Warped Diffusion for Latent Differentiation Inference

Masahiro Nakano **Hiroki Sakuma** **Ryo Nishikimi**
Ryohei Shibue **Takashi Sato** **Tomoharu Iwata** **Kunio Kashino**
NTT Corporation

Abstract

This paper proposes a Bayesian nonparametric diffusion model with a black-box warping function represented by a Gaussian process to infer potential diffusion structures latent in observed data, such as differentiation mechanisms of living cells and phylogenetic evolution processes of media information. In general, the task of inferring latent differentiation structures is very difficult to handle due to two interrelated settings. One is that the conversion mechanism between hidden structure and often higher dimensional observations is unknown (and is a complex mechanism). The other is that the topology of the hidden diffuse structure itself is unknown. Therefore, in this paper, we propose a BNP-based strategy as a natural way to deal with these two challenging settings simultaneously. Specifically, as an extension of the Gaussian process latent variable model, we propose a model in which the black box transformation from latent variable space to observed data space is represented by a Gaussian process, and introduce a BNP diffusion model for the latent variable space. We show its application to the visualization of the diffusion structure of media information and to the task of inferring cell differentiation structure from single-cell gene expression levels.

1 INTRODUCTION

Importance of differentiation inference - In real-world data analysis, there are often situations in which one wishes to reveal potential generative mechanisms with hidden diffusion structures in the observation data. A symbolic example would be the biological differentiation mechanism of a group of living cells (Yuan et al., 2023; Moravec et al., 2023; Braccini, 2023; Xiang et al., 2023; Kim et al., 2020; Matsumoto et al., 2017; Zou et al., 2016; Akmaev et al., 2000). It is hoped that if further progress is made in elucidating the mechanisms of cell differentiation, the realization of artificial organs and artificial cells may become a reality in the near future. As another example, in natural language and art (including painting, sculpture, music), which have developed over a long history, it is an important social issue to reveal the hidden phylogenetic evolutionary process behind them (Kanojia et al., 2019; Shu et al., 2017; Enright and Kondrak, 2011; Milani et al., 2016; Laubach et al., 2012; Dias et al., 2013, 2012). In response to these demands, this paper proposes a method to infer such latent diffusion structures from observational data alone. The discussion that follows will culminate in cell differentiation as a motivational case study, but we would like to emphasize that our method is applicable to many other applications (as shown in Section 4).

Challenges in differentiation inference - This paper will focus on two particular difficulties:

- **Unknown topology** - The potential topology of the diffusion structure representing differentiation (e.g., how many branches occur in the diffusion) is unknown, except in special cases (e.g., with expert knowledge). Therefore, for input data of diverse quantity and quality, there is a need for a mechanism whereby the analytical method itself infers the potential topology in a data-driven manner.
- **Unknown observation mechanism** - There is an unknown black-box transformation between the latent differentiation structure and the observation data. For example, in the task of inferring cell differentiation, the typical observational information, gene expression levels, is only one aspect of the complex chemistry in the cell. Therefore, it is very difficult to accurately describe the transformation mechanism in advance, which should essentially be estimated in a data-driven manner.

Proceedings of the 27$^{\text{th}}$ International Conference on Artificial Intelligence and Statistics (AISTATS) 2024, Valencia, Spain. PMLR: Volume 238.

As a result, the differentiation inference task is very challenging because these two unknown issues have a mutually adverse effect on each other.

Our strategy - We propose a Bayesian nonparametric (BNP) differentiation inference method as a framework that can handle both of the above-mentioned issues at once. We introduce an a prior model for the former problem of unknown topology using a Dirichlet diffusion process (Neal, 2003). We also introduce an a priori model of black-box information transformation using Gaussian processes (Rasmussen and Williams, 2005) for the latter problem of unknown observational information mechanisms. By constructing a hierarchical BNP model with these two modules and inferring the model posterior probabilities, we can simultaneously estimate the hidden latent differentiation structure and the black box observational information transformation.

