into individual samples using genetic demultiplexing. To facilitate this, we generated bulk RNA-seq libraries to demultiplex multiome samples and bulk ATAC-seq to demultiplex scRNA-seq multiome samples sequenced to a targeted depth of 50 million reads. Variant calling and genotyping were performed in accordance with the GATK RNAseq short variant discovery (SNPs + indels) pipeline. Multiome samples were demultiplexed using Souporcell within Demuxafy v2.0.1. Ambient RNA was detected and removed using Cellbender v0.3.0. Doublets were detected based on genotype using Souporcell and separately using scDblFinder both using Demuxafy v2.0.1. Six scRNA-seq and seven multiome libraries were annotated including reference to a previously published whole brain single-cell atlas,[80] using celltypist v.1.6.3 with the developing human brain ('DHB') model. Annotated neuronal and radial glia populations were included for integration totalling 50,388 cells and nuclei with a final median genes of 3034.

METHOD DETAILS

RegVelo model specification

Approaches for inferring RNA velocity models describe splicing dynamics through biophysical models like systems of ordinary differential equations (ODEs).[171] Such models describe the transcription of precursor (unspliced) RNA u at rate α, followed by splicing into mature (spliced) mRNA s with rate β; spliced mRNA degrades at a rate γ, manifesting in the first-order approximation of a chemical master equation for gene g as

$$\begin{aligned} \dot{u}_g &= \frac{du_g(t)}{dt} = \alpha_g \mathbb{I}_{\{t<t_s\}} - \beta_g u_g(t) \\ \dot{s}_g &= \frac{ds_g(t)}{dt} = \beta_g u_g(t) - \gamma_g s_g(t), \end{aligned} \qquad \text{(Equation 1)}$$

where $\mathbb{I}_{\{t<t_s\}}$ denotes the indicator function and t_s is a switch time. Throughout this work, we assume N_C cells and N_G genes.

Previously, we have devised two different parameter inference schemes based on either Expectation-Maximization (EM) or variational inference (VI). In these frameworks, we assumed gene transcription during an induction phase ($t < t_s$), followed by a repression phase ($t > t_s$). Although this dynamical modeling paradigm has enabled the recovery of biological insight, both models neglect gene-gene dependencies and assume a piecewise constant transcription rate for each gene, thereby overlooking the dynamic changes in the transcription rate itself. Biological systems fulfill these assumptions only approximately,[25,172] however, and exhibit more complex dynamics, instead.

Upstream regulators govern gene transcription, a process neglected so far. To include these non-linear dynamics in our model of transcriptional dynamics, we define gene- and cell-specific transcription rates α_{gn} that we parametrize with a neural network α. In this setup, we assume that mature mRNA of upstream regulators and a gene-specific base transcription rate describe the transcription rate via

$$\alpha_{gn} = h\left(\sum_{j=1}^{N_G} W_{gj} s_j^{(n)} + b_g\right) \qquad \text{(Equation 2)}$$

where h is a nonlinear activation function, $s^{(n)} \in \mathbb{R}_{\geq 0}^{N_G}$ denotes the n-th sample, and $b_g \in \mathbb{R}$ and $W \in \mathbb{R}^{N_G \times N_G}$ the base transcription rate and gene regulatory network (GRN) weight matrix, respectively.

The GRN weight matrix represents gene regulation, informed by a prior gene regulation graph G, curated from public datasets like cistrome[35] or learned from single-cell, bulk epigenetic, or multi-ome sequencing data through GRN inference methods. We represent the prior regulation graph G as a binary, skeleton matrix that encodes the interactions between transcription factors (TFs) and target genes and provide two strategies for learning GRN structures from these priors: hard and soft constrained inference. In the case of hard constraints, G restricts the non-zero entries of the weight matrix W to edges present in the prior graph, i.e.,

$$W_{j,k} = \begin{cases} 0 & \text{if } G_{j,k} = 0, \\ \omega \in \mathbb{R} & \text{otherwise} \end{cases}.$$

Conversely, for soft constraints, we allow learning additional connections regularized by the prior graph G with

$$R_{\text{prior}}(W; G) = \|W \odot (1 - G)\|_2, \qquad \text{(Equation 3)}$$

to favor TF-target connections present in the prior; $\odot$ denotes the Hadamard product, i.e., element-wise product operation. The soft constraint, thus, allows for correcting missing links in the prior gene regulation graph.

In a realistic biological system, cells often undergo several internal orthogonal processes simultaneously like cell cycle versus differentiation, for example. In such cases, using a single latent time and a deterministic ODE system fails to model these biological

processes jointly. Additionally, determining the number of biological processes and modeling them requires some prior information. Thus, similar to scVelo and veloVI, we estimate gene-specific latent times t_g and aim to identify a single latent time axis for each gene, combined with a global regulatory function to best describe the observed data.

The velocity function, defined by Equation 1 and Equation 2, governs cell state evolution, inducing a continuous vector field $(\dot{u},\dot{s})$. Given an initial condition (u_0,s_0), we simulate dynamics along each gene-specific latent time axis to estimate the integrated value for each gene and each cell as

$$\begin{aligned}\bar{u}^{(g)}(t_{ng}) &= u_g(0) + \int_0^{t_{ng}} \dot{u}_g\big(s(t_g),u(t_g)\big)dt_g \\ \bar{s}^{(g)}(t_{ng}) &= s_g(0) + \int_0^{t_{ng}} \dot{s}_g\big(s(t_g),u(t_g)\big)dt_g.\end{aligned} \qquad \text{(Equation 4)}$$

We simulate these state solutions for each gene efficiently through a parallelized numerical integration solver dopri5 implemented in the *torchode*[173] Python package.

Model assumptions

We assume multiple time axes, representing different biological processes, but time-independent regulatory principles, i.e., a constant GRN weight matrix W - a modeling choice that allows us to couple multiple gene dynamics. Similar to scVelo and veloVI, we assume an initial gene expression of $u(t_0) = s(t_0) = 0$ at $t_0 = 0$ and that each gene is on the same time scale. Here, we follow scVelo and veloVI and set $t_{\max} = 20$. Lastly, we define the nonlinear activation function h in Equation 2 as the softplus function

$$h(x) = \log(1 + e^x),$$

thus assuming that TFs predict the transcription rate of target genes based on additive effects.

RegVelo generative process

As described above, we assume each gene has a specific latent time. We have already established in recent work that the inferred, gene-specific latent times have a low-rank structure, implying that we can generate the latent time matrix from multiple independent components that control cell dynamics. Therefore, our model does not only couple genes through a shared regulatory principle encoded by W but also, like veloVI, ties the gene-specific latent time via a local low-dimensional latent variable.

For each cell, we draw a low-dimensional latent variable z from an isotropic Gaussian, i.e.,

$$z^{(n)} \sim \mathcal{N}(0, I_d),$$

where I_d denotes the d-dimensional identity matrix. Here, we use $d = 10$ dimensions throughout this study. We define the gene-specific latent time as a function of this latent variable defined by

$$t_{ng} = [T(z^{(n)})]_g,$$

with $T : \mathbb{R}^d \to (0,1)^{N_G}$ parametrized by a fully connected neural network. Utilizing the time information, we derive the state solution for each cell and gene through Equation 4. Finally, we assume that the conditional normal distributions parametrized by θ

$$\begin{aligned}p_\theta\left(u_g^{(n)} \middle| z^{(n)}\right) &= \text{Normal}\left(\bar{u}^{(g)}(t_{ng}), \left(\sigma_g^u\right)^2\right) \\ p_\theta\left(s_g^{(n)} \middle| z^{(n)}\right) &= \text{Normal}\left(\bar{s}^{(g)}(t_{ng}), \left(\sigma_g^s\right)^2\right)\end{aligned}$$

generates the observed data. Here, θ includes transcriptional regulation (W, b), kinetic parameters (β, γ), and neural network parameters.

Like veloVI, we consider the observed data $\{(s^{(n)}, u^{(n)}) | n \in \{1, \ldots, N_C\}\}$ as nearest-neighbor smoothed expression data, which is also used as input for scVelo. RegVelo, like veloVI, assumes that the smoothed spliced and unspliced abundances have been min-max scaled to the unit interval $[0,1]$, independently for each gene. We also assume that the smoothed expression, representing an average of random variables, follows a sampling distribution centered on a mean value and is approximately normal.

RegVelo inference

RegVelo aims to estimate the following parameters: (1) point estimates of kinetic parameters, including gene regulation weights summarized in the GRN matrix, base transcription, degradation, and splicing rate constants; (2) point estimates of the parameters of all neural networks; (3) a posterior distribution over the latent variable z, to model the underlying dynamic process. We use variational inference to approximate the posterior distribution and estimate all other parameters by maximizing the evidence lower bound (ELBO). Consequently, we model RNA velocity probabilistically by calculating it as a function of the variational posterior distribution of latent variables.

Variational posterior

We introduce the variational distribution q_ϕ parametrized by a neural network with parameter set ϕ to approximate the evidence function $\log p_\theta(u,s)$

$$\log(p_\theta(u,s)) \geq \mathrm{ELBO}(\theta,\phi;u,s) = -\mathrm{KL}\left[q_\phi(z|u,s)\middle|p(z)\right] + \mathbb{E}_{q_\phi(z|u,s)}[\log(p_\theta(u,s|z))] \quad \text{(Equation 5)}$$

by maximizing the ELBO. We assume independent samples, allowing us to factorize the distribution $q_\phi(z|u,s)$ and $p_\theta(u,s|z)$ as

$$q_\phi\left(z|u,s\right) = \prod_{n=1}^{N_C} q_\phi\left(z^{(n)}|u^{(n)},s^{(n)}\right)$$
$$p_\theta(u,s|z) = \prod_{n}^{N_C} p_\theta(u^{(n)}|z^{(n)}) p_\theta(s^{(n)}|z^{(n)}).$$

Dynamics regularization

RegVelo solves the dynamics through a numerical integrator, as the kinetic process is too complex to solve analytically. Avoiding poorly conditioned dynamics improves the accuracy of numerical integration.[174] Therefore, we add penalties on the regularity of the velocity function $F : \mathbb{R}^{2N_G} \to \mathbb{R}^{2N_G}$ to ensure smoothness.

Here, we define the regularity as the total derivative of the velocity function F with respect to time, i.e.,

$$\frac{dF(x)}{dt} = \nabla_x F(x) F(x) = J(x) F(x),$$

using the chain rule and with $x = (u,s)$ and the Jacobian matrix $J(x)$. To simplify the dynamics, we regularize the Jacobian matrix and velocity field to encourage small values: we use L2 regularization to penalize large entries of velocity entries, promoting a more regular vector field, and L1 regularization to penalize large entries in the Jacobian matrix to foster sparse gene regulation. Therefore, we define the regularization terms

$$L_{\text{Jacobian}} = \frac{1}{N_C}\sum_{n=1}^{N_C} \|\mathrm{diag}(\mathrm{sigmoid}\circ(Ws^{(n)}+b))W\|_1 \quad \text{(Equation 6)}$$

$$L_{\text{velocity}} = \sum_{n=1}^{N_C} \|\mathrm{softplus}\circ(Ws^{(n)}+b) - \beta u^{(n)}\|_2, \quad \text{(Equation 7)}$$

where, diag and sigmoid denote the operator to convert a vector into a diagonal matrix and the sigmoid activation function, respectively, and $\circ$ the operator to apply a scalar function component-wise. As the complexity of gene dynamics in our model stems primarily from gene regulation, we only regularize the Jacobian matrix and the unspliced RNA velocity to focus on aspects directly influenced by regulation.

Further, assuming that upstream regulators govern each gene transcription fully, we penalize large values of base transcription b with

$$L_{\text{base}}(b) = \|b + b^*\|_2, \quad \text{(Equation 8)}$$

to keep the transcription rate near zero when upstream regulation effects diminish, i.e., $Ws = 0$. We fixed $b^* = 10$ in this model. Finally, we define the dynamic regularization loss function as the sum of L_{Jacobian}, L_{velocity} and L_{base}.

RegVelo's objective function

Combining Equation 3, Equation 5, Equation 6, Equation 7, and Equation 8, our objective includes the three parts

$$L(\theta,\phi;u,s) = -\mathrm{ELBO}(\theta,\phi;u,s) + \underbrace{\lambda_1 R_{\text{prior}}(W;G)}_{\text{prior regularization}} + \underbrace{\lambda_2 L_{\text{Jacobian}}(\theta;s) + L_{\text{velocity}}(\theta;u,s) + L_{\text{base}}(b)}_{\text{dynamics regularization}},$$

with tunable hyperparameters λ_1 and λ_2: λ_1 controls the strengths of adding prior knowledge and λ_2 governs the strengths of the L1 regularization of the Jacobian; increasing λ_2 promotes greater sparsity in the learned GRN. By default, we set $\lambda_1 = 1$ and $\lambda_2 = 0$.

Network architecture and optimization

All neural networks are fully connected feedforward networks, with ReLU activation functions for hidden layers and softplus-transformations of the last layer to enforce non-negative kinetic parameters. We initialize the weight matrix W as a zero matrix, kinetic parameters, including splicing and degradation rates, as a constant values, and base transcription b and all neural network parameters through the default implementation in PyTorch (torch.nn.init.xavier_uniform_); following veloVI, we initialize a unit splicing rate and estimate degradation based on a steady-state assumption. We use stochastic gradients along with the AdamW[175] optimizer to optimize the loss function.

Related work

Recently, several studies have explored integrating GRN with RNA velocity to better understand cellular dynamics. Below, we briefly describe each method and highlight how they differ from our proposed framework, RegVelo.

scKINETICS

scKINETICS[176] proposes a statistical framework to jointly estimate gene regulatory networks and gene expression velocity. It models changes in gene expression using a simple linear system:

$$v = Ax.$$

where x denotes the expression vector of all genes (both targets and TFs), and A represents the regulatory weight matrix, i.e. GRN, with A_{ij} describing the effect of expression of TF j on target i. The method does not solve the system directly. Instead, it treats v as a latent variable and applies an EM algorithm to iteratively infer both v and A. The inferred regulatory matrix A is then used to define cell velocities.

Notably, scKINETICS does not estimate latent time, solve a full ODE system, or generate gene expression. This model is limited to estimate GRN matrix A and does not incorporate splicing and degradation kinetics.

TFvelo

TFvelo,[32] like scKINETICS, jointly infers a GRN and expression velocity via EM framework. It introduces two key improvements compared to scKINETICS. First, given a target gene g, TFvelo assumes that the RNA velocity $\frac{dy_g(t)}{dt}$ is determined by the expression level of involved TFs $X_g \in \mathbb{R}^{n_{TF}\times 1}$ and the target gene expression itself y_g, n_{TF} denotes the number of TFs. For each gene, TFvelo models cell velocity by considering the regulation term and the degradation term γ_g:

$$\frac{dy_g(t)}{dt} = W_g X_g(t) - \gamma_g y_g(t)$$

Second, instead of treating velocity as the latent variable, TFvelo applies a gradient matching-like methods[177] by first fitting time-dependent target gene expression with parametric function (e.g., sine function), and then differentiating them with respect to time to approximate gene-wise velocity. This reformulates the velocity equation as a simple regression problem, enabling inference of the GRN matrix W directly.

Despite these improvements, TFvelo does not incorporate a biophysical model of splicing dynamics or solve the underlying ODEs in a generative framework.

Velorama

In contrast to above approaches, Velorama[146] does not estimate RNA velocity directly. Instead, it leverages either RNA velocity vectors or pseudotime to construct directed acyclic graphs (DAG) that capture the temporal ordering of cells. It then infers GRNs using Granger causality along the DAG structure. In this way, Velorama focuses on learning regulatory interactions guided by directional information rather than modeling RNA kinetics.

Relation to RegVelo

Our approach RegVelo builds on a variational inference framework[14] to jointly infer each cell's latent time, RNA velocity and GRN. Compared to listed methods, RegVelo is the only method to directly integrate the original RNA velocity framework[12] with gene regulation through a biophysical model of transcription and splicing. Unlike scKINETICS and TFvelo, RegVelo explicitly solves the full system of ODEs with numerical solver, allowing for generative modeling of gene expression and capturing dynamic behavior across lineages.

Moreover, RegVelo leverages the CellRank[16,17] framework to connect inferred gene regulatory mechanisms with cell fate probabilities, enabling a more interpretable and principled connection between regulatory dynamics and lineage decisions. While scKINETICS and TFvelo approximate regulatory dynamics and velocity, they do not support simulation of perturbed gene expression. Although scKINETICS proposes a concept of TF-wide in silico perturbation screening based on TF activity, it lacks a systematic framework to interpret perturbation effects across lineages.

Downstream tasks

Fitted abundance value

Let $u^*(n)$ and $s^*(n)$ be the random variable representing posterior predicted abundance values for unspliced and spliced RNA in cell n, respectively. We denote their posterior distribution by $p(u^*(n)|u^{(n)}, s^{(n)})$ and $p(s^*(n)|u^{(n)}, s^{(n)})$ and define as

$$p(u^*(n)|u^{(n)}, s^{(n)}) = \int p_\theta(u^*(n)|z^{(n)})(z^{(n)}|u^{(n)}, s^{(n)})dz$$
$$p\left(s_n^*\middle|u^{(n)}, s^{(n)}\right) = \int p_\theta\left(s^*(n)|z^{(n)})(z^{(n)}|u^{(n)}, s^{(n)}\right)dz.$$

The fitted values for unspliced and spliced abundances are the means of these posterior distributions, i.e., $\mathbb{E}_{p(u^*(n)|u^{(n)},s^{(n)})}[u^*(n)]$ and $\mathbb{E}_{p(s^*(n)|u^{(n)},s^{(n)})}[s^*(n)]$, respectively. We estimate the mean value by sampling from the posterior distribution and taking the average.

Gene-wise latent time

We compute latent time for each gene and cell as $\mathbb{E}_{q_\phi(z^{(n)}|u^{(n)},s^{(n)})}\left[T(z^{(n)})_g\right]$ and estimate the expectation value by Monte Carlo sampling,[178] where we sample from the posterior distribution of $z^{(n)}$ before calculating the average value of $T(z^{(n)})_g$.

RNA velocity

We define the velocity of a specific gene in a particular cell as the function depending on the latent variable $z^{(n)}$. To sample the velocity from the latent posterior distribution, we

1. sample $z^{(n)}$ from $q_\phi(z^{(n)}|u^{(n)},s^{(n)})$,
2. calculate the latent time $t_{ng} = T(z^{(n)})_g$,
3. compute velocity in gene g and cell n using

$$v_{gn} = \beta_g \bar{u}^{(g)}(t_{gn}) - \gamma_g \bar{s}^{(g)}(t_{gn})$$

These steps provide velocity samples from the posterior predictive velocity distribution.

Velocity uncertainty estimation

As a variational inference-based model, veloVI[14] defines an intrinsic uncertainty method to quantify uncertainty in each cell through sampling velocity vectors from the posterior distribution and calculate the variance of the cosine similarity of the samples; as an extension of veloVI, RegVelo follows the same approach.

Representing the underlying GRN

The gene regulation function (Equation 2) describes the regulation of unspliced RNA abundance of upstream regulators. We approximate the gene regulation function in the neighborhood of a specific s_n by first-order Taylor expansion

$$\text{softplus} \circ (Ws + b) \approx \text{softplus} \circ (Ws^{(n)} + b) + \text{diag}(\text{sigmoid} \circ (Ws^{(n)} + b))W(s - s^{(n)}).$$

The coefficient of the first-order term denotes the strength of linear regulation. We define the GRN for cell n as

$$\text{diag}(\text{sigmoid} \circ (Ws^{(n)} + b))W.$$

To define cell type-specific GRNs, let $\mathcal{S}(c)$ denote the set of cells of type c

$$\sum_{j \in \mathcal{S}(c)} \text{diag}\left(\text{sigmoid} \circ \left(Ws^{(j)} + b\right)\right)W.$$

Estimation of uncertainty in gene regulation

We estimated uncertainty of regulatory weight W by training the model multiple times (N) to the same input data and aggregating the resulting regulatory matrices, denoted as $\{W^{(1)}, W^{(2)}, \ldots, W^{(N)}\}$, where each $W^{(j)} \in \mathbb{R}^{N_G \times N_G}$ and N_G is the number of genes.

