Dynamical information enables inference of gene regulation at single-cell scale

Stephen Y Zhang *[1] and Michael P H Stumpf[1]

[1]Mathematics and Statistics, University of Melbourne, Australia

Abstract

Cellular dynamics and emerging biological function are governed by patterns of gene expression arising from networks of interacting genes. Inferring these interactions from data is a notoriously difficult inverse problem that is central to systems biology. The majority of existing network inference methods work at the population level and construct a static representations of gene regulatory networks; they do not naturally allow for inference of differential regulation across a heterogeneous cell population. Building upon recent dynamical inference methods that model single cell dynamics using Markov processes, we propose locaTE, an information-theoretic approach which employs a localised transfer entropy to infer cell-specific, causal gene regulatory networks. LocaTE uses high-resolution estimates of dynamics and geometry of the cellular gene expression manifold to inform inference of regulatory interactions. We find that this approach is generally superior to using static inference methods, often by a significant margin. We demonstrate that factor analysis can give detailed insights into the inferred cell-specific GRNs. In application to two experimental datasets, we recover key transcription factors and regulatory interactions that drive mouse primitive endoderm formation and pancreatic development. For both simulated and experimental data, locaTE provides a powerful, efficient and scalable network inference method that allows us to distil cell-specific networks from single cell data.

1 Introduction

Cell identity is controlled by dynamic gene regulation networks. Identifying and characterising gene regulatory interactions is therefore a central aim of transcriptomic studies. A frequently cited advantage of single cell assays is that population heterogeneity is captured, allowing for the detection and study of rare cellular phenotypes that would otherwise be unobservable from bulk studies [1], as well as variations within groups of cells which may be conventionally regarded as being of the same "type". The rising popularity of single cell data has spurred rapid advances in data analysis and modelling methods such as trajectory inference [2], dimensionality reduction [3], and network inference [4, 5].

In contrast to bulk assays where only population-averaged expression can be measured, single cell data generates vastly more observations. These datasets can thus be harnessed to tackle the "large-p small-N problem" [6] in which the number of predictions outnumber the number of samples [7]. This problem has been plaguing gene regulatory network inference because the number of predictions grows quadratically with the number of genes considered. In this vein, a collection of methods have been developed for network inference from single cell data, some of which build on methods originally intended for bulk expression data, and others which are specifically tailored to single cell data. The recent surveys in [8, 9, 5] provide an overview of the current methodology. The vast majority of network inference methods aim to reconstruct a single *static* network [5, 8] that describes the set of possible interactions occurring within an observed population. Given the biological importance of cellular heterogeneity, a natural expectation is that variations in transcriptional state may correspond to variations in (cell state dependent) regulatory interactions which cannot be represented as static networks [7], and indeed has been observed empirically in previous studies [10, 11].

*stephenz@student.unimelb.edu.au

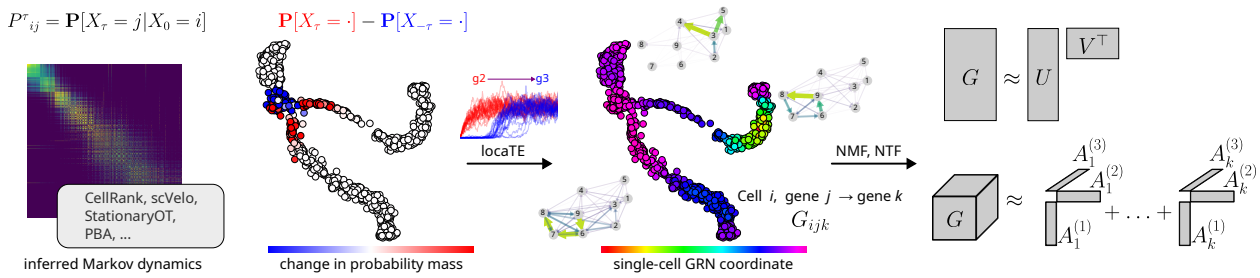


Figure 1: **Cell-specific network inference from estimated dynamics.** LocaTE takes as input a transition matrix P that encodes inferred cellular dynamics as a Markov chain on the cell state manifold. By considering the coupling $(X_\tau, X_{-\tau})$, locaTE produces an estimate of transfer entropy for each cell i and each pair of genes (j, k).

There is clear need for methods that can learn cell-specific networks [7, 12], recently leading to new methods [13, 14]. These methods seek to infer multiple networks (e.g. one per cell) from single-cell resolved data, thus allowing us to learn about variation in regulatory relationships over time or between conditions. Such methods rely on neighbourhood or cluster information to construct networks that are cell, time, or context-specific. Existing methods for cell-specific network inference have focused on inferring undirected networks. If we hope to infer *causal* or *directed* interactions, then information about underlying dynamics in a cell population is essential. To this end, many methods for static network inference take advantage of pseudotemporal orderings to inform network inference [15, 16, 17, 18, 19]. The one-dimensional nature of pseudotime necessarily imposes a total ordering over cells, resulting in a loss of finer structure of the cell state manifold. Recent work on trajectory inference from a Markov process viewpoint [20, 21, 22, 23] depart from this framework and are able to model more complex dynamics, free from assumptions on the trajectory topology and without coarse graining.

Here we propose to adopt the transfer entropy (TE) [24] as an information-theoretic measure of causality. Transfer entropy, like other information-theoretical measures, is a model-free framework which has been used widely in contexts such as neuroscience where abundant time-series data are available. In a Gaussian setting it reduces to the linear Granger causality [25]. Transfer entropy has been used to predict static networks using a pseudotemporal ordering [15, 16]. We demonstrate that the transfer entropy can be adapted to infer cell-specific networks without imposing a prescribed ordering of cells.

Inference of cell-specific networks helps us to address problems in developmental biology, where it is often the transient dynamics that are the most interesting since these are likely to correspond to major epigenetic changes [7]. Our approach is tailored to such problems since we make no restricting simplifying assumptions: the cell state space is treated as a continuum, and a priori no partitioning or clustering is required to group or classify cells.

2 Method

Local transfer entropy of expression dynamics We represent the transcriptional state of a cell as a vector of expression levels $X = (X^{(1)}, \ldots, X^{(N)})$. Although transfer entropy does not make assumptions on the specific form of a model, for the sake of illustration let us consider a model of cellular dynamics where the expression level of each transcript $X^{(i)}$ is governed by a stochastic differential equation (SDE) of the form

$$\mathrm{d}X_t^{(i)} = f^{(i)}(X_t^{(1)}, \ldots, X_t^{(N)})\mathrm{d}t + \sigma^{(i)}(X_t^{(1)}, \ldots, X_t^{(N)})\mathrm{d}B_t. \tag{1}$$

Note that this framework includes the chemical Langevin equation (CLE) [26], a biophysically derived model of single cell expression dynamics. Under the Itô interpretation [27, 28], for a small time interval τ we may write approximately,

$$X_{t+\tau}^{(i)} = X_t^{(i)} + f^{(i)}(X_t^{(1)}, \ldots, X_t^{(N)})\tau + \sigma^{(i)}(X_t^{(1)}, \ldots, X_t^{(N)})\sqrt{\tau}Z^{(i)}, \tag{2}$$

where the $Z^{(k)}$ are independent and identically distributed Gaussian random variables. It is straightforward to condition on $\{X_t^{(i)} = x\}$:

$$X_{t+\tau}^{(i)} | \{X_t^{(i)} = x\} = x + f^{(i)}(\{X_t^{(k)}\}_{k \neq i}, x)\tau + \sigma^{(i)}(\{X_t^{(k)}\}_{k \neq i}, x)\sqrt{\tau} Z^{(i)}, \tag{3}$$

For any other gene $X^{(j)}, j \neq i$, following [29, Section 4.2], the *transfer entropy* with time lag τ is defined to be

$$\mathrm{TE}_\tau(X^{(j)} \to X^{(i)}) := \mathbf{E}_{X_t^{(i)}} \left[\mathrm{I}(X_{t+\tau}^{(i)}, X_t^{(j)} | X_t^{(i)}) \right], \tag{4}$$

where

$$\mathrm{I}(X, Y | Z) := \mathbf{E}_{X,Y} \left[\log \rho_{XY|Z} - \log(\rho_{X|Z} \otimes \rho_{Y|Z}) \right] \tag{5}$$

is the mutual information of two random variables X, Y conditional on a third variable Z. Under the discrete approximation, the transfer entropy will be positive whenever $f^{(i)}$ or $\sigma^{(i)}$ depend on $X^{(j)}$, since then one has $\mathrm{I}(X_{t+\tau}^{(i)}, X_t^{(j)} | X_t^{(i)}) > 0$. In the biophysically relevant setting of the CLE, the functions f and σ share the same functional dependence on $\{X^{(k)}\}_k$ when $X^{(i)} = x$, and so for infinitesimal time-scales τ a non-zero transfer entropy implies regulation, i.e. $\partial_j f^{(i)} > 0, \partial_j \sigma^{(i)} > 0$. In the practically relevant setting where the time-scale τ is short, the data processing inequality [30] guarantees that direct interactions will dominate indirect interactions, i.e. for a feed-forward loop $X^{(i)} \to X^{(j)} \to X^{(k)}$, one has that $\mathrm{I}_{ik} < \min\{\mathrm{I}_{ij}, \mathrm{I}_{jk}\}$ [31].

The concept of transfer entropy has previously been used for inference of static networks where pseudotemporally ordered expression data are available [15, 16]. However, in these methods the expectations in (5), (4) are taken with respect to the set of lagged tuples of cells along a total ordering. This introduces a strong dependence on the input pseudotemporal ordering, and also results in loss of resolution: any variation in regulatory activity over the expression space is integrated out to yield a static network. This can be problematic since biological trajectories may exhibit bifurcating, cyclic or converging trajectories, potentially requiring multiple runs of GRN inference on different subsets of a dataset.

We propose to take for each observed cell $\hat{X}$ the expectation (5) with respect to the evolution described by the transition kernel P^τ of (1) starting from some neighbourhood $\mathcal{N}(\hat{X})$ of the cell. For short time scales τ and an appropriately sized neighbourhood $\mathcal{N}(\hat{X})$ (i.e. large enough to take expectations, but not so large as to lose local detail), the resulting dynamics are therefore local. This is best understood as a conditional transfer entropy,

$$\mathrm{TE}_\tau \left[X^{(j)} \to X^{(i)} \mid X_0 \in \mathcal{N}(\hat{X}) \right] := \mathbf{E}_{X_0 \in \mathcal{N}(\hat{X})} \mathrm{I}(X_\tau^{(i)}, X_0^{(j)} | X_0^{(i)}).$$

This enables us to infer regulatory interactions that are specific to the local cell context. We remark that some information-theoretic static inference methods [4, 15] condition on a third gene in order to remove indirect interactions. Instead of conditioning separately on each potential indirect interaction, our framework conditions on a local region of the expression space. This can be roughly understood as conditioning on some (potentially non-linear) function of the overall gene expression profile that is determined by the construction of the cellular manifold (e.g. in the case of PCA, this would be an affine combination).

Finally, since we model dynamics as a general Markov process, no assumptions on the ordering of cells or topology of trajectories is made or necessary.

Inference in practice In practice, information about the drift and diffusion terms [27] in (1) are unavailable; dynamical information must instead be *inferred* from observed expression data. This *trajectory inference* problem has been widely studied [21, 32, 2]. Since we are interested in cell-specific networks, we restrict our focus to a class of methods that model cellular dynamics as a Markov process involving drift and diffusion over a manifold of cell states [20, 22, 33, 23]. Let $\mathcal{X} = \{x_i\}_{i=1}^N$ be a sample from a population of cells, where $x_i \in \mathbb{R}^M$ is the vector of mRNA expression values for cell i. Let $\mathcal{M}$ be a graph constructed from $\mathcal{X}$ that can be thought of as approximating some underlying manifold of cell states [34]. We assume we have access to a transition matrix $P \in \mathbb{R}^{N \times N}$ that is supported on $\mathcal{M}$ and that adequately describes the ("true") dynamics of (1). In practice, as we do in Section 3, we could use a range of recently reported methods to estimate P. Then $\mathcal{M}$ equipped with P encodes a discrete Markov chain that approximates the true cellular dynamics. In what follows, we will write $[\![n]\!] = \{1, \ldots, n\}$ for $n \in \mathbb{N}$.

For any cell $x \in \mathcal{M}$, we can construct (using e.g. a uniform distribution, truncated Gaussian, etc.) a probability distribution supported on its neighbourhood, π_0^x. We can then consider a Markov process $X_t, t \geqslant 0$ started from $X_0 \sim \pi_0^x$ and evolving under P. For some fixed τ the resulting *coupling* is

$$(X_0, X_\tau) \sim \mathrm{diag}(\pi_0^x) P^\tau =: \gamma_\tau^x. \tag{6}$$

For a pair of genes, (j, k), we quantify the local transfer entropy for the causal relationship $j \to k$ from knowledge of the coupling γ_t^x and gene expression states,

$$\mathrm{TE}\left[X_0^{(j)} \to X_\tau^{(k)}\right] = \mathbf{E}_{(X_\tau, X_0) \sim \gamma_\tau^x}\left[\mathrm{I}(X_0^{(j)}, X_\tau^{(k)} | X_0^{(k)})\right], \tag{7}$$

Repeating this computation for all cells, $x_i \in \mathcal{M}$, and gene pairs, $j, k \in [\![M]\!]$, we arrive at a tensor, $\hat{G} \in \mathbb{R}^{N \times M \times M}$, such that, $\hat{G}_{ijk}$, is the transition entropy score of $j \to k$ in the neighbourhood of cell, x_i. Since the TE score matrix, $\hat{G}_i$, for each cell is learned from a local neighbourhood, it is potentially noisy. Furthermore, it is impractical (due to limited data and computational resources) to condition on potential indirectly interacting genes [15]. Thus, TE scores may contain noise and spurious signals resulting from indirect interactions. To deal with this, we first filter interaction scores using context likelihood of relatedness (CLR) [35] and then solve a manifold-regularised optimisation problem to both de-noise the interaction signals and attenuate noise and indirect interactions [15, 13].

Context likelihood of relatedness. Given raw TE scores $\hat{G}$, we use the context likelihood of relatedness algorithm [36] to filter scored interactions. In the original paper, the CLR algorithm was shown to be effective in filtering out indirect interactions in the context of mutual information scores. For a pair of genes (i, j) and given a interaction score matrix A, compute z_i (resp. z_j) to be the z-score of A_{ij} with respect to $A_{i\cdot}$ (resp. $A_{\cdot j}$). Then we define the *weighted* CLR score for (i, j) to be

$$\widetilde{A}_{ij} = \frac{A_{ij}}{2}\sqrt{\max(0, z_i)^2 + \max(0, z_j)^2}. \tag{8}$$

Applying CLR filtering along the first axis of $\hat{G}$, we obtain the filtered tensor, $\widetilde{G}$. We modify the original approach of [36] and weigh the CLR score by the initial MI value $\hat{G}_{ij}$. This is important since CLR was originally designed to filter interactions in static networks. In cell-specific networks, very few edges may be "active" in any given cell's context, and so entire rows or columns of $\hat{G}$ may consist only of noise. Computing z-scores along those rows or columns would otherwise put both noise and signal on the same scale.

