DYNAMICAL SYSTEMS THEORY INFORMED LEARNING OF CELLULAR DIFFERENTIATION LANDSCAPES

A PREPRINT

Addison Howe *
Department of Engineering Sciences and Applied Mathematics
Northwestern University
Evanston, IL 60208, USA

Madhav Mani *
Department of Engineering Sciences and Applied Mathematics
Northwestern University
Evanston, IL 60208, USA
NSF-Simons Center for Quantitative Biology
Northwestern University
2205 Tech Drive
Evanston, IL 60208, USA

September 20, 2024

ABSTRACT

Successful development from a single cell to a complex, multicellular organism requires that cells differentiate in a coordinated, organized manner, in response to a number of chemical morphogens. While the molecular underpinnings may be extremely complex, the resulting phenomenon, in which a cell decides between one fate or another, is relatively simple. A body of work, rooted in dynamical systems theory, has formalized this notion of cellular decision making as flow in a landscape, in which cells evolve according to gradient-like dynamics within a potential that changes shape in response to a number of signals. This mathematical realization of Waddington's landscape suggests that certain quantifiable characteristics of cellular differentiation can be captured in relatively simple mathematical models, defined by an underlying potential function and a mapping of prescribed signals to their effect on the landscape. We provide a framework leveraging tools of machine learning to infer such a parameterized landscape model directly from gene expression data. The resulting model provides an intuitive, visualizable, and interpretable model of cellular differentiation dynamics. While previous approaches have successfully constructed models capturing cellular decision making, these typically model dynamics in an abstract space, rely on the identification of a discrete number of cell types, and significant prior knowledge of qualitative features of decision-making. In contrast, our data-driven approach infers a governing landscape atop a manifold within expression space, and thus describes the dynamics of interest within a space with direct, biological meaning.

Keywords machine learning · cellular differentiation · dynamical systems

Introduction

Modern technological advances allow us to study biological systems at the level of individual cells. Single-cell RNA sequencing (scRNA-seq) results in a transcriptomic profile for every observed cell, each consisting of thousands of genes. Even in the case of more targeted studies in which only a handful of genes are profiled, the resulting data from

*Corresponding authors `madhav.mani@gmail.com`, `addisonhowe2025@u.northwestern.edu`

these experiments are often high-dimensional enough to require nontrivial analysis. Much progress has been made in the way of discerning the different cell types present in such high-dimensional data, providing insight into the process of cellular differentiation in various contexts. Cellular differentiation, in which a progenitor cell develops into one of a number of more mature cell types, is a phenomenon essential to development, from the formation of the embryo, consisting of a diverse population of functional cells derived from a single cell, to the continuous production of blood cells from a pluripotent stem cell. The identification of new cell types and their hierarchy in a developmental tree is often the ultimate achievement of sequencing studies, and treated as tantamount to a functional understanding of the phenomenon. However, such an enumeration is not the end of the story, and it is less obvious how one begins to make sense of the *dynamical* information captured in such experiments. That is, while we might be able to confidently identify the different cell types into which a progenitor cell may differentiate, an understanding of the route through gene space that a cell takes in the process of making that decision, and the mechanisms by which such trajectories are biologically controlled, remain elusive. Understanding the intricacies of this path, and in particular the role of chemical signals in shaping it, can be at least as valuable as knowing the discrete cell types appearing along the way.

Making strides in our understanding of these dynamics, and the ways in which they depend on signaling, would facilitate our ability to design and control patterns of cellular differentiation and development. Development involves a complex, orchestrated series of events, each regulated by an array of chemical signals. These signals coordinate cellular differentiation and direct the proper number of cells, at the right time, to transition between states. This interplay between signaling and cell type coordination is the basis of synthetic biology, and our ability to successfully reproduce particular aspects of development depends in part on a full understanding of the connection between signaling and differentiation dynamics.

A number of approaches have been used to shed light on the dynamics of cellular differentiation. Measures of RNA velocity [1, 2] estimate the direction within gene space that a cell is moving, directly from sequencing data, assuming a simple model of regulatory dynamics. They do not, however, provide a direct link from chemical signals to cell fate decision making. Classical models of gene regulatory networks describe the interplay of genes using a system of differential equations, and thus explicitly model high-dimensional dynamics. However, the true form of these equations are generally unknown in the context of biology, and these models must therefore rely on either a gross simplification of reality, or suffer from a large number of parameters.

An alternative approach frames the problem in geometric terms, and models cellular decision making as flow in a landscape, an idea that dates back to Waddington, who introduced the "epigenetic landscape" [3]. This metaphor envisions a cell as a ball in a hilly terrain, which rolls downhill through valleys corresponding to particular cell states. Along the way, a valley may split into two diverging paths, representing a point at which a cell chooses between alternate fates. Eventually, the cell comes to reside in a particular valley, or basin, corresponding to its terminal state. Different wells represent the different cell fates that a primitive cell may eventually take. The tendency of the rolling ball to remain in a valley when slightly perturbed captures the biological notion of canalization.

In Waddington's landscape, the axis capturing depth (labeled t in Figure 1a) is often associated with a notion of developmental time, and is understood through our impression that gravity attracts the ball, thereby producing a directed, downward flow. In embryonic development, for instance, where there is a clear temporal direction, this interpretation is sensible, and one can treat time as a parameterization of depth in the landscape. Biologically speaking, it is now understood that within the embryo tissues synthesize and secrete signaling molecules, in a manner dependent on their state, that themselves elicit changes in those cells receiving the signal. As such, the interplay between the states of cells within a tissue and the signals they secrete and sense is what drives the self-organized and time-ordered dynamics of the nascent embryo. In contrast, *in vitro* studies of such dynamics that measure the response of cells to experimentally prescribed signaling timecourses attempt to decouple the dynamics of states and signals. While such simplified experimental scenarios preclude a deeper understanding of the exact nature of the coupling between signals and states, they allow us to study in quantitative precision the map from signals to cell state changes.

In these experimental settings, the parameter of interest is not time, but rather the set of governing signals. Now, depth in Waddington's landscape can be viewed as an axis along which one or more control parameters vary, thereby changing the states available to a developing cell. These states are still represented by the basins in each cross section of the terrain. In Figure 1a, each cross section is a one-dimensional curve, or potential. As the signal varies, the potential changes shape, and bifurcations can now occur in response to a change in signal. The notion of downhill flow is only realized given a prescription of a signaling timecourse. That is, a temporal function describing the signal dynamics induces an extension of the one-dimensional potential along a temporal axis, reproducing Waddington's illustration. A more general picture, however, entirely disregards a temporal parameterization, and depicts *all* possible realizations of the potential across the signaling, or control, axis. In the case of a one-dimensional potential, changing smoothly in response to a single signal, the resulting landscape would look much like Waddington's picture, but without a direction

of downward tilt. The interpretation now is that all cellular dynamics take place in the horizontal direction, and as the signal changes, the dynamics change in response to the new potential.

In recent years, a body of work has mathematically formalized, using the language of dynamical system theory, this notion of Waddington's landscape in which signals parameterize a family of potential functions [4, 5, 6, 7, 8]. In particular, the work of Rand et al. uses the theory of a simple, well-understood class of dynamical systems, called Morse–Smale systems, to enumerate all of the ways in which a three-attractor dynamical system, governed by two parameters, can undergo generic bifurcations. This work has been shown to be applicable to developmental systems, as a way to understand the most basic developmental decision, in which a cell chooses between two alternate fates [8].

However, such applications typically rely on a prior understanding of the developmental system in question. In particular, one must have knowledge of the cellular decision structure *a priori*, in order to choose a particular mathematical ansatz for the class of landscape suitable for the system. This knowledge may be available for certain model systems, to which decades of research have brought some degree of understanding. Otherwise, a sense of the decision structure may be deduced from careful analysis of the phenomenon. For example, in Sáez et al. [8], a preliminary assessment of experimental data suggests a particular decision structure, motivating the use of one of the forms enumerated by Rand et al..

Here, we investigate an approach to inferring a parameterized landscape model using a machine learning architecture, in which a time-independent, underlying potential function is parameterized by a neural network, and "tilted" in response to a number of time-dependent external signals. We begin by providing background on the mathematical formalization of a decision landscape as a potential that induces gradient dynamics and changes in response to signals. Then, we describe a *de novo* neural network-based model architecture that is designed to represent such a landscape. We show that this model can be used to infer essential features of an underlying landscape, given synthetic timecourse data modeling an experimental settings. Finally, we apply this model to a real dataset detailing cellular differentiation in the context of an *in vitro* system modeling the early mouse epiblast.

Background

In what follows, we describe an approach to modeling cellular decision making that is based on the inference, via a universal function approximator, of a potential, or landscape, that changes shape in a highly restricted manner, in response to a number of signals. For any given signal, the gradient of the potential defines the flow of cells through the landscape. Our use of such a parameterized landscape is largely inspired by the theory provided by Rand et al. [9] and utilized by Sáez et al. [8] in their construction of a decision landscape as a model of mouse stem cell differentiation. Below, we summarize the formalization of parameterized landscape models and their essential mathematical theory, emphasizing our use of gradient dynamical systems. We then discuss two parameterized landscape archetypes, presented in Sáez et al. [8], which we will use as synthetic testing grounds of our model.

Gradient systems as landscape models of differentiation

Mathematically, a *parameterized landscape* can be expressed as a potential function $\phi(\boldsymbol{x};\boldsymbol{p})$ defined over a phase space, $\boldsymbol{x} \in \Omega \subseteq \mathbb{R}^d$, and parameterized by a vector $\boldsymbol{p} \in \Gamma \subseteq \mathbb{R}^r$. As $\boldsymbol{p}$ changes, the landscape changes shape. The gradient of the potential induces a flow according to the vector field

$$\boldsymbol{F}(\boldsymbol{x};\boldsymbol{p}) = -\nabla_{\boldsymbol{x}}\phi(\boldsymbol{x};\boldsymbol{p}). \tag{1}$$

A priori, the landscape parameter vector need not carry any biological significance, however, we will shortly associate its role of altering the shape of the landscape with a set of signals affecting the biological system, and it is appropriate to think of the parameters as functions of these signals.

In the context of developmental biology, we view the flow of cells within a parameterized landscaped as a model of cellular differentiation. We associate particular cell states to its attractors, and interpret movement of cells from one attractor to another as a cell fate decision. This movement may be driven by stochastic fluctuations causing cells to escape one basin of attraction and to enter another. Alternatively, a change in the landscape, resulting from some change in the governing signals, might result in the disappearance of an occupied basin, causing cells there to relocate as they are no longer in the vicinity of an attractor. In either case, the particular route that cells take as they flow through the landscape is defined by the stable and unstable manifolds of the saddles and attractors, with saddles serving as demarcations of a decision point. Cells flow toward a saddle, then veer off in one direction or another, following a particular unstable manifold, to ultimately arrive at a new attractor.

Rand et al. [9] present parameterized landscapes as mathematical objects that derive from dynamical systems describing gene networks. They restrict their consideration to biologically relevant systems, namely those with a finite

number of fixed points and no recurrent behavior, a class of dynamical systems captured by *gradient-like Morse–Smale systems*. They then use the mathematical theory of these systems to enumerate the ways in which a three-attractor landscape can undergo generic bifurcations, in response to two control parameters. Their geometric approach provides a finite number of equivalence classes of dynamical systems, each of which encapsulates a particular *decision structure*, a simple connected graph encoding which attractors are connected. A connection between two attractors signifies that a cell can reach one attractor from the other, either as a result of a bifurcation, or by stochastically escaping one basin of attraction and entering another. These connections thereby represent a cellular transition, or *decision*.

While this work relies on an assumption of a small number of control parameters, it applies to systems of arbitrary dimension, and offers a minimal phase space representation of the relevant dynamics [9]. That is, dynamics that encapsulate the transition of cells between three attractor states in arbitrary dimension can be represented by dynamics in a two-dimensional system, with the low-dimensional dynamics given by a suitable normal form. This reduction, however, is based on an association of the dynamical system's fixed points to cell states, and in general does not provide a direct, global interpretation of the axes in the low-dimensional space. In particular, the new axes, corresponding to the variables of the normal form, lack a direct connection to the original phase space dimensions, i.e. genes.

Sáez et al. [8] utilize the enumeration and normal forms provided in Rand et al. [9] to construct a parameterized landscape model of cellular differentiation in an *in vitro* experimental context relating to mouse embryonic stem cell differentiation. This application illustrates an effective use of geometric normal forms, which capture global, qualitative dynamics, to infer quantitative features relating to the role of signaling in a biological system. They represent a five-state decision structure, consisting of two consecutive binary decisions, by algebraically stitching together two normal forms so that each governs dynamics in one half of the plane. In this construction, the x and y coordinates on which the landscape is defined lack an inherent biological meaning. Rather, each landscape attractor is associated with a cell type, and cells in the landscape are discretely classified in accordance with the basin of attraction they fall into. They then infer, for each half-plane region, a map from signals to the landscape parameters appearing in each of the normal forms, thereby learning the effective role of signals in altering the underlying landscape. The map from signals to their effect on the landscape dynamics offers an amount of additional interpretation to the landscape axes, as at least locally a given signal can be associated with a direction in the two-dimensional abstract phase space, namely that direction in which it biases the flow of cells. However, the abstract coordinate system still precludes a global interpretation that relates flow in the landscape directly to genetic factors.

This motivates an approach that attempts to infer a parameterized landscape atop a low-dimensional, but biologically meaningful, phase space. We provide a data-driven framework with which one can infer simultaneously a parameterized landscape, inducing gradient dynamics, along with a mapping of signals to their dynamical effect. While cellular decision making may take place in a high-dimensional gene space, and while the infrastructure that we outline can be applied to systems of arbitrary dimension, we restrict our consideration primarily to the case of inferring two-dimensional potentials, in systems where it is reasonable to assume that two effective signals are responsible for regulating the decision process. We assume that the relevant dynamics, in which a cell decides between one of two states, can be captured in a two-dimensional manifold embedded within the higher-dimensional space, and that on this manifold the dynamics are gradient-like. Moreover, the separation between the underlying landscape defined on the manifold, and the effect of signals in tilting that landscape, provides a powerful interpretation of the system, in which the manifold is exactly the space in which cells are controllable via signaling dynamics. By taking a data-driven approach to inferring an underlying manifold, a potential defined atop it, and a mapping from signals to tilts, we are able to model cellular decision making within a space that has both a direct interpretation with respect to the ambient expression space, as well as an interpretation with respect to the relevant signals.