For each putative edge from transcription factor j to target gene k, we define the normalized uncertainty score as:

$$U_{jk} = \frac{median\left\{\left|W_{jk}^{(1)} - \tilde{W}_{jk}\right|, \left|W_{jk}^{(2)} - \tilde{W}_{jk}\right|, \ldots, \left|W_{jk}^{(N)} - \tilde{W}_{jk}\right|\right\} + \varepsilon}{\left|\tilde{W}_{jk}\right| + \varepsilon}$$

where $\tilde{W}_{jk} = \text{median}\left\{W_{jk}^{(1)}, W_{jk}^{(2)}, \ldots, W_{jk}^{(N)}\right\}$ is the median edge weight, and ε is a data-dependent constant added for numerical stability, computed as the 10th percentile of all absolute median weights $\left|\tilde{W}_{jk}\right|$. This quantity reflects the relative variability of edge weights across runs. Small values of U_{jk} indicate stable and reliable predictions; higher values suggest greater uncertainty.

Therefore, In the inferred GRN, we characterize each edge with two values: the value in the W matrix denotes the direction and strength of the TF-gene regulation, and the corresponding bootstrap uncertainty value serves as the confidence level of that regulatory edge.

Evaluating calibration of bootstrap based uncertainty

We followed a previous study to evaluate calibration.[179] Specifically, we created ten cumulative bins corresponding to the lowest $10\%, 20\%, \ldots, 100\%$ of estimated uncertainty values and assigned point estimates of TF-gene interactions to these bins based on their uncertainty. For each bin, we calculated overlap AUPRC and AUROC using the corresponding gold standard, considering only interactions present in both the bin and the gold standard. We used cumulative binning because non-cumulative bins could include very few gold-standard interactions, which would make AUPRC and AUROC estimates noisier.

Simulation of perturbation effects

RegVelo generates cell dynamics based on gene regulation. By modifying the learned GRN in a trained RegVelo model, we simulate the in silico perturbed dynamics. Assuming we want to perturb the function of the TF I in a system, we mask the corresponding column of the weight matrix to generate a perturbed weight matrix

$$\widehat{W}_{j,k} = \begin{cases} W_{j,k} & \text{if } k \neq I, \\ 0 & \text{if } k = I \end{cases}.$$

The perturbed weight matrix defines a velocity function for numerical integration and induces a perturbed velocity v^*. We systematically compare the perturbed velocity with the original fitted velocity v using the following metrics:

Local perturbation effects

We define the perturbation effects on each cell by computing the cosine dissimilarity

$$1 - \cos(v^*, v).$$

Cell fate perturbation effects

We quantify the perturbation effects on cell fate decisions by passing the different velocity estimates to the typical CellRank workflow: using the VelocityKernel in CellRank and assuming m terminal states, we define the original cell fate probability matrix $\Pi \in \mathbb{R}^{N_C \times m}$ and perturbed probability matrix $\Pi^* \in \mathbb{R}^{N_C \times m}$, where each row represents a cell, and each column represents a terminal state. We can formulate the cell fate perturbation effects for the k-th terminal state as the t test statistics

$$t = \frac{\overline{\Pi}^*_{:,k} - \overline{\Pi}_{:,k}}{\sqrt{2\frac{S^2}{N_C}}},$$

with sample means of fate probabilities toward the k-th terminal state $\overline{\Pi}^*_{:,k}$ and $\overline{\Pi}_{:,k}$, and the pooled variance S^2 of the cell fate probabilities. To further quantify the depletion likelihood of each TF to the terminal state k, we used the normalized Mann-Whitney U statistic

$$\ell_d = \frac{1}{N_C^2}\sum_{n=1}^{N_C}\sum_{n^*=1}^{N_C}\mathbb{I}\left(\Pi_{:,k}(n) > \Pi^*_{:,k}(n^*)\right),$$

where $\mathbb{I}(\Pi(n) > \Pi^*(n^*))$ denotes the indicator function

$$\mathbb{I}(\Pi(n) > \Pi^*(n^*)) = \begin{cases} 1 & \text{if } \Pi(n) > \Pi^*(n^*) \\ 0 & \text{if } \Pi(n) \leq \Pi^*(n^*). \end{cases}$$

This statistic quantifies the probability that, in the predicted perturbation case, the cells will have lower cell fate probabilities toward the terminal state k than in the original unperturbed case.

Cell density perturbation effects

While depletion likelihood quantifies changes in fate probabilities, it does not reflect alterations in cell densities at terminal states. To address this limit, we introduce a complementary method to measure density shifts. Specifically, we use CellRank's VelocityKernel to compute two transition matrices: one from the original (control) velocity field, and one from the perturbed velocity field. For each matrix, we simulated random walks, starting from the initial state set C_{start}. After a fixed number of steps N_S, we record the frequency with which each terminal-state cell $j \in C_{\text{term}}$ is reached. These simulations are repeated N_R times to gain statistical properties.

Let $h_c^{(j)}$ denote the number of times the terminal cell j is visited across all simulations under the original vector field. The predicted density $\rho_c^{(j)}$ for terminal cell j in the control condition is defined as:

$$\rho_c^{(j)} = \frac{h_c^{(j)}}{N_R \cdot |C_{\text{start}}|}.$$

Similarly, let $\rho_p^{(j)}$ denote the predicted density under the perturbed vector field. To quantify the effect of perturbation on terminal cell j, we compute the density change likelihood using a scaled Boltzmann factor[180]:

$$\ell_{\Delta\rho} = \sigma\left(\frac{\rho_p^{(j)} - \rho_c^{(j)}}{k}\right),$$

where σ denotes the Sigmoid function, and scaling factor $k = \frac{1}{\sqrt{N_R \cdot |C_{\text{start}}|}}$ normalizes simulation variability.[181] This formulation yields a smooth, bounded likelihood that captures the effect of TF perturbation on terminal-state cell densities, analogous to MELD likelihood in Perturb-seq data. We simulated $N_R = 1000$ random walks overall, each terminated after $N_S = 100$.

Cell differentiation potential score

Consistent with previous studies defining the differentiation potential (DP) score,[10] we define a cell's DP as the entropy calculated over the matrix of cell fate probabilities.

$$\text{DP}_n = -\sum_{k=1}^{m}\Pi_{n;k}\log_2\left(\Pi_{n;k}\right).$$

Benchmarking metrics

Velocity inference

We assess inferred velocities based on the average Pearson correlation between the ground truth and predicted gene velocity.

Velocity estimation consistency

To assess the robustness of RNA velocity estimates across varying sequencing depths, we introduce the concept of velocity estimation consistency VEC, defined as the average pairwise correlation of cell velocity estimates across different sequencing depths: Let $\mathcal{D} = \{\mathcal{D}_1, \mathcal{D}_2, \ldots, \mathcal{D}_{N_D}\}$ be the set of datasets with different sequencing depths; let $v_k^{(n)} \in \mathbb{R}^{N_g}$ denote the velocity vector for cell n in dataset $\mathcal{D}_k$. For each cell n and pair of datasets (j,k), we then compute the correlation between the corresponding velocity vectors and take their average to find the velocity estimation consistency score VEC, i.e.,

$$\mathrm{VEC}(n) = \frac{2}{N_D(N_D - 1)} \sum_{1 \le j < k \le N_D} corr\left(v_j^{(n)}, v_k^{(n)}\right).$$

Latent time prediction

We define the latent time τ_n of cell n as the mean value of its gene-specific latent times, i.e.

$$\tau_n = \frac{1}{N_G} \sum_{g=1}^{N_G} t_{ng},$$

where t_{ng} denotes the latent time value in cell n and gene g. The aggregation of gene-level latent times into a cell-level or global latent time assumes that gene-specific temporal dynamics lie on a shared low-dimensional manifold; both RegVelo and veloVI infer these time estimates on the gene level jointly within a shared latent representation (z), thereby constraining the overall dynamics to this manifold. Under this assumption, averaging the gene-level times provides an approximation of a cell's progression along the latent trajectory.

Global pseudotime inference

The means over gene-specific latent time denote a local summary statistic of each cell. To compute a global, i.e., cell-level, pseudotime that reflects the overall position of each cell along the manifold of the fitted dynamics, we devised a graph-based global latent time ordering methods based diffusion pseudotime (DPT): we first compute pairwise distances d between cells i and j based on the correlation of their gene-specific latent time vectors, specifically:

$$d_{ij} = 1 - \mathrm{corr}\left(t_{i,:}, t_{j,:}\right).$$

From this distance matrix, we construct a k-nearest neighbor graph K to define a transition matrix $\mathcal{P}$ as

$$\mathcal{P}_{ij} = \frac{K_{ij}}{\sum_k K_{ik}}.$$

The diffusion map embedding $\Psi(n)$ of cell n is then defined as

$$\Psi(n) = (\lambda_1 \varphi_1(n), \ldots, \lambda_d \varphi_d(n)), \mathcal{P}\varphi_\ell = \lambda_\ell \varphi_\ell, \ell = 0, 1, \ldots, d.$$

Here φ_ℓ is the ℓ-th eigenvector, $\lambda_\ell \in [0,1]$ its eigenvalue, and d denotes the number of the diffusion components. Finally, the global pseudotime $\tilde{\tau}_n$ of cell n is the distance to the root cell r in diffusion space, i.e.,

$$\tilde{\tau}_n = \|\Psi(n) - \Psi(r)\|_2.$$

Here, we selected the root cell as the cell closest to the initial states ($t_g = 0$) across all genes, by minimizing mean squared distance:

$$r = arg\ \min_n \frac{1}{N_G} \sum_{g=1}^{N_G} (t_{ng} - 0)^2.$$

GRN inference

We benchmarked the GRN inference based on the area under the receiver operator characteristic curve (AUROC) using a set of ground truth TF-target connections E^* with scikit learn's roc_auc_score function.[182] Specifically, we defined the ROC curve using the following procedure:

1. Rank all possible TF-target pairs (j,k) based on the absolute value of their predicted edge weight $W_{j,k}$ in the inferred GRN.
2. for the TF-target pair (j,k), we define the ground truth label $y(j,k)$ as

$$y(j,k) = \begin{cases} 1 & \text{if } (j,k) \in E^* \\ 0 & \text{if } (j,k) \notin E^* \end{cases}$$

3. for thresholds $\tau \in [\tau_{\min}, \tau_{\max}]$, we define the true positive and false positive rates based on the estimator

$$\hat{y}(j,k|\tau) = \begin{cases} 0 & \text{if } |W_{j,k}| \leq \tau \\ 1 & \text{if } |W_{j,k}| > \tau \end{cases}$$

Here, we chose

$$\tau_{\min} = \min_{j,k \in \{1,\ldots N_G\}} |W_{j,k}|$$
$$\tau_{\max} = \max_{j,k \in \{1,\ldots N_G\}} |W_{j,k}|.$$

Parameter identifiability

Variational inference is sensitive to the parameter initialization because the ELBO is a nonconvex objective function.[183] We, thus, ensured the stability of estimates like the velocity, latent time, or gene regulatory network under different initialization by measuring the correlation of the estimates for each cell across multiple model runs. Here, we ran each probabilistic model overall five times and relied on Pearson correlation for all estimates except latent time where we computed the Spearman correlation to account for the ranked nature of the feature.

Velocity consistency

We expect the inferred vector field to be coherent in a uni-directional trajectory such as the cell cycle. We quantify a consistency score ζ for each cell n as the mean correlation of its velocity $v^{(n)}$ with its neighboring cell velocities, i.e.,

$$\zeta(n) = \frac{1}{|\{j \in \mathcal{N}(n)\}|} \sum_{j \in \mathcal{N}(n)} \mathrm{corr}\left(v^{(n)}, v_j\right),$$

where $\mathcal{N}(n)$ indicates the neighborhood of cell n and corr the Pearson correlation.

Cross-boundary correctness

Predicted cell state transitions need to align with ground truth cell state changes. Assuming we know the correct transition from a source cluster $\mathcal{C}_A$ to target cluster $\mathcal{C}_B$, we use the cross-boundary correctness (CBC) score to evaluate if our velocities align with the correct direction.

To compute the CBC score, we first identify the boundary between $\mathcal{C}_A$ as $\mathcal{C}_B$ as the set of all cells in $\mathcal{C}_A$ with at least one neighbor in $\mathcal{C}_B$

$$\{n \in \mathcal{C}_A | \exists j \in \mathcal{N}(n) : j \in \mathcal{C}_B\}.$$

We then define the heuristic velocity $v^{(h)}_{A \to B}(n)$ of observation $n \in \mathcal{C}_A$ as

$$v^{(h)}_{A \to B}(n) = \frac{1}{|\{j \in \mathcal{N}(n) \cap \mathcal{C}_B\}|} \sum_{j \in \mathcal{N}(n) \cap \mathcal{C}_B} (x_j - x_n).$$

Following, we calculated the CBC score as the Pearson correlation between the inferred velocity $v^{(n)}$ and heuristic velocity $v^{(h)}_{A \to B}(n)$. This score assesses the forward transition accuracy; we denote it as $\mathrm{CBC}^{(f)}(n)$.

This definition of the CBC score only quantifies if the velocity of boundary $\mathcal{C}_A$ cells correctly points to their target state but fails to capture $\mathcal{C}_B$ observations pointing toward their ancestor state. To quantify this incorrect flow, we consider the heuristic velocity from $\mathcal{C}_B$ to $\mathcal{C}_A$ and defined the backward CBC score $\mathrm{CBC}^{(b)}(n)$ as

$$\mathrm{CBC}^{(b)}(n) = -\mathrm{corr}\left(v^{(n)}, v^{(h)}_{B \to A}(n)\right),$$

with Pearson correlation corr.

Intuitively, the forward CBC quantifies how well the predicted velocities align with the expected transition from a source cluster $\mathcal{C}_A$ to a target cluster $\mathcal{C}_B$, thereby reflecting agreement with the correct direction of progression. In contrast, the backward CBC evaluates how strongly velocities in the target cluster $\mathcal{C}_B$ point away from the source cluster $\mathcal{C}_A$, such that higher backward CBC indicates fewer reverse transitions. An ideal velocity field is characterized by high forward CBC, indicating correct directional progression, and high backward CBC, indicating the reduced biologically implausible backflow. Together, these metrics provide a complementary assessment of both the alignment and directional correctness of velocity predictions.

Terminal state identification

CellRank[16,17] infers terminal states based on state change estimates combined with Markov chain theory. To benchmark different trajectory inference methods we rely on CellRank 2's terminal state identification[17] (TSI) score that quantifies how faithfully a method allows the recovery of terminal states with increasing number of macrostates that represent regions of the phenotypic manifold that cells are unlikely to leave. Given the ground-truth dynamics, CellRank recovers all terminal states of a given system and one terminal state for each macrostate considered.

We quantified TSI performance through the TSI score introduced by CellRank 2. First, we used CellRank's VelocityKernel κ to estimate cell-cell transition probabilities based on the inferred RNA velocities and subsequently defined macrostates and identified terminal states. We represented the predicted terminal states as the macrostate with stability exceeding the threshold τ. Then, for each

kernel matrix built upon the velocity estimates, we considered the step function f_κ that maps the number of macrostates to the number of correctly predicted terminal states. Considering m terminal states, an optimal method identifies terminal states according to

$$f_{\text{opt}}(j) = \begin{cases} j, & \text{if } j < m \\ m, & \text{if } j \geq m. \end{cases}.$$

Based on these definitions, we calculated the TSI score as the area under the curve f_κ relative to the area under the curve f_{opt}. We calculated the TSI score for thresholds τ taken from 20 evenly spaced values of the unit interval, i.e., $\tau = \frac{j}{20}, j \in \{0,\ldots,20\}$. Finally, we relied on a one-sided Welch's t test to identify the significantly best TSI performance.

Cell fate estimation consistency

To evaluate the robustness of cell fate predictions under varying sequencing depths and lineage complexities, we define the cell fate estimation consistency $\mathcal{F}$ as the agreement of fate probabilities across corresponding cells in dataset variations $\mathcal{D} = \{\mathcal{D}_1, \mathcal{D}_2, \ldots, \mathcal{D}_{N_D}\}$ of varying sequencing depth or lineage complexity; we assume consistent cell ordering across the N_D datasets. Let $\Pi^{(j)} \in \mathbb{R}^{N_C \times N_L}$ denote the cell fate probability matrix for dataset $\mathcal{D}_j$ across N_{textL} lineages. For each dataset pair $(\mathcal{D}_j, \mathcal{D}_k)$ and lineage ℓ we then compute the cell fate estimation consistency $\mathcal{F}(j,k)$ as

$$\mathcal{F}_\ell(j,k) = \rho\left(\Pi^{(j)}_{:,\ell}, \Pi^{(k)}_{:,\ell}\right),$$

with Spearman rank correlation ρ.

Gene ranking score

Each approach for inferring cellular dynamics allows identifying and ranking potential lineage-associated genes of every lineage with CellRank or, to compare to competing approaches, by relying on least action path (LAP) analysis. To assess this ranking, we curated a list of known lineage markers and regulators that an optimal method would rank the highest. Therefore, following the CellRank 2 method comparison workflow, we evaluated the performance of each method as follows: first, consider a lineage, a set of known lineage-associated genes D, and a method m. Each method m assigns a ranking to a gene $g \in G$, denoted by $r^{(m)}(g)$, and, for each threshold $\mathrm{T} \in \{1,\ldots,|G|\}$, we counted how many lineage-associated genes rank among the top T genes with

$$\varphi^m(\mathrm{T}) = |\{g | r^{(m)}(g) \leq \mathrm{T} \wedge g \in D\}|.$$

Next, to quantitatively evaluate the rankings, for each method m, we computed the area under the curve

$$\mathrm{AUC}(m) = \sum_{\mathrm{T}=1}^{|G|} \varphi^m(\mathrm{T}).$$

Since an optimal method ranks the lineage-associated genes highest, its AUC* is

$$\mathrm{AUC}^* = \frac{|D|(|D|+1)}{2} + (|G| - |D|)|D|.$$

Finally, we evaluate the performance of methods relative to an optimal ranking through

$$\mathrm{AUC}_{\text{rel}}(m) = \frac{\mathrm{AUC}(m)}{\mathrm{AUC}^*}.$$

Hyperparameters sensitivity analysis

We have benchmarked our model's sensitivity to both regularization parameters under the setting soft_constraint=True, using a dataset simulated with dyngen and different versions of the corresponding GRNs as a prior: by systematically corrupting the ground-truth network skeleton at varying rates (0.2, 0.4, 0.6, 0.8, and 1.0), we created ten prior GRNs for each corruption setting. For each corruption level, we applied RegVelo to infer model parameters across a range of regularization magnitudes at 10-fold increments (10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1, and 10).

Additionally, we created two datasets with different numbers of terminal states to account for the effects of lineage complexity: Dataset 1 (two terminal states) and Dataset 2 (nine terminal states). We systematically evaluated the models using several downstream metrics, including velocity, latent time, GRN inference, and model identifiability.

Selecting appropriate model using side information

RegVelo is required to define hyperparameters including soft_constraint and two regularization parameters. The soft_constraint parameter determines whether the model freezes the GRN based on the prior skeleton (False) or allows it to be updated (True). Both regularization parameters are used only when the model updates the GRN (soft_constraint=True).

To tune these hyperparameters on real datasets, we recommend selecting optimal values based on the predictive power of the estimated kinetic features (e.g., velocity and latent time) relative to known side information, such as pseudotime, developmental stage numbers, cell stemness, terminal states, and cell state transitions. Specifically, we used the following metrics to guide model selection.