Smoothing and denoising The tensor $\widetilde{G}$ contains a noisy matrix of filtered interaction scores, one for each measured cell. For notational convenience we write $\widetilde{G}$ as $\widetilde{G} \in \mathbb{R}^{N \times M^2}$, i.e. each row is a length-M^2 unfolded score matrix. We propose to solve the optimisation problem $G = \arg\min_X \mathcal{L}(X; L, \widetilde{G})$ where

$$\mathcal{L}(X; L, \widetilde{G}) = \frac{1}{2}\sum_{i=1}^{N} w_i \|X_{i\cdot} - \hat{G}_{i\cdot}\|_2^2 + \frac{\lambda_1}{2}\,\mathrm{tr}(X^\top L X) + \lambda_2 \sum_{i=1}^{N} w_i \|X_{i\cdot}\|_1, \tag{9}$$

λ_1, λ_2 are hyper-parameters, w is a vector of cell weights, and L is the symmetric graph Laplacian on $\mathcal{M}$. The term associated with λ_1 is a manifold regularisation term corresponding to an expectation that regulatory relationships should vary smoothly with changes in cell state, i.e. $\mathrm{tr}(X^\top L X)$ is large for rapidly fluctuating X. The term associated to λ_2 is a Lasso [6] term that encourages X to be sparse, reflecting our knowledge of biological networks [37, 13]. Together, the objective (9) encourages both parsimony and sharing of information along the cell state manifold. The problem is a case of L1-L2-regularised least squares, and can be solved efficiently using an alternating scheme which we describe in the appendix.

Using a backward transition matrix Since above we consider couplings of the process at times $(0, \tau)$, TE scores may reflect the underlying dynamics up to a shift in time. To remedy this, we instead consider couplings symmetric in time, i.e. between $(-\tau, \tau)$. While time-reversal of a diffusion-drift process away from equilibrium is not well-posed, in practice, suitable backward operators have been constructed by transposing the transition matrix [22, 38] (see Appendix A.2 for further discussion of this). Given such an approximate backward operator, Q, a time-symmetric coupling, $\gamma_\tau^x = (Q^\top)^\tau \,\mathrm{diag}(\pi_0^x) P^\tau$, can be constructed.

Extracting a static network Cell-specific networks can be aggregated across a cellular population to produce a static network. Mathematically, averaging over individual cell-specific networks can be understood as applying the so-called "tower" property of conditional expectation. In practice, we observe that inferred cell-specific networks become less informative near the endpoints of trajectories, since temporal information is lost there. Similarly, averaging may result in signals from small subpopulations of cells to be diluted. Thus, some reweighting strategies will be useful to infer appropriate static networks.

Computational considerations The problem of cell-specific network inference is at least $\mathcal{O}(NM^2)$ for N cells and M genes, since each cell-gene-gene triplet must be considered. For each triplet (i, j, k) locaTE calculates the joint distributions of $(X_\tau^{(k)}, X_0^{(j)}, X_0^{(k)})$ under the coupling $\gamma^{(i)}$. While in the worst-case (if γ were dense) this would have complexity $\mathcal{O}(N^2)$, in all our applications we consider sparse γ where transitions only occur between nearby cells, in which the complexity is $\mathcal{O}(|\gamma^{(i)}|) \ll N^2$. The CLR filtering step is also $\mathcal{O}(NM^2)$, and the regression step primarily involves iterative matrix products of sparse $N \times N$ against $N \times M^2$. We implemented locaTE using the Julia programming language [39] to take advantage of its speed for numerical computation.

We note also that with exception of the regression step which is global, all computational steps are trivially parallelised across cells. Furthermore, the calculation of TE scores can alternatively be parallelised across gene pairs. This allows locaTE to take advantage of large scale parallelisation when available, potentially drastically reducing the compute time required. Finally, we found in practice that a large portion of computational burden arose from the construction of the joint distribution $(X_\tau^{(k)}, X_0^{(j)}, X_0^{(k)})$. Since this is effectively an accumulation operation, this can be significantly accelerated using GPU computation. We observe speedups of $100 - 200\times$ using a NVIDIA V100 card[1]. Other steps in the locaTE pipeline (regression, factor analysis) amount to linear algebra operations and are also amenable to GPU acceleration.

3 Results

We first discuss two examples in simulation where the true regulatory interactions are known. To illustrate the utility of inferring cell-specific networks, we consider examples where the network is rewired across the course of a trajectory. In Appendix B.2 we summarise results for a wider range of "simple" trajectories, where a static network is sufficient to represent the underlying system.

3.1 Simulated data: branch-dependent logic

Overview We first discuss an example in which we seek to infer cell-specific networks from simulated single-cell data. To illustrate the utility of cell-specific inference, we consider a scenario where a static network cannot capture the dynamics. In the network in Figure 2(a, b) a bifurcation driven by a toggle-switch feeds into one of two "gene expression modules". Modules A and B involve the same genes $\{g6, g7, g8, g9\}$, but with completely different interactions among the genes: the flow of information in each module is the mirror image of the other.

A static network (in the graphical sense) is therefore insufficient to describe this system, since the directionality of some interactions depends on the cellular context. Alternatively, context dependence can be understood as arising from the presence of higher-order interactions. Consider gene 7, which may either be activated by gene 6 (in module A) or by gene 8 (in module B). This could be understood conceptually as a *hyperedge*, since the corresponding Boolean rule would be

$$g7 \leftarrow (g6 \wedge (g4 \wedge \neg g5)) \vee (g8 \wedge (\neg g4 \wedge g5)).$$

Inference of higher-order interactions is known to be an even more challenging statistical problem than inference for pairwise interactions [37, 4]. Without trying to address this problem in generality, cell-specific networks allow us to disentangle interactions for a certain subclass of higher-order interactions: those which can be understood to be locally first-order, conditional on some latent variable that is encoded in the cellular

[1] TE calculation for each cell in the Hayashi et al. mESC-Endoderm example [40] took 5ms (GPU, NVIDIA V100) vs. 847ms (CPU single core, Intel Xeon Gold 6242 @ 2.8 GHz)

context. Finally, since existing cell-specific network inference methods [13, 14] infer undirected edges only, such methods are unable to distinguish the direction of information flow in this example and thus would predict identical networks for either branch.

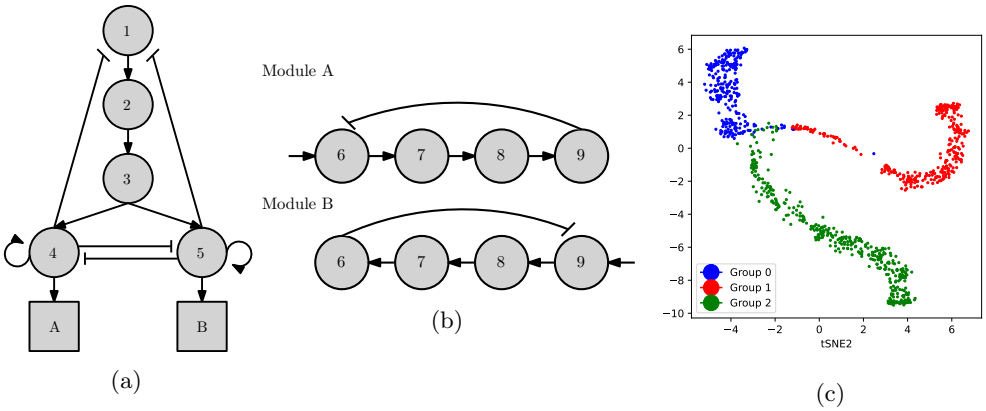


Figure 2: **Example bifurcating network.** (a) 5-gene bifurcating subnetwork feeding into two modules A, B involving the same 4 genes $g6, g7, g8, g9$. (b) 4-gene modules A, B each involving different regulatory interactions involving the same species. (c) tSNE embedding of simulated cell profiles, coloured by inferred diffusion pseudotime cluster.

Simulation We simulate the network in Figure 2 using the BoolODE software package [5]. This uses a chemical Langevin equation scheme for simulating biologically plausible expression dynamics: $\mathrm{d}X_t = f(X_t)\mathrm{d}t + g(X_t)\mathrm{d}B_t$ where f, g have specific forms dictated by the reaction network. From 1,000 independent realisations of the system, we produced a set of 1,000 sampled cells by sampling a cell state from each trajectory at a time chosen uniformly at random. As a measure of ground truth interactions for each cell x_i, we computed the corresponding Jacobians $J_{ijk} = \partial_j f_k(x_i)$. In Appendix B.1 we list the Boolean rules used to implement this network.

Inferring dynamics As input our method requires a cell-state transition matrix P. We construct a "ground truth" matrix P from the velocity of sampled cells in the expression space ("RNA velocity") as follows. For each observed cell, x_i, we calculate the corresponding velocity vector, $v_i = f(x_i)$. From this, we can construct transition probabilities to a neighbourhood of x_i (in gene expression space),

$$P_{ij} = \frac{\exp\left(\frac{\langle x_j - x_i, v_i\rangle}{\sigma}\right)}{\sum_{k\in\mathcal{N}[i]}\exp\left(\frac{\langle x_k - x_i, v_i\rangle}{\sigma}\right)}\mathbf{1}\{j \in \mathcal{N}[i]\} \tag{10}$$

where $\mathbf{1}$ denotes a 0-1 indicator. We set $\mathcal{N}$ to be the k-NN neighbourhood with $k = 25$, and chose the bandwidth σ following the median rule of [22]. We also considered other approaches for constructing P: as in [38], the inner product can be replaced with cosine similarity or Pearson correlation between $x_j - x_i$ and v_i in order to estimate transition probabilities depending only on the orientation of v_i.

In addition to velocity-based methods, we consider a transition matrix based on diffusion pseudotime [41] in which corresponds to a random walk biased in the direction of increasing pseudotime. We also consider the methods StationaryOT [33] and PBA [20], which were shown to have good performance for inferring stochastic dynamics without depending on pseudotime or velocity estimates.

Inferring causal interactions We apply the method described in Section 2 with a backward matrix obtained via the transpose (see Section A.2), for a range of parameter values $(\tau, \lambda_1, \lambda_2)$. To construct the initial neighbourhood distribution π_0^x we use quadratically regularised optimal transport [42], which produces a local neighbourhood density similar in effect to a truncated Gaussian. In Figure 3(a) we show the

individual cell-specific networks obtained for $\tau = 3, \lambda_1 = 25, \lambda_2 = 10^{-3}$ represented as vectors against the pseudotemporal ordering, and in Figure 3(b) we show the averaged networks over the clusters delineated in Figure 2(c). We observe overall that the the de-noised result G resembles the ground truth. Furthermore, the averaged networks for module A and module B are directed and reflect the mirrored connectivities described in Figure 2(b). Averaging networks is convenient for summarising the inference output for cell-specific networks, but obscures the cell-specific nature of the inference result. In Figure 3(c) we show the interaction scores for the two edges $g6 \to g7, g7 \to g6$ which are specific to modules A and B respectively. We observe that the flow of information between genes 6 and 7 is reversed on opposite branches, demonstrating that locaTE is able to resolve directed edges in a cell-specific manner.

Comparison to other methods To date, literature for the problem of inferring cell-specific GRNs is limited. CeSpGRN [13] is perhaps the method most similar to locaTE in terms of the problem it seeks to address, namely inferring a $N \times G \times G$ array of cell-specific gene-gene interaction scores. It is limited to undirected interactions due to use of the (Gaussian) Graphical Lasso. For a larger set of comparisons we also consider methods for static network inference. We consider three information-theoretic methods: TENET [16] and Scribe [15], both of which infer static directed networks from pseudotime-ordered single-cell data, as well as PIDC [4] which infers an undirected network using partial information decomposition. We also considered SCODE [17] and (for the experimental datasets) GRISLI [19] which are ODE-based models for directed inference.

Cell-specific edge detection For each triple (i, j, k) with $i \in [\![N]\!]$ and $j, k \in [\![M]\!]$, we treat the problem of detecting an edge $j \to k$ in cell x_i as a binary classification problem with some threshold q. To construct the set of true positives, we consider the matrix ΠJ, where Π is a neighbourhood transition matrix such that $\Pi_{i\cdot} = \pi_0^{x_i}$. The motivation for considering this instead of simply using the raw Jacobians J is that cell-specific interactions are necessarily inferred by leveraging neighbourhood information. Strict cell-wise comparison to a ground truth would be overly stringent and sensitive to small perturbations in the expression space. By computing ΠJ, the ground truth signal is smoothed over the same neighbourhood as used for inference, allowing for more robust assessment of classifiers. We classify all potential edges for which $(\Pi J)_{ijk} > 0.5$ to be "true" edges.

In Figure 3(d) we show the Precision-Recall (PR) curves calculated for locaTE, as well as raw transfer entropy (TE), transfer entropy followed by CLR filtering (TE+CLR), and the method CeSpGRN [13] which infers cell-specific undirected networks. We find that the CLR filtering step improves upon raw TE scores in terms of prediction performance, and that locaTE improves substantially upon both TE and TE+CLR. This demonstrates that both the CLR filtering and regularised regression step are essential parts of the inference approach: this can be understood as a combined effect of (1) removal of indirect interactions using the CLR algorithm, and (2) sharing of information along the cellular manifold and attenuation of noise using the smooth and sparse regression. We also compute a ground-truth static network by aggregating the cell-specific ground truth networks and calculate a static network AUPRC, corresponding to the best-case performance for a static method, i.e. if all cells were estimated to have the same static network.

To investigate the performance of locaTE further we use the BoolODE package to generate 10 datasets using different kinetic parameters for each simulation [5, Methods]. Thus, variations across datasets result both from the random sampling of cells, as well as differences in the simulated biophysics. For each dataset, we infer cell-specific networks using locaTE using several methods for trajectory inference. We consider three approaches using velocity information: dot-product, cosine similarity, and Pearson correlation. We also consider StationaryOT and PBA, as well as a transition kernel based on diffusion pseudotime [41]. In Figure 3(e) we summarise performance in terms of AUPRC ratio, that is the AUPRC for inferred cell-specific networks relative to that of a random predictor.