Two parameterized landscape archetypes

The landscape model constructed by Sáez et al. [8] motivates our data-driven approach. We utilize the algebraic forms from which they construct a landscape *ansatz*, as well as the *in vitro* experimental data they provide, as respective synthetic and real-life testing grounds of our model. In this section, we describe these algebraic forms, termed the *binary choice* and the *binary flip*, and the specific decision structure that each encapsulates. Each form is in example of a particularly simple type of parameterized landscape, in which the essential structure of the landscape is fixed, and the influence of the parameters in altering the landscape is linear.

The binary choice landscape (Figure 1b) is given by

$$\phi_{bc}(x, y; \boldsymbol{p}) = x^4 + y^4 + y^3 - 4x^2y + y^2 + p_1x + p_2y \tag{2}$$

and admits fold bifurcations (a.k.a. saddle-node bifurcations) as p_1 and p_2 vary. In a generic fold bifurcation, a saddle point and an attractor or repeller either coalesce and vanish (the subcritical case) or appear together (the supercritical case). For this particular ansatz, there is a central region of parameter space where three stable fixed points exist. These

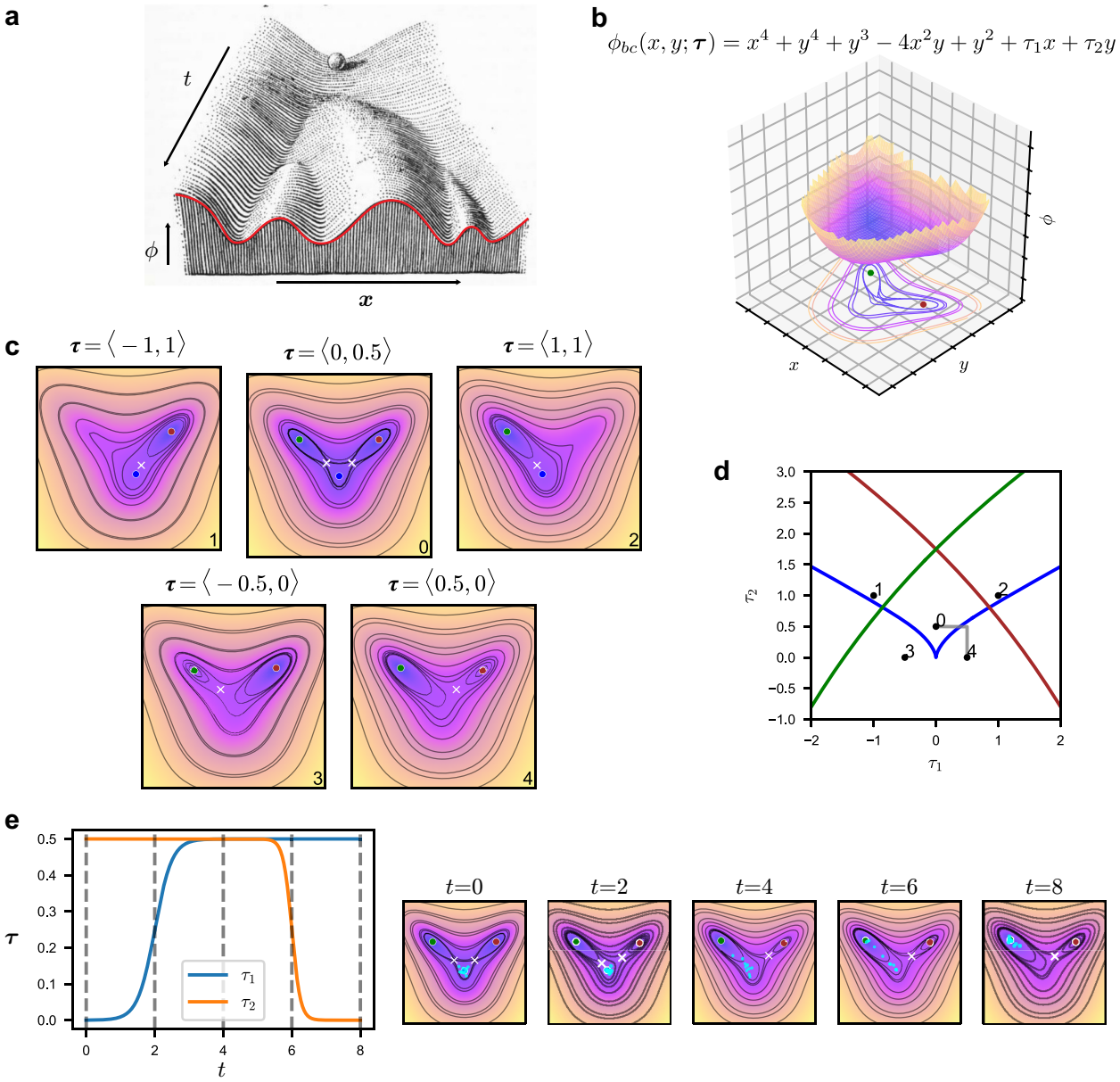

Figure 1: **A parameterized landscape models cellular differentiation dynamics, formalizing a notion of Waddington's landscape metaphor. (a)** Waddington's original illustration of the epigenetic landscape [3], with annotations added. The horizontal axis encapsulates the state of a cell, while depth corresponds to a notion of developmental time. A parameterized landscape, at a given value of the parameters, can be understood as a cross section of Waddington's landscape. The four valleys in the highlighted one-dimensional curve are analogous to the basins in a two-dimensional potential. **(b)** A representative parameterized landscape, the binary choice landscape [8]. The static potential function is given by $\tilde{\phi}_{bc}(x, y) = x^4 + y^4 + y^3 - 4x^2y + y^2$. The landscape tilts as the vector $\boldsymbol{\tau}$ changes, with the potential increasing in the direction of $\boldsymbol{\tau}$. **(c)** Heatmaps of the landscape given different values of the tilt $\boldsymbol{\tau}$. Each plot corresponds to a position in parameter space, as marked in panel (c). Minima are denoted with a circle, and saddle points with a cross. **(d)** The potential ϕ_{bc} exhibits fold bifurcations at particular values of $\boldsymbol{\tau}$. The color of each fold curve corresponds to the particular fixed point that vanishes/appears as $\boldsymbol{\tau}$ passes through it. **(e)** A timecourse of the parameters specifies the landscape's shape over time, and cells in the landscape move accordingly. In this way, the collection of potentials across the temporal axis can be understood as a higher-dimensional version of Waddington's illustration, with $\boldsymbol{x}$ two-dimensional instead of one-dimensional.

are configured such that a central attractor is separated from two peripheral attractors, with saddles in between. The unstable manifold of a saddle connects the central attractor to one of the two peripheral ones. As p_1 and p_2 vary, the fixed points move and the saddles may coalesce with the attractors, resulting in the bifurcation diagram shown in Figure 1c.

The binary flip landscape is given by

$$\phi_{bf}(x, y; \boldsymbol{p}) = x^4 + y^4 + x^3 - 2xy^2 - x^2 + p_1 x + p_2 y. \tag{3}$$

In this case, the configuration of the attractors and saddles is different. One saddle separates a central attractor from the two peripheral ones, and another saddle separates the two peripheral attractors from each other (see Figure S2). While saddle-node bifurcations can still occur, there is also the possibility of a *heteroclinic flip* bifurcation. In this global bifurcation, the unstable manifold of one saddle coalesces with the stable manifold of another, so that there is a heteroclinic orbit connecting one saddle to the other. While all attractors persist smoothly through the bifurcation, there is a sudden change in the unstable manifold of the saddle separating the central and peripheral attractors, as it jumps abruptly from connecting the saddle to one instead of the other.

The binary choice and binary flip landscapes each model a distinct decision structure. An essential difference between these forms is a connection between the two peripheral attractors. In the case of the binary choice, a bifurcation that results in the vanishing of a peripheral attractor leads cells in its vicinity to return to the central attractor; there is no route directly to the remaining periphery. In the binary flip, however, the peripheral attractors are connected via the unstable manifold of a saddle, and this provides a route from the vicinity of one periphery to the other.

The forms of Eqs. 2 and 3 share a commonality in the way that the parameters alter the underlying landscape. In both cases, p_1 and p_2 are coefficients of linear terms. In this way, they effectively "tilt" the landscape in the x and y directions, respectively. We will refer to such parameterized landscapes, in which the parameters $\boldsymbol{p}$ all act as linear coefficients, as *tiltable landscapes*. Formally, we can express a tiltable landscape as the sum of a nonlinear portion $\tilde{\phi}(\boldsymbol{x})$ that does not depend on the parameters, and a linear term:

$$\phi(\boldsymbol{x}; \boldsymbol{\tau}) = \tilde{\phi}(\boldsymbol{x}) + \boldsymbol{x}^T \boldsymbol{\tau} \tag{4}$$

where $\boldsymbol{\tau}$ is a vector of *tilt coefficients* (not to be confused with the use of τ, above, as a pseudo-temporal parameter of Waddington's landscape). Moving forward, we restrict our consideration to these tiltable landscapes, in which all parameters appear as coefficients of linear terms, and therefore will use $\boldsymbol{\tau}$ to refer to the landscape parameters.

In [8] the landscape parameters (p_1 and p_2, or $\boldsymbol{\tau}$, to use our notation) serve as intermediates between a set of signals in the system and their effect on the underlying landscape. Conceptually, this captures the notion that a cell interprets a given set of chemical signals, which then informs, or biases, its fate decision. Formally, we can think of the tilt $\boldsymbol{\tau}$ as a function of the signal vector $\boldsymbol{s} \in \mathbb{R}^{n_{\text{sigs}}}$, where n_{sigs} is the number of signals present in the system, and write

$$\boldsymbol{\tau} = \psi(\boldsymbol{s})$$

where $\psi : \mathbb{R}^{n_{\text{sigs}}} \to \mathbb{R}^d$ is a function describing how a cell processes the signal. We can then express the tiltable landscape in terms of the signals directly:

$$\phi(\boldsymbol{x}; \boldsymbol{s}) = \tilde{\phi}(\boldsymbol{x}) + \boldsymbol{x}^T \psi(\boldsymbol{s}). \tag{5}$$

Results

A model architecture encapsulating a tiltable landscape

We construct a model architecture to capture gradient dynamics of stochastic cells in a landscape that changes in response to a set of external signals, what we term a *Parameterized Landscape Neural Network*, or PLNN. This model can be broken down into three core modules, each of which captures an aspect of a tiltable landscape: an underlying static potential function, a mapping describing the effect of signals on the landscape, and a noise kernel capturing stochasticity. Throughout, we denote the learnable parameters of this model by $\boldsymbol{\theta}$. A schematic of this architecture is shown in Figure 2a.

The first module is the *potential module*, $\boldsymbol{\Phi} = \boldsymbol{\Phi}_{\boldsymbol{\theta}}$, and it is intended to represent an underlying static potential $\tilde{\phi}(\boldsymbol{x})$. Biologically speaking, this is the zero-signal state of the landscape. In order to represent a smooth, nonlinear potential function, we use a neural network architecture as part of our construction. We denote by $\boldsymbol{\Phi}_{\boldsymbol{\theta}}^{nn}$ a neural network whose weights and biases are among the learnable model parameters $\boldsymbol{\theta}$. In order to guarantee that the potential is smooth, we use a smooth activation function. Additional details concerning the specific architecture are provided in the SI (see S2).