Time prediction

Suppose we know the pseudotime or developmental stage, denoted as t_{pseudo} and t_{real} respectively, we assess how well the estimated latent time correlates with the known time using Spearman correlation:

$$\rho(\hat{t}, t_{real}) \text{ or } \rho(\hat{t}, t_{pseudo}),$$

where $\hat{t}$ denotes the RegVelo estimated cell specific latent time.

Stemness prediction

Given established cell differentiation potential or stemness scores, we compute the Spearman correlation between these values and the cell DP scores derived from the cell fate probability matrix obtained using the *VelocityKernel* in CellRank.[17]

Dynamics prediction

Suppose we know the correct terminal states and cell state transitions. We used TSI[17] and CBC[17] respectively to evaluate how accurately that model predicted known dynamic side-information.

Evaluating the impact of the GRN on dynamics inference

We developed two baseline models based on a pre-trained RegVelo model to assess the significance of GRNs in velocity inference. We created the first baseline, the *random GRN* model, by shuffling the estimated GRN from the pre-trained RegVelo model and used it as input for a new model. We retrained the original and new model but kept the GRN fixed. This approach allows assessing how the regulatory network affects the model's cellular velocity and latent time estimates.

In the second baseline model, the *remove GRN* model, we removed all regulatory influences. As with the first baseline, we fixed the modified GRN and trained the model until convergence. This model served as a control to evaluate the impact of entirely omitting regulatory information on velocity inference.

We quantified the performance of these baseline models compared to the original RegVelo model by calculating the CBC score and the Spearman correlation of latent time estimates with ground truth. To gain statistical significance, we repeated this process 30 times for each model and used the Wilcoxon signed-rank test to compare the original model with each baseline, *i.e.*, original versus random GRN and original versus removed GRN. This evaluation allowed us to determine the critical role that GRNs play in accurately modeling cellular dynamics.

Robustness to prior GRN estimates

RegVelo relies on a prior gene regulatory graph, and since this prior knowledge is usually inaccurate, the method must be robust to such imperfections. To assess the robustness of our method to inaccuracies in prior gene regulatory graphs, we utilized dyngen-simulated datasets[41] as described in the Datasets section. Following, we generated corrupted prior gene regulatory graphs by randomly replacing different fractions of existing gene regulations with nonexistent ones based on the original gene regulation graph used in the dyngen simulations.

We simulated five levels of corrupted fractions $\{0.2, 0.4, 0.6, 0.8, 1.0\}$, each generating five different corrupted prior graphs to account for corruption randomness. We ran RegVelo with each of these corrupted prior graphs and evaluated performance based on the AUROC score of the inferred GRN, and velocity and latent time correlation. We used these metrics to determine how well RegVelo maintained accuracy and reliability despite the corruption in the prior knowledge.

Evaluating the impact of the prior GRN and learned edge weights on driver prediction

During RegVelo driver prediction, there are two factors that play an important role: the prior GRN structure and the RegVelo learned edge weights from datasets. In the first comparison, to assess the impact of prior GRN structure to the driver ranking in RegVelo, we trained the RegVelo model while keeping the GRN fixed as the binary prior GRN. At the same time, we generated a randomized GRN by randomly shuffling the connections between TFs and target nodes. We then trained both models and evaluated the resulting driver ranking.

In the second comparison, we assessed the effects of RegVelo learned edge weight. We developed three models to compare with. In the first baseline model, we trained the RegVelo model with the GRN fixed as the prior GRN skeleton, using edge weights learned by SCENIC+. In the second model, we trained a full version of RegVelo, allowing it to learn the edge weights. In the third model, based on the trained RegVelo model, we shuffled the learned edges on the skeleton to create a randomized baseline.

After training all RegVelo models, we simulated the perturbed dynamical systems and calculated the depletion likelihoods to rank lineage driver genes. We quantified the performance of driver ranking using AUROC. Each model category was trained ten times to account for randomness.

Quantifying perturbation effects

Dynamo-based perturbation prediction

To perform perturbation prediction with dynamo,[23] we followed the corresponding tutorial provided in the Python package's documentation. In the first step, dynamo estimates cell velocities; we used dynamo.tl.dynamics to estimate kinetic rates and used

dynamo.tl.reduceDimension to perform the dimensional reduction. Next, we used dynamo.tl.cell_velocities with argument basis='pca' to project cell velocities to the PCA space. We further learned a vector field function based on this projection with dynamo.vf.VectorField with argument basis='pca'.

Dynamo provides two methods to simulate perturbation effects: *KO* or *expression perturbation*. For *KO*-based perturbation simulation, the method directly removes the regulation of the TF through masking the Jacobian matrix and generates a perturbed vector field. To assess downstream perturbation effects, we first used the dynamo-inferred velocities and used dynamo.pd.KO with argument store_vf_ko=True to predict the velocities after perturbing a certain TF, we then applied our cell density perturbation scoring method using the original and perturbed velocity fields.

For *expression perturbation*-based simulations, dynamo estimates perturbation effects by simulating perturbed gene expression and calculating the perturbation shift matrix, corresponding to the difference between the perturbed and the original gene expression. We computed the perturbation shift matrix on each cell through dynamo.pd.perturbation and quantified the perturbation score on each cell as the inner product between the perturbation shift and velocity matrices. Finally, we calculated the average perturbation score within each terminal cell type.

CellOracle-based perturbation prediction

To perform perturbation prediction with CellOracle,[26] we followed the tool's tutorial: after defining the oracle object with oracle = celloracle.oracle(), we injected the base gene regulatory graph via oracle.import_TF_data, calculated the PCA representation through oracle.perform_PCA and performed gene expression imputation with oracle.knn_imputation. Next, we estimated a GRN for each cell type with oracle.get_links and set the regularization parameter of Bayesian ridge regression alpha=10 to prevent overfitting during GRN inference. Finally, we filtered edges of the GRNs with links. filter_links and fitted the GRNs for perturbation simulation by oracle.fit_GRN_for_simulation.

To perform TF perturbation simulations with CellOracle, the method requires a pseudotime-based reference vector field defined in 2D space. We used Gradient_calculator to calculate the reference gradient based on a pseudotime or latent time inferred by velocity models. We then defined grid points with gradient.calculate_p_mass and gradient.calculate_mass_filter and projected the reference gradient onto the grid points using gradient.calculate_gradient. For each TF, oracle.simulate_shift simulated the shift vector, and oracle.estimate_transition_prob estimated the cell transition probability matrix; we relied on n_neighbors=30 neighbors and calculated the shift matrix in PCA space with oracle.calculate_embedding_shift. Next, we calculated the inner product between the perturbed shift vectors and reference velocity vectors in 2D space with calculate_inner_product and calculate_digitized_ip. Finally, we calculated the perturbation score within each lineage using get_negative_PS_p_value.

Data simulation

Toy simulation

We simulated a toy GRN containing six genes and three toggle switch network motifs (Figure S1A): following a previous simulation method based on Hill functions,[184] we used 12 stochastic differential equations (SDEs) to describe regulatory OR-gate dynamics among six genes A, B, C, D, E, and F, where toggle switches positively regulate each other.

$$du_A = \left[\frac{\kappa_-^{h_-}}{s_B(t)^{h_-}+\kappa_-^{h_-}} - \beta_A u_A(t)\right]dt + \sigma_s dW_t$$

$$ds_A = [\beta_A u_A(t) - \gamma_A s_A(t)]dt + \sigma_s dW_t$$

$$du_B = \left[\frac{\kappa_-^{h_-}}{s_A(t)^{h_-}+\kappa_-^{h_-}} - \beta_B u_B(t)\right]dt + \sigma_s dW_t$$

$$ds_B = [\beta_B u_B(t) - \gamma_B s_B(t)]dt + \sigma_s dW_t$$

$$du_C = \left[\alpha_{max}\left(\frac{s_A(t)^{h_+}}{s_A(t)^{h_+}+\kappa_+^{h_-}} + \frac{\kappa_-^{h_-}}{s_D(t)^{h_-}+\kappa_-^{h_-}}\right) - \beta_C u_C(t)\right]dt + \sigma_s dW_t$$

$$ds_C = [\beta_C u_C(t) - \gamma_C s_C(t)]dt + \sigma_s dW_t$$

$$du_D = \left[\alpha_{max}\left(\frac{s_A(t)^{h_+}}{s_A(t)^{h_+}+\kappa_+^{h_-}} + \frac{\kappa_-^{h_-}}{s_C(t)^{h_-}+\kappa_-^{h_-}}\right) - \beta_D u_D(t)\right]dt + \sigma_s dW_t ds_D = [\beta_D u_D(t) - \gamma_D s_D(t)]dt + \sigma_s dW_t$$

$$du_E = \left[\alpha_{max}\left(\frac{s_B(t)^{h_+}}{s_B(t)^{h_+}+\kappa_+^{h_-}} + \frac{\kappa_-^{h_-}}{s_F(t)^{h_-}+\kappa_-^{h_-}}\right) - \beta_E u_E(t)\right]dt + \sigma_s dW_t ds_E = [\beta_E u_E(t) - \gamma_E s_E(t)]dt + \sigma_s dW_t$$

$$du_F = \left[\alpha_{max}\left(\frac{s_B(t)^{h_+}}{s_B(t)^{h_+}+\kappa_+^{h_-}} + \frac{\kappa_-^{h_-}}{s_E(t)^{h_-}+\kappa_-^{h_-}}\right) - \beta_F u_F(t)\right]dt + \sigma_s dW_t$$

$$ds_F = [\beta_F u_F(t) - \gamma_F s_F(t)]dt + \sigma_s dW_t,$$

with $dW_t \sim \mathcal{N}(0,1)$ independently and identically distributed, and σ_s denoting the system noise level; based on a previously suggested noise level setting, we set $\sigma_s = 1$ for simulation. κ represents the dissociation constant set to $\kappa_+ = 4$ for activation and $\kappa_- = 2$ for inhibition. For hill coefficients h, we used $h_+ = 2$ and $h_- = 2$. These parameters result in a system that generates highly variable gene expression dynamics across all six genes, enabling us to study the effects of gene regulation.

We then simulated a time vector with 1500 Poisson-distributed time steps with parameter $\lambda = 1$ as suggested by scVelo. To align with scVelo's parameter simulation strategy, we sampled kinetic parameters $\theta = (\alpha_{\max}, \beta, \gamma)$ from a log-normal distribution where $\log(\theta) \sim \mathcal{N}(\mu, \Sigma)$. We set $\mu = (5, 0.5, 0.125)$ and defined a symmetric covariance matrix

$$\Sigma = \begin{bmatrix} 0.4^2 & 0.4^2 \cdot 0.2 & 0.4^2 \cdot 0.2 \\ 0.4^2 \cdot 0.2 & 0.4^2 & 0.4^2 \cdot 0.8 \\ 0.4^2 \cdot 0.2 & 0.4^2 \cdot 0.8 & 0.4^2 \end{bmatrix}$$

We simulated the dynamics with the SDE solver *torchsde*[185] with the Euler-Maruyama method[186] and sampled parameters and time values 100 times to generate 100 datasets under different conditions.

Dyngen simulation

Dyngen[41] allowed us to simulate gene expression dynamics in different ways with high-dimensional GRN and consider the effect of translation processes. To use dyngen for benchmarking, we followed the tool's tutorial with default settings to perform a single-cell dataset simulation. We simulated single-cell datasets with a bifurcating trajectory by setting the backbone through backbone <- backbone_bifurcating(); we considered the GRN and RNA velocity simultaneously by setting compute_cellwise_grn=TRUE and compute_rna_velocity=TRUE.

Following the guidelines in the dyngen tutorial, we simulated datasets with varying lineage complexity by tuning the modification_ points simulation parameter. We used four values $\{1, 2, 4, 6\}$ which generated simulated scRNA-seq datasets with 1, 3, 7, and 9 terminal states, respectively. To mimic real datasets, we simulated a limited number of regulators, randomly ranging from 50 to 80, and 800 target genes. In total we have simulated 1000 cells per dataset. This setup enabled us to benchmark methods across different levels of lineage complexity and perform a comprehensive evaluation. For each parameter setting, we generated 30 distinct datasets to account for simulation variability.

Simulating shallow sequencing depth

To simulate shallow-sequencing depths, we collected BAM files from Smart-seq3 datasets. We then used SAMtools view to downsample a specified proportion of total reads from each BAM file through the -s parameter with values $\{0.05, 0.1, 0.3, 0.5\}$. For each proportion, we then ran velocyto using the command velocyto run with the mm10 reference annotation[187] to generate.loom files containing both spliced and unspliced read counts.

QUANTIFICATION AND STATISTICAL ANALYSIS

General processing

We conducted all analyses using Scanpy v1.10.3[158] and scVelo v0.3.2,[13] using default parameters unless specified otherwise. We applied scVelo to filter out genes expressed in fewer than ten cells in the spliced and unspliced counts before normalizing their gene expression. If not stated otherwise, we selected the top 1000 highly variable genes (HVGs) based on gene dispersion parameters. We performed these filtering and normalization steps using scvelo.pp.filter_and_normalize. Next, we computed the top 30 PCA components and constructed a nearest-neighbor graph with 30 neighbors using scanpy.tl.pca and scanpy.pp.neighbors, respectively. We then calculated first-order moment matrices as smoothed spliced and unspliced gene expression on the nearest neighbor graph using scvelo.pp.moments. All velocity models take these moment matrices as input.

We applied gene-wise min-max scaling to preprocess both spliced and unspliced counts. Following the preprocessing employed by scVelo and veloVI, we applied scVelo's deterministic model and kept genes with a positive coefficient of determination $R^2 > 0$ for further RNA velocity inference because RegVelo's spliced equation with positive kinetic parameters β and γ implies a correlation between spliced and unspliced read counts; negative R^2 values indicate poor model fit, suggesting the gene may not be informative for dynamical analysis. However, even if some transcription factors (TFs) do not meet the velocity gene selection criteria, their regulatory functions remain crucial for understanding broader gene regulation during differentiation. Therefore, we retained all TFs contained in the highly variable gene set unless specified otherwise.

We downloaded lists of TFs for humans and mice from the cisTarget database of the SCENIC software,[18,24] and for zebrafish, we retrieved the corresponding information from the AnimalTFDB v4.0 database.[188] Finally, if not specified otherwise, we used the *soft constraint* mode to allow RegVelo to learn additional edges in the GRN.

Trajectory inference benchmark

We inputted all velocity estimations into the typical CellRank v2.0.6[16,17] workflow for downstream analyses. To compare velocity estimations, we used only CellRank's VelocityKernel to compute the transition matrix for macrostates estimation. Otherwise, we combined the VelocityKernel and ConnectivityKernel with weights 0.8 and 0.2, respectively, as shown in the CellRank tutorials. To

benchmark trajectory inference performance, we used the CBC score provided by the kernel method cbc and the TSI score supplied by the GPCCA estimator tsi[17]; the Benchmarking metrics section provides the relevant details.

Driver gene analysis

For each velocity-based method, we rely on CellRank to infer lineage-associated genes, following the tutorials for the VelocityKernel. Further, we computed terminal states and their corresponding fate probabilities using the GPCCA estimator. We used the compute_lineage_drivers function to identify genes that correlate with specific lineages.

To identify putative cell cycle activation-related drivers, we utilized cell cycle markers to compute cell cycle scores CCS for the G1, G2/M, and S phases by applying scvelo.tl.score_genes_cell_cycle. We defined the cell cycle score through

$$\frac{CCS_S - \min(CCS_S)}{\max(CCS_S) - \min(CCS_S)} - \frac{CCS_{G2/M} - \min(CCS_{G2/M})}{\max(CCS_{G2/M}) - \min(CCS_{G2/M})}$$

Finally, we calculated Pearson correlations between the expression levels of all TFs and this score to identify potential cell cycle regulators.

Perturbation screening

We simulated regulon knock-outs for each TF to identify lineage drivers through perturbation screening and estimated the resulting perturbed vector fields. After obtaining these perturbed velocities, we incorporated the perturbed and original velocities into the canonical CellRank workflow.

We used the GPCCA estimator[189] to calculate fate probabilities and compare these probabilities with Pearson correlation, allowing us to quantify the effect of each perturbation on the lineages. To ensure the robustness of these perturbation predictions, we ran RegVelo five times, each with a different initialization for the perturbation simulations. Finally, after assessing the effects of perturbation on each gene and terminal state, we averaged the results from all five runs to obtain the final prediction.

Benchmarking related methods

We followed the provided tutorials for each velocity method[13,14,29,31,32] to infer cell velocities. For scVelo, we first used scvelo.tl.recover_dynamics to estimate kinetic parameters and applied scvelo.tl.velocity with argument mode="dynamical" to calculate velocities. For UniTVelo v0.2.5.2, we estimated velocities with unitvelo.run_model with the argument FIT_OPTION="1". In the case of VeloVAE v0.1.2, we initialized the model using velovae.VAE with argument dim_z=10, then used velovae.train to estimate velocities. We also ran a full Bayesian VeloVAE model with rate prior set as 'alpha': (0, 1.0), 'beta': (0, 0.5); 'gamma': (0, 0.5), and initialized the model with additional argument full_vb=True.

For TFVelo v1.0, we estimated the velocities with TFvelo.tl.recover_dynamics. We then extracted the estimated GRN from adata.varm['fit_weights_final'], and used this GRN for the subsequent GRN benchmark. For cell2fate v1.0.0, we first applied cell2fate.utils.get_max_modules to estimate the number of modules, and estimated velocities through a cell2fate.Cell2fate_DynamicalModel model, trained with the train() method.

We followed the provided tutorials for each GRN inference method.[26,32,49,155] For CellOracle v0.20.0, we used the fit_All_genes function to compute the GRN, and for GRNBoost2 (from arboreto v0.1.6), we used the arboreto.algo.grnboost2 function to calculate the GRN. Importantly, GRNBoost2 does not estimate gene self-regulation but Pearson correlation always assigns self-regulation to every gene. To make the approach comparable to our method, we followed a previous GRN inference benchmark[42] and removed all edges corresponding to self-regulation from the predicted and ground truth GRNs when calculating the AUROC score. Since spliceJAC v0.0.1 requires pre-defined cell cluster information as input, we first ran scanpy.tl.leiden to find cell clusters, before applying splicejac.tl.estimate_jacobian for GRN estimation.

To benchmark RegVelo's driver ranking against other combinations of GRN inference (SCENIC+ v1.0a2,[24] Pando v1.1.1[25]), regulon activity (ULM[110] and AUCell[18] both from Rom decoupler v2.10.0, Linger v1.0.0[28]) and trajectory inference pipelines (scVelo,[13] DPT,[7] moscot v0.4.2[51]), we followed the official tutorials for each regulon activity inference method.[18,28,110] We employed the *decoupler*[110] package for estimating regulon activity due to its standardized implementation of multiple activity inference methods. Based on a prior GRN, we used the function decoupler.mt.ulm to compute regulon-activity for ULM, and decoupler.mt.aucell for AUCell.

LINGER is a deep learning–based method for inferring GRN from paired single-cell chromatin accessibility and gene expression data, enabling the estimation of TF-regulon activity. After preprocessing the single-cell multiome dataset following the LINGER tutorial, we inferred regulon activity: First, we used LINGER_tr.get_TSS to load transcription start site (TSS) locations for each gene and paired them with regulatory elements within a $\pm 200kb$ genomic window. We then applied LINGER_tr.RE_TG_dis pair regulatory elements with their target gene. Next, we trained the LINGER, using LINGER_tr.training with parameters activef='ReLU' and method='scNN'. To derive the population-level GRN, we inferred all potential regulatory interactions with LL_net.TF_RE_binding, LL_net.cis_reg, and LL_net.trans_reg. Finally, we estimated TF-regulon activity in gene expression data using the pre-trained GRN model and the function TF_activity.regulon.