These results show that the trajectory reconstructions used as input to locaTE can affect the inference result: the dot-product velocity kernel outperforms the cosine and correlation velocity kernels. This is not unexpected, since the cosine and correlation kernels depend only on the orientation of the velocity vector and not on its magnitude – thus losing information. We find that StationaryOT performs similarly to cosine and correlation kernels, while PBA and the pseudotime kernel perform worst. However, we find that all locaTE-based methods outperform the undirected method CeSpGRN, indicating that the availability of even potentially poor dynamical information can lead to improved inference of directed interactions. Finally,

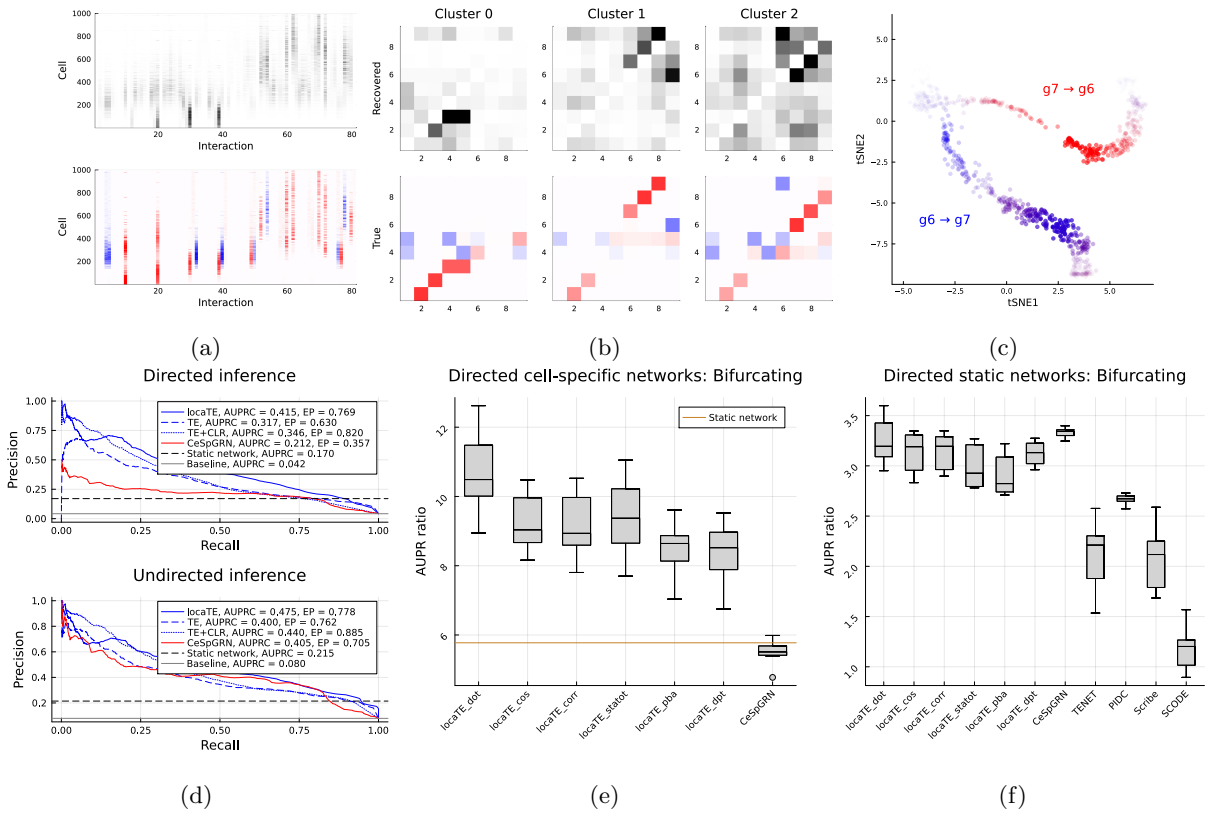


Figure 3: **Results for simulated bifurcating network.** (a) LocaTE output as a $N \times M^2$ matrix in order of increasing pseudotime (top), shown alongside the ground truth Jacobians (bottom). (b) Interactions inferred by locaTE averaged by cluster (top) shown against ground truth interactions averaged by cluster. (c) Interaction scores for $g7 \to g6$ and $g6 \to g7$ shown for each cell on the tSNE embedding. (d) Precision-Recall (PR) curves for cell-specific edge detection from $\hat{G}$ (raw TE values), $\tilde{G}$ (TE + CLR values), or G (locaTE). Area under PR curve (AUPRC) and early precision at 10% recall (EP) is shown. (e, f) AUPRC ratio for locaTE and other methods across 10 sampled simulation datasets, in the cell-specific (e) and the static (f) setting. In the cell-specific setting, we show as a horizontal line the AUPRC ratio achieved by perfect knowledge of the static network.

we find that all methods perform at least as well as the static network baseline, illustrating the utility of cell-specific network inference methods in general.

Static networks as aggregates As most methods in the literature deal with the static inference problem, we investigate the performance of locaTE by averaging local networks across cells to form a static network (see Figure S11 for representative examples). We show AUPRC ratios in Figure 3(f) for networks inferred by locaTE, as well as other selected static inference methods. In this setting we find that locaTE performs similarly well across trajectory inference methods, and that CeSpGRN performs marginally better. This can be partially understood, since the aggregated networks for modules A and B together in Figure 2 would be indistinguishable from the corresponding undirected network (as we find in the next example, this is not always the case). Competing static methods all perform worse, with PIDC performing best, and SCODE performing worst. This is broadly in keeping with the conclusions of [5].

Downstream factor analysis Instead of using the de-noising regression described previously, we can also apply techniques from factor analysis to the tensor of TE values. The insight here is that methods such as non-negative matrix factorisation (NMF) and non-negative tensor factorisation (NTF) seek a simplified, low-rank description of the data. In the context of cell-specific GRNs, we are faced with an $N \times G \times G$ array.

By applying NMF (resp. NTF) to the flattened $N \times G^2$ matrix (resp. $N \times G \times G$ tensor), we effectively seek groups or *modules* of interactions which are active together in some region of the cell state space, where the regulatory state of a given cell is represented as a linear combination of some inferred archetypal regulation patterns.

We derive a numerical scheme for finding matrix and tensor low-rank representations of the TE score matrix that incorporates the same smoothing and sparsity priors as (9). This can be used as a drop-in replacement for the (full-rank) problem (9). We describe these in detail in Appendix A.4 (NMF) and Appendix A.5 (NTF). In Figure 4, we illustrate the utility of NMF and NTF for finding simplified representations of scGRNs from locaTE. First applying locaTE-NMF with 8 factors, we find that the recovered coefficients are localised to windows of pseudotime. Importantly, pseudotime information is only used for visualisation here, and these results suggest locaTE-NMF is able to extract meaningful factors from geometry alone. Examining the corresponding atoms, we find that they also correspond to distinct groups regulatory interactions that are differentially active across the population. For instance, atom 1 corresponds to the "trunk" of the bifurcation ($g2 \to g3$ and $g3 \to \{g4, g5\}$), and atoms 4-8 correspond different sections along the branches.

Tensor factorisations differ from matrix factorisations in that effectively the learned atoms have an additional constraint of being separable. Because of this added structure, tensor decompositions are often found to have better properties (identifiablity of factors, robustness to noise [43]) and thus are favoured where possible. We apply locaTE-NTF to find 16 rank-one factors of the scGRN tensor, and we applied k-means to cluster the rank-one atoms into 8 clusters based on their activities. We found that the resulting clusters largely resemble the ones found by NMF. Unlike NMF however, the clusters can be decomposed into smaller rank-one components, and are arguably less noisy than the atoms found by NMF owing to the additional constraint. In Figure 4(c) we show how clusters 1 and 3 from the NTF representation split into rank-one components, demonstrating how the rank constraint leads to finer grained representations of regulatory states.

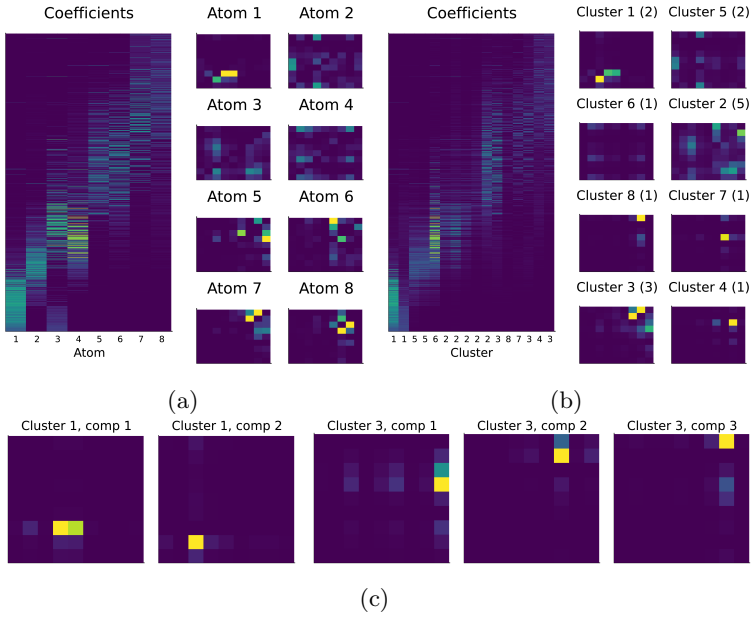


Figure 4: **Factor analysis extracts groups of differentially regulated genes.** Factor analysis results for scGRNs using (a) locaTE-NMF with $k = 8$ factors and (b) locaTE-NTF with $k = 16$ factors; the number of rank-one factors contributing to each cluster for NTF are shown in parentheses. (c) Rank-one atoms found by locaTE-NTF corresponding to clusters 1 and 3.

3.2 Simulated data: switch-like behaviour

We next consider a scenario where differential regulation occurs along a linear trajectory instead of a bifurcation. This inference problem is more subtle than a bifurcating process given the lack of a clear

branching or clustering structure upon which regulatory interactions depend. We consider a model network depicted in Figure 5: notably the network either contains a feed-forward loop $g2 \rightarrow g3 \rightarrow g4$ or $g2 \rightarrow g4 \rightarrow g3$, depending on the position of a cell along the trajectory. The interaction $g2 \rightarrow g4$ is first *indirect* and then becomes *direct*: the nature of this differential regulation is such that static GRN inference methods would observe a superposition of these interactions, i.e. $g2 \rightarrow g3 \rightarrow g4$ as well as $g2 \rightarrow g4$. As an example, this could potentially lead to the $g2 \rightarrow g4$ interaction being erroneously identified as an indirect interaction and pruned, if filtering heuristics such as CLR or the data-processing inequality were used [36, 16]. The utility of cell-specific networks is that the added resolution would allow these two effects to be disentangled.

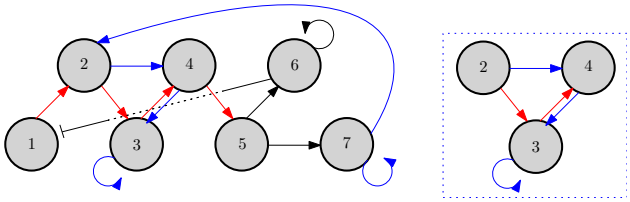


Figure 5: **Example switch network.** Blue (resp. red) interactions are active when $(g6 \wedge (\neg g5))$ is FALSE (resp. TRUE). Notably, gene 2 interacts both directly and indirectly with gene 4, depending on the system state.

We implement the network again using BoolODE and simulate 1,000 cells. As before, we apply locaTE using a transition kernel derived from the simulated velocity information and the same parameters as in the bifurcating dataset. We find from Figure 6(a, b) that the inferred interactions recapitulate the true directed interactions. In Figure 6 we find that locaTE significantly outperforms competing methods in terms of AUPRC, for both cell-specific and static networks. We find in particular that locaTE substantially outperforms CeSpGRN in both settings for this example, in contrast to the previous example where CeSpGRN performed well for static inference.

Together, these two simulated examples demonstrate that locaTE is able to use dynamical information in the form of transition matrices to improve inference of directed interactions, often substantially. Unsurprisingly, better estimates of dynamics lead to better results for network inference. Perhaps what is more remarkable is that even after aggregating cell-specific networks produced by locaTE, the resulting static networks generally outperform competing static inference methods.

3.3 mESC differentiation dataset using StationaryOT

We consider the dataset of Hayashi et al. (2018), [40], consisting of 456 cells and 100 TFs. This dataset is a time-series of mouse embryonic stem cells differentiating through to primitive endoderm, sampled at 0, 12, 24, 48, 72h. Cells in this differentiation process follow a linear trajectory, as can be seen in Figure 7. To infer trajectories, we apply StationaryOT [33] using quadratically-regularised optimal transport to ensure a sparse transition kernel. Using this transition matrix, we infer cell-specific networks with locaTE, resulting in a $456 \times 100 \times 100$ array of interaction scores.

Since no ground truth is available for cell-specific networks in this experimental dataset, we are only able to validate the output of locaTE against a static ground truth. In [18] the authors constructed a reference network from known regulatory interactions in the embryonic stem cell atlas from pluripotency evidence (ESCAPE) database [44], which consists of 19 regulators and 100 targets. Starting from the array of cell-specific interaction scores, we obtain a static 100×100 GRN by averaging interaction scores across cells and computed precision-recall curves relative to the ground truth interactions (Figure 7(c)). Cells from the last 10% of pseudotime were left out from averaging since informative dynamics are lost at the end of the observed trajectory. For comparison, we run CeSpGRN, which infers cell-specific undirected networks, and a collection of static inference methods. We compared a subnetwork of the inferred networks to the ESCAPE reference. In the directed setting, we find that locaTE outperforms other methods in terms of AUPRC, and reasonably well in terms of EP (0.496, compared to a best EP of 0.553 for TENET and baseline of 0.343). For undirected inference locaTE is out-performed by SCODE (AUPRCs of 0.453 and 0.483 respectively). Together with the observation that SCODE performs poorly for directed inference suggests that, despite

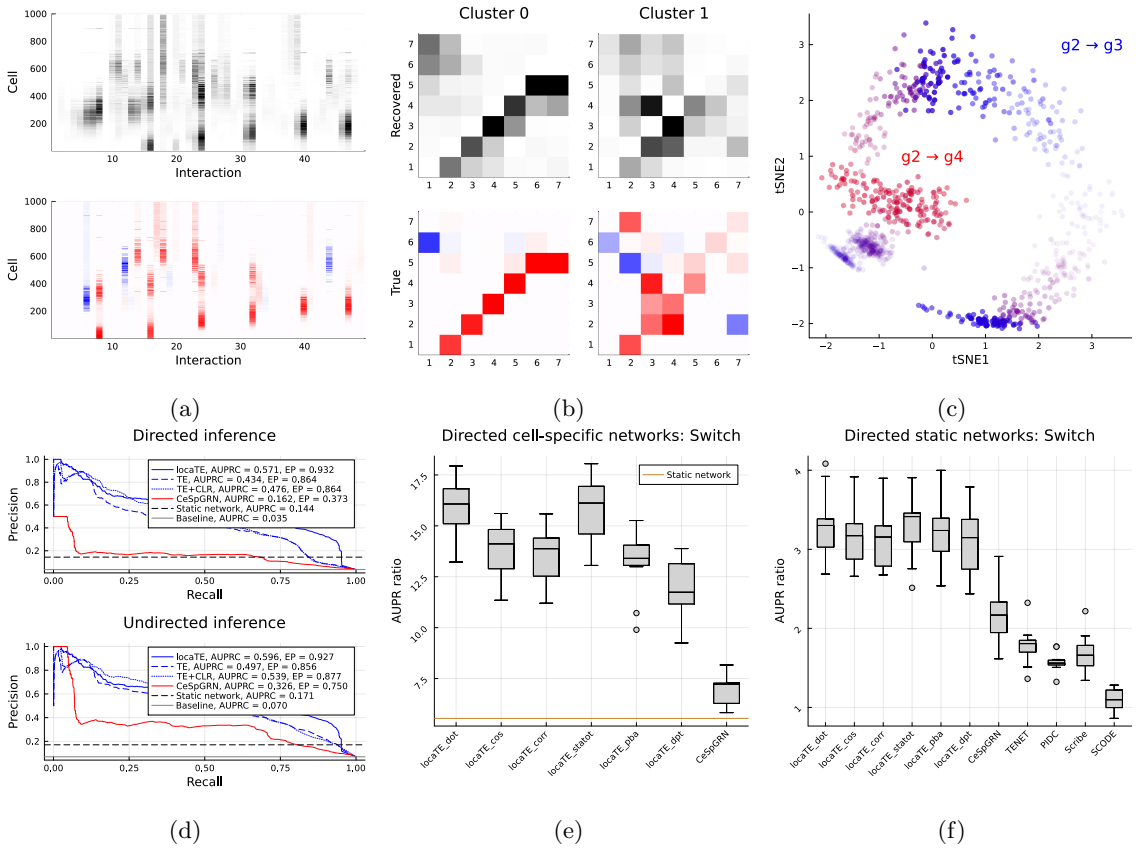
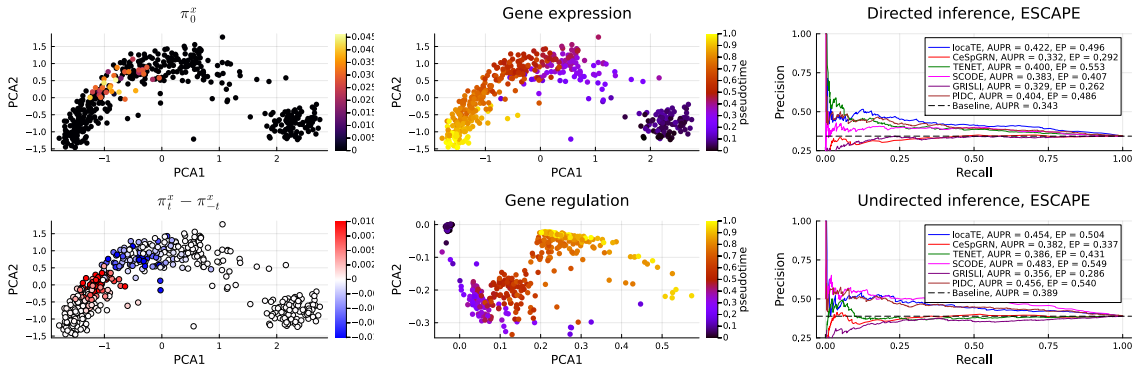


Figure 6: **Results for simulated switch network.** (a) LocaTE output as a $N \times M^2$ matrix in order of increasing pseudotime (top), shown alongside the ground truth Jacobians (bottom). (b) Interactions inferred by locaTE averaged by cluster (top) shown against ground truth interactions averaged by cluster. (c) Interaction scores for $g2 \rightarrow g3$ and $g2 \rightarrow g4$ for each cell on the tSNE embedding. (d) Precision-Recall (PR) curves for cell-specific edge detection from $\hat{G}$ (raw TE values), $\tilde{G}$ (TE + CLR values), or G (locaTE). Area under PR curve (AUPRC) and early precision at 10% recall (EP) is shown. (e, f) AUPRC ratio for locaTE and other methods across 10 sampled simulation datasets, in the cell-specific (e) and the static (f) setting. In the cell-specific setting, we show as a horizontal line the AUPRC ratio achieved by perfect knowledge of the static network.

being a method for directed inference, SCODE may be prone to detecting interactions but with the wrong directionality; PIDC [4] is the second best performing method in terms of AUPRC (0.456).