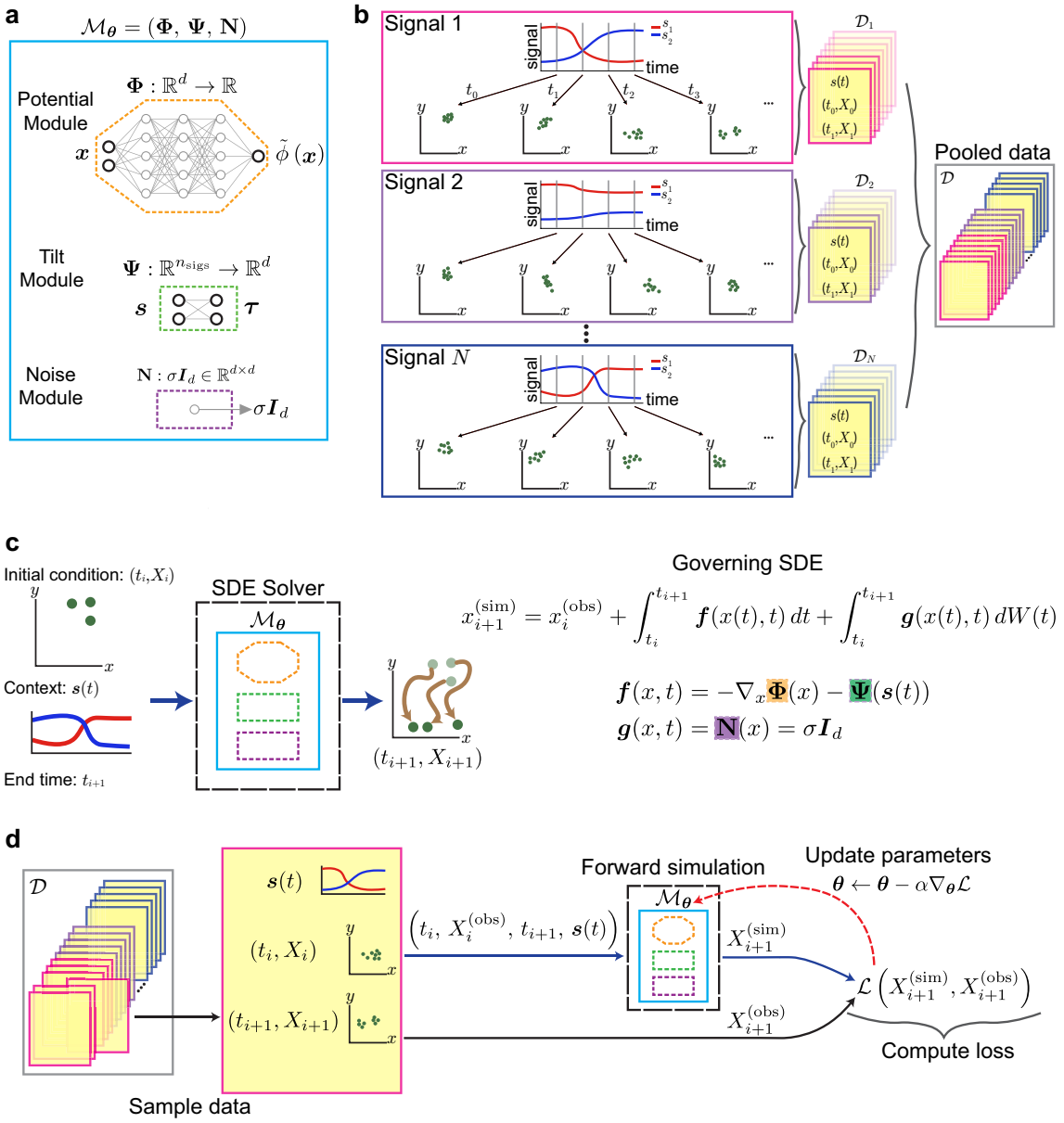

Figure 2: **A proposed model architecture and data modality, and the training process through which model parameters may be inferred. (a)** The PLNN model architecture. Three components capture the essential features of a tiltable landscape. The learnable parameters $\boldsymbol{\theta}$ of the model $\mathcal{M}_{\boldsymbol{\theta}}$ include the hidden weights and biases of the neural network $\boldsymbol{\Phi}$, the weights of the linear transformation $\boldsymbol{\Psi}$, and the scalar noise parameter σ. **(b)** Training data comes from a number of experiments in which an ensemble of cells are sampled at a set of timepoints under the condition of a particular signal profile. Two samples taken at different times constitute an initial/final state datapoint, represented by a single "card." The datapoints from each experiment are pooled together to form the training dataset $\mathcal{D}$. Note that within a given experiment all datapoints have the same signal profile $\boldsymbol{s}(t)$. **(c)** (Right) The governing SDE for cells in the ensemble. The state of a cell at a future timepoint can be written as the integration of a deterministic and stochastic function, each built from components of the model $\mathcal{M}$. This equation is vectorized across all cells in the ensemble. (Left) The inputs and outputs of a model landscape simulation. The inputs are an initial ensemble state X_i at time t_i, a signal timecourse defined over an interval $[t_i, t_{i+1}]$, and the end time of the interval, t_{i+1}. The model $\mathcal{M}$ is wrapped in a SDE solver that internally solves the SDE forward in time in order to sample a trajectory for each cell in the ensemble. The output is the simulated final state X_{i+1} corresponding to time t_{i+1}. **(d)** The training procedure for a PLNN. A datapoint is processed as in (c) to return a *simulated* final state $X_{i+1}^{(\text{sim})}$. This is compared to the *observed* final state $X_{i+1}^{(\text{obs})}$ using a suitable loss function. Backpropagation is then used to compute the gradient of this loss with respect to the model parameters, which are then updated. In practice, this procedure is vectorized and performed in batches.

We regularize the potential module by including an additional *confinement term*, denoted Φ_0. This regularization is meant to guarantee that trajectories are confined and do not escape to infinity during the training process. We define

$$\Phi_0(\boldsymbol{x}) = C_{\text{conf}}\|\boldsymbol{x}\|^4 \tag{6}$$

where C_{conf} is a non-negative constant that serves to greaten or lessen the degree of regularization. (Here, the confinement factor C_{conf} is a hyperparameter, and not a learned model parameter.) Then, the complete potential module is given by the sum

$$\boldsymbol{\Phi}_{\boldsymbol{\theta}}(\boldsymbol{x}) = \Phi_0(\boldsymbol{x}) + \boldsymbol{\Phi}^{nn}_{\boldsymbol{\theta}}(\boldsymbol{x}). \tag{7}$$

The second module captures the effect of signals on the underlying landscape, and we denote it by $\boldsymbol{\Psi} = \boldsymbol{\Psi}_{\boldsymbol{\theta}}$. In general, this module defines a transformation that maps a vector of signal values $\boldsymbol{s} \in \mathbb{R}^{n_{\text{sigs}}}$ to a vector $\boldsymbol{\tau} \in \mathbb{R}^d$, the effective tilt of the landscape. In parallel with Equations 4 and 5, given a tilt vector $\boldsymbol{\tau}$ or a signal vector $\boldsymbol{s}$, the height of the tilted landscape is given by

$$\phi(\boldsymbol{x}; \boldsymbol{\tau}) = \boldsymbol{\Phi}(\boldsymbol{x}) + \boldsymbol{x}^T\boldsymbol{\tau} \tag{8}$$

or

$$\phi(\boldsymbol{x}; \boldsymbol{s}) = \boldsymbol{\Phi}(\boldsymbol{x}) + \boldsymbol{x}^T\boldsymbol{\Psi}(\boldsymbol{s}), \tag{9}$$

respectively.

By design, the resulting tilt $\boldsymbol{\tau} = \boldsymbol{\Psi}(\boldsymbol{s})$ acts linearly on the underlying landscape. It is in addition to this restriction that we assume the signal processing function $\boldsymbol{\Psi}$ is linear, and write

$$\boldsymbol{\Psi}(\boldsymbol{s}) = \mathbf{A}\boldsymbol{s} \tag{10}$$

where $\mathbf{A} \in \mathbb{R}^{d\times n_{\text{sigs}}}$. Enforcing a linear processing of the signals in addition to a linear effect of the resulting vector on the landscape passes all nonlinearities to the potential module $\boldsymbol{\Phi}$ representing the underlying landscape $\tilde{\phi}$. To learn the transformation $\boldsymbol{\Psi}$ is to learn the effect of each signal on each dimension of the underlying landscape. This transformation is the same across the entire landscape, and raises the question as to how to interpret the impact of the signal within the vicinity of each fixed point. By subsuming all of the nonlinearities under the potential module, the interpretation is that the structure of the landscape will inform the exact nature of the role of the signal in local regions of phase space. It is entirely reasonable, of course, to posit that cells do not process an exogenous chemical signal in a linear fashion. In particular, we note that in the work of Sáez et al. [8], the inclusion of a memory kernel, whereby a prolonged signal would continue to influence the system even after it was halted, proved necessary for the model to accurately and robustly fit the data. More realistic models of signal processing, including saturation and memory effects, can in theory be built into the model architecture and subsequently inferred (see Discussion).

Finally, the third component of our model captures the stochasticity of the system, and we denote it by $\boldsymbol{N} = \boldsymbol{N}_{\boldsymbol{\theta}}$. In general, this module maps a cell's state $\boldsymbol{x}$ to a noise kernel, $\Sigma(\boldsymbol{x}) \in \mathbb{R}^{d\times d}$. We make a number of simplifying assumptions with regard to the system noise, but these assumptions may be weakened without much additional effort. We will assume that stochasticity in each dimension is the result of an independent 1-dimensional Wiener process, and that the noise is additive (i.e. it does not depend on the state). Moreover, we assume that a global parameter σ governs the magnitude of isotropic noise in the system. Together, these assumptions result in a diagonal noise kernel

$$\boldsymbol{N}_{\boldsymbol{\theta}}(\boldsymbol{x}) = \sigma\boldsymbol{I}_d \tag{11}$$

where $\boldsymbol{I}_d$ is the $d \times d$ identity matrix.

To summarize, our complete model is parameterized by the weights and biases of the neural network $\boldsymbol{\Phi}_{\boldsymbol{\theta}}$, the weights of the linear transformation $\boldsymbol{\Psi}_{\boldsymbol{\theta}} : \mathbb{R}^{n_{\text{sigs}}} \to \mathbb{R}^d$, and the global noise parameter σ. Further specifics of the model architecture, including the number of network layers and choice of activation functions in $\boldsymbol{\Phi}$, and the potential for a general noise kernel, can be found in the SI (see S2).

The architecture as described provides the essential components of a stochastic dynamical system, and allows us to simulate the evolution of an ensemble of cells within a changing potential. From this model, we derive a stochastic differential equation (SDE) and then sample cell trajectories from this system. In this way, we formulate a generative model that we can train via backpropagation and standard machine learning techniques.

The modules of the PLNN provide all of the information needed to define an SDE governing the evolution of an ensemble of cells within the represented landscape, over an interval of time $t \in [0, T]$. A general SDE contains a drift and a diffusion term, denoted $\boldsymbol{f}(t, x)$ and $\boldsymbol{g}(t, x)$, respectively, and can be written in the form

$$d\boldsymbol{x}(t) = \boldsymbol{f}(t, \boldsymbol{x}(t))dt + \boldsymbol{g}(t, \boldsymbol{x}(t))dW(t)$$

where

$$\boldsymbol{f} : [0, T] \times \mathbb{R}^d \to \mathbb{R}^d$$
$$\boldsymbol{g} : [0, T] \times \mathbb{R}^d \to \mathbb{R}^{d\times d_w}$$

and W is a d_w-dimensional Wiener process [10].

In this case, the drift term $\boldsymbol{f}$ is the deterministic flow induced by the potential, tilted in response to a particular signal. For a signal profile $\boldsymbol{s}(t)$, this flow is given by the negative gradient of (9):

$$\boldsymbol{f}(t, \boldsymbol{x}(t)) = -\nabla_{\boldsymbol{x}} \boldsymbol{\Phi}(\boldsymbol{x}(t)) - \boldsymbol{\Psi}(\boldsymbol{s}(t)). \tag{12}$$

Under the assumption of isotropic additive noise, $d_w = d$ and the diffusion term is given by the output of the noise module $\boldsymbol{N}$:

$$\boldsymbol{g}(t, \boldsymbol{x}(t)) = \boldsymbol{N}(t, \boldsymbol{x}(t)) = \sigma \boldsymbol{I}_d.$$

With these equations, given an initial cell state $\boldsymbol{x}_0 = \boldsymbol{x}(0)$, we can sample a future state $\boldsymbol{x}(t)$ by solving the SDE forward in time, using the Euler-Maruyama method or a higher-order method such as Heun's method [11]. However, we are interested in the evolution of not just a single cell, but an ensemble of cells. Therefore, we must simulate an array of SDEs

$$d\boldsymbol{x}_c(t) = \boldsymbol{f}(t, \boldsymbol{x}_c(t))dt + \boldsymbol{g}(t, \boldsymbol{x}_c(t))dW_c(t) \tag{13}$$

where $c = 1, \ldots, N_{\text{cells}}$ indexes across the total number of cells. Importantly, each cell is subject to an independent d-dimensional Brownian motion W_c. We utilize the numerical differential equation package `Diffrax` [12], within the JAX computing ecosystem [13], to solve the SDE vectorized across an ensemble of cells.

A landscape model inferred from synthetic data

We begin by showing that we can infer a nonlinear potential function and the way in which it is tilted in response to two signals. We do this using *in silico* experiments where the potential we wish to infer, denoted ϕ^*, is known *a priori*, and used to generate synthetic data on which our model will be trained. To this end, we leverage the binary choice and binary flip landscapes described above, in Eqs. 2 and 3, and define two ground-truth parameterized landscape dynamical systems

$$\phi_1^*(x, y; \boldsymbol{s}) = x^4 + y^4 + y^3 - 4x^2y + y^2 + s_1 x + s_2 y \tag{14}$$

and

$$\phi_2^*(x, y; \boldsymbol{s}) = x^4 + y^4 + x^3 - 2xy^2 - x^2 + s_1 x + s_2 y. \tag{15}$$

The asterisks here are meant to emphasize that these equations define the ground-truth data generating process, and encapsulate both the structure of the underlying, static landscape $\tilde{\phi}^*$, as well as the mapping Ψ^* of signals to tilt parameters, which we take to be the identity map. We can break down the definitions of ϕ_1^* and ϕ_2^* in order to highlight the separation between the static landscape and the effect of tilts:

$$\begin{aligned} \phi_1^*(\boldsymbol{x}; \boldsymbol{s}(t)) &= \tilde{\phi}_1^*(\boldsymbol{x}) + \boldsymbol{x}^T \Psi_1^*(\boldsymbol{s}(t)) \\ \tilde{\phi}_1^*(\boldsymbol{x}) &= x^4 + y^4 + y^3 - 4x^2y + y^2 \\ \Psi_1^*(\boldsymbol{s}) &= \mathbf{A}_1^* \boldsymbol{s}, \quad \mathbf{A}_1^* = \boldsymbol{I}_2 \end{aligned} \tag{16}$$

and

$$\begin{aligned} \phi_2^*(\boldsymbol{x}; \boldsymbol{s}(t)) &= \tilde{\phi}_2^*(\boldsymbol{x}) + \boldsymbol{x}^T \Psi_2^*(\boldsymbol{s}(t)) \\ \tilde{\phi}_2^*(\boldsymbol{x}) &= x^4 + y^4 + x^3 - 2xy^2 - x^2 \\ \Psi_2^*(\boldsymbol{s}) &= \mathbf{A}_2^* \boldsymbol{s} \quad \mathbf{A}_2^* = \boldsymbol{I}_2. \end{aligned} \tag{17}$$

Written in this form, our objective can be stated clearly. We aim to approximate the function $\tilde{\phi}^*$ with a neural network, and infer the matrix $\mathbf{A}^*$, in both cases the identity matrix $\boldsymbol{I}_2$.

We synthesize a training dataset from the ground-truth data generating process, illustrated in Figure 2b. Taking $\phi^* = \phi_1^*$ or $\phi^* = \phi_2^*$, we perform a series of *in silico* experiments in which cells are observed at discrete points in time as they flow through the changing landscape (see Methods). Each individual experiment involves simulating an ensemble of two-dimensional cells in the landscape, under the influence of a prescribed signal profile drawn from a prior distribution of feasible signal functions. The state of the ensemble is recorded at a number of sampling timepoints, from which we compile a set of consecutive timepoint-state pairs of the form

$$(t_0, X_0;\ t_1, X_1;\ \boldsymbol{s}(t)) \tag{18}$$

where X_0 and X_1 are the states of the ensemble at times t_0 and t_1, respectively. In addition, we record the parameterization of the signal profile, $\boldsymbol{s}(t)$, so that we may compute the value of the signal at any time t. The data generated across all of the performed experiments are pooled, resulting in a synthetic training dataset $\mathcal{D}$.