We estimated cell fate probabilities using CellRank,[17] relying of the VelocityKernel, PseudoTimeKernel, or RealTimeKernel, depending on the data view we used – RNA velocity, pseudotime, or real-time information – to quantify cell-cell transition probabilities;

we used the GPCCA estimator[190] to compute cell fate probabilities. Finally, we ranked candidate driver regulators by correlating these probabilities with the estimated regulon activity.

To analysis our SDE-based simulation dataset, we implemented two distinct models to compare the Jacobian (J-L1) to the weight matrix (w-L1) regularization, each incorporating a distinct L1 loss function: for J-L1, we defined the loss function L as

$$L = -\mathrm{ELBO} + \lambda L_{\mathrm{Jacobian}},$$

with

$$L_{\mathrm{Jacobian}} = \frac{1}{N}\sum_{n=1}^{N} \|\mathrm{diag}(\mathrm{sigmoid}\circ(Ws_n + b))W\|_1,$$

where the operator $\circ$ applies a scalar function component-wise, and for w-L1

$$L = -\mathrm{ELBO} + \lambda |W|,$$

Simulated dataset – SDE simulation

where $|W|$ denotes the L1 norm of the weight matrix.

We tested each model with three distinct regularization parameter settings $\lambda \in \{0, 0.5, 1\}$ to evaluate the models' performance across different levels of regularization strength. We ran both models on each dataset and calculated the Spearman correlation of estimated latent time and ground truth, and AUROC to evaluate GRN prediction accuracy accordingly.

Simulated dataset – Dyngen simulation

For simulating datasets with dyngen,[41] we ran RegVelo with a fully connected prior GRN. After training and estimating velocities, latent time and GRN, we calculate velocity correlation, latent time prediction and GRN inference accuracy (*i.e.*, AUROC), with dyngen-defined ground truths.

Tuning regularization parameters

To benchmark the RegVelo's sensitivity to both regularization parameters (λ_1 and λ_2) under the setting soft_constraint=True, we used two dyngen-simulated datasets with two and nine terminal states, respectively, and constructed prior GRNs by systematically corrupting the ground-truth network skeleton at varying rates (0.2, 0.4, 0.6, 0.8, and 1.0). For each corruption level, we simulated five different prior GRNs, and applied RegVelo to infer model parameters across a range of regularization magnitudes at 10-fold increments (10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1, and 10). We calculated several downstream metrics, including velocity correlation, latent time prediction and GRN inference accuracy, to assess model performance; relatedly, we also calculated pair-wise correlations of estimated velocities, latent times and GRNs under different runs to evaluate model identifiability.

Mouse hematopoietic stem and progenitor cell differentiation

We followed the standard preprocessing pipeline and selected the top 3,000 HVGs. We then trained RegVelo three times to benchmark its performance against other methods. For consistency, RegVelo was trained without any prior GRN in both GRN inference benchmark and calibration analysis.

GRN benchmark

We utilized GRN benchmark pipeline, BEELINE[42]'s defined ground truth cell type specific GRN, which were curated from ENCODE,[191] ChIP-Atlas[192] and ESCAPE[193] datasets for ChIP-seq data from the hematopoietic cell types. We also incorporated BEELINE's defined GRN benchmark metric, the early precision rate (EPR),[42] to evaluate prediction precision.

Mouse neural crest development – Smart-seq2

We followed the original preprocessing pipeline to preprocess the dataset,[194] and our standard preprocessing pipeline for all other preprocessing followed. We then applied SCENIC[18] to the normalized dataset to infer the GRN with its default preprocessing and parameter settings.

Prior GRN definition

We followed a well-established guideline to run the Command Line Interface (CLI) version of SCENIC, using pySCENIC v0.12.1.[157] The pipeline involves three main steps: First, we used GRNBoost2 with the pyscenic grn command to infer the GRN, generating an adjacency matrix between TFs and target genes for regulon prediction. Second, we annotated motifs based on a curated list the cisTarget database and executed pyscenic ctx with the—all_modules parameter. Finally, we used the pyscenic aucell function to calculate regulon activity before binarizing the GRN by setting the weights of nonzero edges to 1 and all others to 0; we ran RegVelo with default settings and this binary graph as the prior gene regulatory graph.

Assessing dynamics across developmental stages

Using the preprocessed dataset, we defined four subsets of increasing lineage complexity by splitting the dataset based on developmental stages:

- Scale 1: neural crest cells (NCC), mesenchyme, satellite glia (SatGlia), and sensory neurons (sensory)
- Scale 2: scale 1 supplemented by Schwann cells (SC), endoneurial fibroblasts (enFib), and melanocytes

- Scale 3: scale 2 supplemented by gut neuron and Gut glia
- Scale 4: scale 3 supplemented by chromaffin cells (ChC) and sympathetic neurons (Symp).

For each scale we sought to compute terminal identification (TSI) and cross-boundary correctness (CBC) scores, using RegVelo, veloVI,[14] and scVelo's EM model.[13] For TSI evaluation, we defined 11 terminal states based on the original publication[194] – melanocytes, endoneurial fibroblasts, Schwann cells, mesenchyme, sensory neurons, chromaffin cells, satellite glia, gut glia, gut neuron, sympathetic neurons, and boundary cap cells (BCC) – to compute the TSI. For CBC comparisons we relied on transitions from the hub – a separating pool of cycling progenitors identified by the original work[194] and indicative of future cell state; we considered cell transitions toward the six lineages sensory neurons, satellite glia, Schwann cells, gut glia, gut neuron and chromaffin cells.

To validate estimation consistency, we computed the cell fate estimation consistency across the four scales.

Mouse neural crest development – sci-RNA-seq3

The original sci-RNA-seq datasets[195] include 500 million cells from 83 embryos and exhibit substantial batch effects. We therefore selected data from embryo 41 and extracted the neural crest-derived cell types, including dorsal root ganglion neurons, enteric neurons, melanocytes, myelinating Schwann cells, myelinating Schwann cells (Tgfb2+), neural crest (PNS glia), neural crest (PNS neurons), olfactory ensheathing cells, and otic sensory neurons. We then applied SCENIC to the normalized dataset using its default processing steps and parameter settings to infer the GRN.[117–124]

Prior GRN definition

We followed the exact same SCENIC-based[18] workflow as described for the smart-seq2 mouse neural crest development dataset to construct the prior GRN and run RegVelo.

Downstream analysis

We inputted the estimated velocity into the downstream CellRank workflow to compute ten macrostates and defined "dorsal root ganglion neurons", "enteric neurons", "melanocyte cells", "myelinating Schwann cells", "myelinating Schwann cells (Tgfb2+)" and "otic sensory neurons" as the terminal states.

Cell cycling dataset – U2OS cell line

After training the RegVelo model and estimating velocities, we calculated the CBC score with the ground truth differentiation directions, including G1 phase to S phase and S phase to G2/M phase; the original studies provided the phase label information.

We calculated the velocity consistency score using scvelo.tl.velocity_confidence. To further validate our results, we used the fluorescent ubiquitination-based cell-cycle indicator (FUCCI) derived score[39] as the proximal ground truth of cell cycling time and correlated it with inferred cell latent time using Spearman correlation.

Evaluate model robustness on shallow sequence depth

We performed downsampling with SAMtools[196] following the procedure that we have introduced, and compared RegVelo with scVelo and veloVI in terms of their inferred latent time accuracy, velocity consistency and the resulting CBC. Finally, we used the velocity estimation consistency score to assess the robustness of velocity estimates across different sequencing depths.

Global time ordering

RegVelo, together with scVelo and veloVI estimates the gene-wise latent time matrix $T \in \mathbb{R}^{N_C \times N_G}$. We constructed a cell-cell k-nearest neighbor graph using correlation distance on the gene-wise latent time matrix, and computed DPT from a root cell selected as the one with the lowest average gene-time. The resulting DPT pseudotime for each cell provides a globally temporal ordering.

Evaluating GRN inference

To evaluate GRN predictions through our AUROC score, we first defined the proximal ground truth GRN for the regulators by retrieving ground truth edges from ChIP-Atlas,[197] this resource provides putative binding scores of TF-target pairs derived from ChIP-seq datasets across different cell lines. We set the links with a MACS2 score greater than 500 as the proximal ground truth links; this stringent significance threshold, which corresponds to a q-value less than 10^{-50}, ensures that our defined proximal ground truth links are reliable.

To calculate the AUROC score for each TF, we followed our pipeline outlined in the Benchmark metrics section. To provide a comprehensive evaluation, we used the mean AUROC score across TFs with more than 50 targets in the cycling dataset to benchmark the GRN prediction.

Cell cycling dataset – RPE1 cell line

We applied the same preprocessing steps and evaluation metrics to the FUCCI-RPE1 data as in the FUCCI-U2OS analysis. Previous work has provided the phase label information on FUCCI-RPE1 data.[23,198]

Transcription rate prediction

Since FUCCI-RPE1 is based on scEU-seq,[198] which includes metabolic labeling information to capture time-resolved nascent RNA, it enables experimental quantification of gene-specific transcription rates. We used the transcription rates provided by the original study as ground truth and applied RegVelo to estimate transcription rates at the single-cell level. We then computed the Spearman correlation between RegVelo's inferred transcription rates and the ground truth for each gene.

For benchmarking against CellDancer,[156] we strictly followed the official pipeline described in their documentation.[156] Specifically, we first used the cd.adata_to_df_with_embed function to convert the AnnData object into the required input format. We then applied cd.velocity with the parameter permutation_ratio=0.1 to estimate gene-specific kinetic parameters for each cell.

Pancreatic endocrinogenesis

To infer the gene regulatory graph as prior knowledge for RegVelo, we began by utilizing the 10x multiome pancreatic endocrine datasets[51] of the same mouse line (Ngn3-Venus fusion) and the same embryonic stages (E14.5 and E15.5) to the scRNA-seq dataset used in the scVelo study.[50] We then applied Pando[25] to these 10x multiome datasets to infer the GRN with its default preprocessing and parameter settings.

Prior GRN definition

Pando's GRN inference consists of two steps: first, it scans for motifs in candidate cis-regulatory element (CRE) regions on the genome with the find_motifs function. Second, it infers the GRN by integrating gene expression and peak accessible information using the infer_grn function. The first step requires TF-motif binding information and position weight matrices (PSM). Here, we used the mouse_pwms_v2 dataset, which collects mouse motifs from the CIS-BP database and corresponding PSMs, from the R package chromVARmotifs[199] to provide this information. We inferred the prior GRN estimate through XGBoost[200] and used Signac[201] to link peaks with genes during the second step. We binarized the constructed GRN, *i.e.*, set weights of nonzero edges to 1 and 0 otherwise; we used this binary graph as the prior gene regulatory graph for RegVelo.

Downstream analysis

We inputted the estimated velocity into the downstream CellRank workflow, used eight macrostates, and defined "alpha", "beta", "delta", and "epsilon" as the terminal states.

epsilon cell substate identification

We defined sub-clusters in epsilon cells by first subsetting to the annotated epsilon cells from the pancreatic endocrine dataset. We then computed PCA embedding of this subset and constructed a neighborhood graph using 30 PCs and 30 neighbors with scanpy.tl.pca and scanpy.pp.neighbors functions. Next, we identified the sub-clusters using the Leiden algorithm (scanpy.tl.leiden). We defined the sub-cluster with the highest average alpha cell fate probabilities as State A (Figure S4B). To further characterize both sub-clusters, we used the Wilcoxon rank-sum test to identify differentially expressed genes.

Outbound score calculation

We used CellRank for computing transition probabilities from RegVelo-estimated velocities and define the outbound score to quantify the number of epsilon cells in terminal versus outbound states: let $P \in \mathbb{R}^{N_C \times N_C}$ be the cell-cell transition probability matrix computed by CellRank, where P_{jk} denotes the probability of cell j transitioning into cell k. Let $\mathcal{T} \subset \{1, \ldots, N_T\}$ be the index set of terminal state cells. For epsilon cells j, we calculated inbound and outbound probabilities

$$\text{Inbound}(j) = \sum_{k \in \mathcal{T}} P_{jk},$$

$$\text{Outbound}(j) = \sum_{k \notin \mathcal{T}} P_{jk}.$$

to define the outbound score as

$$\text{Outbound score}(j) = \text{Outbound}(j) - \text{Inbound}(j)$$

Pathway enrichment analysis

We calculated the ductal cell specific GRN and predicted targets via RegVelo. For each of the top five cell cycle regulators, we performed GO enrichment analysis using the enrichGO function from the ClusterProfiler[202] R package, with the parameters ont = "BP", pvalueCutoff = 0.1, and qvalueCutoff = 0.1. We focused on biological process GO terms with an FDR adjusted $P < 0.1$.

For GSEA analysis, we used the full ranking list of weights between downstream genes and each top cell cycle regulator. We focused on *hallmark* pathways and ran the analysis using the fgsea function from the *fgsea*[203] R package with default parameters.

Minimum cell numbers and sequence depth analysis

To determine the minimum number of cells and sequencing depth required for reliable modeling, we analyzed the sensitivity of dynamic inference to both factors. We assessed the impact of dataset size by downsampling to 500, 1,000, 2000 and 3,000 cells. In parallel, for each subset, we further downsampled the read counts to 5%, 10%, 25%, 50%, and 75% of the original sequencing depth via *_downsample_per_cell* function in scanpy.[158] To account for stochastic variation, we repeated each downsampling condition three times. We evaluated the effect of these dataset alternations on model performance using two metrics: the number of identified terminal states and velocity estimation consistency.

Human hematopoiesis

We conducted all analyses on the dataset preprocessed by the original study.[23] All other preprocessing followed our standard preprocessing pipeline.

Prior GRN inference

We curated prior gene regulatory graphs from two sources: putative gene regulation based on multi-omics datasets and experimental ChIP-seq datasets. We used a publicly available putative GRN inferred by the multiome GRN inference model *Dictys* on multi-omics bone marrow mononuclear cells.[27] Since this putative GRN may be incomplete, we also curated a GRN from human hematopoiesis ChIP-seq datasets of the UniBind database.[36,77] By merging both GRNs, we created a prior gene regulatory graph for RegVelo input.

We ran RegVelo with soft constraints, with unregularized ($\lambda_2 = 0$) and regularized ($\lambda_2 = 1$) settings for comparison. We inputted the estimated velocity into the downstream CellRank workflow to compute seven macrostates and defined "Mon", "Bas", "Ery", "Meg" and "Neu" as the terminal states.

Method comparison

We compared all velocity estimations with generic trajectory inference benchmark procedures, using the PCA representation as the input to calculate the CBC score. We calculated the forward CBC score and backward CBC score for each method by the kernel method cbc, with ground-truth transitions: "HSC" to "GMP-like", "HSC" to "MEP-like", "GMP-like" to "Mon", "MEP-like" to "Ery", "MEP-like" to "Meg" and "MEP-like" to "Bas". We further used Welch's t test to assess if the log ratio of CBC score of the regularized RegVelo model and the other model including unregularized RegVelo, scVelo, and veloVI models was significantly higher than zero.

To compare how RegVelo and veloVI predicted uncertainty in terminal states of the human hematopoiesis dataset, we first calculated the median uncertainty in progenitor cells (HSC, MEP-like, and GMP-like) and normalized the terminal state uncertainties based on this median. We then calculated the $\log_2$ ratio of the normalized uncertainties between veloVI and RegVelo. We applied Welch's t test to assess whether veloVI's uncertainty was significantly higher than RegVelo's in terminal states.

To evaluate the estimated gene rankings, for each system that we analyzed, we manually compiled a list of known driver TFs from the literature for each lineage (Table S2). For lineage-associated genes, we curated genes from CellMarker 2.0[204] and other relevant literature.[205]

Dynamo-based analysis

To predict the drivers using dynamo, we strictly followed the analysis pipeline provided by the tool's tutorial.[23] We calculated moment matrices using dynamo-inferred neighbor graph and group label="time". Following, we used the dynamo function dynamo.tl.dynamics to estimate cell velocities with arguments group="time", one_shot_method="sci_fate", model="deterministic".

With the stable fixed points given by the dynamo tutorials, we identified putative lineage drivers using dynamo's least action path analysis. Firstly, we projected the dynamo velocity field onto the PCA space with the function dynamo.tl.cell_velocities with basis="pca" and learned a vector field function based on this projection via the function dynamo.vf.VectorField with basis="pca". Next, we defined the terminal states as the 30 nearest neighbors computed in UMAP space of each stable fixed point using the function dynamo.tools.utils.nearest_neighbors. We computed the least actions path starting from each cell in the HSC cluster to each cell in the clusters labeled "Mon" or "Ery" with dynamo.pd.least_action. We further identified the genes correlated with the path using dynamo.pd.GeneTrajectory and ranked the lineage-associated genes with dynamo.vf.rank_genes.

Zebrafish neural crest development

Zebrafish Smart-seq3 and perturb-seq data analysis

To preprocess our Smart-seq3 data, fastq files for each Smart-seq3 pool library were merged into read 1, read 2, index 1, and index 2 four fastq files. Next, we applied zUMIs[160] to the fastq files with the expected barcodes file containing the Nextera v2 index combinations, the GRCz11 Ensembl 105 reference genome built by STAR v2.7.3a[206] and the parameters including barcode binning of 1, no automatic barcode detection, and the minimum number of reads per cell of 0.

Dynamic inference via RegVelo. We used the ultra-deep full-length Smart-seq3 scRNA-seq dataset during neural crest cell development for RegVelo implementation. We first filtered out genes expressing fewer than ten cells in spliced and unspliced matrices. To fully model the transcription regulation dynamics, we selected the top 8,000 genes for further downstream analysis.

We utilized the neural crest GRN[100] built by SCENIC+ using the 10x multiome datasets as the input for RegVelo. After training models, we input inferred velocity estimations into the CellRank workflow, proceeded with eight macrostates and defined "facial mesenchyme", "pharyngeal arch 2", "mNC hox34" and "pigment" as the terminal states in this system.

Evaluating terminal state identification and driver ranking at shallow sequence depth. To evaluate model robustness across different sequencing depths, we performed read-level downsampling of Smart-seq3 data on our zebrafish neural crest using SAMtools[196] to generate datasets at six sequencing levels: 0.5%, 1%, 5%, 10%, 30%, and 50% of the original read depth. Because SAMtools downsampling reduces the number of reads per cell rather than removing cells, we applied an additional filtering step to exclude cells with fewer than 5,000 reads after downsampling. This cutoff was chosen to prevent excessive cell loss (>90%) at the lowest depth (0.5%), retaining at least approximately 30% of cells for downstream benchmark. To ensure fair comparison of driver gene rankings across sequencing depths, we retained all TFs that were present in any dataset, even if they did not meet the filtering criteria at specific depths. After generating the downsampled datasets, we applied the TSI and driver gene ranking analyses to benchmark model performance under varying sequencing depths, and used the cell fate and velocity estimation consistency score to assess the robustness of estimation.

Driver gene prediction benchmarking. Besides our canonical lineage driver identification through TF expression via CellRank,[17] we also correlated TF activity with cell fate probabilities to rank TFs.

To compare RegVelo with different combinations of GRN and trajectory inference method, we first applied Pando,[25] SCENIC+[24,100] and Linger[28] to the preprocessed 10x neural crest multiome[100] dataset to infer the neural crest GRN, following our standard processing pipeline. For SCENIC+[24,100] and Pando-inferred TF regulons, we used AUCell[18] and ULM[110] to quantify TF activity in our Smart-seq3 gene expression data. For LINGER, we followed our standard pipeline to calculate TF activity in the Smart-seq3 zebrafish gene expression data using the TF_activity.regulon function. Following, we calculated the correlation of each regulon activity with cell fate probabilities estimated by VelocityKernel, PseudoTimeKernel, or RealTimeKernel, and ranked the genes according to their correlation coefficients within each lineage. Finally, we evaluated the identified putative drivers in facial mesenchymal and Pigment lineage using classical drivers as ground truth. Since the GRN estimated by LINGER did not include any facial mesenchymal drivers, we benchmarked its driver gene ranking only for the pigment cell lineage.