Since locaTE infers cell-specific networks, we can ask questions at finer resolution than the population scale. To demonstrate the utility of this type of analysis, we use locaTE-NMF with $k = 8$ factors to find regulatory modules which are active at different stages along the trajectory (Figure 8(a)). Hierarchical clustering of the inferred factor activities produces three groups, corresponding to "early", "mid", and "late" trajectory stages. For each cluster we compute a representative network by taking weighted averages of NMF factors, and the top 0.25% of edges for each stage are shown in Figure 8(b). It is clear that the nature of the inferred regulatory interactions changes drastically between stages. Among the top inferred interactions several are the canonical ones previously reported to be associated with maintenance of pluripotency and embryonic development, including Nanog-Zfp42 [45], Foxd3-Nanog [46] and Sall4-Klf2 [47] which are active early in the trajectory. Top inferred interactions for the late trajectory include Epas1 (Hif2a)-Pou5f1 (Oct4) which is known to play a role in embryonic development [48], and Epas1 (Hif2a)-Bhlhe40 (Dec1) which plays a role in suppression of Deptor [49], a stemness factor known to be downregulated during ES differentiation [50].

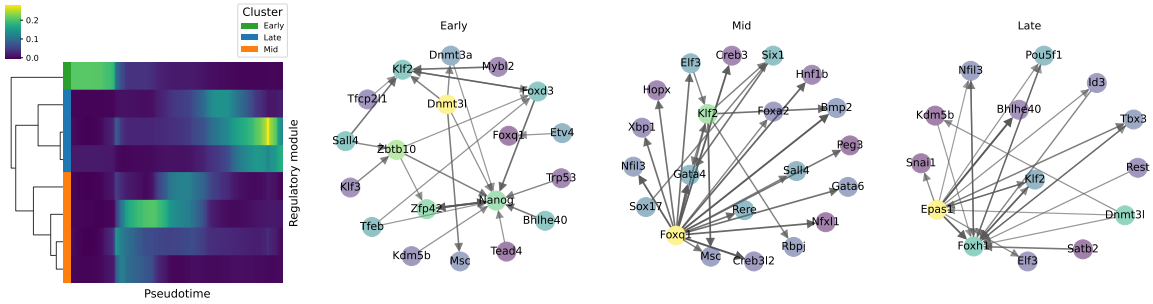
For each stage, we rank potential regulators by their out-edge eigenvector centrality. In Figure S16(a) we



(a) (b) (c)

Figure 7: **ES-endoderm development dataset.** (a) PCA embedding of 456 cells along an ES-endoderm differentiation trajectory, coloured by (top) example neighbourhood density (bottom) change in density along evolution of Markov process. (b) PCA embedding of (top) expression profiles (bottom) cell-specific interaction networks, coloured by diffusion pseudotime. (c) Precision-recall curves against the ESCAPE reference for static inference, both directed and undirected.

show the 25 top-ranked genes for each stage. We find that many of the top genes are known to be associated to stem cell development, including Nanog, Pou5f1 (Oct4), Sox2 [46], Gata6 [51], Zfp42 [45]. These gene lists are input to Metascape [52] to find corresponding Gene Ontology (GO) annotations (Figure S16). Among the gene set annotations found for the early trajectory, we find that enriched pathways with the highest confidence are those associated with pluripotency and embryonic morphogenesis. On the other hand, annotations for the mid and late trajectory largely correspond to cell fate commitment, endoderm formation, and organ development. These annotations reflect the developmental progression of embryonic stem cell differentiation to primitive endoderm.



(a) (b)

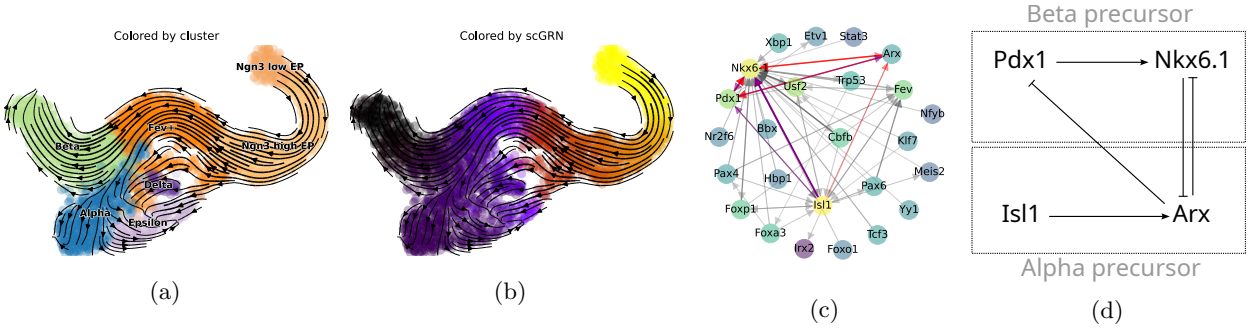
Figure 8: **Factor analysis of locaTE results for ES-endoderm dataset.** (a) Regulatory module activities found using locaTE-NMF, grouped by activity along the trajectory. (b) Inferred networks (top 0.25% of all edges) for early, mid and late modules. Nodes are coloured by out-edge eigenvector centrality.

3.4 Pancreas development dataset with RNA velocity

Finally, we consider a murine pancreatic development dataset of Bastidas-Ponce et al. [53], where aspects of the cellular dynamics are captured in the form of RNA velocity estimates. The dataset consists of 2,531 cells, and we filter for 82 transcription factors following the analysis in [19]. We apply the scVelo [54] package to produce RNA velocity estimates, and CellRank [22] with a cosine kernel to produce a transition matrix encoding the inferred dynamics on the cell set. Applying locaTE with the RNA-velocity derived transition matrix, we obtain a $2,531 \times 82 \times 82$ array of cell-specific interaction scores. In Figure 9 we show a

UMAP embedding of the full dataset with streamlines visualising RNA-velocity dynamics superimposed. To illustrate the heterogeneity in the networks inferred by locaTE, we also colour cells by their corresponding cell-specific GRN using the first diffusion component in "scGRN-space". This reveals a clear gradient along the differentiation trajectory, suggesting that the networks output by locaTE capture a progression in regulatory interactions over developmental time. We find that the strongest-scoring interactions involved the genes Fos and Egr1 which are well known to play a role in stress response pathways [55]. Fos expression has been documented to be an dissocation-induced artifact in scRNA-seq [56], and we therefore conservatively filter out interactions involving Fos or Egr1 in subsequent analyses.

As a first summary of the inference output, we produce an 80×80 static GRN by averaging, and as previously we exclude the last 10% of pseudotime. Figure 9(c) shows the top 1% of edges from this network. We observe that the genes with highest out-edge eigenvector centrality includes Nkx6-1, known to be a master regulator involved in the control of pancreatic beta cell development [57, 58], as well as Isl1 and Pdx1 which are known to be determinants of alpha and beta cell fate [59]. In [57] the authors reported evidence supporting a 4-gene network (henceforth referred to as the α/β network) involving Nkx6-1, Pdx1, Isl1 and Arx that drives alpha and beta fate determination (Figure 9(d)). We find that all 4 genes have high out-edge centrality in the inferred static network, and furthermore we find that the α/β network is recapitulated in the top-1% edges inferred network, as highlighted by the coloured edges in Figure 9(c).



(a) (b) (c) (d)

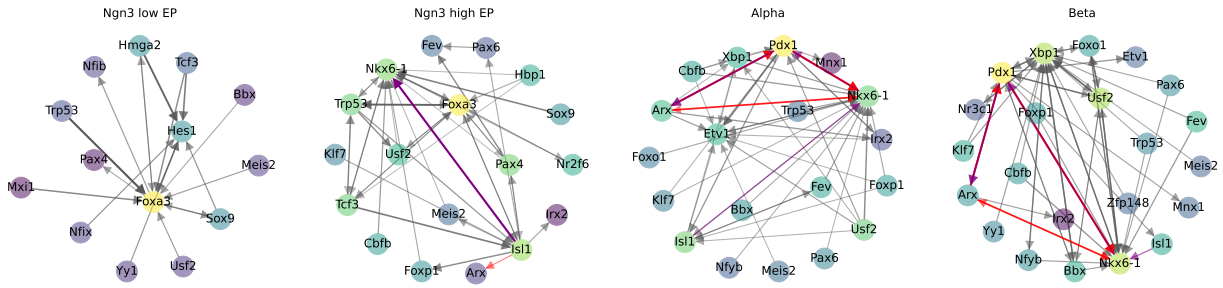
Figure 9: **Pancreatic development dataset.** (a, b) UMAP embedding of murine pancreatic development dataset [53], with smoothed RNA velocity vector field shown. Cells are coloured according to their cluster annotation and inferred cell-specific GRN in a one-dimensional diffusion embedding. (c) Static GRN constructed by averaging over first 90% of pseudotime. Top 1% of edges are shown, after filtering for stress response genes. Direct (indirect) edges belonging to the α/β subnetwork are shown in red (purple). Nodes are coloured by out-edge eigenvector centrality. (d) α/β subnetwork driving development of pancreatic alpha and beta cells [57].

In this dataset annotations for cell types are available. To investigate differences in regulatory state between subpopulations, we produce cluster-averaged networks for each celltype. In Figure 10(a) we show top 1% edges of cluster-averaged, filtered networks for the Ngn3− EP (endocrine progenitor), Ngn3+ EP, beta and alpha clusters (see Appendix S17 for remaining clusters). We note that the inferred network for Ngn3− EP cells includes Sox9-Hes1. Sox9 is known to play a key role in maintenance of pancreatic progenitor cell state by regulating Hes1 [60], an inhibitor of Neurog3 (Ngn3) [59]. This interaction is no longer present in the Ngn3+ network, further suggesting its role in the mechanism for maintaining the Ngn3− cell state. This interaction also does not appear in the static GRN in Figure 9(c), illustrating the utility of cell-specific networks in capturing localised signals in a dataset which would be otherwise lost by population averaging. Out-edge eigenvector centrality of the Ngn3− EP cluster identifies additional potential regulators including Foxa3, known to be implicated in endodermal organ formation [61] and Hmga2, a chromatin-associated protein expressed during development that does not directly regulate downstream genes but may have widespread indirect effects via chromatin interactions [62].

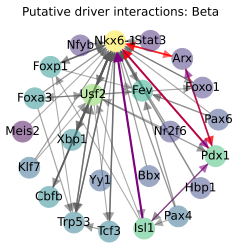
The clusters corresponding to alpha and beta committed cells yield networks that are distinct from the progenitor clusters. In particular, we find that the α/β network is prominent in both subpopulations. This reflects the key role that these genes are hypothesised to play in the fate determination mechanism that underlies this dataset. To investigate this at a finer scale we visualised expression levels and interaction scores

for the Nkx6.1-Arx interaction in Figure 10(c). We observe a fate decision boundary from expression levels that reflects the competition between Nkx6.1 and Arx corresponding to beta and alpha fate. The interaction scores for both the Nkx6.1-Arx edge and the entire α/β subnetwork are also strongest near this boundary. Examining again the out-edge eigenvector centrality, we identify Xbp1 as a top regulator specific to the alpha and beta subpopulations, recently found to be critical to maintenance of beta-cell identity [63].

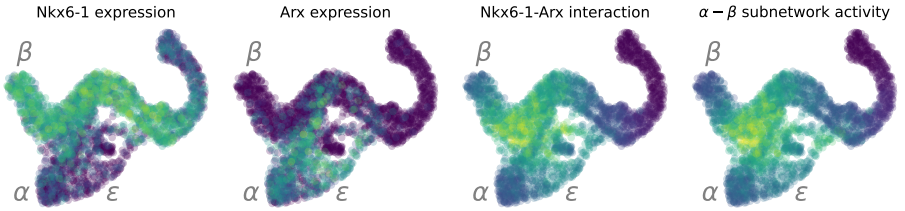
Finally, modelling of cellular dynamics as Markov processes naturally allows estimation of fate commitment in terms of absorption probabilities [20, 22, 33] and so examining correlation of gene interactions with fate commitment can be used to construct a putative "lineage driver" network that is correlated to a cell fate of interest. In Figure 10(b) we construct such a network for the beta cell lineage. As we expect, the α/β network is strongly correlated to beta cell fate.



(a)



(b)



(c)

Figure 10: **Inferred developmental GRNs.** (a) Averaged scGRNs from the pancreatic dataset are shown for selected clusters corresponding to Ngn3−, Ngn+ endocrine progenitor, alpha and beta cells respectively. Top 1% of edges are shown, after filtering for edges involving stress response gene Fos and Egr1. (b) Putative driver interactions associated with beta-cell fate. Direct (indirect) edges belonging to the α/β subnetwork are shown in red (purple). Nodes are coloured by out-edge eigenvector centrality. (c) Nkx6.1-Arx expression and interaction scores in UMAP coordinates, and average activity of the 4-gene α/β subnetwork of [57].

4 Conclusion

Motivated by recent developments in dynamical inference for single-cell data, we have presented a framework which enables reconstruction of cell-specific, causal networks. Given a transition kernel that approximates underlying cellular dynamics, our method relies on computing the transfer entropy (TE) [24, 29] for pairs of genes to measure causal relationships and makes no assumptions on linearity of interactions, the distribution of gene expression counts, nor the topology of the trajectory.

By using neighbourhoods in gene expression space we can effectively infer cell-specific networks without going to the extreme $N = 1$ case. How to pool information across cells most effectively will ultimately decide on the success of such methods and the insights they provide. Temporal ordering helps enormously, but we are still, by and large, not able to collect meaningful single cell transcriptomic time-course data. Pseudo-temporal

ordering or some of the many versions of RNA velocity entering the literature can provide useful information, as can, related multi-omic data, such as scATAC-seq.