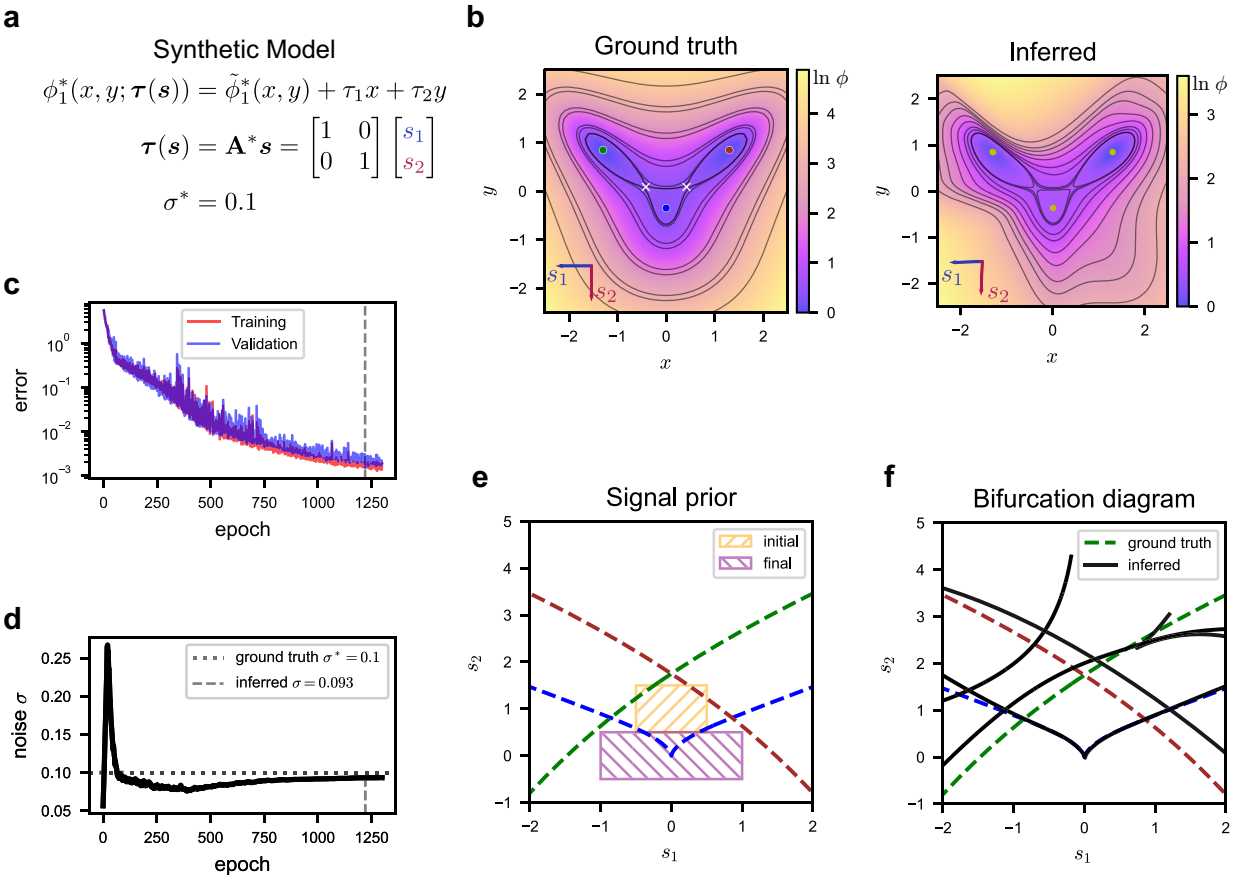

Figure 3: **The binary choice landscape provides a synthetic testing ground for inferring a parameterized landscape. (a)** The algebraic form of the binary choice system, the ground-truth model generating the data used for training. This system tilts as the signals s_1 and s_2 vary. In the ground-truth system, the signals map identically to the tilt parameters, and we specify a low noise level $\sigma^* = 0.1$. **(b)** The ground-truth and inferred landscapes. The effects of the signals s_1 and s_2 are depicted as arrows in the lower left of each plot, pointing in the direction in which flow is biased when a unit signal vector is prescribed. The landscapes agree qualitatively, with larger disparities occurring in regions containing few cells from the training data, as evidenced by the differences in the contours shown. **(c)** The value of the training and validation loss, or error, over the course of training. The dashed line indicates the epoch corresponding to the optimal model, exhibiting the minimal validation loss. Early halting occurred due to the validation loss not improving for a number of consecutive epochs. **(d)** The evolution of the inferred noise parameter σ during the training process. The true value, σ^*, is denoted with a horizontal dotted line. The solid vertical line corresponds to the optimal model found during training, with an inferred $\sigma \approx 0.093$. An initial phase of model training witnesses a significant increase in σ, before decreasing towards the true value. **(e)** The bifurcation diagram of the true underlying parameterized landscape ϕ_1^*, with the color of each fold curve corresponding to the color of the fixed point in panel (b) that vanishes when the signal passes through the curve. The hatched rectangles illustrate the range of sigmoidal signal functions used to generate the synthetic training data. They indicate the regions of parameter space from which the initial and final signal values are drawn, so that $\boldsymbol{s}(t)$ passes from the central region into the lower region, crossing the fold curve corresponding to the bifurcation of the central attractor. **(f)** A comparison of the bifurcation diagrams corresponding to the ground-truth and inferred landscapes. Dashed lines correspond to the true model. Solid lines correspond to the inferred model. There is good agreement between the true and inferred fold curve with a cusp (depicted in blue) since this is the bifurcation captured by the training data. In addition, two inferred fold curves appear to approximate those corresponding to the peripheral attractors. Additional fold curves indicate the presence of anomalous minima in the landscape at certain signal values.

Using the synthetic data generated in the ground-truth landscape, we then train a PLNN using the procedure described in Methods and summarized in Figure 2c,d. The training procedure involves simulating an ensemble of cells, from an initial condition at time t_i, to a later time t_{i+1}. The model parameters are updated based on the value of a specified loss function, that compares the *simulated* final condition to the *observed*, or ground-truth final condition, in a distributional sense. To this end, we utilize a loss function that computes from empirical samples an unbiased estimate of the maximum mean discrepancy [14] between two distributions (see Methods).

Figure 3 shows results in the case of approximating the binary choice landscape, ϕ_1^*. The inferred landscape is shown alongside the ground-truth landscape, in 3b. Qualitatively, the inferred landscape resembles the ground-truth, with three fixed points separated by two saddles. We expect that the difference between the actual and inferred landscapes will be larger in the regions of state space with a lower density of observed cells across all of the training data. In those regions that are not in the support of the training data, we expect a poorer fit. This conjecture is supported by the level sets of the inferred landscape, which become more distorted away from the central region of attractors. The inferred linear map of signals to tilts is also depicted within the landscape plots, as arrows showing the direction in which each basis signal vector biases the flow of cells in the landscape. Note that this direction is opposite the resulting tilt vector, as we have defined it. The inferred model captures both the magnitude and direction of the effect of both signals.

In regards to the noise parameter σ, the training process does appear to converge toward the true value $\sigma^* = 0.1$, as shown in Figure 3d. Notably, we observe a transient phase at the outset of training during which the noise level increases past the true value, before converging back towards it. We suspect that this phase is necessary to allow cells to occupy a wider region of phase space, without which there is no chance of correctly inferring the landscape topology.

The agreement between the ground-truth and inferred parameterized landscapes is further evidenced by comparing the bifurcation diagrams of the two dynamical systems. Figure 3f depicts the fold curves of the ground-truth and inferred landscapes, which constitute the locus of signal values at which the landscape exhibits a saddle-node bifurcation.

There is particularly close agreement with respect to the fold curve corresponding to the bifurcation of the central attractor, depicted as a dashed, blue line, and containing a dual cusp point. This is explained by the particular signal profiles used to generate the training data. As shown in Figure 3e, the initial signal value falls within a region of signal space almost guaranteeing the presence of the central attractor, whereas the final value is largely confined to a region where the central attractor has vanished in a bifurcation with one of the neighboring saddle points. Thus, the bifurcation of the central attractor is well-captured in the training data, whereas the bifurcations of either peripheral attractor are not. In light of this, it is perhaps surprising that there is still relatively good agreement between the fold curves denoting the peripheral attractor bifurcations (shown in red and green). That they exist at all, and agree qualitatively, is indicative of the robust and generic structure of the archetypal landscape forms.

The bifurcation diagram contains additional fold curves, indicating the presence of additional fixed points in the inferred model. These additional minima occur in regions outside of the support of the training data. This suggests again that while anomalies may occur, they tend to do so in regions of phase space that are never occupied by cells during the training process, and therefore for which there is no information to inform the landscape shape.

Landscape inference is sensitive to temporal sampling resolution

A number of factors influence our ability to infer a landscape system from data. Among these is the frequency of temporal sampling, which must be sufficient to capture the transition of cells between states. This is an important consideration, especially for systems in which the transition rate of cells is high. Here we examine the ways in which model inference suffers when the temporal sampling rate is insufficient.

Figure 4 shows the results of training models on a series of three different datasets, each one generated using the binary choice system given by (16), but with varying rates of sampling. Each dataset is constructed from a number of synthetic experiments over the interval $t \in [0, 20]$, with snapshots taken at sampling intervals of $\Delta T \in \{5, 10, 20\}$. Each dataset is composed of a different number of experiments, in order to ensure that the total number of snapshots is the same in each case. For example, the dataset corresponding to $\Delta T = 10$ is composed of twice as many experiments as the dataset using $\Delta T = 5$. Each of the three training datasets is depicted, highlighting the qualitative difference between them. Sparser sampling results in fewer cells captured in the transition between attractors. That is, the unstable manifolds of the saddles are poorly resolved.

We trained a number of models on each dataset, and assessed the performance of each on a held-out evaluation dataset, generated via the same process as the training data. Note that in each case, models were trained for up to 2000 epochs, with training halted if the validation loss did not improve for 100 consecutive epochs. We test each model by applying it to each of the datapoints (i.e. consecutive snapshots) in the evaluation dataset, and computing the

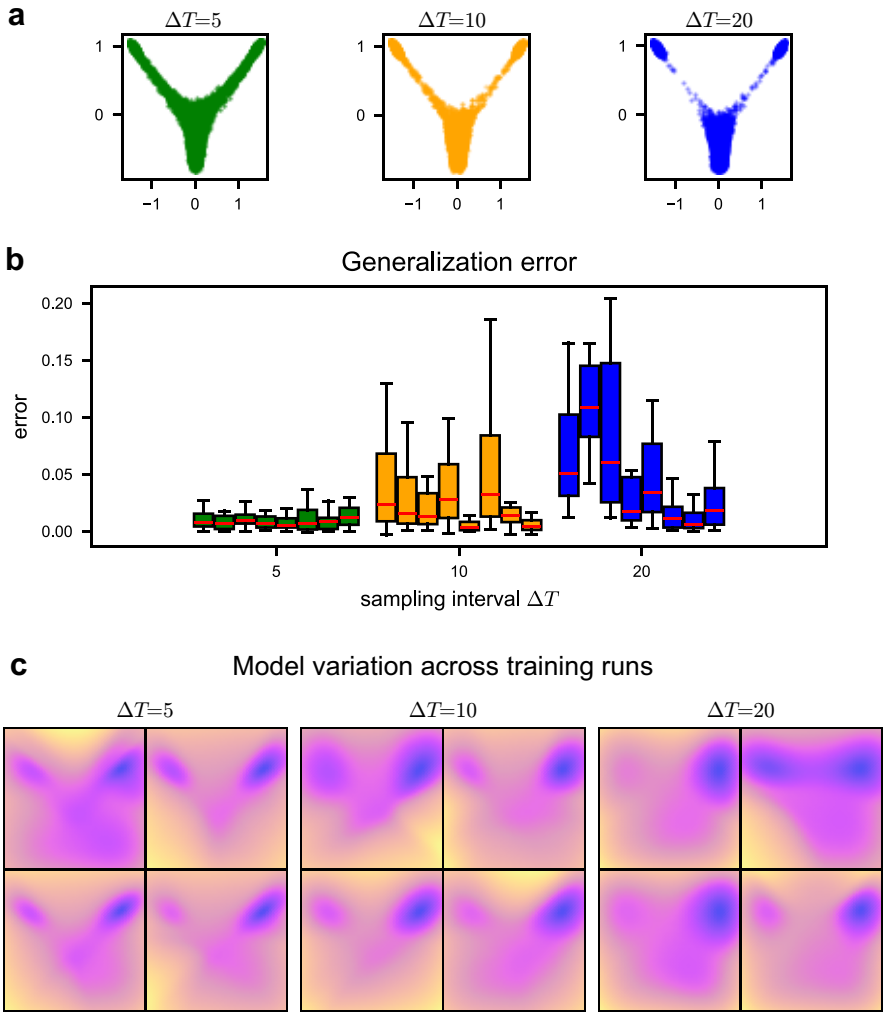

Figure 4: **Accurate model inference requires sufficiently dense sampling in time. (a)** Three datasets generated from the binary choice landscape, simulated over the interval $t \in [0, 20]$, with cells sampled at varying intervals. The plots display *all* cells making up a dataset, consisting of pooled experimental data. Each dataset consists of a different number of experiments, so that the total number of consecutive timepoint samplings is consistent. Although the same number of cells are depicted in each case, a shorter sampling interval ΔT, defined as the duration between snapshots, provides more information about the routes between attractors. **(b)** For each dataset, 8 models were trained and then evaluated against a held-out dataset. Each box and whisker plot corresponds to a single model, and depicts the distribution of the generalization error, or loss, across all evaluation datapoints. Note that outliers are excluded. Models trained on data with a higher rate of sampling perform better and more consistently. **(c)** Depictions of four of the eight inferred models, for each of the three sampling intervals ΔT. For each value of ΔT, the plots are arranged lexicographically and correspond to the first four box plots of each color in panel (b). Models inferred from data with low sampling resolution may recapitulate attractor states, corresponding to regions of the data with a high-density of observations, but may fail to correctly capture transition routes between states, corresponding to the stable and unstable manifolds of saddles.

corresponding loss value. In this way, we assess a model by examining the distribution of the loss across the evaluation dataset, providing a measure of generalization error. Models trained on more densely-sampled data ($\Delta T = 5$) appear to perform better on the evaluation dataset, with nearly all datapoints resulting in a loss below 0.05 for all such models (see Figure 4b). While the median loss of models trained on the dataset with a longer sampling interval of $\Delta T = 10$ is consistently below 0.05, there is more variability in model performance, with half of the models incurring a loss greater than 0.05 on a significant proportion of the evaluation data. Notability, three of the models trained in this case appear to perform just as well as the models trained on more frequently sampled data.