We evaluated the effect of GRN structure on ranking driver genes with RegVelo by shuffling the gene symbols on the nodes of the GRN to randomize its structure, simulating the perturbed dynamical system and calculating the depletion likelihood to rank the lineage driver genes.

Temporal dynamics ablation. We focused on the downstream regulation of nr2f5 and ets1 to illustrate how RegVelo links early drivers to lineage identity. For each TF, we ranked its targets by RegVelo-inferred weights, selected the top 10, and calculated Pearson correlations with CellRank lineage fate probabilities using the compute_lineage_drivers function. We defined targets with significant correlation ($q - value < 0.05$) and literature support for facial mesenchyme as lineage-associated genes. We visualized expression trajectories of TFs and their targets along latent time using cellrank.pl.gene_trends.

To evaluate temporal dynamics, we randomly downsampled the dataset into four subsets with the same number of cells, each covering progressively more developmental stages. We chose 220 cells for all subsets, matching the total number of cells from the earliest three stages. For each subset, we ran RegVelo five times and averaged TF weights to the lineage-associated genes across runs.

In silico gene expression perturbation prediction. We used RegVelo to generate both fitted values for spliced and unspliced readouts based on the original and perturbed dynamical systems. For each cell, we summed these values to represent the predicted gene expression before applying a Wilcoxon rank-sum test to identify differentially expressed genes; we computed $\log_2$ transformed fold changes (FC) between the perturbed and original gene expression profiles.

To estimate gene expression programs, we applied non-negative matrix factorization (NMF) to the concatenated matrix of original and perturbed gene expression values. We used the sklearn.decomposition.NMF function with n_components = 5 to compute the NMF components.

Perturbation effects prediction and benchmarking. We applied dynamo and CellOracle on the zebrafish dataset following the generic steps described in the Quantifying perturbation effect section. When incorporating dynamo inferred velocities into the CellRank workflow, we computed two transition matrices using the VelocityKernel and ConnectivityKernel and combined them with 0.8 and 0.2 weights, respectively; we computed eight macrostates and defined "facial mesenchyme", "pharyngeal arch 2", "mNC hox34" and "pigment" as the terminal states.

The CellOracle perturbation analysis required information on a base GRN and pseudotime; we used the SCENIC+-inferred GRN[106] as the base GRN structure. For pseudotime estimates, we estimated pseudotime with scVelo in an unsupervised manner, using scvelo.tl.recover_dynamics and scvelo.tl.velocity to estimate kinetics, followed by scvelo.tl.latent_time to estimate the latent time for each cell.

Quantification of in vivo perturbation effects. The zebrafish Perturb-seq data were processed following an established workflow.[100] Briefly, raw data were preprocessed with CellRanger using SC5P-R2 chemistry, aligning reads to the reference genome that incorporated mCherry and Citrine genes along with GRCz11 Ensembl 105. Cellular genotypes were assigned based on sgRNA counts, with ambient RNA contamination and doublets removed.

Cell clustering was performed using Seurat, and cell state annotations were enhanced by integrating multiome-gene expression (GEX) data with reciprocal PCA (RPCA).[207] Neural crest clusters were isolated for further clustering and visualized using the PHATE method.[208] Perturbation effects were quantified using MELD v1.0[126] by comparing each knockout condition to the non-targeting control, MELD then calculated depletion-enrichment likelihoods on the 0–1 scale. Latent time was inferred from the multiome dataset[100] via RPCA and averaging latent time of neighboring cells. Within specific lineages, cells were categorized into early and late stages based on median latent time.

The MELD score leverages density estimates of transcriptomics-derived manifolds and is, thus, ideal for continuous developmental data, without being unaffected by the granularity of discrete clustering.[126] Since the MELD score is a relative likelihood, it depends on the global distribution of all conditions in the dataset. For a given lineage, a value below 0.5 often indicates depletion, whereas a high score does not necessarily imply enrichment as it may result from either true enrichment or depletion in other lineages.

To rigorously assess whether a perturbation genuinely enriches or depletes a specific terminal cell state beyond random expectation, we developed a statistical test to evaluate the significance of observed MELD scores in a lineage-specific manner: first, we use CellRank 2's PseudotimeKernel to identify terminal states across the Perturb-seq dataset. Next, for each pair of gene knockout and corresponding control, we compute the average MELD likelihood within these terminal states. Since MELD likelihood is a relative measure, we assess the statistical significance of the perturbation effect using a permutation test. Specifically, for a given TF-lineage

(terminal state) pair, let ℓ_{obs} denote the observed mean MELD likelihood in the terminal state. We then generate a null distribution by randomly sampling the same number of cells as in the terminal state from the entire dataset for $N = 1000$ times, and compute their mean MELD likelihoods $\{\ell^{(1)}, \ell^{(2)}, ..., \ell^{(N)}\}$. Through this null distribution, we calculate the one-sided empirical p-value to determine whether the observed MELD score in the true terminal states is significantly higher (enrichment-like) or lower (depletion-like) to random expectation:

$$p_{emp}^{dep-like} = \frac{\sum_{i=1}^{N} \mathbb{I}(\ell^{(i)} \leq \ell_{obs})}{N}$$

$$p_{emp}^{enr-like} = \frac{\sum_{i=1}^{N} \mathbb{I}(\ell^{(i)} \geq \ell_{obs})}{N}$$

Perturbation benchmark. For each TF j and lineage m, we defined a ground truth label matrix $g(j,m)$ based on the empirical p-values from the MELD permutation test, indicating significant enrichment-like or depletion-like effects as follows:

$$g(j,m|\alpha) = \begin{cases} +1 & \text{if } P_{emp}^{enr-like}(j,m) \leq \alpha \\ -1 & \text{if } P_{emp}^{dep-like}(j,m) \leq \alpha \\ 0 & \text{otherwise} \end{cases}.$$

Here, α denotes a predefined significance threshold (0.05). We further used $P_{rgv}(j,m)$ and $\ell_{\Delta\rho}(j,m)$ to denote the test p-value and density change likelihood predicted by RegVelo for j-th TF and m-th lineage, respectively. We defined the following estimator:

$$\hat{g}(j,m|\alpha) = \begin{cases} +1 & \text{if } P_{rgv}(j,m) \leq \alpha \text{ and } \ell_{\Delta\rho}(j,m) > 0 \\ -1 & \text{if } P_{rgv}(j,m) \leq \alpha \text{ and } \ell_{\Delta\rho}(j,m) < 0 \\ 0 & \text{otherwise} \end{cases}.$$

We then calculated true positive (TP), false positive (FP) and false negative (FN) rate as follows:

$$TP = \sum_{j,m} \mathbb{I}(g(j,m) \neq 0 \wedge g(j,m) = \hat{g}(j,m)),$$

$$FP = \sum_{j,m} \mathbb{I}(\hat{g}(j,m) \neq 0 \wedge g(j,m) \neq \hat{g}(j,m)),$$

$$FN = \sum_{j,m} \mathbb{I}(g(j,m) \neq 0 \wedge \hat{g}(j,m) = 0).$$

Finally, we computed Precision and Recall as:

$$\text{Precision} = \frac{TP}{TP + FP}, \text{Recall} = \frac{TP}{TP + FN}$$

Since we also applied the cell density perturbation scoring method to dynamo KO, we computed the corresponding Precision and Recall values for dynamo as well. Additionally, to establish a random baseline, we generated randomized predictions of $\hat{g}(j,m|\alpha)$ by assigning values uniformly at random from $-1, 1, 0$ for each TF-lineage pair. Precision and Recall were then calculated based on these random estimators, we repeated this process for ten times to obtain baseline performance.

Because all model training was based on preprocessed Smart-seq3 data, we limited the evaluation to KO panels where the corresponding TFs included in the preprocessed and filtered Smart-seq3 dataset. As a result, we excluded the '*etv2*', '*erf*', and '*smarcc1a*' panels, along with any multiple TF knockout panels that include these TFs: '*fli1a_etv2*', '*fli1a_erf_erfl3*', and '*erf_erfl3*'. We excluded the *mitfa_tfec_inhouse* panel from benchmarking due to its use of IVT-synthesized sgRNA, which differs from the synthesis method used for all other knockout panels.[100] Therefore, for performance evaluation, we focused on two subsets: (i) single TF knockout predictions, and (ii) combined single and multiple TF knockout predictions (excluding the filtered panels), given the limited number of multiple TF knockout cases.

TF binding site prediction

To predict the TF binding sites, we used the interpretation of previously trained deep learning models[100] by ChromBPNet,[209] de novo motif discovery results[100] predicted by TFmodisco-lite[209] and hit calling results[100] by Finemo.[209] To predict the putative sox10 TF binding sites, we used the TFmodisco-inferred SoxE motif that had a q-value of 0.0044 compared to the DANIO-CODE[210] Sox10 motif (by TomTom[211]). Finemo analysis predicted that this inferred motif had hit events in the cluster mNC non-hox at chr4:6031867-6031883 and chr4:6019773-6019789 in the *tfec* nearby regions with the global hit coefficients of 3.86×10^{-5} and 9.28×10^{-5} and the hit cosine similarity of 0.84 and 0.91, respectively. This cosine similarity was computed between the untrimmed contribution weight matrix (CWM) of this motif and the contribution score where this motif hits. For visualization, we used Signac[201] to plot the pseudobulk chromatin accessibility tracks of the multiome ATAC data, and used the WashU epigenome Browser[212] to

visualize the ChromBPNet profile contribution scores, which indicated the base-resolution importance of DNA sequences for chromatin accessibility.

Biotin ChIP-seq and CUT&RUN data processing and analysis

The raw data were trimmed by trim galore, mapped by Bowtie2[161] onto the GRCz11 reference genome, deduplicated by Sambamba markdup,[213] and downsampled to equal read numbers by Sambamba view. To identify reproducible peaks, peak calling was first performed using MACS2[163] callpeak with the q-value cutoff of 0.01 between the Sox10-Avi immunoprecipitated samples and the matching input control, or with the q-value cutoff of 0.1 between the ELF1 antibody samples and IgG controls. Using IDR,[138] peaks were identified as the reproducible peaks from duplicates for each stage ($q < 0.05$, by Irreproducible Discovery Rate, IDR). FIMO[139] was used to predict ETS-family motif hits in Elf1 CUT&RUN reproducible peaks with the *p* value cutoff of 0.0001. The bigwig tracks and peak bed files were visualized using Signac[201] and IGV.[214]

Human limb development

We analyzed the single-cell myogenesis dataset using the version preprocessed by the original study.[79] We performed all additional preprocessing steps using our standard pipeline. For downstream RegVelo analysis, we used the SCENIC-inferred gene regulatory network (GRN) from the original study as the prior GRN.[79]

Model selection via real-time information

The original study included an attribute, adj_stage, which encodes the developmental stage of each cell. Using this information, we applied the ModelComparison class to select an appropriate RegVelo model with side_information = 'Real_Time'. We then benchmarked the selected model against veloVI and scVelo on cell state transition prediction, using CBC as the evaluation metric. Following the original study, we defined two ground truth cell state transition paths: PAX3+MyoProg → MyoB1 → MyoC1 → , and PAX3+PAX7+MyoProg → MyoB2 → MyoC2.

Perturbation analysis

Based on the original study,[79] we defined PAX7+MyoProg1, MYH3+MyoC, and MYL3+MyoC as terminal states. We screened the top driver genes for each state and recovered *MSC* as a key regulator for PAX7+MyoProg1.[79] To assess regulatory effects, we predicted gene expression changes following *MSC* regulon perturbation and compared the reconstructed expression profiles before and after perturbation. We focused on three maturation markers reported in the original study,[79] *MYOG*, *MYH3*, and *TNNT1*, as they were the only genes that passed our velocity gene filtering criteria.

Human embryonic hindbrain development

Integration scRNA-seq and single-cell multiome dataset

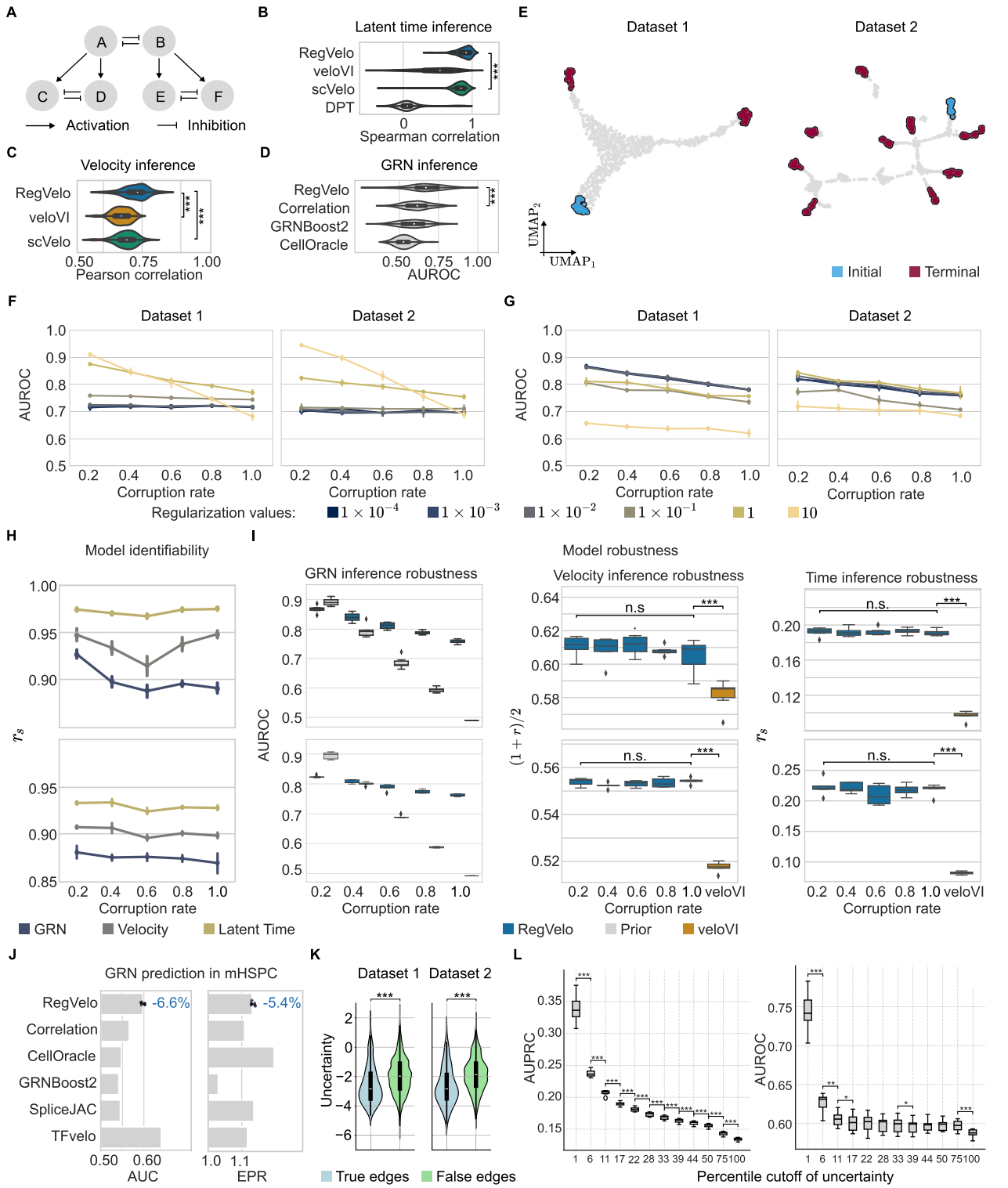
We combined a previously published whole brain single cell atlas with our own curated whole brain multiome dataset,[80] selecting cells from embryos less than 7 post-conception weeks (pcw) in order to capture defined trajectories from radial glia through neuroblast lineages. We integrated a final dataset of 850,848 cells and nuclei using scvi-tools v.1.2.0 with covariates of the experimental platform, cell preparation (cells or nuclei) and whether FACS was used for nuclei sorting. We built an scVI model with parameters num_hidden = 256, n_latent = 50, num_layers = 3, num_markers = 2000, dispersion = 'gene-batch' at a resolution of 2. We selected clusters where the majority of cells were manually labelled to derive from the hindbrain (labelled as Medulla, hindbrain, Cerebellum, Pons) into a final dataset of 52,914 cells and nuclei. This was re-integrated with the same scVI parameters at a resolution of 1. We constructed gene regulatory networks on multiome data using SCENIC+.

Model selection via cell-state transition

Based on the known differentiation hierarchy of hindbrain neuron cells,[215] we defined the known cell state transitions including: HBN → PCG, HBN → DI, HBN → DSN and HRG → VMN, and used this known cell state transition to select the candidate RegVelo model. We defined the neuron cell types including 'PCG', 'DI', 'DSN' and 'VMN' as the terminal states.

CellPress
OPEN ACCESS

Supplemental figures



(legend on next page)

Figure S1. Refining and testing RegVelo using simulated datasets, related to Figure 1

(A) Graphical representation of the GRN used for data simulation; each letter identifies one gene.

(B–D) Comparison of inferred latent times (B), velocities (C), and GRN (D) with ground truth using RegVelo, veloVI, scVelo (EM model), or diffusion pseudotime; ($n = 100$ each; one-sided Welch's t test. Latent time: $p = 5.41 \times 10^{-5}$; velocities: RegVelo versus veloVI $p = 7.13 \times 10^{-8}$, RegVelo versus scVelo $p = 3.31 \times 10^{-9}$; GRN: $p = 1.91 \times 10^{-5}$).

(E) UMAP of 1,000 cells of two *dyngen* simulated datasets with varying terminal states.

(F and G) Comparison of GRN prediction using six levels of regularization parameters, including λ_1 (F) and λ_2 (G) with altered prior GRN (STAR Methods) in two dyngen-simulated datasets: Dataset 1 (two terminal states); Dataset 2 (nine terminal states). Mean performance 95% confidence interval (CI), $n = 5$ corrupted priors per rate.

(H) Pairwise correlation of GRN (yellow), velocity (brown), and latent time (purple) estimates across RegVelo runs ($n = 5$) with altered prior GRNs (STAR Methods) in Dataset 1 (top) and Dataset 2 (bottom).

(I) Benchmark model robustness in Dataset 1 (top) and Dataset 2 (bottom); Left: GRN inference accuracy of corrupted prior GRNs (gray) versus RegVelo's estimated GRNs (blue). Middle and Right: Pearson correlation between ground truth and estimated statistics—velocities and latent time, compared with veloVI (orange). Box plots show the median (center line), interquartile range (hinges), and whiskers extending to 1.5× the interquartile range ($n = 5$). One-sided Welch's t test. Dataset 1: (velocity: $p = 1.21 \times 10^{-6}$, time: $p < 10 \times 10^{-11}$); Dataset 2: (velocity: 4.76×10^{-10}, time: 1.11×10^{-9}); n.s.: not significant.

(J) GRN benchmarking on the mouse HSPC dataset[43] via AUC (AUROC) and EPR metrics ($n = 3$ runs). We report the performance "delta" in AUC scores of RegVelo compared with the top competitor for this metric, TFvelo, and the performance "delta" in EPR scores of RegVelo compared with the top competitor for this metric, CellOracle.

(K) Comparison of GRN edge uncertainty (true versus false edges; one-sided Welch's t test, Dataset 1: $n_{edges} = 8{,}502$, $p < 1 \times 10^{-10}$; Dataset 2: $n_{edges} = 44{,}704$, $p < 1 \times 10^{-10}$).

(L) RegVelo learned GRN edge weights from the mouse HSPC dataset were grouped into cumulative bins based on bootstrap uncertainty. For each bin, the AUPRC (left) and AUROC (right) were calculated against the gold-standard network.