Biology is dynamic and a static representation of a network can never do justice to what we observe phenotypically. To discern molecular causes, however, will require experimental and theoretical advances to progress in lock-step. The power of single cell data, and the insights we can gain from such data, crucially rely on flexible statistical analyses that allow us to dissect the differences between gene regulatory networks (and programs) and how they differ between cells.

Directions for future work include investigating approaches using which local information can leveraged to deal with technical zeros (dropout) in single cell data. We investigated the application of data-imputation methods for the pancreatic dataset, however, we found that imputation worsened performance for all inference methods considered. This effect has been reported previously being due to spurious signals being amplified [64] between non-interacting genes. The problem of inferring cellular dynamics, upon which application of methods such as locaTE crucially depend, is far from solved. In particular, accurately modelling and quantifying RNA velocity remains challenging in practice [65, 38]. Metabolic labelling assays may also provide a more accurate source of dynamical information [23]. Lastly, a further route for improving locaTE and network inference in general is to use relevant prior information to guide network inference.

Acknowledgements

The first author would like to thank Geoffrey Schiebinger, Léo Diaz and Wilder Scott for insightful discussions. The first author acknowledges support from the Australian Government Research Training Program and an Elizabeth and Vernon Puzey Scholarship.

A Method

A.1 Code availability

An open-source Julia implementation of locaTE is available at `https://github.com/zsteve/locaTE.jl`.

A.2 Constructing a backward operator

It is straightforward to write:

$$\begin{aligned} Q_{ij} = \mathbf{P}[X_{-t} = x_j | X_0 = x_i] &= \frac{\mathbf{P}[X_0 = x_i | X_{-t} = x_j]\mathbf{P}[X_{-t} = x_j]}{\mathbf{P}[X_0 = x_i]} \\ &= \frac{(P^t)_{ji}\mathbf{P}[X_{-t} = x_j]}{\mathbf{P}[X_0 = x_i]} \end{aligned}$$

If P encoded a reversible Markov chain and we chose $\mathbf{P}[X_{-t} = \cdot]$ to be the stationary distribution, this would give us time-reversal at equilibrium. Since we are interested in the behaviour of the process away from equilibrium, we must prescribe $\mathbf{P}[X_{-t} = \cdot]$ away from equilibrium. In practice, we find that taking $\mathbf{P}[X_{-t} = \cdot] = \text{Unif}$ works well – this is equivalent to simply taking the transpose of P and rescaling, as done in [22]. We refer the reader to the analysis presented in Section 4.3 of [38] for a more detailed discussion on the transpose as a backward operator and its interpretation in the continuous limit.

A.3 (Smoothness, sparsity)-regularised regression

The problem (9) is convex and a straightforward numerical algorithm for its solution can be derived using the alternating direction method of multipliers (ADMM) [66]. With auxiliary variables W, Z with the same dimensions as X, the corresponding ADMM scheme is

$$X^{(k+1)} = (\alpha + \lambda_1 L + \rho I)^{-1}(\alpha \hat{G} + \rho(Z^{(k)} - W^{(k)})) \tag{11}$$

$$Z^{(k+1)} = \arg\min_{Z} \lambda_2 \|Z\|_1 + \frac{\rho}{2}\|X^{(k+1)} - Z^{(k)} + W^{(k)}\|_2^2 = \text{prox}_{\rho^{-1}\lambda_2\|\cdot\|_1}(X^{(k+1)} + W^{(k)}) \tag{12}$$

$$W^{(k+1)} = W^{(k)} + X^{(k+1)} - Z^{(k+1)} \tag{13}$$

where $\rho > 0$ is the ADMM relaxation parameter (we take this to be 0.05), $\alpha = \text{diag}(w)$ and

$$\text{prox}_{\lambda\|\cdot\|_1}(x) = \text{sgn}(x)(|x| - \lambda)_+,$$

interpreted elementwise.

A.4 Regularised non-negative matrix factorisation

Now we consider the same problem setting as (9), but this time we also require that the $N \times M^2$ matrix of recovered interactions G be low rank. For a target rank k, we seek "tall" matrices $U \in \mathbb{R}^{N\times k}, V \in \mathbb{R}^{M^2\times k}$ such that $G \approx UV^\top$. That is, for a cell i, its scGRN is given by a linear combination of the columns of V with coefficients $U_{i1}, \ldots, U_{ik}$:

$$G_{i\cdot} = (UV^\top)_{i\cdot} = \sum_{\ell=1}^{k} U_{i\ell} V_{\cdot\ell}.$$

We consider a general loss function of the form

$$\begin{aligned} \mathcal{L}(U, V; L, G) = &\frac{1}{2}\|UV^\top - G\|_2^2 + \frac{\alpha}{2}\,\text{tr}(VU^\top LUV^\top) - \beta\langle H, UV^\top\rangle \\ &+ \frac{\lambda_1}{2}\,\text{tr}(U^\top K_1 U) + \mu_1\|U\|_1 \\ &+ \frac{\lambda_2}{2}\,\text{tr}(V^\top K_2 V) + \mu_2\|V\|_1. \end{aligned}$$

We explain the loss function term by term. The first term encourages a fit to the matrix G of raw TE values. We impose a penalty on the smoothness of the low-rank reconstruction (w.r.t. the graph Laplacian L) with weight α. We add an (optional) affine term on the reconstruction, where the entries of the matrix H can be thought of as a prior on which interactions may occur. Positive (negative) entries of H would correspond to prior on edge existence (non-existence). Priors could be constructed from gene-gene correlations, prior knowledge, or additional assay data (e.g. scATAC or ChIP data). In addition, we can impose Tikhonov and sparsity penalties on the coefficients and atoms.

One can derive multiplicative update equations for the gradient descent, following the approach of [67]. For positivity-preserving updates, we decompose any Laplacian matrices into their positive and negative parts:

$$\begin{aligned} L &= D - A, \\ K_1 &= D_1 - A_1, \\ K_2 &= D_2 - A_2. \end{aligned}$$

$$\begin{aligned} U &\leftarrow \frac{U \odot (GV + \alpha AUV^\top V + \lambda_1 A_1 U + \beta HV)}{UV^\top V + \alpha DUV^\top V + \lambda_1 D_1 U + \mu_1} \\ V &\leftarrow \frac{V \odot (G^\top U + \alpha VU^\top AU + \lambda_2 A_2 V + \beta H^\top U)}{VU^\top U + \alpha VU^\top DU + \lambda_2 D_2 V + \mu_2}. \end{aligned}$$

A.5 Regularised non-negative tensor factorisation

Let G have dimensions $N \times M \times M$, but for notational convenience we do our derivations in a general setting. Write for convenience $X = S \times_{i=1}^{d} A^{(i)}$, where the multilinear product is defined as

$$(A \times_k B)_{i_1,\ldots,i_{k-1},j,i_{k+1},\ldots,i_d} = \sum_{i_k=1}^{n_k} A_{i_1,\ldots,i_k,\ldots,i_d} B_{j,i_k}.$$

In order to derive a multiplicative update scheme, all matrices need to be decomposed into their positive and negative parts. We write $X_{(k)}$ for the matricization of X along its k-th mode. We want to derive multiplicative update scheme for the problem

$$\begin{aligned} \mathcal{L}(S, \{A^{(i)}\}_{i=1}^3) = &\frac{1}{2}\|X - G\|_2^2 + \frac{\alpha}{2}\operatorname{tr}(X_{(1)}^\top L X_{(1)}) - \beta\langle H, X\rangle \\ &+ \sum_{i=1}^{3} \frac{\lambda_i}{2}\operatorname{tr}((A^{(i)})^\top L^{(i)} A^{(i)}) \\ &+ \sum_{i=1}^{3} \mu_i \|A^{(i)}\|_1 \end{aligned}$$

The final two terms are straightforward to differentiate. For the first term, note that

$$\partial_{A^{(k)}} \frac{1}{2}\|X - G\|_2^2 = \partial_{A^{(k)}} \frac{1}{2}\|X_{(k)} - G_{(k)}\|_2^2$$

Now employ the identity $X_{(k)} = \left[S \times_{i=1}^{d} A^{(i)}\right]_{(k)} = A^{(k)}\left[S \times_{i\neq k} A^{(i)}\right]_{(k)}$. Let $B^{(k)} = \left[S \times_{i\neq k} A^{(i)}\right]_{(k)}$ for convenience, then one has

$$\partial_{A^{(k)}} \frac{1}{2}\|X - G\|_2^2 = A^{(k)} B^{(k)} (B^{(k)})^\top - G_{(k)} (B^{(k)})^\top.$$

The second term is more involved. First consider the case $k = 1$, so $X_{(1)} = A^{(1)} B^{(1)}$.

$$\begin{aligned} \partial_{A^{(1)}} \frac{1}{2}\langle X_{(1)}, L X_{(1)}\rangle &= \partial_{A^{(1)}} \frac{1}{2}\operatorname{tr}\left((B^{(1)})^\top (A^{(1)})^\top L A^{(1)} B^{(1)}\right) \\ &= L A^{(1)} B^{(1)} (B^{(1)})^\top. \end{aligned}$$

For $k \neq 1$, note that if we "re-fold" $LX_{(1)}$ along mode 1, we get

$$S \times_1 (LA^{(1)}) \times_{j>1} A^{(j)}.$$

So,

$$\begin{aligned}
\frac{1}{2}\langle X_{(1)}, LX_{(1)}\rangle &= \frac{1}{2}\langle S \times_{i=1}^{d} A^{(i)}, S \times_1 (LA^{(1)}) \times_{j>1} A^{(j)}\rangle \\
&= \frac{1}{2}\langle A^{(k)}B^{(k)}, A^{(k)}\left[S \times_1 (LA^{(1)}) \times_{j>1, j\neq k} A^{(j)}\right]_{(k)}\rangle \\
&= \frac{1}{2}\langle A^{(k)}B^{(k)}, A^{(k)}\tilde{B}^{(k)}\rangle,
\end{aligned}$$

where $\tilde{B}^{(k)} = \left[S \times_1 (LA^{(1)}) \times_{j>1, j\neq k} A^{(j)}\right]_{(k)}$. Then,

$$\partial_{A^{(k)}} \frac{1}{2}\langle A^{(k)}B^{(k)}, A^{(k)}\tilde{B}^{(k)}\rangle = \frac{1}{2}A^{(k)}\left(\tilde{B}^{(k)}(B^{(k)})^\top + B^{(k)}(\tilde{B}^{(k)})^\top\right).$$

For the third term,

$$\partial_{A^{(k)}}\langle H, X\rangle = H_{(k)}(B^{(k)})^\top.$$

Combining the gradient expressions with the decomposition of Laplacians into positive and negative parts as done for NMF, we arrive at the update scheme

$$\begin{aligned}
A^{(1)} &\leftarrow \frac{A^{(1)} \odot (G_{(1)}(B^{(1)})^\top + \lambda_1 W_1 A^{(1)} + \alpha W A^{(1)} B^{(1)}(B^{(1)})^\top + \beta H_{(1)}(B^{(1)})^\top)}{A^{(1)}B^{(1)}(B^{(1)})^\top + \lambda_1 D_1 A^{(1)} + \alpha D A^{(1)} B^{(1)}(B^{(1)})^\top + \mu_1} \\
A^{(i)} &\leftarrow \frac{A^{(i)} \odot (G_{(i)}(B^{(i)})^\top + \lambda_i W_i A^{(i)} + \alpha A^{(i)} C_- + \beta H_{(i)}(B^{(i)})^\top)}{A^{(i)}B^{(i)}(B^{(i)})^\top + \lambda_i D_i A^{(i)} + \alpha A^{(i)} C_+ + \mu_i}, \quad i > 1,
\end{aligned}$$

where for each $i > 1$ we define for convenience

$$\begin{aligned}
C_+ &= (\tilde{B}_+^{(i)}(B^{(i)})^\top + B^{(i)}\tilde{B}_+^{(i)})/2 \\
C_- &= (\tilde{B}_-^{(i)}(B^{(i)})^\top + B^{(i)}\tilde{B}_-^{(i)})/2.
\end{aligned}$$

B Results details

B.1 Simulated data

For each boolean network considered, simulated datasets were generated using the BoolODE package [5], modified to record additionally the velocity vector of simulated cells as well as the Jacobian matrix of expression state. We provide below the boolean rules that we used for simulating the two examples in the main text; all other simulated trajectories were produced using rules provided as part of the BoolODE package.

Boolean rules: bifurcating

$$\begin{aligned}
&g1 \leftarrow \neg(g4 \vee g5) \\
&g2 \leftarrow g1 \\
&g3 \leftarrow g2 \\
&g4 \leftarrow ((g3 \vee g4) \wedge (\neg g5)) \\
&g5 \leftarrow ((g5 \vee g3) \wedge (\neg g4)) \\
&g6 \leftarrow ((g4 \wedge (\neg(g5 \vee g9))) \vee ((g7 \vee g6) \wedge (g5 \wedge (\neg g4)))) \\
&g7 \leftarrow ((g6 \wedge (g4 \wedge (\neg g5))) \vee (g8 \wedge (g5 \wedge (\neg g4)))) \\
&g8 \leftarrow ((g7 \wedge (g4 \wedge (\neg g5))) \vee (g9 \wedge (g5 \wedge (\neg g4)))) \\
&g9 \leftarrow (((g8 \vee g9) \wedge (g4 \wedge (\neg g5))) \vee (g5 \wedge (\neg(g4 \vee g6))))
\end{aligned}$$

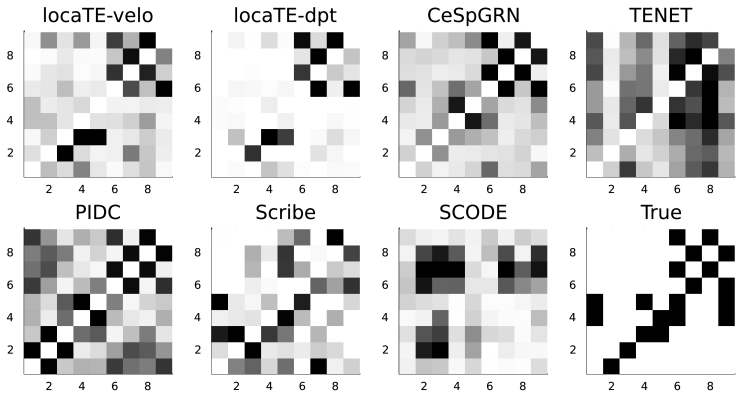


Figure S11: Best static networks inferred for bifurcating trajectory.

Boolean rules: switch

$$
\begin{aligned}
g1 &\leftarrow (\neg g6) \\
g2 &\leftarrow ((g1 \wedge (\neg(g6 \wedge (\neg g5)))) \vee (g7 \wedge ((g6 \wedge (\neg g5))))) \\
g3 &\leftarrow ((g2 \wedge (\neg(g6 \wedge (\neg g5)))) \vee ((g4 \vee g3) \wedge ((g6 \wedge (\neg g5))))) \\
g4 &\leftarrow ((g3 \wedge (\neg(g6 \wedge (\neg g5)))) \vee (g2 \wedge ((g6 \wedge (\neg g5))))) \\
g5 &\leftarrow ((g4 \wedge (\neg(g6 \wedge (\neg g5))))) \\
g6 &\leftarrow ((g5 \vee g6)) \\
g7 &\leftarrow (g5 \vee (g7 \wedge (g6 \wedge (\neg g5)) \wedge (\neg g2)))
\end{aligned}
$$

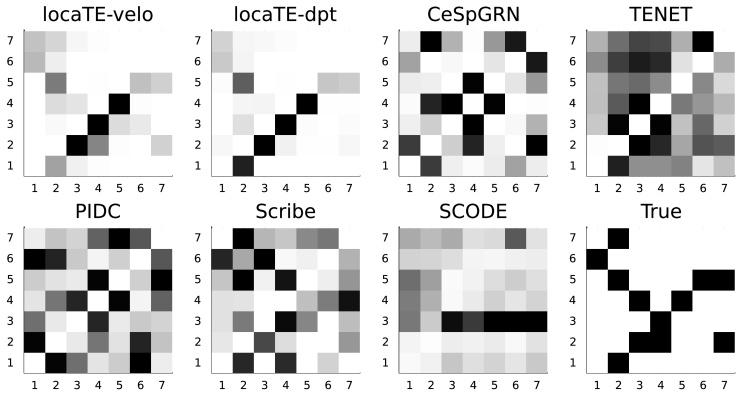


Figure S12: Best static networks inferred for switch-like trajectory.