Finally, in the case of $\Delta T = 20$, the training data consists of only the initial and terminal condition for each synthetic experiment, and the resulting data is much sparser in regions between attractors. Still, a number of peregrinating cells are captured, and while models trained in this case again perform more variably than those trained on denser data, it remains the case that some models perform well, with a majority of datapoints resulting in a loss below 0.05.

Identifying a two-dimensional decision manifold in an *in vitro* dataset

Having shown that the model architecture is able to adequately capture elements of a synthetic cellular decision process, we now turn our consideration to the case of real data. Our purpose here is to demonstrate a framework with which a low-dimensional landscape model can be inferred from real data, and to illustrate the necessary considerations and current limitations of this approach. We start by outlining a general approach to working with high-dimensional data capturing a simple binary decision. We then apply this methodology to an *in vitro* dataset.

Figure 5 summarizes a general procedure for handling high-dimensional data capturing a binary decision. We assume that high-dimensional timecourse data is collected from a number of experiments, each corresponding to a particular prescribed signal, or experimental condition (the differently colored scatterplots in Figure 5a). Ideally, we want to identify a two-dimensional manifold within this high-dimensional expression space along which the essential dynamics take place, and on which we can infer a landscape potential. Our search for a *two*-dimensional landscape is based on an assumption that there are two signals controlling the system, or at least the particular decision of interest, and that these signals serve to bias the flow of cells in particular directions along the manifold. Then, these signals can be viewed as tangent vectors to the manifold. One can imagine applying any one of a number of manifold inference techniques in order to determine a suitable two-dimensional decision manifold. As a first approximation, and for the sake of simplicity, we illustrate a linear approach, using principal component analysis (PCA) to define a linear manifold onto which we project the cells. While this approach will likely fail to identify more subtle, nonlinear nuances in the dynamics, it should be sufficient to capture the essential transition from an initial state to one of two subsequent states. Moreover, this approach illustrates a method that preserves an interpretation of cells in terms of the high-dimensional state space, in this case given by the principal axes found through PCA, and does not directly rely on any assignment of cell type or other qualitative procedure to associate cells with a particular state.

Under the assumption that the data capture a binary decision involving three cell states, these states should be discernible when the data is pooled across all experimental conditions (as suggested by the contours shown in Figure 5b), and these states, as three attractors in phase space, will define a 2-dimensional plane. While cells may transition along paths that are not restricted to this plane, the projection of the cells onto the plane should capture an approximation of the dynamics. Rather than attempting to identify the attractor states directly, we instead isolate the cells at the start and end of the experiment, as we expect these cells to be primarily settled around attractors that correspond to the initial and terminal cell states, respectively. We then construct a plane through these states using PCA. We apply PCA to the initial and terminal cells, and define the linear decision manifold by the plane spanned by the first two PCs and passing through the average cell (see Figure 5c). Once we determine this decision manifold, we project all of the observed cells onto it, and take the coordinates of each cell within this manifold as the (x, y) landscape coordinates (Figure 5d-e). We then infer a landscape model defined on this manifold, as illustrated in Figure 5f.

In order to demonstrate this approach on real data, we apply our framework to the *in vitro* dataset originally detailed in [8]. This dataset captures an array of experiments, in which mouse embryonic stem cells (mESCs) were exposed to two chemical signals, WNT and FGF, for different durations over a span of five days. Starting at Day 2 (D2), the experimenters used flow cytometry to profile the expression levels of five marker proteins at intervals of 12 hours. From the resulting data, they identified five primary cell states, arranged in two consecutive binary decisions. We utilize the five-dimensional protein expression dataset and infer a landscape model for each of the two binary decisions captured therein. We begin by summarizing the two binary decisions.

After two days of exposure to FGF, mESCs take on an identity similar to that of early epiblast cells. This epiblast-like (EPI) identity serves as the initial state of cells in the system. Removing exogenous FGF results in EPI cells transitioning to an anterior neural progenitor (AN) identity. On the other hand, the activation of WNT signaling at D2 via addition of CHIRON99021 (CHIR) causes the EPI cells to instead transition to a caudal epiblast (CE) state,

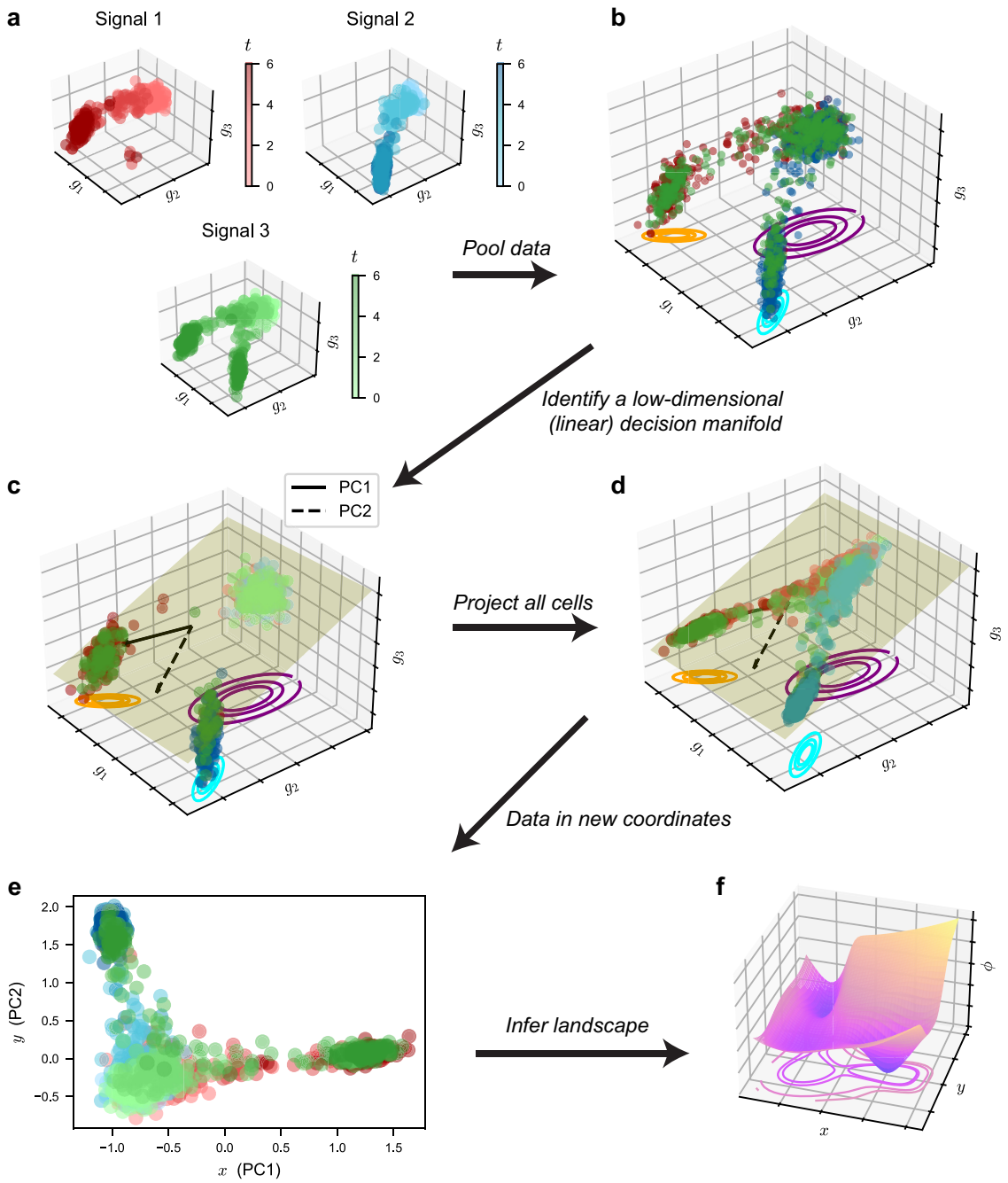

Figure 5: **Identifying a two-dimensional decision manifold that captures the relevant cellular dynamics and underlies an inferred landscape model. (a)** A collection of experiments, in which an ensemble of potentially high-dimensional cells have been sampled at a number of timepoints. Here we illustrate the profiled cells as points in a 3-dimensional gene space, sampled at times $t \in \{0, 2, 4, 6\}$. Each experimental condition is shown in a different color, and depict cells moving noisily near the surface of a sphere. **(b)** The experimental conditions, when pooled, are assumed to lie along a curved manifold within the high-dimensional gene space, with identifiable initial and terminal cell types. Here, the plotted contours capture three identifiable clusters, found via a clustering algorithm, e.g. a Gaussian Mixture Model. **(c)** The subset of cells across all experimental conditions captured at the initial and final times are used to determine a linear decision manifold, an approximation to the supposed nonlinear, true manifold. We illustrate this step using the first two principal components found via PCA, which define a plane passing through the average cell. Ideally, the cell states should be distinguishable along these axes. **(d)** Once the linear decision manifold is found, we project all cells, including those sampled at intermediate times, onto this plane, capturing the relevant transitions. **(e)** The resulting two-dimensional data, in this case given by the coordinates of each cell along the first two principal components, serves as training data with which to infer a PLNN. **(f)** An inferred landscape model overlays the decision manifold, and offers an intuitive picture of the dynamics in an interpretable space.

passing through an intermediate transitory (Tr) state. The CE state is shared between the two binary decisions, and marks the central attractor for the second. Withdrawing both the FGF and WNT signal results in CE cells transitioning to a posterior neural progenitor (PN) state, while a sustained WNT signal causes them to take on a paraxial mesoderm identity (M). Whereas in the first decision all cells either transition to the AN state or to the CE state (a binary choice), in this second decision, some cells may transition to the PN state while others transition to the M state (a binary flip), with the relative proportion governed by the duration of FGF signaling in addition to WNT.

An essential assumption in the original work, and in our treatment of the data, is a complete specification of the signal profile used in each experiment. The level of CHIR (in effect, the WNT signal) is treated as a binary function taking a value of either 0 or 1. That is, the WNT signal is either entirely absent, or fully saturated. On the other hand, the level of FGF varied between three discrete levels: off, partially active due to endogenous production, and fully active due to an exogenous signal. We use the same numerical prescription for the FGF signal function as in the original work, with the off, partial, and saturated levels given by 0, 0.9, and 1, respectively.

We restrict our attention to the first decision. The case of the second decision is detailed in the SI. In order to infer a landscape capturing the first binary decision, in which EPI cells transition to either the AN or CE state, we first recapitulate the cell classification algorithm detailed in the original work, assigning a cell type label to each cell (see S8). We then isolate those cells relevant to the initial transition, namely the EPI, CN, Tr, and CE cells.

Next, using the general procedure described above, we reduce the dimensionality of the data, moving from the five-dimensional protein expression space to a two-dimensional space on which we can infer a PLNN. We apply PCA to the subset of cells sampled at day 2 and day 3.5, taking these times to correspond to the start and end of the first binary decision, since by day 3.5 all cells have transitioned to the AN or CE states. We then use the first two principal components to define the coordinates of a two-dimensional plane onto which we project all of the observed cells. This plane serves as a linear approximation to a nonlinear decision manifold.

Two subsets of the resulting two-dimensional data, corresponding to the experimental conditions NO CHIR and CHIR 2–3, are shown in Figure 6a. The signal profile and resulting frequencies of each cell type are displayed above a series of snapshots of the data. Below the frequency plots we show the temporal evolution of the data in the dimensionally-reduced x, y space. The first column below each frequency plot displays a density plot of the data in this x, y space, capturing the evolution of the cell ensemble. The coloring included in the central columns, in accordance with each cell's nominal type, allows us to associate regions of the (x, y) space, at each timepoint, to a particular cell state. Importantly, the data displayed in this column, which is effectively used as input to the model during training, is colored independently from the landscape inference process, and the model is agnostic with respect to the cell type label. The rightmost columns show snapshots of a *predicted* flow of cells, generated from the inferred landscape model, described in the following section.

The extent to which the linear approximation to the decision manifold adequately captures the binary decision is questionable. It does appear that the initial population of EPI cells remains relatively stable under a constant FGF signal, and that when provided a CHIR signal, transitions from the left half of the plane to the right. On the other hand, when the exogenous signal halts, the distribution of EPI cells extends further into the upper-left region of the plane. However, there is not a clear separation between the EPI and AN cells when projected onto the linear decision manifold.

An inferred *in vitro* cellular decision landscape

With the dimensionally-reduced data in hand, we train a PLNN using a subset of the experimental conditions, holding out additional subsets for model validation and evaluation (see Table 1). Details of the training procedure are provided in Methods and the SI. The resulting landscape is shown in Figure 6b. We plot the landscape tilted in response to a number of relevant signals and identify local minima within the extent of the data. In the presence of FGF without a CHIR signal, $\boldsymbol{s} = (s_{\text{CHIR}}, s_{\text{FGF}}) = (0, 1)$, we identify four local minima in the inferred landscape. One of these (rightmost) is alone within a basin that coincides with a high density region occupied by CE and CE-transitioning (Tr) cells. The additional three minima are clustered together in a region that initially corresponds to EPI cells, from D2 to D3, and is then inhabited by AN cells, from D3.5 onwards.