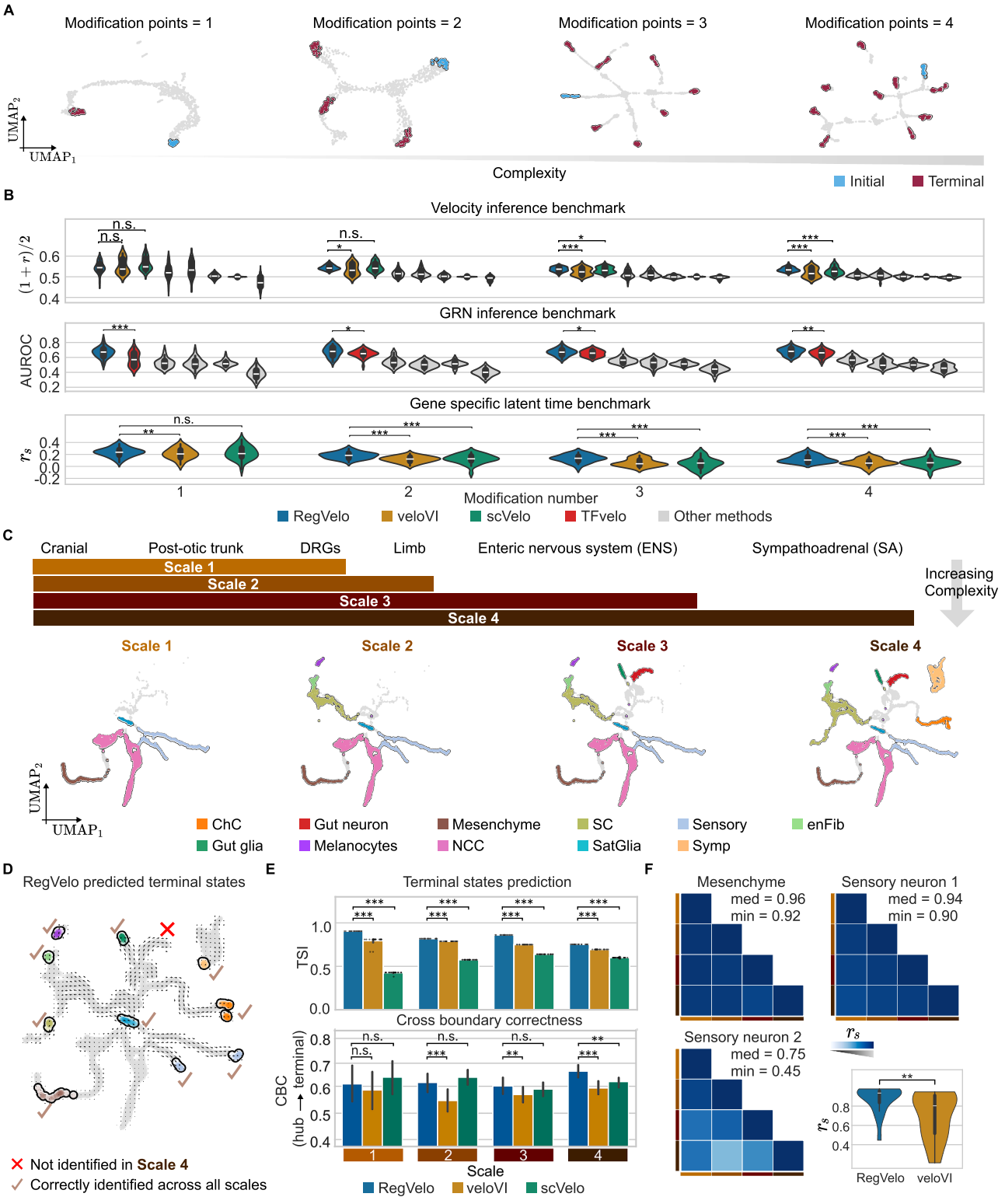


Figure S2. Benchmark RegVelo with varying lineage complexity, related to Figure 1

(A) UMAP embedding of 1,000 *dyngen* simulated cells with a varying number of terminal states, *dyngen* simulated true velocity is used to determine the initial and terminal states via CellRank.[17] *Modification points* are the model parameters of *dyngen* to control lineage complexity.[41]

(legend continued on next page)

(B) Comparison of estimated velocities (top), GRN (middle), and latent time (bottom) against ground truth. Box plots indicate the median (center line) and interquartile range (hinges). (RegVelo versus other methods; one-sided paired *t* test; $n = 30$ each; Velocity: Modification point = 1, scVelo: $p = 0.99$, veloVI: $p = 0.83$; Modification point = 2, scVelo: $p = 0.84$, veloVI: $p = 0.01$; Modification point = 3, scVelo: $p = 0.09$, veloVI: $p = 0.0001$; Modification point = 4, scVelo: $p = 0.0002$, veloVI: $p = 1.84 \times 10^{-6}$. GRN inference benchmark: Modification point = 1, TFvelo: $p = 7.69 \times 10^{-6}$; Modification point = 2, TFvelo: $p = 0.027$; Modification point = 3, TFvelo: $p = 0.016$; Modification point = 4, TFvelo: $p = 0.028$. Latent time benchmark: Modification point = 1, scVelo: $p = 0.28$, veloVI: $p = 0.004$; Modification point = 2, scVelo: $p = 9.83 \times 10^{-6}$, veloVI: $p = 8.692 \times 10^{-8}$; Modification point = 3, scVelo: $p = 1.53 \times 10^{-6}$, veloVI: $p = 8.25 \times 10^{-9}$; Modification point = 4, scVelo: $p = 2.56 \times 10^{-6}$, veloVI: $p = 3.17 \times 10^{-6}$. Other velocity inference methods include (from left to right): VeloVAE (fullvb), VeloVAE (vae), TFvelo, cell2fate, UniTVelo; other GRN inference methods, including (from left to right): GRNBoost2, CellOracle, spliceJAC, Pearson correlation.
(C) Defining four complexity scales based on developmental stages (top); and corresponding UMAP embeddings (bottom; Scale 1: $n = 3,926$ cells; Scale 2: $n = 5,139$ cells; Scale 3: $n = 6,273$ cells; Scale 4: $n = 8,821$ cells)
(D) Using RegVelo and CellRank identifies nine out of ten terminal states in Scale 4.
(E) Barplot of TSI (left) and CBC (right) score for RegVelo, veloVI, and scVelo across four scales. Line styles and colors indicate different methods. (One-sided paired *t* test; TSI: $n = 20$, $p < 1 \times 10^{-10}$ for all scales. RegVelo versus other methods, CBC: Scale 1: $n = 103$, veloVI $p = 0.33$, scVelo $p = 0.69$; Scale 2: $n = 280$, veloVI $p = 0.00019$, scVelo $p = 0.91$; Scale 3: $n = 429$, veloVI $p = 0.008$, scVelo $p = 0.249$; Scale 4: $n = 614$, veloVI $p = 4.29 \times 10^{-10}$, scVelo $p = 0.00014$). Error bars: 95% CI.
(F) Pairwise correlation of fate probabilities per lineage across complexity scales. Comparing the fate probability estimation consistency between RegVelo and veloVI ($n = 6$, one-sided Welch's *t* test, $p = 0.006$). scVelo was excluded as it failed to identify the Mesenchyme terminal state.

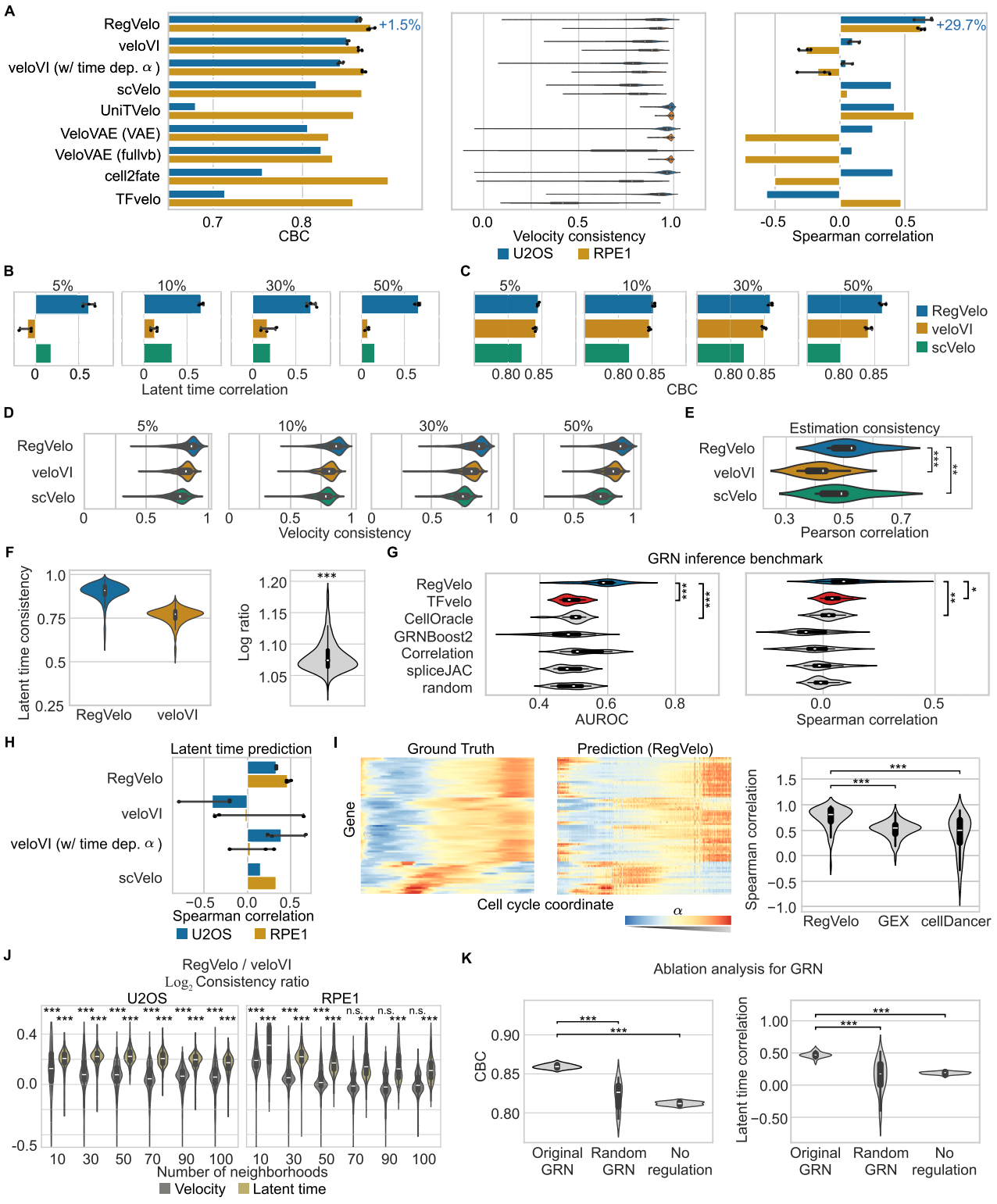


Figure S3. Benchmarking RegVelo against competing approaches on cell cycling datasets, related to Figure 1

(A) CBC score of RegVelo and veloVI (n = 3 models), and six other velocity inference methods (left), velocity consistency (middle), and correlation between ground truth and inferred latent times (right). Box plots indicate the median (center line), interquartile range (hinges), and 1.5× interquartile range (whiskers) (U2OS cell

(legend continued on next page)

line: $n = 1,146$ cells; RPE1 cell line: $n = 2,793$ cells). We also report the improvement in the average CBC values of RegVelo over the top competitor, veloVI, based on results from both cell lines.

(B–D) Latent time correlation (B), CBC (C), and velocity consistency (D) of RegVelo, veloVI ($n = 3$ models) under varying reads downsampling level.

(E) Comparison of velocity estimation consistency of RegVelo, scVelo, and veloVI ($n = 6$, one-sided Welch's t test; RegVelo versus scVelo: $p = 0.0019$; RegVelo versus veloVI: $p = 2.13 \times 10^{-5}$)

(F) Latent time consistency (left) and $\log_2$ ratio (right) of RegVelo's and veloVI's consistency score. Box plots indicate the median (center line), interquartile range (hinges), and 1.5× interquartile range (whiskers) ($n = 1,146$; one-sided Welch's t test, $p < 1 \times 10^{-10}$).

(G) Comparison of GRNs inferred by GRN inference methods and a random baseline to a ChIP-Atlas-derived proximal ground truth GRN. We assessed performance based on an AUROC (left) and Spearman correlation between ChIP-Atlas-derived TF-target binding score and weight estimates (right). We used a one-sided Welch's t test to assess significance ($n = 15$ TFs, one-sided Welch's t test, AUROC: RegVelo versus TFvelo $p = 8.7 \times 10^{-6}$, RegVelo versus CellOracle $p = 1.19 \times 10^{-6}$; Spearman correlation: RegVelo versus TFvelo $p = 0.019$, RegVelo versus CellOracle $p = 0.0016$). Box plots indicate the median (center line), interquartile range (hinges), and 1.5× interquartile range (whiskers).

(H) Latent time prediction with graph-based ordering (STAR Methods) of RegVelo ($n = 3$) versus three other velocity models in two cell lines data.

(I) Ground-truth and predicted transcription rate on cell cycle coordinate defined in original study[198] (left); Spearman correlation between ground truth and inferred transcription rate from RegVelo and CellDancer, or directly using gene expression (GEX).

(J) Comparison of RegVelo and veloVI model based on cell-wise velocity consistency on both cell lines across different numbers of neighborhoods when calculating the moment matrix (U2OS cell line: $n = 1,146$ cells; RPE1 cell line: $n = 2,793$ cells; one-sided Welch's t test, $p < 1 \times 10^{-10}$).

(K) CBC score (left) and Spearman correlation between ground truth and estimated latent time (right), using original GRN (original), randomly shuffled GRN (random), or no GRN (No regulation) to run RegVelo in the U2OS cell line. ($n = 30$ runs, one-sided Welch's t test, $p < 1 \times 10^{-10}$); box plots indicate the median (center line), interquartile range (hinges), and 1.5× interquartile range (whiskers).

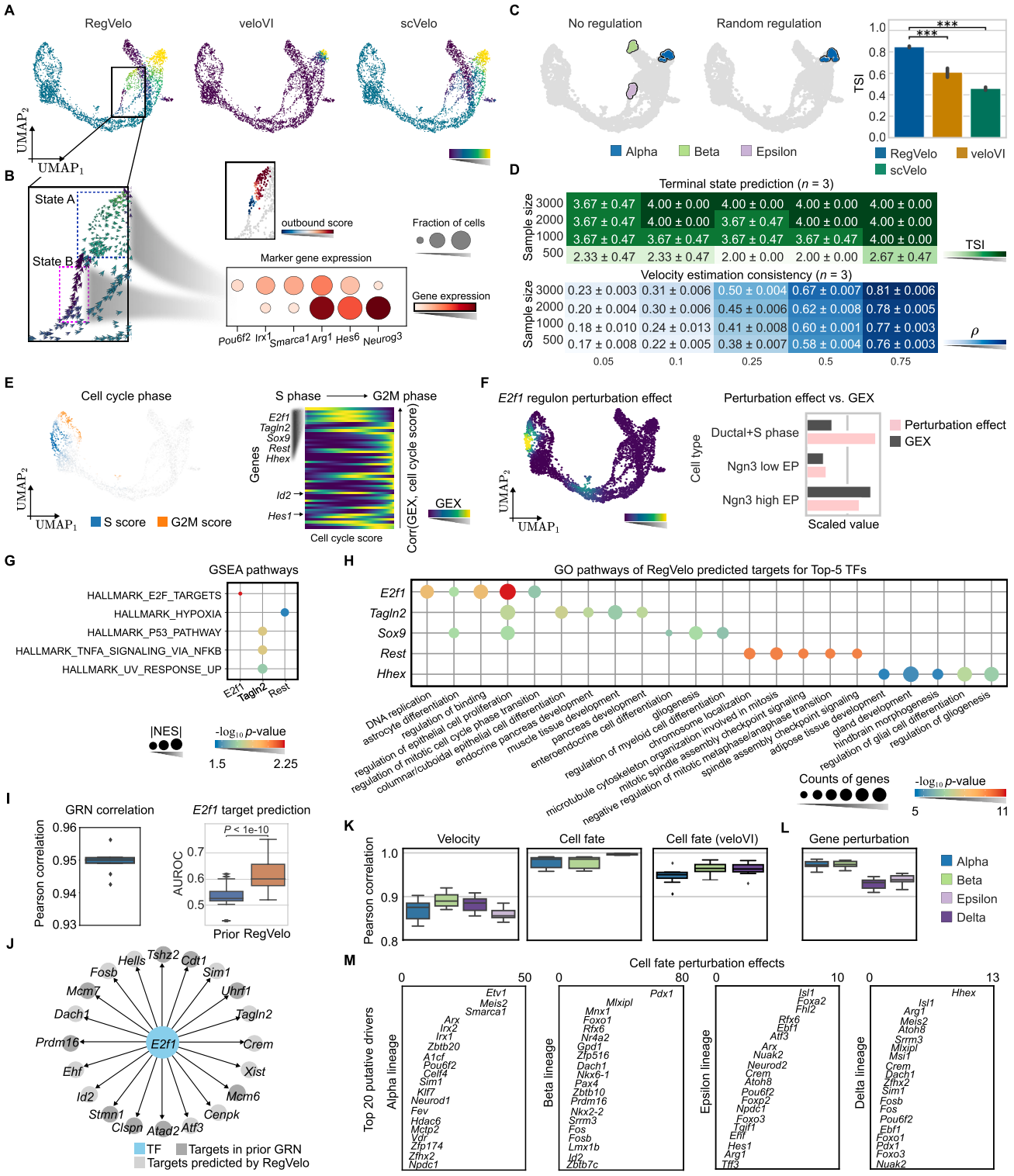


Figure S4. Unraveling cell fate dynamics and drivers in pancreatic endocrinogenesis using RegVelo, related to Figure 2

(A) UMAP of pancreatic endocrinogenesis data, colored by fate probabilities using velocity estimates from RegVelo (left), veloVI (middle), and scVelo (right).

(B) UMAP embedding of velocity stream (left) and outbound score (STAR Methods; right upper) from (A) zoomed in on epsilon cells and marker gene expression in two substates (bottom right; STAR Methods).

(legend continued on next page)

(C) UMAP colored by identified terminal states using CellRank with RegVelo when omitting any regulation (left) or relying on a random GRN (middle; STAR Methods); Comparison of the TSI (right) score using RegVelo (blue), veloVI (orange), and scVelo (green) ($n = 20$ runs). ($n = 20$; one-sided Welch's t test; RegVelo versus veloVI: $p = 9.23 \times 10^{-8}$; RegVelo versus scVelo: $p < 1 \times 10^{-10}$; Error bars: 95% CI.)
(D) Number of identified terminal states and velocity estimation consistency of RegVelo under varying numbers of downsampled cells and percentage of retained counts.
(E) UMAP colored by inferred S and G2M phases based on cell cycle score derived from gene expression (left); Smoothed gene expression along the cell cycling staging for the top 50 cell cycle score-correlated TFs, sorted by correlation from high to low (right).
(F) UMAP colored by local perturbation effects of *E2F1* regulon (left); Comparison of perturbation score and expression of *E2F1* in different cell types (right).
(G) Enriched MSigDB hallmark pathways for the top five TFs' targets predicted by RegVelo, identified via GSEA analysis. (Permutation test; Benjamini and Hochberg (BH) method adjusted p value.) No enriched pathways identified for *Sox9* and *Hhex*.
(H) Top-5 enriched GO biological process pathways for the top five TFs' targets predicted by RegVelo (Fisher's exact test; BH-adjusted p value).
(I) Correlation of RegVelo-inferred GRN derived from pairs of runs of the RegVelo model (left, $n = 5$ runs); Comparison of the AUROC of the GRN prediction using the prior GRN (blue) and RegVelo-inferred GRN (orange) (right, STAR Methods). Box plots indicate the median (center line), interquartile range (hinges), and 1.5× interquartile range (whiskers) ($n = 100$ comparisons; one-sided Wilcoxon signed-rank test; $p < 1 \times 10^{-10}$).
(J) Top 20 targets of *E2F1* predicted by RegVelo.
(K and L) Pearson correlation of RegVelo inferred velocities (K, left), CellRank-computed cell fate probabilities based on RegVelo's (K, middle) and veloVI's (K, right) velocity estimates, and cell fate perturbation score across repeated model fits (L). Box plots indicate the median (center line), interquartile range (hinges), and 1.5× interquartile range (whiskers) ($n = 5$ runs each).
(M) Top 20 putative lineage drivers ranked by depletion strength, with the y axis representing the negative t-statistics.