Dimensionality reduction was performed using PCA and a k-NN constructed with $k = 25$. Diffusion pseudotime was then assigned with the root cell taken to be the simulated cell with the earliest simulation time. Various transition kernels P were then constructed as described below.

Velocity kernel As described in the main text, the velocity kernel was constructed from evaluations of the drift term of the chemical Langevin SDE using the `cellrank.tl.kernels.{ DotProductScheme, CorrelationScheme, CosineScheme }` classes within CellRank [22], with the previously computed k-NN graph and bandwidth σ estimated using the median heuristic.

Diffusion pseudotime kernel The diffusion pseudotime (DPT) kernel was constructed from the previously estimated diffusion pseudotime ordering using the `cellrank.tl.kernels.PseudotimeKernel` class within CellRank.

Optimal transport kernel Cells in the top 92.5%-97.5% of pseudotime were assigned a negative flux rate R_i so that the net negative flux was -50. Remaining cells were assigned a positive flux rate to satisfy the zero net flux requirement, $\sum_i R_i = 0$. StationaryOT [33] with quadratically regularised optimal transport was applied with $\Delta t = 1.0, \varepsilon = 0.05$. The cost matrix was taken to be the matrix of pairwise graph distances constructed from the k-NN graph. To prevent short-circuiting resulting from spurious edges, edges were ranked by their random-walk betweenness centrality and the top 5% of edges were pruned.

Population balance analysis (PBA) kernel Population balance analysis [20] was applied using the previously constructed k-NN graph, the same flux rates R_i as used in the optimal transport kernel, and diffusivity $D = 1.0$.

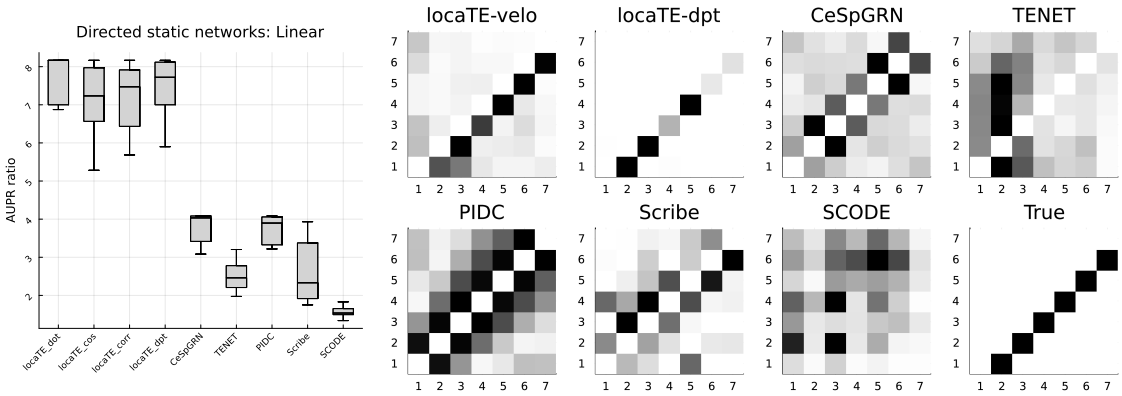
Inferring scGRNs LocaTE was applied using the simulated expression data and inferred transition kernels P. Expression values were discretised using the Bayesian blocks algorithm as implemented in [4]. The chemical Langevin simulation is a continuous approximation to biophysically realistic discrete expression dynamics, and as such produces expression values near zero. Expression values $< 10^{-0.5}$ were set to zero to prevent many bins from being created near zero. A neighbourhood kernel was constructed using quadratically regularised optimal transport, and the backward kernel was constructed from P as described in Section A.2. A grid-search was performed over $\tau \in \{1, \ldots, 5\}$, $\lambda_1 \in \{1, 2.5, 5, 10, 25\}$, $\lambda_2 \in \{0, 0.001, 0.005, 0.01, 0.025\}$. For comparison, CeSpGRN [13] was applied with a grid search performed over $|D(i)| \in \{5, 25, 50, 100, 250\}$, $\sigma \in \{0.05, 0.1, 0.25, 0.5, 1.0\}$, and $\lambda \in \{0.005, 0.01, 0.025, 0.05, 0.1\}$.

Evaluating scGRNs For each simulated dataset, ground truth scGRN networks were constructed by recording the Jacobian for each sampled cell state. Inferred scGRNs were evaluated against this ground truth by constructing Precision-Recall curves using the approach described in the main text (Section 3.1). Static GRNs were constructed by averaging scGRNs over the cell population.

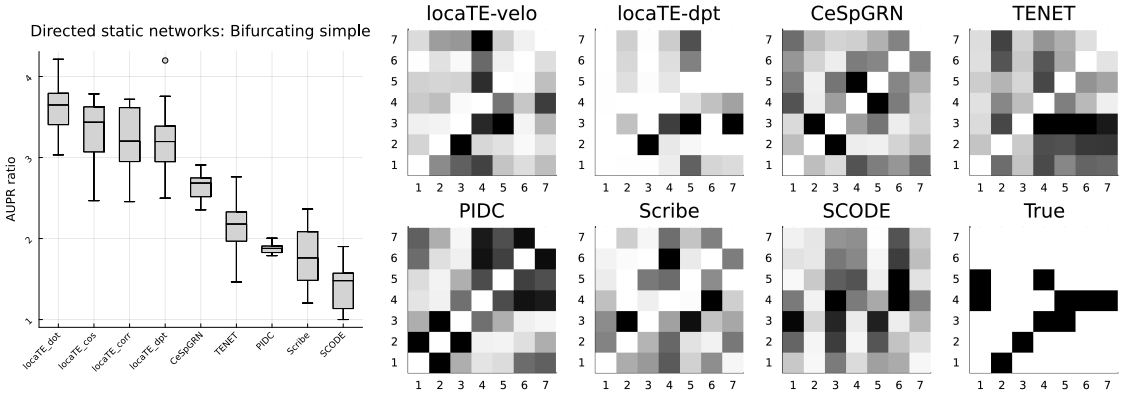
For comparison, PIDC [4], TENET [16], Scribe [15] and SCODE [17] were also applied to infer static networks. For TENET, we report the best results as measured by AUPRC ratio for a range of history lengths $\in \{1, \ldots, 8\}$.

B.2 Results: simple trajectories

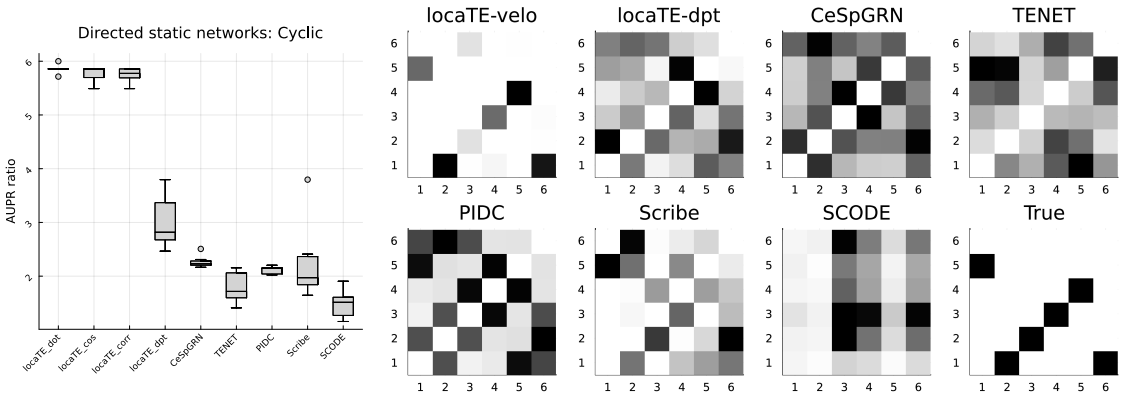
We applied locaTE with several choices of kernel to the simulated trajectory examples from [5]. For comparison, we also applied CeSpGRN, TENET, PIDC, Scribe and SCODE. A grid search was performed for locaTE, CeSpGRN and TENET, and default parameters were used for the remaining methods. Performance for each method was assessed in terms of the AUPRC ratio for directed edge inference. We summarise performance with boxplots below, and show representative examples of the recovered static networks.



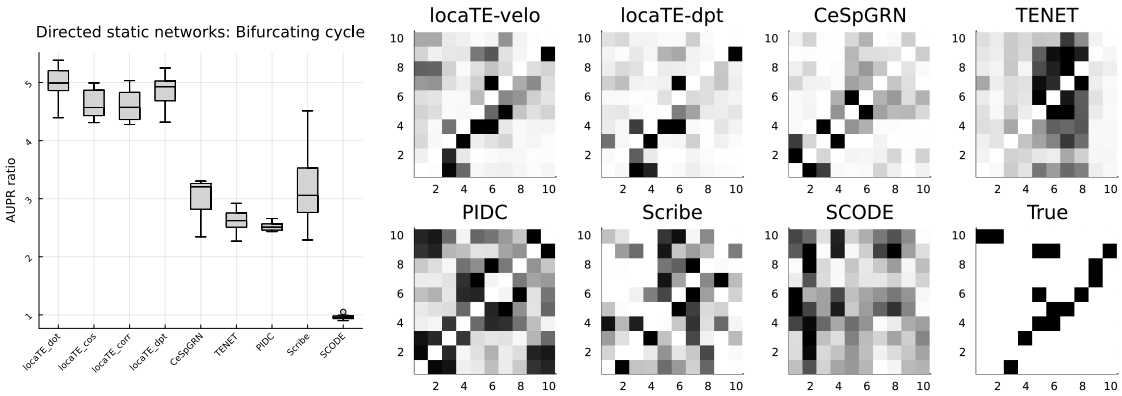
(a) Linear network, 1000 cells.



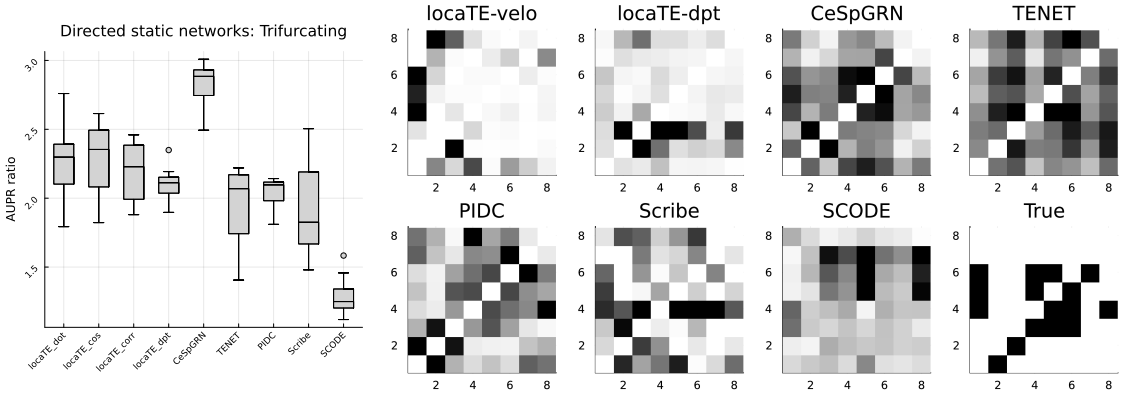
(b) Simple bifurcating network, 1000 cells.



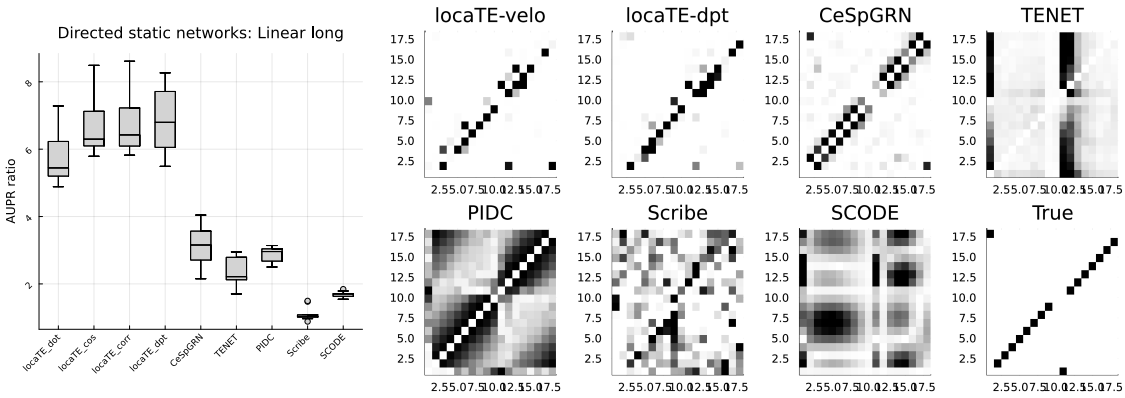
(c) Cycling network, 1000 cells.



(a) Bifurcating cycling network, 1000 cells.



(b) Trifurcating network, 1000 cells.



(c) Linear long network, 1000 cells.

B.3 mESC dataset

Log-transformed expression values for 100 TFs and pseudotime estimates were fetched from the SCODE example "data1" [17]. Cells were embedded into the top 25 principal components and a k-NN graph was constructed with $k = 25$. Using this k-NN graph, a cost matrix C of squared pairwise shortest-path distances was calculated using the Floyd-Warshall algorithm. StationaryOT [33] was then applied using this cost matrix, treating cells in the final 10% of pseudotime as sink cells. The flux rate R_i for sink cells was taken to be uniform such that $\sum_{i \in \text{sinks}} R_i = -50$, and similarly for non-sink cells so that $\sum_{i \notin \text{sinks}} R_i = 50$. Quadratically regularised optimal transport was applied with $\varepsilon = \overline{C}$, i.e. the mean value of C. The resulting transition matrix was used as the forward transition matrix for locaTE. The ESCAPE reference network of [18] was retrieved from `https://github.com/gitter-lab/SINGE-supplemental`.

LocaTE was applied to estimate TE using the transition kernel derived from StationaryOT with $\tau = 1$. Expression values were discretised using the Bayesian blocks algorithm as implemented in [4], with artifactually small values ($< 10^{-2}$) set to zero in order to prevent many bins being created near zero. Raw TE values were first filtered using weighted CLR, and denoising regression was applied with $\lambda_1 = 10, \lambda_2 = 10^{-3}$ using the normalised k-NN graph Laplacian.

To apply locaTE-NMF, the array of raw TE values was scaled to have unit 0.9-quantile. The target rank was set to $k = 8$, L was taken to be the normalised k-NN graph Laplacian, and $K_1 = K_2 = I$. We chose $\alpha = 10, \beta = 0, \lambda_1 = \lambda_2 = \mu_1 = \mu_2 = 1$.