Our contributions - **(1) Modeling**: We extend the Gaussian process latent variable model (Lawrence, 2003) to propose a new BNP model for types of data in which the latent variables have diffusion structures. **(2) Inference**: For complex models with continuous diffuse structure in the latent variable, we propose an inference algorithm that approximates it by discrete structures using nested Chinese restaurant processes (Blei et al., 2010a; Knowles and Ghahramani, 2015). This is guaranteed to asymptote to the original continuous diffusion structure under the infinite limit of the nested structure hierarchy.

2 PRELIMINARIES

2.1 Related work

Dimensionality reduction and intrinsic dimension extraction - When our method is viewed as a kind of data dimensionality reduction, its distinguishing feature is that it assumes a diffuse structure in the latent variable space. Conventionally, various methods have been proposed that use various foresight and assumptions on the latent variable space, including the manifold learning (Ghojogh et al., 2023; McInnes et al., 2018), those that use distributions over distances between pairs Cai and Ma (2022); van der Maaten (2014); van der Maaten and Hinton (2008), and embedding local structure (Ghojogh et al., 2021; He and Niyogi, 2003). In this context, the closest framework to our method is an extension of the Gaussian process latent variable model. In Figure 1, we summarize the position of our method among related methods.

Latent differentiation inference - As an inference method for differentiation structure, our method is characterized by its ability to simultaneously estimate latent differentiation and black box transformation to observed information in a data-driven manner based on a unified evaluation criterion (model posterior probability). As exemplified by the challenge to elucidate the mechanism of cell differentiation, the technique for extracting latent diffusion behind the observation data is one of the most recent topics of interest among information processing technologies. An exhaustive and comprehensive survey paper of recent years can be found, for example, in Saelens et al. (2019) (especially, Table 1). Our efforts can be summarized in two ways:

- **Module integration** - One of the most commonly used conventional methods is to perform (1) dimensionality reduction (or clustering) and (2) differentiation estimation of data separately in a pipelined fashion (Schiebinger et al., 2019; Boukouvalas et al., 2018; Herring et al., 2018; Ahmed et al., 2018; Jin et al., 2018; Parra et al., 2019; Moon et al., 2019). Such methods have the disadvantage that cues from later stages of processing cannot be utilized in earlier stages. On the other hand, our method, in the sense of Bayesian Hierarchical Modeling, represents the entire system as a single probabilistic model, allowing each module's cues to be leveraged against each other.
- **Unified criteria** - In order to take advantage of alternating clues for each module, some have proposed an iterative approach to module processing (Diaz et al., 2016; Qiu et al., 2017). However, in such cases, it is non-trivial to know what evaluation criteria to use for iterative iteration as a whole. On the other hand, our method is able to learn by a unified criteria in the Bayesian framework.

2.2 Gaussian process latent variable model

Gaussian process (GP) (Rasmussen and Williams, 2005) - GP is a tool that has come in handy as a prior model of function space in Bayesian analysis and more general probabilistic methods. By definition, GP is a stochastic process, i.e., a collection of random variables $(\boldsymbol{y}_1, \cdots, \boldsymbol{y}_N)$ where $\boldsymbol{y}_n \in \mathbb{R}^D$ $(n \in [N] := \{1, \dots, N\})$, and any subset $(\boldsymbol{y}_{a_1}, \cdots, \boldsymbol{y}_{a_M})$ $(\{a_1, \dots, a_M\} \subseteq [N]])$ of those random variables all have a multivariate normal distribution, that is, all their finite linear combinations have a normal distribution. The *projectivity* property of the multivariate normal distribution allows us to define the joint distribution of an infinite number of random variables as the inverse limit (projective limit) of a finite-dimensional multivariate normal distribution, which can then be used as a stochastic process of infinite dimension (Orbanz, 2009, 2011).