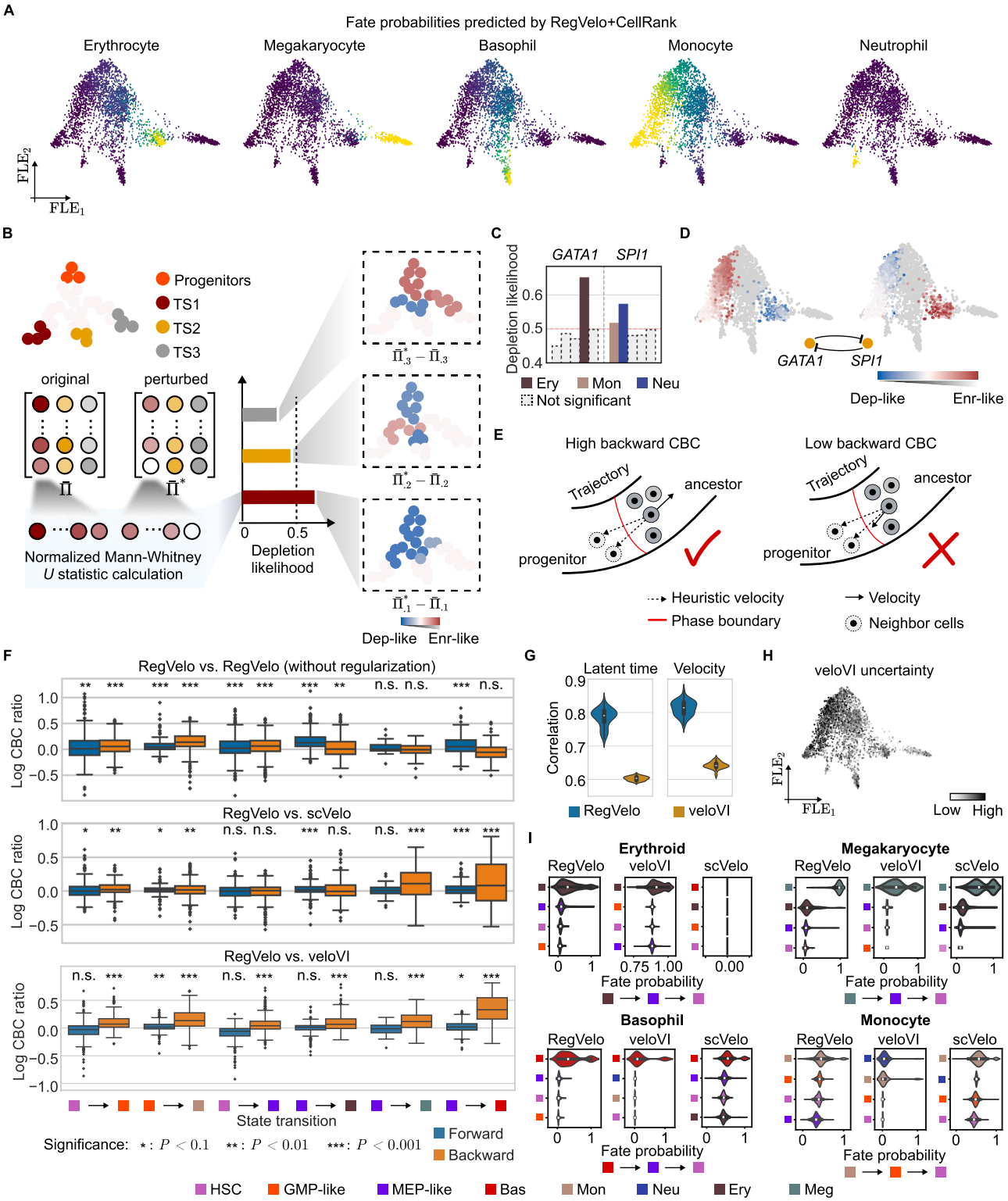


Figure S5. Benchmarking velocity inference for human hematopoiesis, related to Figure 3

(A) CellRank-computed fate probabilities using RegVelo-inferred velocities.

(B) Flowchart for estimating *depletion likelihood* (STAR Methods) from cell fate probabilities. Depletion likelihood is a normalized Mann-Whitney U statistic that quantifies the relative tendency of a cell fate to be depleted upon TF perturbation.

(C) Predicted depletion likelihood for the *GATA1* (left) and *SPI1* (right) (Wilcoxon rank sum test).

(legend continued on next page)

(D) FLE colored by RegVelo's cell-specific perturbation effect after deleting the *GATA1* (left) and *SPI1* (right) regulon without informative prior.

(E) Each panel shows velocity vectors (solid arrows) and heuristic velocities (dashed arrows) across a phase boundary (red line) separating progenitor and ancestor states along a developmental trajectory. A high backward CBC indicates that predicted velocities diverge from the backward heuristic (left), while a low backward CBC indicates closer alignment with it (right).

(F) Log-transformed ratio of CBC score considering both forward and backward flows of cell type transitions of RegVelo with or without Jacobian regularization (top), scVelo (middle), and veloVI (bottom) (HSC, hematopoietic stem cell, Meg, megakaryocyte, Ery, erythrocyte, Bas, basophil, MEP: megakaryocyte and erythrocyte progenitor-like, GMP, granulocyte and monocyte progenitor; one-sided Welch's *t* tests; STAR Methods). Box plots indicate the median (center line), interquartile range (hinges), and 1.5× interquartile range (whiskers). Upper: HSC → GMP-like (forward: $n = 304$ cells, $p = 0.001$; backward $n = 161$ cells, $p = 6.39 \times 10^{-6}$); GMP-like → Mon (forward: $n = 152$ cells, $p = 1.23 \times 10^{-7}$; backward: $n = 376$ cells, $p = 8.24 \times 10^{-38}$). HSC → MEP-like (forward: $n = 309$ cells, $p = 0.0001$; backward: $n = 446$ cells, $p = 3.17 \times 10^{-10}$). MEP-like → Ery (forward: $n = 312$ cells, $p = 6.46 \times 10^{-41}$; backward: $n = 220$ cells, $p = 0.012$). MEP-like → Meg (forward: $n = 16$ cells, $p = 0.157$; backward: $n = 66$ cells, $p = 0.835$). MEP-like → Bas (forward: $n = 161$ cells, $p = 1.33 \times 10^{-6}$; backward: $n = 144$ cells, $p = 0.99$). Middle: HSC → GMP-like (forward: $p = 0.032$; backward: $p = 0.0008$). GMP-like → Mon (forward: $p = 0.031$; backward: $p = 0.0008$). HSC → MEP-like (forward: $p = 0.58$; backward: $p = 0.99$). MEP-like → Ery (forward: $p = 1.71 \times 10^{-5}$; backward: $p = 0.18$). MEP-like → Meg (forward: $p = 0.31$; backward: $p = 6.53 \times 10^{-6}$). MEP-like →Bas (forward: $p = 0.0016$; backward: $p = 1.16 \times 10^{-7}$). Bottom: HSC → GMP-like (forward: $p = 0.99$; backward: $p = 4.93 \times 10^{-16}$). GMP-like → Mon (forward: $p = 0.012$; backward: $p = 1.96 \times 10^{-44}$); HSC →MEP-like (forward: $p = 1$; backward: $p = 2.13 \times 10^{-18}$). MEP-like → Ery (forward: $p = 0.453$; backward: $p = 4.61 \times 10^{-18}$). MEP-like → Meg (forward: $p = 0.655$; backward: $p = 1.02 \times 10^{-7}$). MEP-like → Bas (forward: $p = 0.038$; backward: $P = 1.19 \times 10^{-39}$).

(G) Latent time correlation and velocity estimates consistency across multiple runs using RegVelo and veloVI. Box plots indicate the median (center line), interquartile range (hinges), and 1.5× interquartile range (whiskers) ($n = 5$ runs).

(H) FLE colored by veloVI's cell uncertainty.

(I) Distribution of CellRank's fate probabilities toward four terminal states using RegVelo's, veloVI's, and scVelo's velocities. Ground truth cell state orderings are presented at the bottom of each panel. For each lineage, we show the four cell types with the highest median cell fate probability for each method in the lineage.

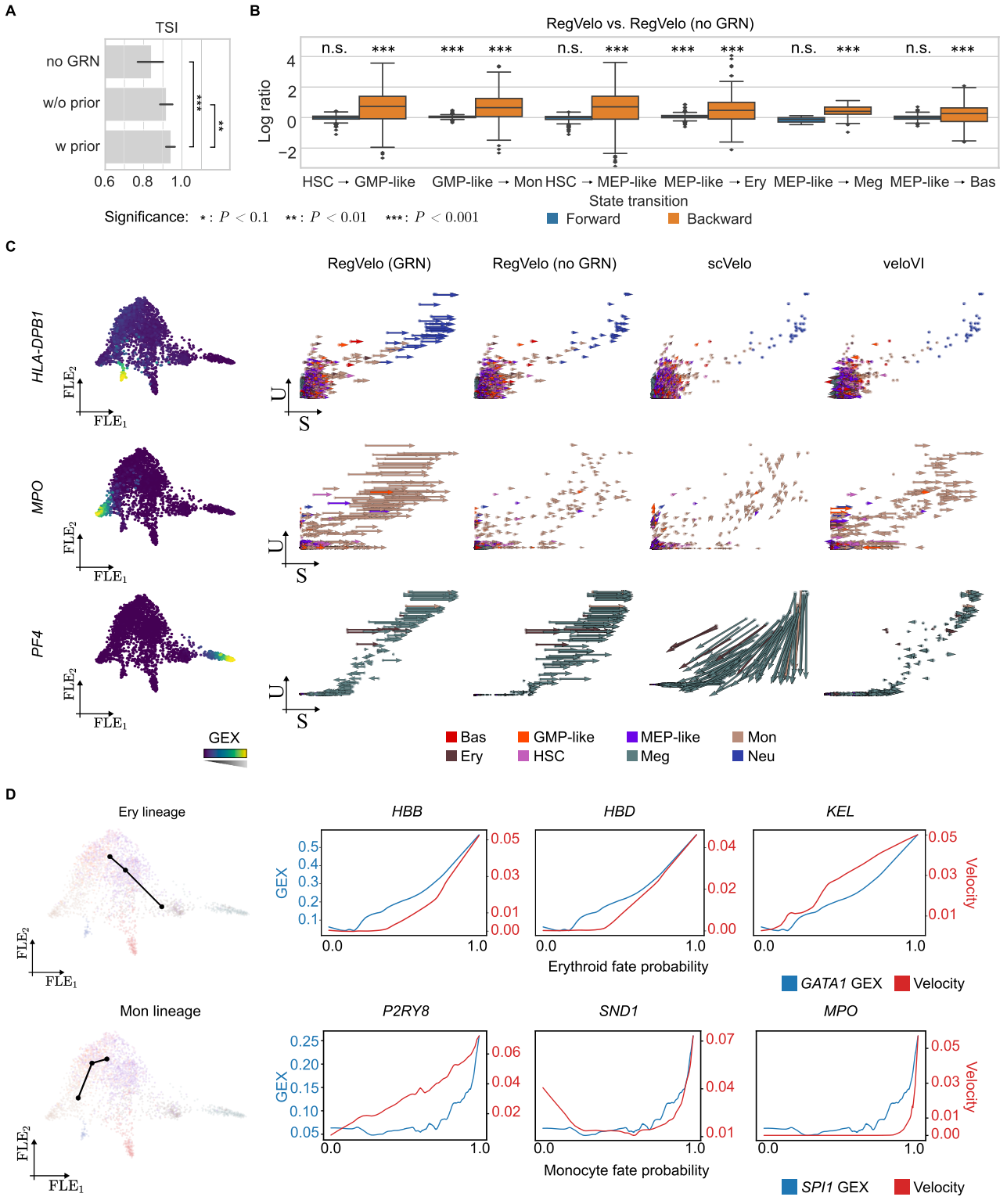


Figure S6. Incorporating gene regulation improves dynamic inference in hematopoiesis, related to Figure 3

(A) TSI scores across three modeling conditions: no GRN, without prior (w/o prior), and with prior (w prior). Incorporating prior knowledge significantly improves TSI. (One-sided paired *t* test, $n = 20$; w prior versus w/o prior: $p = 0.008$; w prior versus no GRN: $p = 0.00029$); Error bars: 95% CI.

(legend continued on next page)

(B) Log-ratio of CBC score of RegVelo, and RegVelo without GRN (no GRN). Box plots indicate the median (center line), interquartile range (hinges), and 1.5× interquartile range (whiskers) (One-sided Welch's *t* tests). HSC → GMP-like (forward: $n = 304$ cells, $p = 0.90$; backward: $n = 161$ cells, $p = 4.74 \times 10^{-10}$); GMP-like → Mon (forward: $n = 152$ cells, $p = 2.78 \times 10^{-7}$; backward: $n = 376$ cells, $p = 2.97 \times 10^{-38}$). HSC → MEP-like (forward: $n = 309$ cells, $p = 0.999$; backward: $n = 446$ cells, $p = 5.13 \times 10^{-27}$). MEP-like → Ery (forward: $n = 312$ cells, $p = 4.87 \times 10^{-13}$; backward: $n = 220$ cells, $p = 7.53 \times 10^{-13}$). MEP-like → Meg (forward: $n = 16$ cells, $p = 0.997$; backward: $n = 66$ cells, $p = 6.25 \times 10^{-13}$). MEP-like → Bas (forward: $n = 161$ cells, $p = 0.64$; backward: $n = 144$ cells, $p = 2.76 \times 10^{-5}$).

(C) FLE embedding of 1947 HSPC cells, colored according to the expression level of marker genes *HLA-DPB1*, *MPO*, and *PF4* (left). Phase portraits of three genes presented by gene-resolved velocities inferred by RegVelo (with and without GRN), scVelo, and veloVI (right). Each cell is colored according to the cell type.

(D) Differentiation trajectories along the erythroid (top) and monocyte (bottom) lineages shown on the FLE embedding (left); gene expression (GEX, blue) and RNA velocity (red) profiles plotted against lineage-specific fate probabilities. Key lineage regulators and targets (*HBB*, *HBD*, *KEL*, and *GATA1* for erythroid; *P2RY8*, *SND1*, *MPO*, and *SPI1* for monocyte) exhibit increasing expression and velocity trends aligned with lineage commitment.

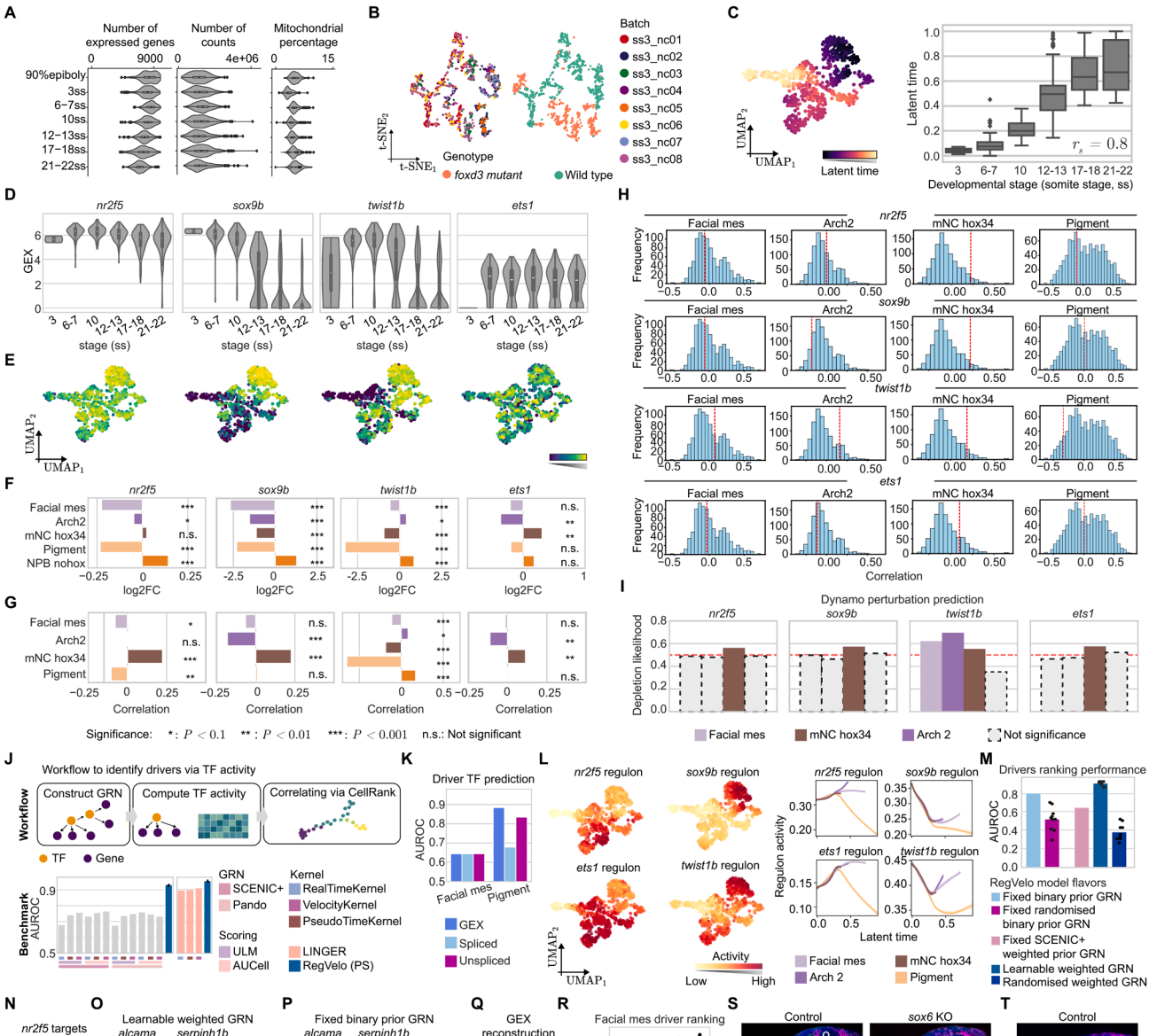


Figure S7. Incorporating gene regulation helps RegVelo to identify early drivers during neural crest development, related to Figure 4

(A) Quality metrics across developmental stages in the Smart-seq3 data.

(B) Eight distinct batches in the Smart-seq3 data overlap in the *t*-SNE embedding, signifying minimal batch effects (left). Distinct cellular patterns are observed between wild-type and *foxd3*-mutant neural crest cells (right).

(C) UMAP colored by the RegVelo-inferred latent time (left); The distribution of the latent time stratified by embryonic developmental stages (right). Box plots indicate the median (center line), interquartile range (hinges), and 1.5× interquartile range (whiskers) (E3: n = 10 cells, E6-E7: n = 105 cells, E10: n = 105 cells, E12-E13: n = 156 cells, E17-E18: n = 146 cells, and E21-E22: n = 183 cells).