The addition of a CHIR signal (green arrow) leads to the bifurcation of the leftmost attractors, leaving only the CE-associated attractor. The inferred effect of CHIR on the landscape is greater in magnitude than that of FGF. A reduction in the level of FGF from 1 to 0.9, representing removal of the exogenous signal, does not appear to result in a significant, qualitative change to the landscape. As it is the level of CHIR that is largely responsible for the dynamics of the first decision, this is unsurprising, and we expect to see a greater role of FGF in altering the landscape in the second case. While the effect on the landscape is minimal, it is at least qualitatively consistent with the distribution of the training data on the decision manifold. As we see from the direction of bias resulting from a unit FGF signal, a

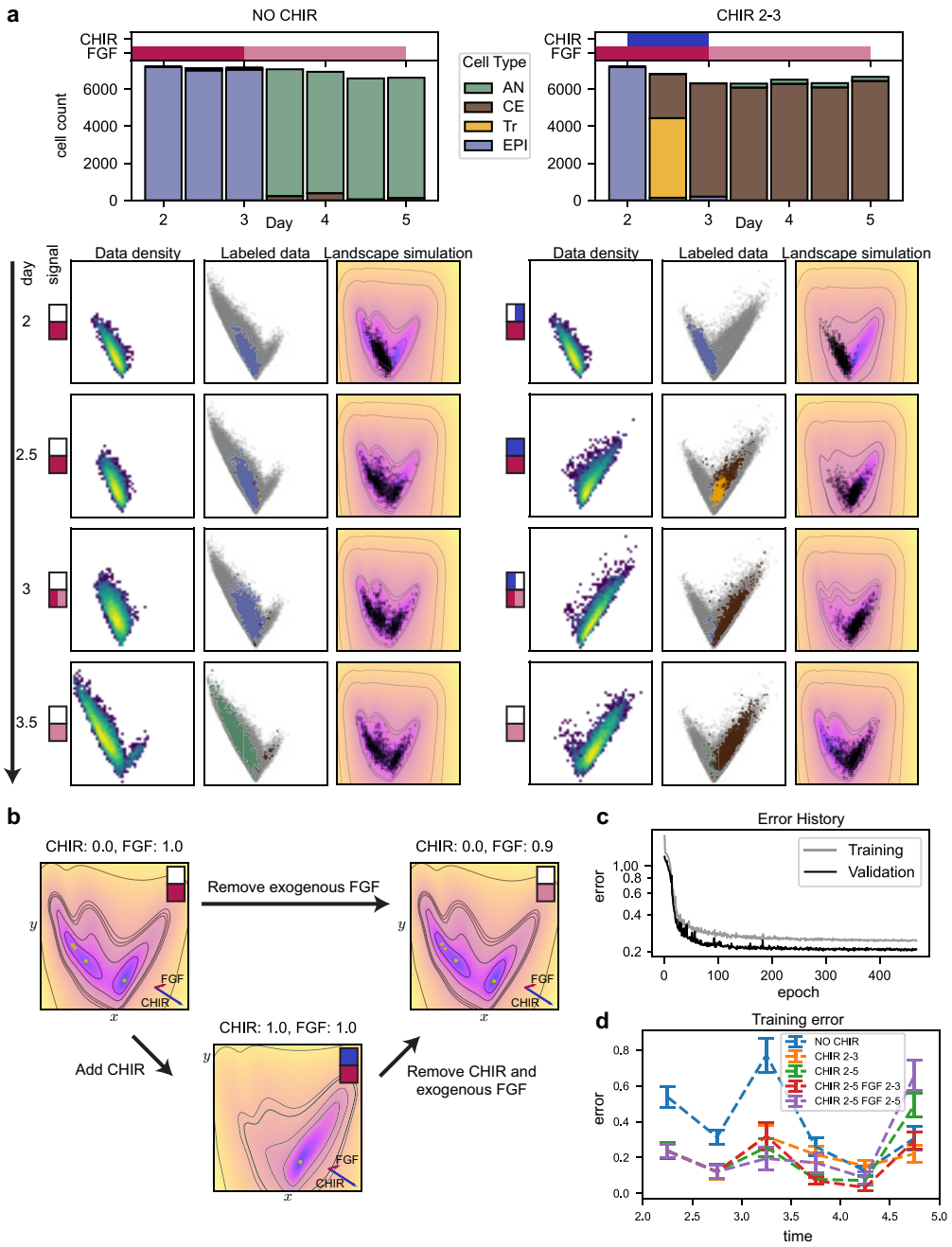

Figure 6: **A parameterized landscape model inferred from real data, consisting of a series of *in vitro* experiments. (a)** Two examples of the experimental conditions used for model training. FGF is applied until D3, at which point the exogenous FGF signal is halted. In the NO CHIR case, CHIR is never present and cells transition from the EPI state to the AN state. In the CHIR 2–3 case, CHIR is applied between D2 and D3, and cells move instead through the Tr state towards the CE state. The resulting proportions of each cell type across time are determined using the method detailed in Section S8. Below, the temporal evolution of the cell ensemble is projected onto the two-dimensional decision manifold identified within the five-dimensional expression space. The leftmost column shows a density plot of the cells at each timepoint. The center column shows a sample of cells, colored according to their nominal type. The background scatterplot, in grey, shows all cells associated with that experimental condition. The rightmost column shows a snapshot of a simulation performed using the resulting model after training. **(b)** The inferred landscape tilted in response to different signals. The effect of CHIR is evident, with its addition leading to the disappearance of a cluster of fixed points in the region occupied by nominal EPI cells. The effect of FGF is less noticeable, and does not appear to qualitatively change the landscape. Arrows in the lower right show the direction of bias resulting from addition of a unit CHIR or FGF signal. **(c)** The training and validation error over the course of model training. The validation error plateaus around the value of 0.2. The training error is higher in this case than the validation error, as the training data includes the NO CHIR condition, which is underfit. **(d)** The error associated to each datapoint in the training dataset, corresponding to a particular condition and time interval. For each datapoint, the MMD loss is averaged over multiple initial conditions and simulations, as described in the text. This average is plotted with error bars showing two standard deviations. The similarity in the error associated with different conditions at early time intervals is due to overlap in the signal profiles, with some signaling conditions deviating only at later times.

removal of FGF results in a slight bias in the negative x direction, consistent with cells transitioning from the initial EPI region to the presumable AN region (see arrows in lower right of Figure 6b panels).

The level of noise inferred by the model is substantial ($\sigma \approx 2.57$), though consistent with the degree of variation of cells around presumable attractors. This degree of noise may inhibit the association of identified minima with specific cell states. For example, the clustered minima in the EPI region may not individually have distinct biological relevance. Instead, their presence suggests a general flattening-out of the landscape in that region, which in combination with the noise level, allows for the cells to occupy a large region of phase space.

After training, we can generate sample trajectories, or simulations, of a cell ensemble within the landscape, under a prescribed signaling timecourse. We do this for both of the signals depicted in Figure 6, and display snapshots of the ensemble at the sampling times (Figure 6a, "landscape simulation" columns). These snapshots show varying degrees of qualitative agreement with the dynamics captured in the training data (left columns). In particular, the simulation of the CHIR 2–3 condition shows a movement of cells from the left branch to the right, in agreement with the training data that capture the movement of EPI cells to a CE state. In contrast, the simulation of the NO CHIR condition fails to capture the fixed nature of the EPI state over the first three sampling points, and shows a spread of cells out of the initial state and into both branches of the landscape. This artifact is likely a result of incomplete data normalization, and the fact that the NO CHIR condition is under-represented in the data, compared to signaling timecourses that include at least an initial CHIR signal. We address the issue of data normalization in more detail below (see Discussion).

During the training process we monitor the average loss computed across the validation data, consisting of four experimental conditions. This assessment takes place at the end of each epoch, and equally weights all datapoints, each corresponding to a pair of consecutive timepoints. The training and validation loss histories are depicted in Figure 6c. While the model appears to converge, the training loss does not converge towards 0, suggesting that elements of the training data are inconsistent with the model. We expect that these inconsistencies are primarily the result of our use of a linear approximation to the decision manifold, and also an overfitting of the data to a particular set of signals, namely those including CHIR.

We withhold two of the eleven experimental conditions from the training procedure as an evaluation dataset, and use these as an out-of-sample test of model performance (see Table 1). Our approach to model evaluation assesses the ability of the model to generate a feasible ensemble state at time t_{i+1} from an initial condition at time t_i, where t_i and t_{i+1} are consecutive sampling times. An alternate approach assesses the ability of the model to reconstruct the full time course of an experimental condition. The distinction between these two evaluation methods is important, as the first serves as a direct evaluation of the computational training procedure on out-of-sample data, while the latter is likely of greater value for practical purposes, for example if one is interested in predicting the complete trajectory of an ensemble under the influence of a particular signal profile.

Figure 7a shows resulting ensemble trajectories for the two evaluation conditions, CHIR 2–4 and CHIR 2–5 FGF 2–4. It is important to note that the signal profiles of these two conditions agree up to day 3, and that the underlying data for these conditions at days 2, 2.5, and 3 consist of the same cells. This accounts for the similarity in the average loss between the first two transition intervals depicted in Figure 7b. Moreover, this data is actually seen during the training process, due to overlap between the signaling conditions making up the training data. Thus, the true test of generalization on held-out data is the performance of the model at later intervals, which we see are less accurate.

Dataset	**Experimental Conditions**
Training	NO CHIR
	CHIR 2–3
	CHIR 2–5
	CHIR 2–5 FGF 2–3
	CHIR 2–5 FGF 2–5
Validation	CHIR 2–2.5
	CHIR 2–3.5
	CHIR 2–5 FGF 2–3.5
	CHIR 2–5 FGF 2–4.5
Evaluation	CHIR 2–4
	CHIR 2–5 FGF 2–4

Table 1: Breakdown of experimental conditions into a training, validation, and evaluation set.

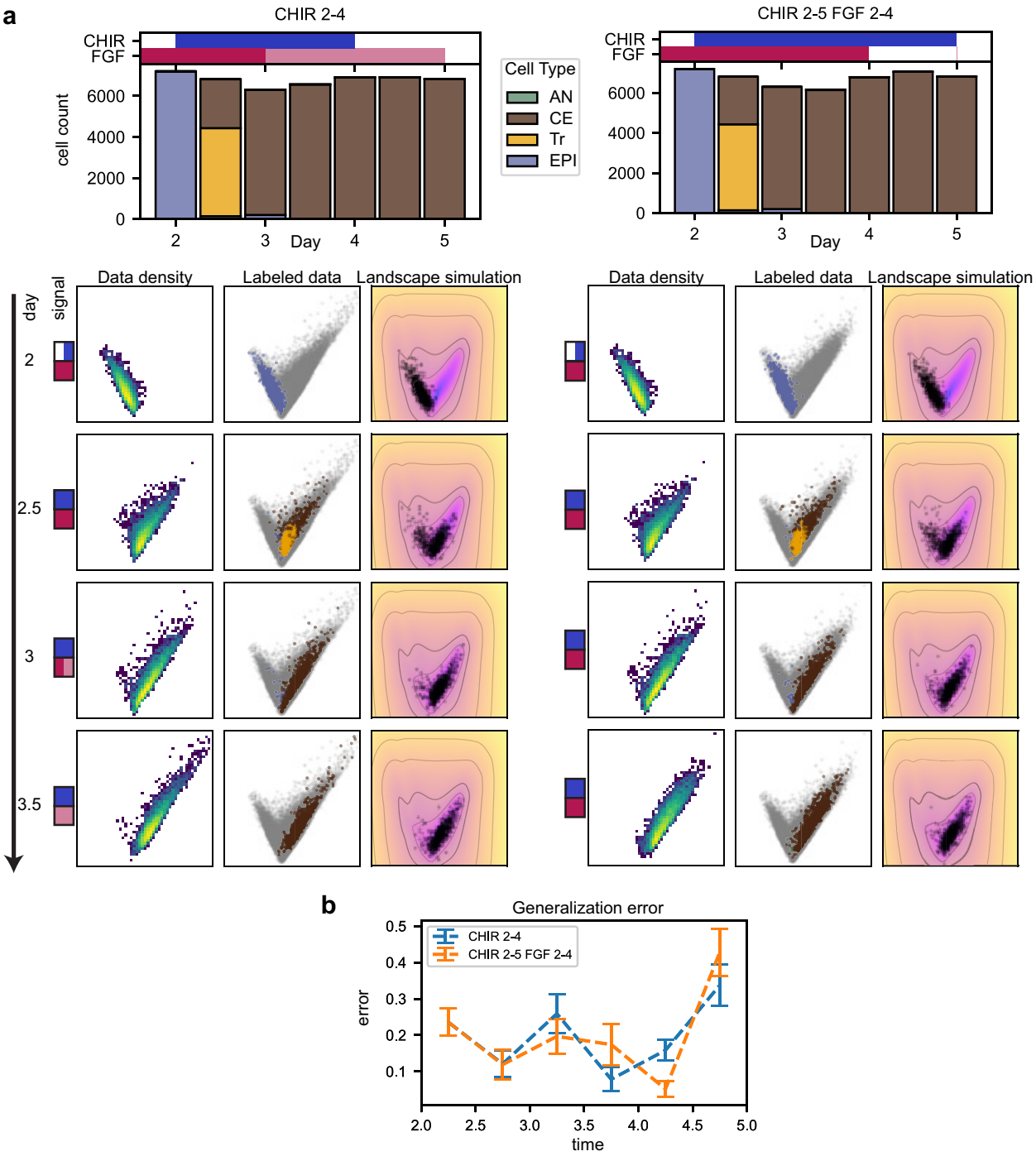

Figure 7: **Evaluation of a trained PLNN model on a held out set of experimental conditions.** **(a)** Experimental conditions held out of the training process and used for model evaluation. Plots are analogous to those depicted in Figure 6a. **(b)** The generalization error corresponding to each held out signal and sampling interval, computed over multiple initial conditions and simulations, as described in the text. This average is plotted with error bars showing two standard deviations. Importantly, the data corresponding to the initial time intervals are seen during the training process, even though the signaling conditions are nominally held out. This is again due to overlap in the signal profiles making up the training and evaluation datasets, with some signals deviating only over the final interval.