GP latent variable model (GPLVM) (Lawrence, 2003) - One fascinating use case for GP is its appli-

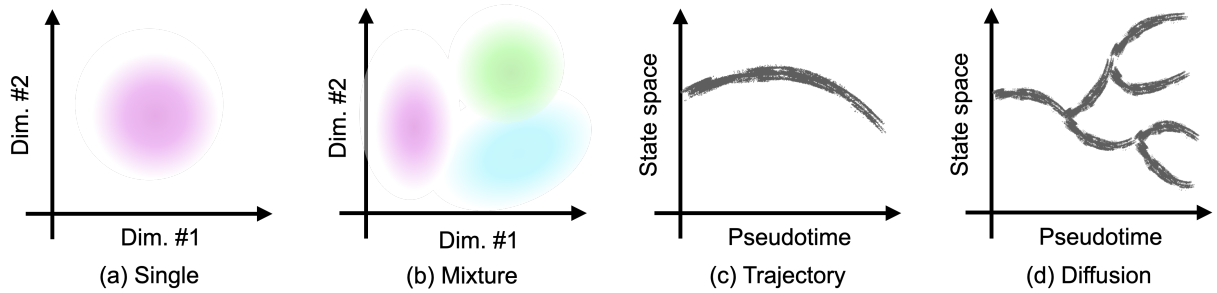

Figure 1: Conceptual comparison of our model and existing models as extensions of the Gaussian process latent variable model (GPLVM) (Lawrence, 2003). **(a) Latent single parametric distribution**: GPLVM originally uses a strategy of modeling the transformation from latent space to observed space by a Gaussian process, based on the assumption that the (often high-dimensional) observed data has an intrinsic parametric distributional structure in the (often low-dimensional) latent space. The history of GPLVM extensions has been developed by modifying modeling regarding the latent space inherent in the data. **(b) Latent mixture**: While the original GPLVM assumed a single parametric distribution in latent space (e.g., a normal distribution in the standard sense), the infinite warped mixture model (iWMM) (Iwata et al., 2013) is an extension to capture cluster structure using a mixture model (e.g., the infinite Gaussian mixture model Rasmussen (2000) using the Dirichlet process (Ferguson, 1973)) on latent space. **(c) Latent trajectory**: Gaussian process dynamical model (Wang et al., 2008) can be used to represent the hidden dynamics of a time series represented by a single path trajectory. **(d) Latent diffusion**: Our model with diffusion structures.

cation to low-dimensional visualization and intrinsic dimension extraction of observation data, which take (generally high-dimensional) observed data and infers the hidden (often low-dimensional) latent structure behind them. Suppose that we have a collection of observations $\boldsymbol{Y} = (\boldsymbol{y}_1, \cdots, \boldsymbol{y}_N)^\top$, where $\boldsymbol{y}_n \in \mathbb{R}^D$. They are associated with a set of latent variables $\boldsymbol{X} = (\boldsymbol{x}_1, \cdots, \boldsymbol{x}_N)^\top$, where $\boldsymbol{x}_n \in \mathbb{R}^Q$ (We typically expect $Q \ll D$). GPLVM assumes that observations are generated by mapping latent variables through a set of smooth functions in which a Gaussian process prior is placed. More specifically, the probability of observations given the latent variables, integrating out the mapping functions, is

$$p(\boldsymbol{Y}|\boldsymbol{X}, \theta) = (2\pi)^{\frac{-DN}{2}} |\boldsymbol{K}|^{\frac{-D}{2}} \exp\left(-\frac{1}{2}\mathrm{tr}(\boldsymbol{Y}^\top \boldsymbol{K}^{-1}\boldsymbol{Y})\right),$$

where $\boldsymbol{K}$ is the $N \times N$ covariance matrix defined by the kernel function $k(\boldsymbol{x}_n, \boldsymbol{x}_m)$ $(n, m \in [N])$, and $\theta = (\alpha, \ell, \beta)$ is the kernel hyperparameter vector related to the RBF kernel with an additive noise term:

$$k(\boldsymbol{x}_n, \boldsymbol{x}_m) = \alpha \exp\left(-\frac{1}{2\ell^2}(\boldsymbol{x}_n - \boldsymbol{x}_m)^\top(\boldsymbol{x}_n - \boldsymbol{x}_m)\right) + \delta_{nm}\beta^{-1},$$

where $\alpha > 0$ is a weight variable, $\ell > 0$ is the length scale, and $\beta > 0$ is the variance of the additive noise. For convenience, we denote the random observation data as $\boldsymbol{Y}|\boldsymbol{X}, \theta \sim \mathrm{GPwarp}(\boldsymbol{X}, \theta)$. The last remaining question in the full modeling of GPLVM is how to set up the prior model $p(\boldsymbol{X})$ of the latent variable $\boldsymbol{X}$. As shown in Figure 2, several methods have been considered to introduce a single parametric model, an infinite mixture model (Iwata et al., 2013), and a trajectory model (Wang et al., 2008), depending on the respective application. This paper examines the design of a new prior model $p(\boldsymbol{X})$ to apply the GPLVM framework to diffusion structures.

2.3 Dirichlet diffusion tree

The Dirichlet diffusion tree (DDT) is one of the standard stochastic processes used to describe diffusion on state space with time. The generative probabilistic model of DDT yields tree topologies and branch times (via a hazard process) as well as latent states at and along branches (via Gaussian diffusion) (Neal, 2003; Shiffman et al., 2018). Roughly speaking, the structure of DDT can be depicted as a sequential evolution of topology and branching time for an existing branching structure. We will therefore describe the structure of DDT in two stages: a random branching structure and a random trajectory on it.

Branching structure - As a standard approach in the context of previous DDP (Neal, 2003; Shiffman et al., 2018), we first introduce a branching rate of $a(t) = \gamma/(1-t)$, which represents the instantaneous chance $a(t)dt/s$ of divergence, where s is the number of particles that have previously traversed the given branch without diverging. This branching rate means that, if a particle is on an existing edge of a tree bounded by time $[t_a, t_b]$ and s particles have previ-

ously taken this path, the likelihood of branching by some time $t > t_a$ is defined by the Poisson process:

$$\begin{aligned}\mathbb{P}\big[\text{branch in } [t_a, t]\big] &= 1 - e^{-(A(t)-A(t_a))/s} \\ &= 1 - \left(\frac{1-t}{1-t_a}\right)^{\gamma/s}, \quad (1)\end{aligned}$$

where $A(t) := \int_0^t a(u)du = -\gamma \log(1-t)$ is the cumulative branching function.

Particle trajectory - We can use the Brownian motion in transforming the above random branching structure into random particle trajectory along virtual pseudotime ($\in \mathcal{T}$) in a state space $\mathcal{X}$ (e.g., for data visualization, typically a one-dimensional or two-dimensional Euclidean space). Specifically, a particle that has reached $\boldsymbol{X}(t)$ at time $t \in (0,1)$ will diffuse to $\boldsymbol{X}(t+dt) = \boldsymbol{X}(t) + \text{Normal}(0,\ \sigma_0^2 \boldsymbol{I} \cdot dt)$ after an infinitesimal amount of time dt, for some base variance σ_0^2, where $\boldsymbol{I}$ is the identity matrix. From the linearity of the normal distribution, integrated over a discrete time interval Δt, we can rewrite the above particle trajectory as

$$\boldsymbol{X}(t+\Delta t) \sim \text{Normal}\left(\boldsymbol{X}(t),\ \sigma_0^2 \boldsymbol{I} \cdot \Delta t\right). \quad (2)$$

Thus, latent state along the tree evolves according to collective Brownian motion, where each branch event signifies the birth of two independent Brownian motion processes (conditioned on their starting location).