(legend continued on next page)

(D) Expression distribution of *nr2f5*, *sox9b*, *twist1b*, and *tfap2a* across embryonic stages. Box plots indicate the median (center line), interquartile range (hinges), and 1.5× interquartile range (whiskers).
(E) UMAP colored by expression levels of *nr2f5*, *sox9b*, *twist1b*, and *ets1*.
(F) Log-transformed fold change of *nr2f5*, *sox9b*, *twist1b*, and *ets1* gene expression across five cell types (terminal or initial states, Wilcoxon rank-sum tests, n.s.: not significant).
(G) CellRank-predicted Pearson correlation between gene expression and cell fate probabilities (Welch's *t* tests). Each bar is colored according to the color code in Figure 4B.
(H) Distribution of the Pearson correlation between gene expression and fate probabilities for each lineage. The red dashed lines indicate the correlation between the corresponding TF (nr2f5, sox9b, twist1b, and tfap2a) expression and cell fate probability (n = 952 genes).
(I) Dynamo-predicted lineage depletion likelihoods of four cranial mesenchymal drivers indicate that *in silico* knockouts of three out of four TFs show no depletion in mesenchymal fates.
(J) Schematic workflow to identify TF drivers via the combination of GRN and trajectory inference methods. First construct the GRN, then compute TF activities, and finally correlate these with fate probabilities inferred by CellRank with Kernels induced by different trajectory inference methods. Benchmark results show the average AUROC for TF driver prediction of both pigment and mesenchymal cell lineages across combinations of GRN methods (SCENIC+, Pando), TF activity scoring methods (ULM, AUCell), and kernel types (RealTimeKernel, VelocityKernel, and PseudotimeKernel). Since LINGER filtered out all facial mesenchymal-related TFs, only pigment rankings are shown.
(K) Driver TF prediction across modalities (GEX, spliced, unspliced) for facial mesenchyme and pigment lineages.
(L) UMAP colored by early drivers' regulon activity inferred by AUCell (left). Regulon activity dynamics over latent time across lineages (right).
(M) Driver TF prediction across five different RegVelo models (n = 10).
(N) Expression dynamics of early drivers and their facial mesenchyme-associated targets. Spliced TF and unspliced target expression are shown; latent time is min–max scaled.
(O and P) ODE-based dynamic simulations under the RegVelo model with (O) a learnable weighted GRN and (P) a fixed binary prior GRN. Red dashed lines show simulated target gene trajectories after TF regulon perturbation, compared with observed (lavender solid line) and fitted unspliced expression (black dashed line). Right: UMAP colored by RegVelo's cell-specific perturbation effect of the facial mesenchymal lineage for each model and TF.
(Q) Gene expression reconstruction performance (R^2) for *nr2f5* and *ets1* target genes under learnable weight versus fixed binary GRN settings.
(R) Top, AUROC of facial mesenchymal (mes) driver ranking increases with broader temporal coverage of training data; we used a fixed prior binary model as baseline (blue bar). Bottom, learned regulatory weights of nr2f5 and ets1 toward facial mesenchyme-associated genes increase with temporal coverage, consistent with their roles in facial mesenchymal fate specification (n = 5 trained models). Each temporal coverage has the same cell number (220 cells).
(S and T) Representative confocal microscopy images of *colec12* (S, facial mesenchymal marker) and *dlx2a* (T; arch 2 marker) expression in control and *sox9b, sox6, zic2b*, and *hoxa2b* crispant zebrafish embryos at the 21-somite stage. e, eye; fm, facial mesenchymal; o, otic.

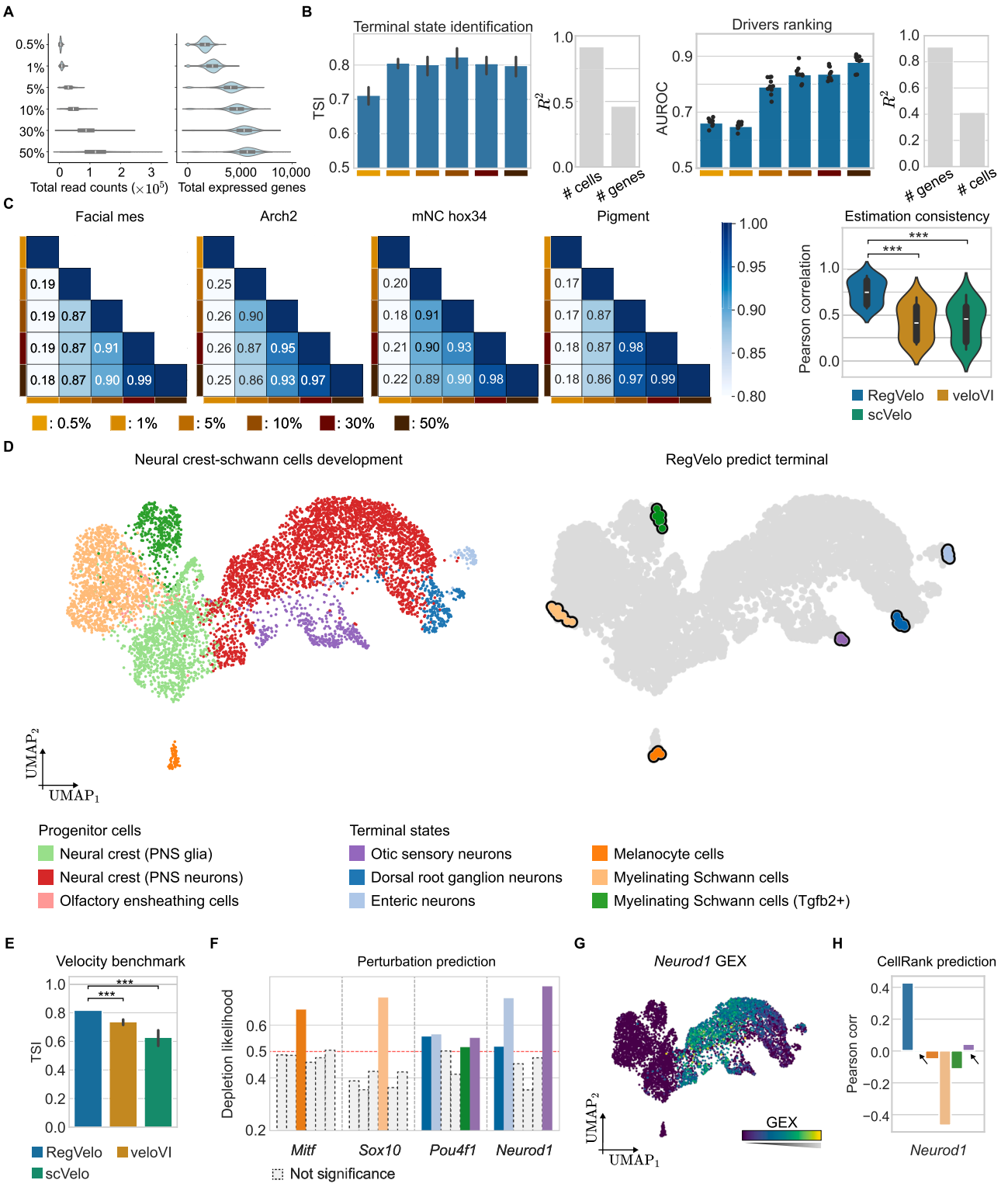


Figure S8. RegVelo gives robust dynamic inference on simulated and real low-coverage neural crest data, related to Figure 4

(A) Violin plot of total read counts (left) and total expressed genes (right) per cell across four different downsampling levels (0.5%, 1%, 5%, 10%, 30% and 50%) (B) Barplot of TSI (left) and driver ranking score (right) for RegVelo across six downsampled zebrafish neural crest datasets (from left to right: 0.5%, 1%, 5%, 10%, 30%, and 50%). And the coefficient of determination (R^2) between number of cells or genes and the metric score, Error bars: 95% CI (n = 10 trained models).

(legend continued on next page)

(C) Pairwise correlation of fate probabilities per lineage when varying downsampling level (left), we did not include downsampling level 0.5% because, at this level, we failed to detect Arch 2 as the terminal states; and comparison of velocity estimation consistency across practical sequence-depth condition (right, 5%–50%), box plots indicate the median (center line), interquartile range (hinges), and 1.5× interquartile range (whiskers). (One-sided Welch's *t* test, $n = 15$, RegVelo versus veloVI: $p = 2.93 \times 10^{-6}$; RegVelo versus scVelo: $p = 6.98 \times 10^{-6}$).

(D) UMAP embeddings of 6788 mouse neural crest-Schwann development cells, colored by cell type annotation of original study (left); CellRank predicted terminal states using RegVelo-inferred velocity (right)

(E) Comparison of the TSI score using RegVelo (blue), veloVI (orange), and scVelo (green) ($n = 20$ runs). (One-sided Welch's *t* test, RegVelo versus veloVI: $p < 1 \times 10^{-10}$; RegVelo versus scVelo: $p < 1 \times 10^{-10}$); Error bars: 95% CI.

(F) Predicted depletion likelihood for the *Mitf, Sox10, Pou4f1*, and *Neurod1* (One-sided Wilcoxon rank sum test).

(G) UMAP colored by *Neurod1* gene expression.

(H) CellRank-predicted correlation of *Neurod1* expression with each lineage.

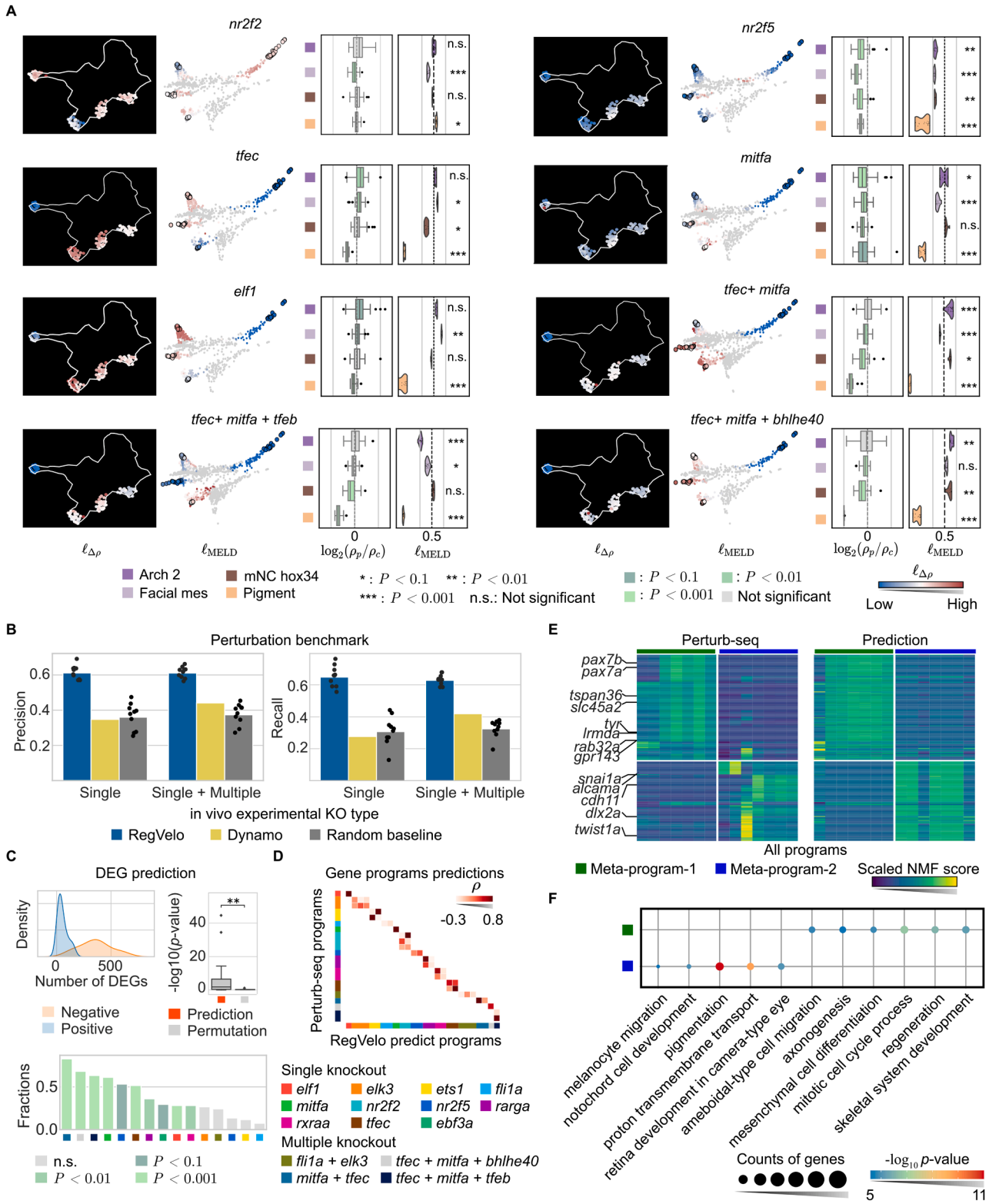


Figure S9. RegVelo predicts perturbation effects on both cell fate composition and gene programs, related to Figure 5

(A) RegVelo simulated density changes and MELD quantified density ratio of four lineages (arch 2, facial mesenchyme, mNC hox34, and pigment). CellRank is applied to both datasets (Smart-seq3 and Perturb-seq) to identify terminal state cells using VelocityKernel and PseudotimeKernel, respectively. Each cell is

(legend continued on next page)

colored by the likelihood value ($\ell_{\Delta\rho}$ and MELD likelihood) (left); and comparison of density changes predicted by RegVelo or quantified by MELD (right). Two-sided Welch's *t* test and permutation test (STAR Methods) are applied to assess statistical significance. We presented only the Perturb-seq results in this study; for the remaining crispants' Perturb-seq results, please refer to Hu et al.[100]

(B) Precision (top) and Recall (bottom) value of correct perturbed lineage prediction using RegVelo and dynamo ($n = 10$ trained RegVelo models).

(C) Comparison of DEG prediction performance. Top left: Density plot of the number of upregulated DEGs and downregulated DEGs in KO. Top right: Boxplot showing log 10-transformed statistical significance (hypergeometric test) of DEG overlap between RegVelo predictions and ground truth compared with permuted control (STAR Methods, $n = 15$, $p = 0.0067$). Bottom: Fraction of correctly predicted DEGs in RegVelo predictions of each KO panel, each bar is colored by their significance level.

(D) Gene program prediction performance. Spearman correlation matrix showing agreement between gene programs inferred by RegVelo and those derived from Perturb-seq. Rows and columns represent matched TF perturbation conditions.

(E) Heatmaps of gene expression programs from Perturb-seq (left) and RegVelo (right). Scaled NMF scores across programs are shown. Annotated genes (left) highlight marker genes specific to pigment cells, pharyngeal arches, and mesenchymal programs. Meta-program-1 and Meta-program-2 are identified via joint clustering of all shared programs.

(F) Top enriched GO biological process pathways for the meta-programs (Fisher's exact test; BH-adjusted *p* value).

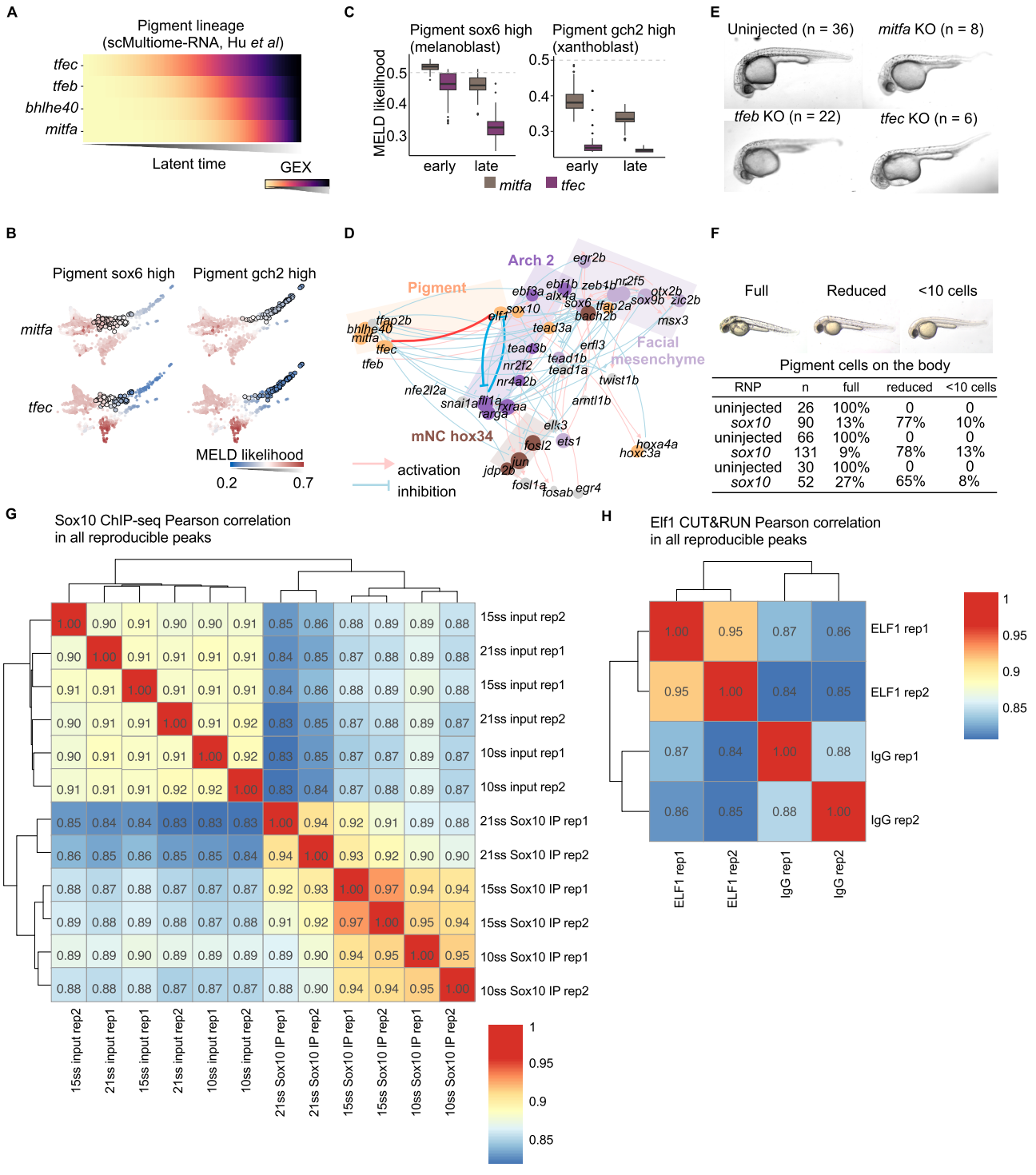


Figure S10. Regulation of pigment drivers during pigmentation, related to Figures 6 and 7

(A) Gene expression patterns from multiome data of representative bHLH factors in the pigment lineage over latent time as computed by the original study.[100]

(B) UMAP embedding of 779 neural crest development cells from the *tfec* knockout and control panel, with sox6-high (melanoblast) and gch2-high (xanthoblast) pigment cells highlighted by circles.

(C) *In vivo* Perturb-seq analysis shows profound depletion effects for *tfec* knockouts compared with *mitfa* in both early and late pigment lineages.

(D) UMAP embedding of the inferred TF-TF network, where the distance represents the Jaccard index between TF pairs. The visualization highlights regulatory interactions between TFs inferred by RegVelo. The dot colors represent the TF-associated lineage as determined by RegVelo perturbation simulations, while the dot size indicates the predicted effect size of perturbations on the corresponding lineage; groups of closely related TFs were manually highlighted.

(legend continued on next page)

(E) Single-gene knockouts of *mitfa, tfeb*, or *tfec* show no delay in pigmentation formation at 36 hpf. The embryo numbers in each condition are labeled above the sub-panels.
(F) Phenotypic quantification of single-gene knockouts of *sox10* and uninjected controls show that most *sox10* crispants have a weak phenotype (reduced pigment) and a strong phenotype (<10 pigment cells) at approximately 48 hpf ($p = 1 \times 10^{-8}$ for all three experiments).
(G) Hierarchical clustering of Sox10 biotin ChIP-seq peak sets based on Pearson correlation scores from duplicate experiments.
(H) Hierarchical clustering of Elf1 CUT&RUN peak sets based on Pearson correlation scores from duplicate experiments.