For comparison, PIDC [4], GRISLI [19], TENET [16], SCODE [17] and CeSpGRN [13] were applied using the same log-transformed expression values and pseudotime estimates. For TENET a history length of 1 was used, following the usage guidelines in its documentation. We found that TENET tended to perform worse with longer history lengths. For SCODE, the rank parameter was set to $D = 4$ which is the recommended value. For CeSpGRN, we took a neighbourhood size $|N(i)| = 25$, bandwidth $\sigma = 5$, and sparsity level $\lambda = 0.1$ in keeping with the suggested parameter ranges in [13, Section 3.1]. All other parameters were taken to be their default values.

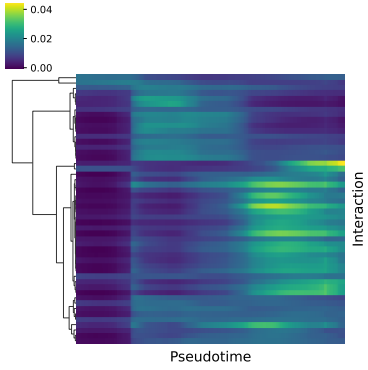
B.4 Pancreatic dataset

Expression data was processed and RNA velocity was estimated following the CellRank tutorial available at `https://cellrank.readthedocs.io/en/stable/cellrank_basics.html`. A transition kernel derived from RNA velocity was produced using `cr.tl.kernels.VelocityKernel` using the cosine similarity scheme. Pseudotime estimates for visualisation were calculated using the function `scvelo.tl.latent_time`. Intersecting the genes that passed filtering with the list of TFs from [19] yielded a subset of 82 TFs. LocaTE was applied using log-transformed expression values of the TF subset and the transition kernel derived from RNA velocity, with $\tau = 1$. All other parameters were chosen to be the same as for the mESC dataset.

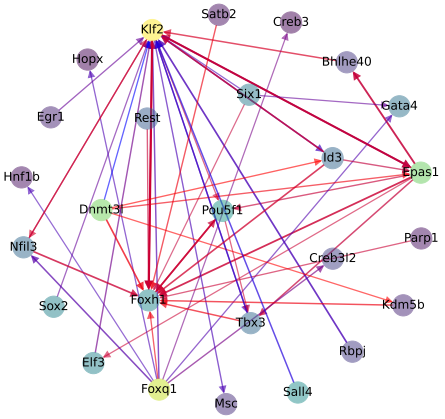
C Additional figures

C.1 mESC dataset

Additional figures for the mESC dataset.

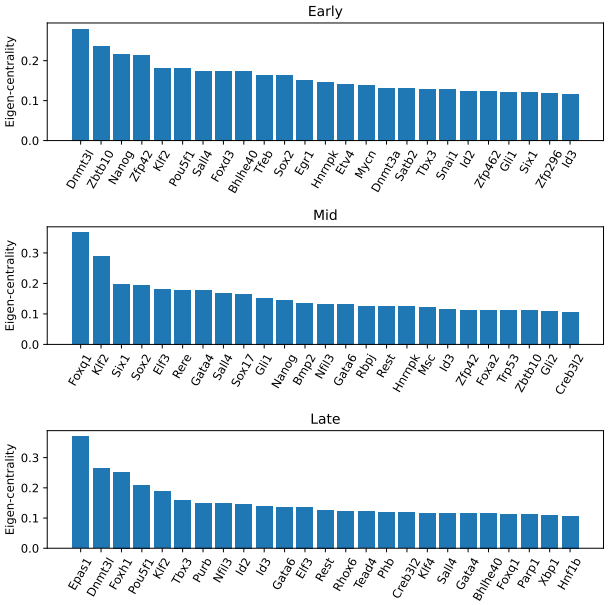


(a)

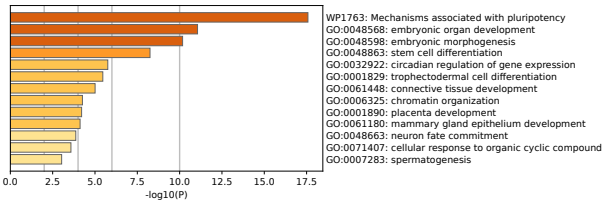


(b)

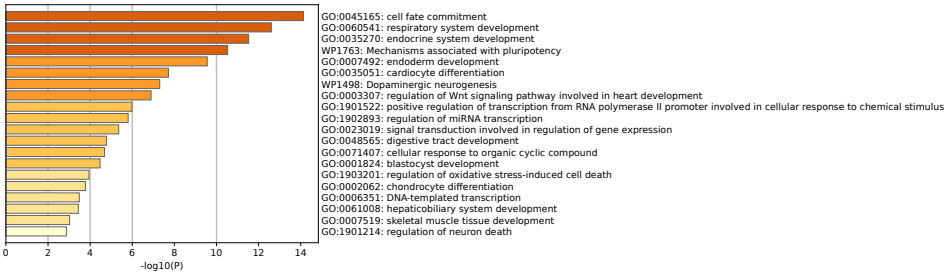
Figure S15: Smoothed plot of top 1% of inferred interactions (from static network) (a) clustered and shown against pseudotime (b) coloured by the time of peak activity, where blue corresponds to earliest, and red corresponds to latest.



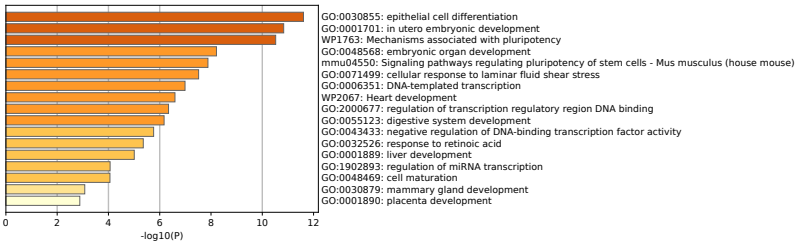
(a)



(b) Early



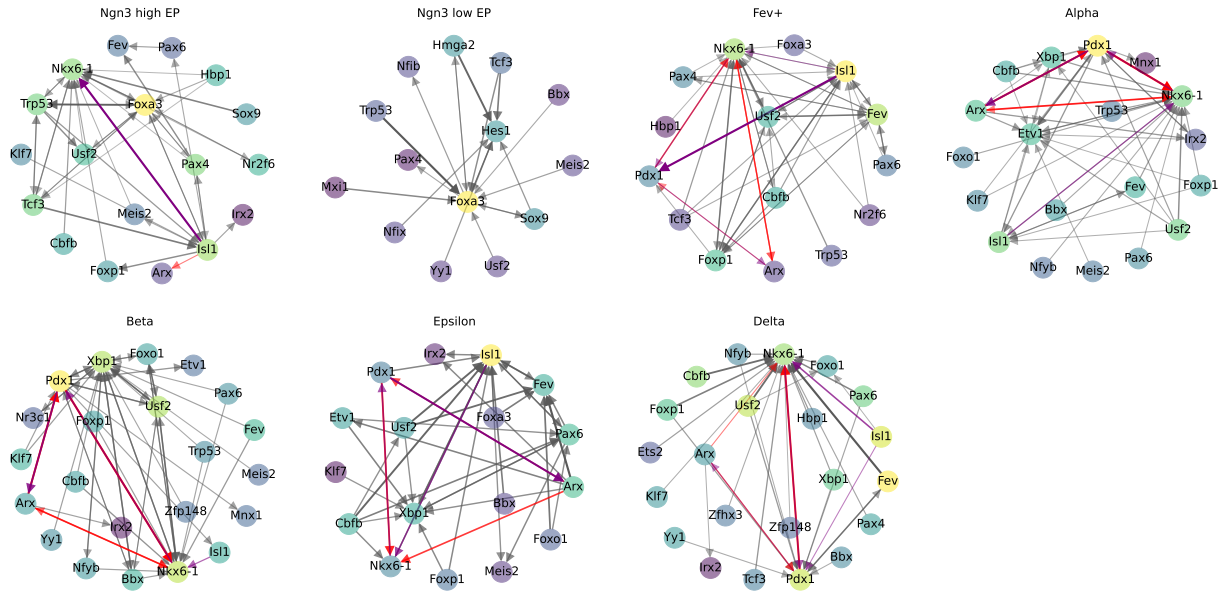
(c) Mid



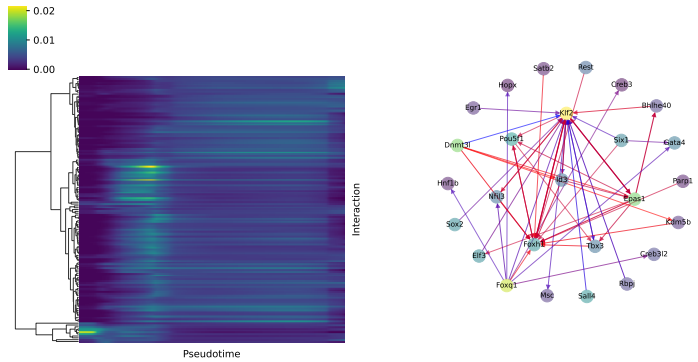
(d) Late

Figure S16: (a) Top 25 TFs for each stage by outgoing eigenvector centrality (b-d) GO terms found by Metascape for top 25 early/mid/late TFs.

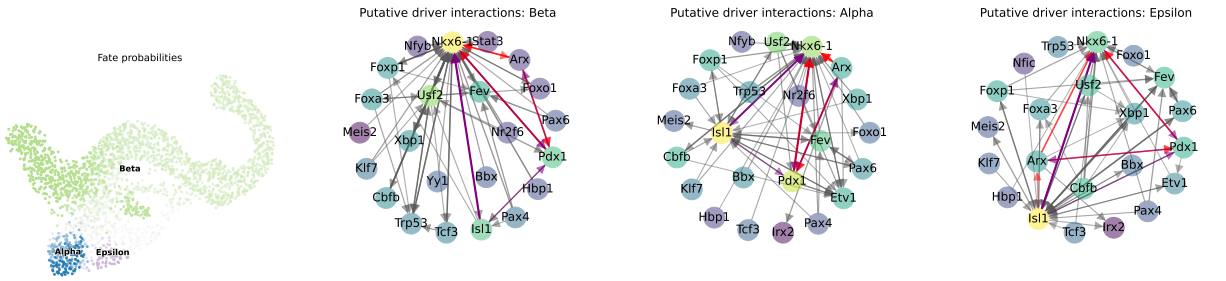
C.2 Pancreatic development dataset



(a)

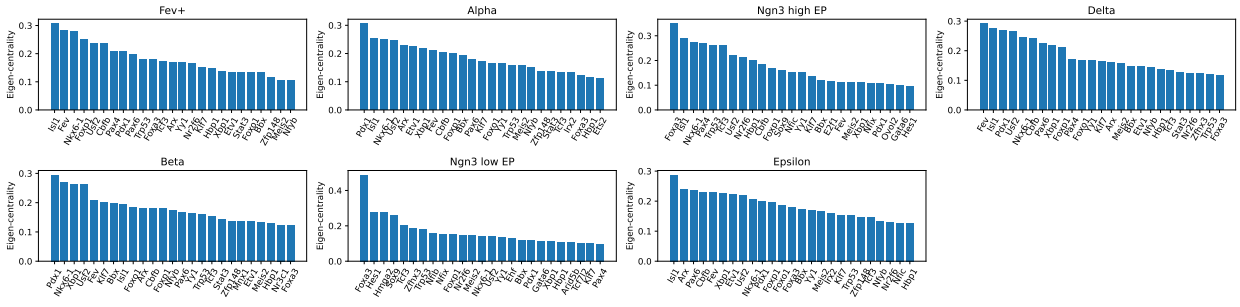


(b)

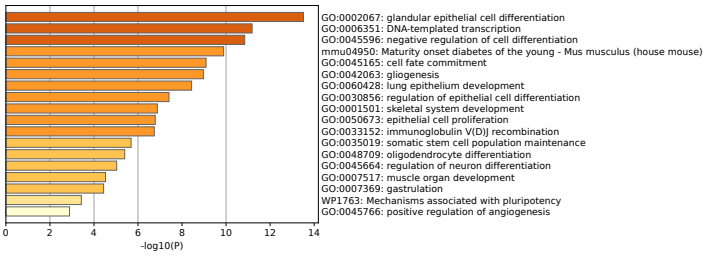


(c)

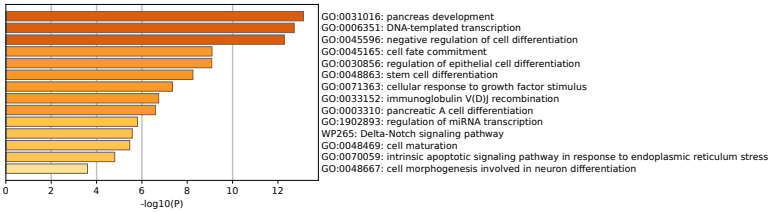
Figure S17: (a) Per-cluster networks (b) Smoothed plot of top 2.5% of interactions clustered and shown against pseudotime, and corresponding static network with edges coloured by time of peak activity (c) Putative driver networks for alpha, beta and epsilon fates.



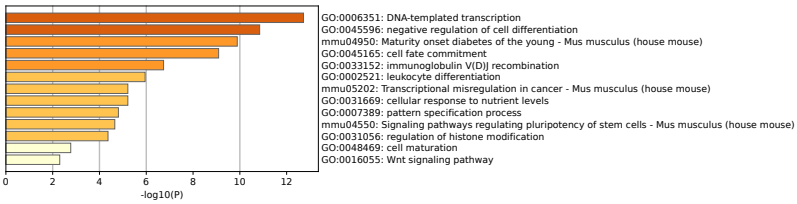
(a)



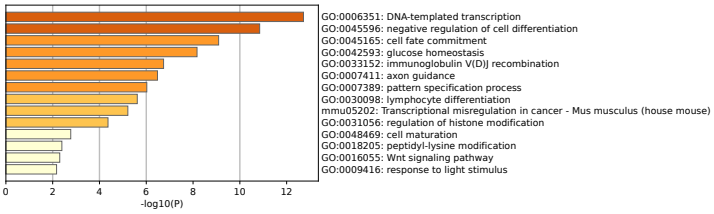
(b) Ngn3- EP cluster



(c) Ngn3+ EP cluster



(d) Beta cluster



(e) Alpha cluster

Figure S18: (a) Top 25 TFs for each stage by outgoing eigenvector centrality (b-e) GO terms found by Metascape for top 25 TFs from Ngn3- EP, Ngn3+ EP, Beta, Alpha clusters.

References

[1] Florian Buettner, Kedar N Natarajan, F Paolo Casale, Valentina Proserpio, Antonio Scialdone, Fabian J Theis, Sarah A Teichmann, John C Marioni, and Oliver Stegle. Computational analysis of cell-to-cell heterogeneity in single-cell rna-sequencing data reveals hidden subpopulations of cells. *Nature biotechnology*, 33(2):155–160, 2015.

[2] Sophie Tritschler, Maren Büttner, David S Fischer, Marius Lange, Volker Bergen, Heiko Lickert, and Fabian J Theis. Concepts and limitations for learning developmental trajectories from single cell genomics. *Development*, 146(12):dev170506, 2019.

[3] Etienne Becht, Leland McInnes, John Healy, Charles-Antoine Dutertre, Immanuel WH Kwok, Lai Guan Ng, Florent Ginhoux, and Evan W Newell. Dimensionality reduction for visualizing single-cell data using umap. *Nature biotechnology*, 37(1):38–44, 2019.

[4] Thalia E Chan, Michael PH Stumpf, and Ann C Babtie. Gene regulatory network inference from single-cell data using multivariate information measures. *Cell systems*, 5(3):251–267, 2017.