In summary, DDT is a generative probabilistic model of random diffusion on the product of the state space $\mathcal{X}$ and pseudotime $\mathcal{T} := [0,1]$. For the nth particle ($n = 1, \ldots, N$), we view $\tau_n : \mathcal{T} \to \mathcal{X}$ as a function that returns the state $\tau_n(t)$ (i.e. realization of $\boldsymbol{X}(t)$ for the n-th particle) at pseudotime t ($\in [0,1]$). For convenience, we denote the collection of the latent trajectories as

$$\tau_1, \ldots, \tau_N \sim \text{DDT}(\gamma, \sigma_0). \quad (3)$$

2.4 Nested Chinese restaurant process

At the end of the preparations, we describe another representation of DDT, the *nested Chinese restaurant process* representation, which is more suited to Bayesian inference. Roughly speaking, DDT is not easy to infer due to its representation of continuous trajectories, so we are trying to come up with a discrete-time approximation representation of it.

Chinese restaurant process (CRP, Figure 2 (a)) - As is well known in the Bayesian nonparametrics literature, CRP is a stochastic process that represents the evolution on the partition of elements and is an equivalent representation of the data clustering assignment

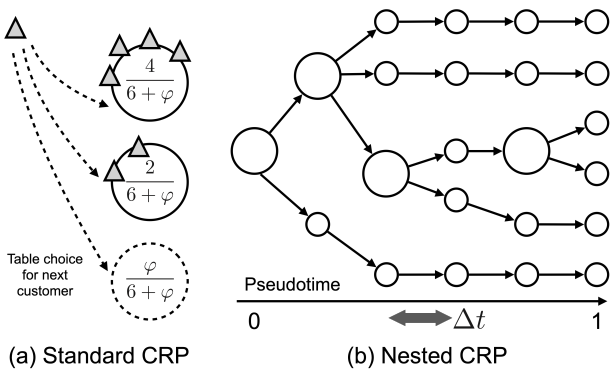

Figure 2: **(a) Standard CRP**: By considering data elements as customers (triangles) and clusters as tables (circles), CRP can generate random partitions through a probabilistic algorithm in which customers choose tables sequentially. **(b) Nested CRP**: By nesting random partitions generated from CRPs, nCRP can represent hierarchical partitions. In the figure above, the large circles represent the tables in which the partitions occur in the hierarchy. An important property to emphasize here is that the continuum limit of nCRP can simulate the DDT diffusion (See Theorem 2.2).

in the Dirichlet process (DP) mixture model (Ferguson, 1973; Rasmussen, 2000). We here consider the data elements as guests and the clusters as tables. Let $\varphi > 0$ be a concentration parameter corresponding to DP. We consider a Markov process in which customers sequentially choose tables as follows: (1) The first customer sits at the first table, and (2) The next and subsequent customers choose their tables according to

- The nth customer chooses the kth existing table with probability $\mathcal{N}_k/(n+\varphi)$, where $\mathcal{N}_k$ is the number of customers already seated at the kth table.
- The nth customer sits at the new table with probability $\varphi/(n+\varphi)$.

In this procedure, the CRP can represent the partitioning of elements by considering data elements corresponding to customers sitting at the same table as belonging to the same cluster. As is well known, the random partitioning generated by CRP can also be expressed as a scaling limit of a finite Dirichlet-Categorical hierarchical mixture model:

Remark 2.1 *For a real variable $\varphi > 0$ and $K \in \mathbb{N}$, we consider the following K-dimensional Dirichlet-Categorical hierarchical model:*

$$\begin{aligned}\pi &\sim \text{Dirichlet}(\varphi/K, \ldots, \varphi/K) \quad (4)\\ Z_n \mid \pi &\sim \text{Categorical}(\pi) \quad (n = 1, \ldots, N). \quad (5)\end{aligned}$$

Then, we regard Z_n as the table index of the n-th customer. Taking the limit $K \to \infty$ recovers CRP with the concentration parameter φ.