Discussion

Our framework for inferring a parameterized landscape dynamical system is an initial attempt at bridging two views of cellular differentiation dynamics, the first focused on the high-dimensional, complex nature of differentiation as it relates to the underlying gene regulatory system, and the second focused on the role of chemical signaling in directing the discrete, hierarchical progression of cells from one state to another. The first view is motivated by the modern paradigm of high-dimensional, high-throughput sequencing data, which offers a view of cellular decision making as it takes place in gene expression space. In this space, cellular dynamics, governed by complex gene-regulatory networks, are highly nonlinear and difficult to model. Decades of progress in both experimental techniques and methods of data analysis have given us the tools to identify differentiated cell types within gene expression space, with some of these tools even providing a degree of insight into dynamics. The role of signaling, however, is obscured in this setting, due to the complex interplay in real systems between cell states and signaling dynamics. The second view focuses on the low-dimensional phenomenon that arises out of the complex genetic program—the sequential transition of cells between discernible states—and utilizes the mathematical formalization of Waddington's landscape to model it. In addition, it emphasizes the role that chemical signaling plays in directing the course of cellular differentiation, with these signals biasing the flow of cells through the landscape, and uses a controlled, albeit simplified, experimental setting to investigate this phenomenon. In such experiments, prescribed signaling timecourses result in particular patterns of cellular differentiation, and the promise is that these patterns can be predicted and controlled through the prescription of a particular signal. Previous approaches have established the value of this framework, including its predictive capability, but have relied on abstract normal forms as the basis of a model, thus preventing a direct connection to the genetic profile of cells.

A bridge between these two approaches is a landscape model governed by a set of signals, that is defined within a low-dimensional manifold embedded in a higher-dimensional ambient expression space. In this low-dimensional manifold, the dynamics are produced by the gradient system defined by a potential, and are in this sense simple, though the potential itself may be complex. Additional complexity is then captured by the way in which the manifold sits in the ambient gene expression space, and the particular parameterization of the manifold provides an interpretation of the space in the context of gene expression. That is, the ultimate goal is a landscape model, offering a simple, low-dimensional description of the relevant dynamics, that maintains interpretability within gene expression space.

In the framework that we have outlined, we have focused on the inference of a two-dimensional potential function and the role of signals in altering its shape. Together, these two elements provide a dynamical system that can be used to describe cell state transitions, provided that there is a two-dimensional representation of those cells. The identification of a two-dimensional decision manifold provides this necessary ingredient, and situates the potential within the ambient gene expression space. In our synthetic experiments, the data have been two-dimensional, and no attempt is made to infer the manifold on which the governing dynamics are given by the gradient of a potential. Rather, we take for granted that the data sit in Euclidean space, and infer the parameterized landscape system directly, as a surface defined over this flat space. In our application to real data, the treatment of the underlying decision manifold is done prior to and in complete isolation from the subsequent dynamical inference. That is, we first identify a two-dimensional manifold onto which we project the observed cells, providing a picture of low-dimensional dynamics, and then use the resulting data as input to our PLNN model. Moreover, we choose to identify a *linear* manifold, that is, a flat plane in gene expression space. Both the assumption of a strictly linear manifold, and the separation between its identification and the inference of the gradient system atop it, offer directions in which we can generalize our approach.

Our current approach is based on the assumption that a linear approximation is sufficient to capture a more complex, nonlinear manifold on which cells move over the course of differentiation. Importantly, we restrict our attention to binary decisions so that the system consists of only three attractor states. In this case, a two-dimensional, flat plane serves as a linear approximation to the decision manifold containing these three points. In systems with a more complex decision structure, for example a sequence of consecutive binary decisions, a single plane will generally be insufficient to encapsulate all of the transitions, as two consecutive binary decisions may take place along entirely different dimensions of gene expression space. That is, the linear approximation of a manifold using a plane is a local one, and the core assumption is that for a cell fate transition involving three states, the local approximation can adequately capture the dynamics of interest. We expect that this assumption on its own is likely to fail in many instances, given the nonlinear nature of gene regulatory networks and their influence on the dynamics of cellular differentiation. The statement that a *single*, two-dimensional plane is sufficient to capture a sequence of cell state transitions is an even stronger one, and we do not expect this to be the case.

An illustration of the binary decision case is shown in Figure 5b, in which the essential dynamics take place on the surface of a sphere. While the true decision manifold would therefore be curved, the plane identified via the procedure described above provides an approximate description that successfully captures the transition. In this toy example, the approximation is suitable. However, it may be the case that for real data the dynamics take place within a

more complex manifold, for which a linear approximation fails to reproduce an accurate representation. In this case, a generalized approach is required, that identifies a more complicated two dimensional manifold. In the case of the *in vitro* mESC dataset, the linear approximation ultimately appeared to be insufficient. One improvement to the framework we have described is a relaxation from the use of a linear manifold to a nonlinear one. While there are a number of methods with which one can infer a nonlinear manifold from data, we argue that this inference should be done in an end-to-end fashion, simultaneously alongside the inference of the landscape itself. That is, we have performed the step of identifying a (linear) manifold prior to and in complete isolation from the subsequent step of inferring the potential function atop it. Because we explicitly assume gradient dynamics, induced by the inferred potential, the dynamics that we are able to infer are already greatly constrained. While the potential is free to change shape, it is entirely possible that the trajectories of observed cells, when projected onto the manifold, do not conform to gradient dynamics, and the only way to observe a gradient flow is to warp the underlying manifold on which the potential is defined. By simultaneously inferring both elements—an underlying manifold and the potential atop it—this warping can take place at the same time as the dynamical inference, and the model can avoid becoming "stuck" in a configuration in which no dynamics are suitable to capture the transitions in the data. We foresee the use of either more advanced methods of nonlinear manifold inference or the application of an autoencoder architecture to learn a low-dimensional phase space representation of the data, directly from the high-dimensional input.

Our framework also involves a number of critical assumptions when it comes to the role of signaling. Some of these assumptions stem from the particular context in which we expect the use of our framework to be most immediately applicable, namely experimental settings in which a signal timecourse is prescribed, and all cells are subject to the same, uniform application of that signal. In this setting, signaling, or at least a subset of the relevant signaling, is assumed to be controllable; an experimenter can specify a particular signaling timecourse and observe the ensuing cell state transitions. This situates us well outside the realm of *in vivo* development, where exact signaling dynamics are largely unknown, let alone controllable, but we have to our advantage the iterative feedback between theory and experimental design. That is, once an initial model is trained on a particular set of signaling timecourses, that model provides a way in which we can probe the additional information gained from the observation of *new* signaling timecourses, motivating additional experiments that can in turn improve the model. We envision a feedback loop, based on the machine learning paradigm of reinforcement learning, where the decision of interest is an informative signaling timecourse.

The *in vitro* experimental setting is vastly different from real development as it takes place, for example, in the nascent embryo. In that context, the signals secreted by a cell, which inform the differentiation process of other, nearby cells, depend on the particular state of the secreting cell, and the resulting feedback between cell state and signaling results in a far more complicated system from the one our model suggests. How, then, can we expect a model such as we have described to be informative in the broader context of real development? One missing piece is the role that space plays in directing development, through the spatial organization of signaling patterns. We foresee future directions that take into account the spatial organization of tissues, and allow for heterogeneous signaling across cell populations.

Our assumptions about how the signal influences the system constitute another set of model constraints. The main assumptions we make in this regard are the statements that the parameterized landscape is *tiltable*, and that the signal is processed in a linear fashion. The notion that merely tilting this landscape is sufficient to capture cell state transitions is motivated by the existence of normal forms, such as the binary flip and binary choice landscapes, which suggest that the generic bifurcations we are interested in can be expressed without higher order terms. The second statement, that signals are processed linearly, is not meant as an assertion that gene expression depends linearly on any set of chemical signals. Rather, it is a modelling assumption that allows us to pass all nonlinearities to the underlying landscape $\mathbf{\Phi}$. In [8], the authors make a similar assumption, expressing the linear coefficients in a parameterized landscape model as an affine transformation of signal levels. In addition, they introduce a nonlinearity in the way that a signal is processed by a cell by including a memory effect. This effect, in which a signal present for an extended duration continues to tilt the landscape even after it is removed, improves the predictive capability of their model. This suggests that we are in fact compromising the accuracy of our model by neglecting such nonlinearities in the function $\mathbf{\Psi}$. Practically, our reason for neglecting any memory effect is to retain a Markovian model for the purpose of convenient simulation. The fact that in our case the landscape height is directly computable from the signal at a given instant means that we can easily simulate the evolution of an ensemble of cells starting from any point in time, without considering the past signaling history. However, in cases in which a memory or other nonlinear effect is essential, it is feasible to introduce additional model parameters as part of the functional form of $\mathbf{\Psi}$ without too much additional difficulty. For example, we could easily compute a memory term for a signal at a given time, from the signal profile $\boldsymbol{s}(t)$, as we already assume that we have full knowledge of the signaling dynamics in the system. This memory term could then be integrated into the function $\mathbf{\Psi}$, with, for example, an additive effect weighted according to an additional model parameter, analogous to the memory parameter introduced in Sáez et al. [8]. The potential for additional complexity in the signal processing function begs the question as to whether it is sensible to take the extreme approach of parameterizing this module using a neural network, just as we do for the module $\mathbf{\Phi}$. While the interpretability of a simple, linear transformation of signals into

tilts is perhaps desirable, we should emphasize that as long as the signal processing function does not depend on the state $\boldsymbol{x}$, its effect on the landscape will always be linear, in accordance with (5).

A number of issues arose in the application of our method to real data. Many of these issues relate to the structural nature of the data, and the way in which it describes a temporal process. These issues offer insight into the limitations that experimental restrictions impose on our modeling framework. The high-throughput methodologies that allow us to profile gene expression in individual cells are not always well-suited to capture temporal dynamics at a high resolution. Indeed, the *in vitro* dataset that we consider offers snapshots of a cell ensemble at only 7 timepoints, separated by 12 hour intervals. While this resolution is sufficient to capture the transition of cells between discernible cell states, it may not be sufficient to capture the full range of dynamics. In addition, while the data capture a number of distinct experimental conditions, each defined by a signaling timecourse, there is a great deal of redundancy between the actual data associated with each experimental condition, as a result of the overlap between the particular signaling timecourses. Specifically, up to a certain timepoint, the same data is shared across multiple experimental conditions, because those experimental conditions correspond to signal profiles that agree up to a certain time. For example, the condition CHIR 2–3 and the condition CHIR 2–4 are identical up to day 3. Thus, the same data describes the D2–D2.5 and the D2.5–D3 transition in each experiment. As currently implemented, this means that our training procedure encounters certain consecutive sampling snapshots more often than others, and thus is prone to overfitting to those snapshots that occur multiple times across the experimental conditions. While we do not address this concern directly in our implementation, reasonable solutions include implementing a weighting scheme for the data, in cases where duplicate snapshots occur. Alternatively, one can address the issue by taking care to remove from the training data duplicate snapshots that correspond to multiple conditions.

Finally, our modeling framework directly samples an ensemble of trajectories under the influence of Langevin dynamics, specified by the system of stochastic differential equations given by (13). This approach to modeling the evolution of an ensemble is essentially agent-based, as we generate a trajectory for every cell in the ensemble. From the set of trajectories, we procure a distributional picture of the ensemble, and ultimately compare the simulated and observed ensembles in this distributional sense. The fact that we are ultimately interested in how a distribution of cells changes over time suggests an alternative approach to modeling the dynamics, which reduces to solving a Fokker–Planck equation describing the time evolution of the probability distribution of cells in phase space. The Langevin approach that we have taken is relatively straightforward to implement, given that we can easily write down the governing SDE and sample trajectories using a standard suite of differential equation solvers. However, the data we are concerned with are inherently distributional in nature, and a direct use of distributional dynamics is more directly appropriate. We foresee a future approach that frames the inference of cellular dynamics generated according to a tilting landscape as a solution to a suitable Fokker–Planck equation. By taking a more explicitly distributional approach, we also aim to capture uncertainty in the inferred landscape. At present, our model does not directly incorporate the fact that particular regions of phase space may not be well-represented in the training data, and therefore in these regions the inferred landscape does not reliably inform dynamics. To this end, a direction of future investigation includes the use of Bayesian inference methods to construct uncertainty estimates of the inferred potential function.

Methods

Generating synthetic data

We use the algebraic forms of the binary choice and binary flip landscapes, given by Eqs. (14) and (15) to generate synthetic datasets with which we can train a PLNN. In both cases, we use the Euler-Maruyama method with a step size of $\Delta t = 0.001$ to simulate $N_{\text{exps}} = 100$ experiments in which $N_{\text{cells}} = 500$ cells evolve in the landscape over an interval of time $t \in [0, 100]$, under the influence of a sigmoidal signal profile. We capture snapshots of the ensemble at intervals of $\Delta T = 10$, including the initial and final states, thereby yielding 10 training datapoints of the form $(t_0, X_0;\, t_1, X_1;\, \boldsymbol{s}(t))$ per experiment.

For each experiment, a particular sigmoidal signal profile is drawn from a prior distribution of signal functions, thereby allowing us to restrict the domain of allowable signals. The specific priors used for the binary choice and binary flip cases can be found in Tables S11 and S13, respectively. The prior for the signal profile in the binary choice case is such that the central attractor bifurcates in nearly all experiments, while the peripheral attractors typically persist. This captures the phenomenon in which cells initially located at the central attractor proceed to move towards a peripheral one after it vanishes. In the binary flip experiments, the prior yields signals capturing the heteroclinic flip bifurcation, as well as the occasional disappearance of attractors in fold bifurcations (see Figure S2).

In each experiment all cells are initialized at the same point in the landscape, near the known location of the central attractor. Then, the simulation runs for a short duration under a given signal, allowing for the randomization of the

ensemble. The state of the ensemble at the end of this burn-in phase is taken as the initial condition of the ensemble at time $t = 0$.

We specify the magnitude of isotropic, homogeneous noise in the system with $\sigma^* = 0.1$ for the binary choice simulations and $\sigma^* = 0.3$ for the binary flip simulations.