[5] Aditya Pratapa, Amogh P Jalihal, Jeffrey N Law, Aditya Bharadwaj, and TM Murali. Benchmarking algorithms for gene regulatory network inference from single-cell transcriptomic data. *Nature methods*, 17(2):147–154, 2020.

[6] Bradley Efron and Trevor Hastie. *Computer Age Statistical Inference, Student Edition: Algorithms, Evidence, and Data Science*, volume 6. Cambridge University Press, 2021.

[7] Michael PH Stumpf. Inferring better gene regulation networks from single-cell data. *Current Opinion in Systems Biology*, 27:100342, 2021.

[8] Kyle Akers and TM Murali. Gene regulatory network inference in single-cell biology. *Current Opinion in Systems Biology*, 26:87–97, 2021.

[9] Shuonan Chen and Jessica C Mar. Evaluating methods of inferring gene regulatory networks highlights their lack of performance for single cell gene expression data. *BMC bioinformatics*, 19(1):232–221, 2018.

[10] Patrick S Stumpf, Rosanna CG Smith, Michael Lenz, Andreas Schuppert, Franz-Josef Müller, Ann Babtie, Thalia E Chan, Michael PH Stumpf, Colin P Please, Sam D Howison, et al. Stem cell differentiation as a non-markov stochastic process. *Cell Systems*, 5(3):268–282, 2017.

[11] Elias Ventre, Ulysse Herbach, Thibault Espinasse, Gérard Benoit, and Olivier Gandrillon. One model fits all: combining inference and simulation of gene regulatory networks. *bioRxiv*, 2022.

[12] Robrecht Cannoodt, Wouter Saelens, Louise Deconinck, and Yvan Saeys. Spearheading future omics analyses using dyngen, a multi-modal simulator of single cells. *Nature Communications*, 12(1):1–9, 2021.

[13] Ziqi Zhang, Jongseok Han, Le Song, and Xiuwei Zhang. Inferring cell-specific gene regulatory networks from single cell gene expression data. *bioRxiv*, 2022.

[14] Xuran Wang, David Choi, and Kathryn Roeder. Constructing local cell-specific networks from single-cell data. *Proceedings of the National Academy of Sciences*, 118(51), 2021.

[15] Xiaojie Qiu, Arman Rahimzamani, Li Wang, Bingcheng Ren, Qi Mao, Timothy Durham, José L McFaline-Figueroa, Lauren Saunders, Cole Trapnell, and Sreeram Kannan. Inferring causal gene regulatory networks from coupled single-cell expression dynamics using scribe. *Cell systems*, 10(3):265–274, 2020.

[16] Junil Kim, Simon T. Jakobsen, Kedar N Natarajan, and Kyoung-Jae Won. Tenet: gene network reconstruction using transfer entropy reveals key regulatory factors from single cell transcriptomic data. *Nucleic acids research*, 49(1):e1–e1, 2021.

[17] Hirotaka Matsumoto, Hisanori Kiryu, Chikara Furusawa, Minoru SH Ko, Shigeru BH Ko, Norio Gouda, Tetsutaro Hayashi, and Itoshi Nikaido. Scode: an efficient regulatory network inference algorithm from single-cell rna-seq during differentiation. *Bioinformatics*, 33(15):2314–2321, 2017.

[18] Atul Deshpande, Li-Fang Chu, Ron Stewart, and Anthony Gitter. Network inference with granger causality ensembles on single-cell transcriptomics. *Cell reports*, 38(6):110333, 2022.

[19] Pierre-Cyril Aubin-Frankowski and Jean-Philippe Vert. Gene regulation inference from single-cell rna-seq data with linear differential equations and velocity inference. *Bioinformatics*, 36(18):4774–4780, 2020.

[20] Caleb Weinreb, Samuel Wolock, Betsabeh K Tusi, Merav Socolovsky, and Allon M Klein. Fundamental limits on dynamic inference from single-cell snapshots. *Proceedings of the National Academy of Sciences*, 115(10):E2467–E2476, 2018.

[21] Geoffrey Schiebinger. Reconstructing developmental landscapes and trajectories from single-cell data. *Current Opinion in Systems Biology*, 27:100351, 2021.

[22] Marius Lange, Volker Bergen, Michal Klein, Manu Setty, Bernhard Reuter, Mostafa Bakhti, Heiko Lickert, Meshal Ansari, Janine Schniering, Herbert B Schiller, et al. Cellrank for directed single-cell fate mapping. *Nature methods*, pages 1–12, 2022.

[23] Xiaojie Qiu, Yan Zhang, Jorge D Martin-Rufino, Chen Weng, Shayan Hosseinzadeh, Dian Yang, Angela N Pogson, Marco Y Hein, Kyung Hoi Joseph Min, Li Wang, et al. Mapping transcriptomic vector fields of single cells. *Cell*, 185(4):690–711, 2022.

[24] Thomas Schreiber. Measuring information transfer. *Physical review letters*, 85(2):461, 2000.

[25] Lionel Barnett, Adam B Barrett, and Anil K Seth. Granger causality and transfer entropy are equivalent for gaussian variables. *Physical review letters*, 103(23):238701, 2009.

[26] Daniel T Gillespie. The chemical langevin equation. *The Journal of Chemical Physics*, 113(1):297–306, 2000.

[27] Crispin Gardiner. *Stochastic Methods: A Handbook For The Natural And Social Sciences*. Springer, 2009.

[28] MA Coomer, L Ham, and MPH Stumpf. Noise distorts the epigenetic landscape and shapes cell-fate decisions. *Cell Syst*, 13(1):83–102, 2022.

[29] Terry Bossomaier, Lionel Barnett, Michael Harré, and Joseph T Lizier. Transfer entropy. In *An introduction to transfer entropy*, pages 65–95. Springer, 2016.

[30] Thomas M. Cover and Joy A. Thomas. *Elements Of Information Theory*. J. Wiley, 2005.

[31] Adam A Margolin, Ilya Nemenman, Katia Basso, Chris Wiggins, Gustavo Stolovitzky, Riccardo Dalla Favera, and Andrea Califano. Aracne: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. In *BMC bioinformatics*, volume 7, pages 1–15. BioMed Central, 2006.

[32] Louise Deconinck, Robrecht Cannoodt, Wouter Saelens, Bart Deplancke, and Yvan Saeys. Recent advances in trajectory inference from single-cell omics data. *Current Opinion in Systems Biology*, 27:100344, 2021.

[33] Stephen Zhang, Anton Afanassiev, Laura Greenstreet, Tetsuya Matsumoto, and Geoffrey Schiebinger. Optimal transport analysis reveals trajectories in steady-state systems. *PLoS computational biology*, 17(12):e1009466, 2021.

[34] Kevin R Moon, Jay S Stanley III, Daniel Burkhardt, David van Dijk, Guy Wolf, and Smita Krishnaswamy. Manifold learning-based methods for analyzing single-cell rna-sequencing data. *Current Opinion in Systems Biology*, 7:36–46, 2018.

[35] Aviv Madar, Alex Greenfield, Eric Vanden-Eijnden, and Richard Bonneau. Dream3: network inference using dynamic context likelihood of relatedness and the inferelator. *PloS one*, 5(3):e9803, 2010.

[36] Jeremiah J Faith, Boris Hayete, Joshua T Thaden, Ilaria Mogno, Jamey Wierzbowski, Guillaume Cottarel, Simon Kasif, James J Collins, and Timothy S Gardner. Large-scale mapping and validation of escherichia coli transcriptional regulation from a compendium of expression profiles. *PLoS biology*, 5(1):e8, 2007.

[37] Sophie Lèbre, Jennifer Becq, Frédéric Devaux, Michael P H Stumpf, and Gaëlle Lelandais. Statistical inference of the time-varying structure of gene-regulation networks. *BMC Systems Biology*, 4(1):130, 2010.

[38] Tiejun Li, Jifan Shi, Yichong Wu, and Peijie Zhou. On the mathematics of rna velocity i: theoretical analysis. *bioRxiv*, 2020.

[39] Jeff Bezanson, Alan Edelman, Stefan Karpinski, and Viral B Shah. Julia: A fresh approach to numerical computing. *SIAM review*, 59(1):65–98, 2017.

[40] Tetsutaro Hayashi, Haruka Ozaki, Yohei Sasagawa, Mana Umeda, Hiroki Danno, and Itoshi Nikaido. Single-cell full-length total rna sequencing uncovers dynamics of recursive splicing and enhancer rnas. *Nature communications*, 9(1):1–16, 2018.

[41] Manu Setty, Vaidotas Kiseliovas, Jacob Levine, Adam Gayoso, Linas Mazutis, and Dana Pe'Er. Characterization of cell fate probabilities in single-cell data with palantir. *Nature biotechnology*, 37(4):451–460, 2019.

[42] T Matsumoto, S Zhang, and G Schiebinger. Beyond $k$nn: Adaptive, sparse neighborhood graphs via optimal transport. Submitted, 2022.

[43] Tamara G Kolda and Brett W Bader. Tensor decompositions and applications. *SIAM review*, 51(3):455–500, 2009.

[44] Huilei Xu, Caroline Baroukh, Ruth Dannenfelser, Edward Y Chen, Christopher M Tan, Yan Kou, Yujin E Kim, Ihor R Lemischka, and Avi Ma'ayan. Escape: database for integrating high-content published data collected from human and mouse embryonic stem cells. *Database*, 2013, 2013.

[45] Wenjing Shi, Hui Wang, Guangjin Pan, Yijie Geng, Yunqian Guo, and Duanqing Pei. Regulation of the pluripotency marker rex-1 by nanog and sox2. *Journal of biological chemistry*, 281(33):23319–23325, 2006.

[46] Guangjin Pan and James A Thomson. Nanog and transcriptional networks in embryonic stem cell pluripotency. *Cell research*, 17(1):42–49, 2007.

[47] Chin Yan Lim, Wai-Leong Tam, Jinqiu Zhang, Haw Siang Ang, Hui Jia, Leonard Lipovich, Huck-Hui Ng, Chia-Lin Wei, Wing Kin Sung, Paul Robson, et al. Sall4 regulates distinct transcription circuitries in different blastocyst-derived stem cell lineages. *Cell stem cell*, 3(5):543–554, 2008.

[48] Kelly L Covello, James Kehler, Hongwei Yu, John D Gordan, Andrew M Arsham, Cheng-Jun Hu, Patricia A Labosky, M Celeste Simon, and Brian Keith. Hif-2α regulates oct-4: effects of hypoxia on stem cell function, embryonic development, and tumor growth. *Genes & development*, 20(5):557–570, 2006.

[49] Hong Doan, Alexander Parsons, Shruthi Devkumar, Jogitha Selvarajah, Francesc Miralles, and Veronica A Carroll. Hif-mediated suppression of deptor confers resistance to mtor kinase inhibition in renal cancer. *Iscience*, 21:509–520, 2019.

[50] Pooja Agrawal, Joseph Reynolds, Shereen Chew, Deepak A Lamba, and Robert E Hughes. Deptor is a stemness factor that regulates pluripotency of embryonic stem cells. *Journal of Biological Chemistry*, 289(46):31818–31826, 2014.

[51] Edward E Morrisey, Zhihua Tang, Kirsten Sigrist, Min Min Lu, Fang Jiang, Hon S Ip, and Michael S Parmacek. Gata6 regulates hnf4 and is required for differentiation of visceral endoderm in the mouse embryo. *Genes & development*, 12(22):3579–3590, 1998.

[52] Yingyao Zhou, Bin Zhou, Lars Pache, Max Chang, Alireza Hadj Khodabakhshi, Olga Tanaseichuk, Christopher Benner, and Sumit K Chanda. Metascape provides a biologist-oriented resource for the analysis of systems-level datasets. *Nature communications*, 10(1):1–10, 2019.

[53] Aimée Bastidas-Ponce, Sophie Tritschler, Leander Dony, Katharina Scheibner, Marta Tarquis-Medina, Ciro Salinno, Silvia Schirge, Ingo Burtscher, Anika Böttcher, Fabian J Theis, et al. Comprehensive single cell mrna profiling reveals a detailed roadmap for pancreatic endocrinogenesis. *Development*, 146(12):dev173849, 2019.

[54] Volker Bergen, Marius Lange, Stefan Peidli, F Alexander Wolf, and Fabian J Theis. Generalizing rna velocity to transient cell states through dynamical modeling. *Nature biotechnology*, 38(12):1408–1414, 2020.

[55] EC Baechler, FM Batliwalla, George Karypis, PM Gaffney, K Moser, WA Ortmann, KJ Espe, S Balasubramanian, KM Hughes, JP Chan, et al. Expression levels for many genes in human peripheral blood cells are highly sensitive to ex vivo incubation. *Genes & Immunity*, 5(5):347–353, 2004.

[56] Léo Machado, Frédéric Relaix, and Philippos Mourikis. Stress relief: emerging methods to mitigate dissociation-induced artefacts. *Trends in Cell Biology*, 31(11):888–897, 2021.

[57] Ashleigh E Schaffer, Brandon L Taylor, Jacqueline R Benthuysen, Jingxuan Liu, Fabrizio Thorel, Weiping Yuan, Yang Jiao, Klaus H Kaestner, Pedro L Herrera, Mark A Magnuson, et al. Nkx6. 1 controls a gene regulatory network required for establishing and maintaining pancreatic beta cell identity. *PLoS genetics*, 9(1):e1003274, 2013.

[58] Brandon L Taylor, Fen-Fen Liu, and Maike Sander. Nkx6. 1 is essential for maintaining the functional state of pancreatic beta cells. *Cell reports*, 4(6):1262–1275, 2013.

[59] H Efsun Arda, Cecil M Benitez, and Seung K Kim. Gene regulatory networks governing pancreas development. *Developmental cell*, 25(1):5–13, 2013.

[60] Philip A Seymour, Kristine K Freude, Man N Tran, Erin E Mayes, Jan Jensen, Ralf Kist, Gerd Scherer, and Maike Sander. Sox9 is required for maintenance of the pancreatic progenitor cell pool. *Proceedings of the National Academy of Sciences*, 104(6):1865–1870, 2007.

[61] Richard I Sherwood, Tzong-Yang Albert Chen, and Douglas A Melton. Transcriptional dynamics of endodermal organ formation. *Developmental dynamics: an official publication of the American Association of Anatomists*, 238(1):29–42, 2009.

[62] Shin-Heng Chiou, Madeleine Dorsch, Eva Kusch, Santiago Naranjo, Margaret M Kozak, Albert C Koong, Monte M Winslow, and Barbara M Grüner. Hmga2 is dispensable for pancreatic cancer development, metastasis, and therapy resistance. *Scientific reports*, 8(1):1–10, 2018.

[63] Kailun Lee, Jeng Yie Chan, Cassandra Liang, Chi Kin Ip, Yan-Chuan Shi, Herbert Herzog, William E Hughes, Mohammed Bensellam, Viviane Delghingaro-Augusto, Mark E Koina, et al. Xbp1 maintains beta cell identity, represses beta-to-alpha cell transdifferentiation and protects against diabetic beta cell failure during metabolic stress in mice. *Diabetologia*, 65(6):984–996, 2022.

[64] Lam-Ha Ly and Martin Vingron. Effect of imputation on gene network reconstruction from single-cell rna-seq data. *Patterns*, 3(2):100414, 2022.

[65] Gennady Gorin, Meichen Fang, Tara Chari, and Lior Pachter. Rna velocity unraveled. *bioRxiv*, 2022.

[66] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, Jonathan Eckstein, et al. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine learning*, 3(1):1–122, 2011.

[67] Daniel D Lee and H Sebastian Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791, 1999.