Nested CRP (nCRP, Figure 2 (b)) (Blei et al., 2010b) - A tree can be viewed as a sequence of nested partitions; generalizing the CRPs to such a sequence yields a distribution over the tree. Specifically, we define a nested CRP (nCRP) by imagining the following scenario for generating the sample. Suppose there are an infinite number of Chinese restaurants with infinite tables in a given city. One restaurant is identified as the root restaurant, and each of its infinite tables has a card with the name of another restaurant on it. Each of these restaurant tables has a card introducing another restaurant, and this structure is assumed to be repeated over $T \in \mathbb{N}$ levels. Note that each restaurant is associated with a level in this tree. The root restaurant is on level 1, the restaurants featured on its table card are on level 2, and so on. The most important thing to emphasize here is that a very interesting property of nCRP is that it is a discrete approximate representation of DDT:

Theorem 2.2 *(See also Theorem 2 in (Knowles and Ghahramani, 2015)). We suppose that Δt is a small nonnegative real number and that, for simplicity, $(1/\Delta t)$ is a natural number. We associate each level s in an $(1/\Delta t)$-level nCRP with pseudotime $(s-1)\Delta t$, and let the concentration parameter at level s be $a((s-1)\Delta t)/S$. Taking the limit $\Delta t \to 0$ recovers DDT with divergence function $a(t)$ (Figure 2 (b)).*

3 WARPED DIFFUSION MODEL

Our goal is to infer potential diffusion structures from (often high-dimensional) observation data. This is motivated, for example, by the intention to elucidate the mechanisms of cell differentiation (how cells transition through state space in pseudotime) from expression level data to various genes in a group of cells. Therefore, it is reasonable to interpret the observation data as corresponding to a single landmark in the latent diffusion structure, from which it is translated into gene expression levels through complex (often black-box) biological mechanisms. We will employ a Bayesian nonparametric approach to such a goal.

3.1 Generative probabilistic model

We will consider Bayesian nonparametric methods for inferring potential diffusion structures, such as cell differentiation, hidden behind high-dimensional data, such as gene expression patterns in a group of cells. Our strategy begins with the GPLVM framework described in Section 2.2. We recall that GPLVM has a model for revealing the hidden latent variable structure behind high-dimensional data; in the design of GPLVM, there is room for introducing whatever foresight knowledge the designer wishes to induce about the prior model for the latent variable structure. Therefore, we can consider the strategy of introducing an a prior model to induce diffusion structures by DDT described in Section 2.3. Based on the above broad strategy, we will construct a detailed model as follows. As Figure 3 shows, our generative probabilistic model is based on a Bayesian hierarchical model of three modules. Suppose that we have a collection of observations $\boldsymbol{Y} = (\boldsymbol{y}_1, \cdots, \boldsymbol{y}_N)^\top$, where $\boldsymbol{y}_n \in \mathbb{R}^D$.

#1 Latent diffusion - First, we introduce the direct product space of the low-dimensional state space $\mathcal{X}$ and the pseudotime $\mathcal{T}$ as a latent variable space in order to represent latent variables that diffuse along a pseudotime axis in the state space. We represent the diffusion structure, i.e., a collection of random trajectories $\tau_n : \mathcal{T} \to \mathcal{X}$ $(n = 1, \ldots, N)$ on the latent variable space based on DDT:

$$\tau_1, \ldots, \tau_N \quad \sim \quad \mathrm{DDT}(\gamma, \sigma_0), \tag{6}$$

where γ and σ_0 are hyperparameters for DDT described in Section 2.3.