Training a PLNN

The training procedure for a PLNN model involves simulating the forward evolution of an ensemble of cells from an initial state, and comparing the resulting *simulated* final state to an *observed* final state that we consider the ground-truth. A schematic of this process is illustrated in Figure 2d. In practice, we assume that the training, or *observed*, dataset is of the form

$$\mathcal{D} = \left\{ (t_0,\, X_0;\, t_1,\, X_1;\, \boldsymbol{s}(t))_j \right\}_{j=1}^{N_{\text{obs}}} \tag{19}$$

That is, each datapoint consists of an initial ensemble state X_0 observed at time t_0, a corresponding final state X_1 at time t_1, and the signal profile $\boldsymbol{s}(t)$ used over the course of the corresponding experiment. It should be noted that the index j includes all observations made across a number of experiments. Here, X_0 and X_1 are data matrices, with each row an observed cell in d-dimensional space. The number of rows in each observed data matrix may vary, as the number of observed cells at each timepoint and in each experiment may change. Moreover, the timepoints t_0 and t_1 need not be consecutive sampling times. For example, if in an experiment observations were made at times $t \in \{1, 2, 3\}$ with corresponding observation matrices $\{X(1), X(2), X(3)\}$, then a datapoint $(1, X(1); 3, X(3); \boldsymbol{s}(t))$ is entirely valid. For the sake of computational efficiency, however, it may be advantageous for all datapoints to consist of equally spaced observations.

Given a datapoint $\left(t_0, X_0^{(\text{obs})}; t_1, X_1^{(\text{obs})}; \boldsymbol{s}(t)\right)$ from the observed dataset, we must simulate a corresponding initial condition $X_0^{(\text{sim})}$ forward in time to obtain a simulated final state $X_1^{(\text{sim})}$. In order to take advantage of the accelerated linear algebra tools provided by JAX [13], we fix the number of simulated cells, $N_{\text{cells}}^{(\text{sim})}$, as a model hyperparameter, so that the simulated data matrices have consistent dimensions. In the case that the observed data matrices *do* share the same number of cells, we may simply set $X_0^{(\text{sim})} = X_0^{(\text{obs})}$, provided that the number of cells does not exceed a memory constraint. In the case that the number of cells is too large, or in the case that the observed data matrices are of inconsistent dimension, we construct the simulated initial condition by randomly sampling rows from the observed data matrix:

$$X_0^{(\text{sim})} \sim X_0^{(\text{obs})}.$$

Using the PLNN architecture described above, we simulate the ensemble of cells $X_0^{(\text{sim})}$ forward in time, constituting a sample trajectory of the system of SDEs in (13). This step of the training process is illustrated in Figure 2c. A number of (stochastic) differential equation solvers are readily available for use in the python package `Diffrax` [12]. We choose to use Heun's method, a second order explicit Runge–Kutta method, which provides sufficient accuracy while maintaining computational efficiency. We must specify as a hyperparameter a timestep Δt, which will impact the numerical accuracy of the trajectory. In order to balance computational efficiency and accuracy, we vary this hyperparameter over the course of training, starting with a large timestep and periodically reducing it after a specified number of epochs. Full details of the particular schedules used in our implementation are included in the SI.

The resulting state at time $t_1 > t_0$ gives the *simulation* final state $X_1^{(\text{sim})}$. We then compare this to the observed final state $X_1^{(\text{obs})}$ and compute the value of the specified loss function $\mathcal{L}\left(X_1^{(\text{sim})}, X_1^{(\text{obs})}\right)$. As we compare the simulated and observed final states in a distributional sense, the dimensions of the matrices $X_1^{(\text{obs})}$ and $X_1^{(\text{sim})}$ need not be consistent. Once the loss is computed, we update the model parameters using gradient updates, as implemented in the optimization package `Optax` [15] within the JAX ecosystem. We use the RMSProp algorithm [16] to perform parameter updates. The values of all hyperparameters, including the learning rate used in the RMSProp algorithm, are provided in the SI.

Choice of loss function

We require a loss function that compares observed and simulated final ensemble states $X_1^{(\text{obs})}$ and $X_1^{(\text{sim})}$ in a distributional sense. To this end, we experimented with two loss functions. The first is a numerical estimate of the KL divergence [17]. The second is based on the maximum mean discrepancy (MMD) which has been used in the context of the two-sample test [14, 18]. Additional mathematical details of both methods can be found in the SI (see Section S6).

Ultimately, we settled on using an MMD-based loss function, as this provided a differentiable function through which we could effectively backpropagate during the training process. While training using the KL divergence estimate

as a loss function did appear to converge, resulting in models similar to those found using the MMD loss, the training process was slower, and we hypothesize that this was a result of the use of a nearest neighbor computation in the calculation of the estimate. The MMD calculation requires a specified kernel function, for which we use a sum of Gaussian kernels with varying bandwidth (details in the SI).

Recapitulation of cell type labeling for *in vitro* data

The cell type labels assigned to the observed cells in the *in vitro* mESC dataset, and appearing in Figures 6 and 7 are determined using the procedure originally described in [8]. We recapitulate this labeling and refer the reader to the original work, as well as the SI, for more details. An important detail, as it pertains to the following section, is the use of a Gaussian Mixture Model (GMM) in the cell type identification process. The algorithm proposed by Sáez et al. [8] associates at each time point t a GMM $\mathcal{G}_t$ with multiple components, each corresponding to a cell type. As part of our recapitulation of the cell type labeling, we construct and retain these GMMs.

Isolation and dimension reduction of *in vitro* binary decision data

We use the cell type labels assigned to each cell in the *in vitro* mESC dataset to isolate those cells relevant to the first binary decision, involving the transition of EPI cells to either the AN or CE state. For each timepoint t, we remove all cells labeled M or PN, replacing these cells with "pseudo-cells" generated from the Gaussian Mixture Model $\mathcal{G}_t$ retained from the cell type labeling process. Specifically, we use the component(s) of the GMM associated with the CE state to generate random samples from a multivariate normal distribution. This allows us to replace the PN and M cells, which have presumably transitioned out of the CE state, with cells that are distributed around a CE attractor as determined by the labeling algorithm. This replacement strategy is necessary in order to respect the relative proportion of cells that have entered the CE attractor. In reality, this state is an intermediate one, and cells may escape it. Our approximation is meant to treat this state as a terminal state, a periphery in a binary decision.

After isolating the data relevant to the initial binary decision, we perform a step of dimension reduction in order to acquire a two-dimensional representation of each cell. Ultimately, we infer a coordinate space using Principal Component Analysis (PCA) and project all cells onto this space. We pool all cells corresponding to D2 and D3.5, using these timepoints as the start and end of the first binary decision. We apply PCA to these cells, and then project *all* of the data across all timepoints onto the plane spanned by the first two principal components and passing through the average cell. The resulting data, the coordinates of each cell in this plane, is then used in the subsequent step of model training.

Code availability

Code implementing the model architecture, including scripts to generate all figure panels, is available on Github. Code recapitulating the cell type labeling process originally detailed in [8], and isolating the data relevant to the first binary decision, can be found in a secondary repository. Original data from the *in vitro* mESC experiments performed by Sáez et al. [8] was requested from and provided by the authors.

Acknowledgments

We thank Meritxell Sáez for providing data. M.M. and A.H. were supported by The National Science Foundation-Simons Center for Quantitative Biology at Northwestern University and the Simons Foundation grant 597491. M.M. is a Simons Investigator. This project has been made possible in part by grant number DAF2023-329587 from the Chan Zuckerberg Initiative DAF, an advised fund of the Silicon Valley Community Foundation. This research was supported in part by grants NSF PHY-1748958 and the Gordon and Betty Moore Foundation Grant No. 2919.02 to the Kavli Institute for Theoretical Physics (M.M.). This research was supported in part through the computational resources and staff contributions provided by the Genomics Compute Cluster which is jointly supported by the Feinberg School of Medicine, the Center for Genetic Medicine, and Feinberg's Department of Biochemistry and Molecular Genetics, the Office of the Provost, the Office for Research, and Northwestern Information Technology. The Genomics Compute Cluster is part of Quest, Northwestern University's high performance computing facility, with the purpose to advance research in genomics.

References

[1] Gioele La Manno, Ruslan Soldatov, Amit Zeisel, Emelie Braun, Hannah Hochgerner, Viktor Petukhov, Katja Lidschreiber, Maria E. Kastriti, Peter Lönnerberg, Alessandro Furlan, Jean Fan, Lars E. Borm, Zehua Liu,

David van Bruggen, Jimin Guo, Xiaoling He, Roger Barker, Erik Sundström, Gonçalo Castelo-Branco, Patrick Cramer, Igor Adameyko, Sten Linnarsson, and Peter V. Kharchenko. RNA velocity of single cells. *Nature*, 560(7719):494–498, August 2018. ISSN 0028-0836, 1476-4687. doi:10.1038/s41586-018-0414-6. URL http://www.nature.com/articles/s41586-018-0414-6.

[2] Spencer Farrell, Madhav Mani, and Sidhartha Goyal. Inferring single-cell transcriptomic dynamics with structured latent gene expression dynamics. *Cell Reports Methods*, 3(9):100581, September 2023. ISSN 2667-2375. doi:10.1016/j.crmeth.2023.100581.

[3] Conrad Hal Waddington. *The Strategy of the Genes*. Allen & Unwin, London, 1957.

[4] Elena Camacho-Aguilar, Aryeh Warmflash, and David A. Rand. Quantifying cell transitions in C. elegans with data-fitted landscape models. *PLOS Computational Biology*, 17(6):e1009034, June 2021. ISSN 1553-7358. doi:10.1371/journal.pcbi.1009034. URL https://dx.plos.org/10.1371/journal.pcbi.1009034.

[5] Francis Corson and Eric Dean Siggia. Geometry, epistasis, and developmental patterning. *Proceedings of the National Academy of Sciences*, 109(15):5568–5575, April 2012. ISSN 0027-8424, 1091-6490. doi:10.1073/pnas.1201505109. URL https://pnas.org/doi/full/10.1073/pnas.1201505109.

[6] Francis Corson and Eric D Siggia. Gene-free methodology for cell fate dynamics during development. *eLife*, 6: e30743, December 2017. ISSN 2050-084X. doi:10.7554/eLife.30743. URL https://elifesciences.org/articles/30743.

[7] Sui Huang. The molecular and mathematical basis of Waddington's epigenetic landscape: A framework for post-Darwinian biology? *BioEssays*, 34(2):149–157, 2012. ISSN 1521-1878. doi:10.1002/bies.201100031. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/bies.201100031. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/bies.201100031.

[8] Meritxell Sáez, Robert Blassberg, Elena Camacho-Aguilar, Eric D. Siggia, David A. Rand, and James Briscoe. Statistically derived geometrical landscapes capture principles of decision-making dynamics during cell fate transitions. *Cell Systems*, 13(1):12–28.e3, January 2022. ISSN 24054712. doi:10.1016/j.cels.2021.08.013. URL https://linkinghub.elsevier.com/retrieve/pii/S2405471221003367.

[9] David A. Rand, Archishman Raju, Meritxell Sáez, Francis Corson, and Eric D. Siggia. Geometry of gene regulatory dynamics. *Proceedings of the National Academy of Sciences*, 118(38):e2109729118, September 2021. ISSN 0027-8424, 1091-6490. doi:10.1073/pnas.2109729118. URL https://pnas.org/doi/full/10.1073/pnas.2109729118.

[10] Patrick Kidger. On Neural Differential Equations, February 2022. URL http://arxiv.org/abs/2202.02435. arXiv:2202.02435 [cs, math, stat].

[11] E. Hairer, S.P. Nørsett, and G. Wanner. *Solving Ordinary Differential Equations I Nonstiff Problems*. Springer, Berlin, second revised edition edition, 2008.

[12] Patrick Kidger. *On Neural Differential Equations*. PhD thesis, University of Oxford, 2021.

[13] James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, George Necula, Adam Paszke, Jake VanderPlas, Skye Wanderman-Milne, and Qiao Zhang. JAX: composable transformations of Python+NumPy programs, 2018. URL http://github.com/google/jax.

[14] Arthur Gretton, Karsten Borgwardt, Malte J. Rasch, Bernhard Scholkopf, and Alexander J. Smola. A Kernel Method for the Two-Sample Problem, May 2008. URL http://arxiv.org/abs/0805.2368. arXiv:0805.2368 [cs].

[15] DeepMind, Igor Babuschkin, Kate Baumli, Alison Bell, Surya Bhupatiraju, Jake Bruce, Peter Buchlovsky, David Budden, Trevor Cai, Aidan Clark, Ivo Danihelka, Antoine Dedieu, Claudio Fantacci, Jonathan Godwin, Chris Jones, Ross Hemsley, Tom Hennigan, Matteo Hessel, Shaobo Hou, Steven Kapturowski, Thomas Keck, Iurii Kemaev, Michael King, Markus Kunesch, Lena Martens, Hamza Merzic, Vladimir Mikulik, Tamara Norman, George Papamakarios, John Quan, Roman Ring, Francisco Ruiz, Alvaro Sanchez, Laurent Sartran, Rosalia Schneider, Eren Sezener, Stephen Spencer, Srivatsan Srinivasan, Miloš Stanojević, Wojciech Stokowiec, Luyu Wang, Guangyao Zhou, and Fabio Viola. The DeepMind JAX Ecosystem, 2020. URL http://github.com/google-deepmind.

[16] T Tieleman and G Hinton. Lecture 6.5-rmsprop: Divide the Gradient by a Running Average of Its Recent Magnitude. *COURSERA: Neural Networks for Machine Learning*, 4(2):26, 2012. URL https://cir.nii.ac.jp/crid/1370017282431050757.

[17] Fernando Perez-Cruz. Kullback-Leibler divergence estimation of continuous distributions. In *2008 IEEE International Symposium on Information Theory*, pages 1666–1670, Toronto, ON, Canada, July 2008. IEEE. ISBN 978-1-4244-2256-2. doi:10.1109/ISIT.2008.4595271. URL http://ieeexplore.ieee.org/document/4595271/.

[18] Gintare Karolina Dziugaite, Daniel M. Roy, and Zoubin Ghahramani. Training generative neural networks via Maximum Mean Discrepancy optimization, May 2015. URL http://arxiv.org/abs/1505.03906. arXiv:1505.03906 [cs, stat].