#2 Latent landmarks - We consider the situation where each particle is observed at a particular time among potential trajectories on pseudotime $\mathcal{T} = [0, 1]$, similar to Shiffman et al. (2018). This is often the case, for example, in the process of cell differentiation, where gene expression levels are measured by destructive manipulation at a particular time. Each measurement procedure can only provide a snapshot of gene expression at a given instant in time in a continuously growing cell. This means that the observation data can only be observed at *landmark* discrete points in a continuously growing, differentiating cell in continuous time. We introduce a probability measure $\mu : \mathcal{T} \to \mathbb{R}_+$ as an intensity function on the pseudotime $\mathcal{T}$. For example, for simplicity, this paper employs an example like the beta distribution $\mathrm{Beta}(1, 5)$, where landmarks are more likely to appear as the pseudotime increases. Then, for each $n = 1, \ldots, N$, we have $t_n \sim \mu$. This means that the nth observed data corresponds to $(\tau_n(t_n), t_n) \in \mathcal{X} \times \mathcal{T}$ on the latent variable space, i,e., the product of the state space $\mathcal{X}$ and the pseudotime $\mathcal{T}$.

#3 Black-box transformation - Finally, we assume that each observation data is generated via a black box transformation by a Gaussian process using latent measurement points. For each $n = 1, \ldots, N$, we set $\boldsymbol{x}_n := (\tau_n(t_n), t_n)$. Then, we draw the observation data $\boldsymbol{Y}$ based on the GPLVM framework, that is,

$$\boldsymbol{Y} \mid \boldsymbol{X}, \theta \quad \sim \quad \mathrm{GPwarp}(\boldsymbol{X}, \theta). \tag{7}$$

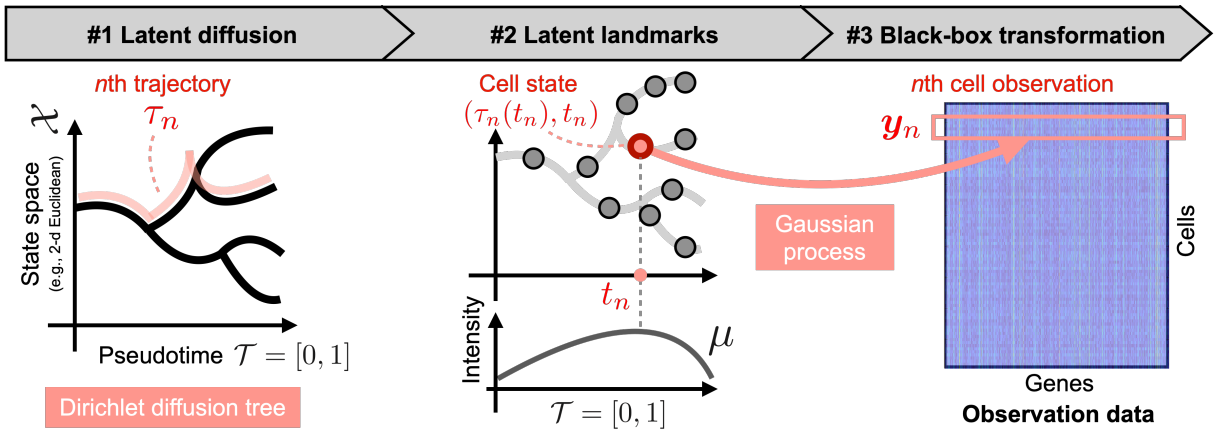

Figure 3: Overview of our generative model. **Left**: From the foresight knowledge of the bioinformatics perspective, we can first assume that certain groups of cells have potential differentiation processes behind them. We model this cell differentiation by DDT, which is a suitable model to model the development of cells differentiating (branching) according to virtual time (horizontal axis) in a latent space (in the example above, one-dimensional Euclidean space as vertical axis). **Middle**: In a population of cells following the DDT differentiation process on a virtual continuous time scale, we often measure gene expression levels at each point by taking discrete measurements (red points in the figure above). This discrete measurement can be modeled by a point process with intensity on virtual continuous time from the differentiation process of DDT. **Right**: A group of cells extracted at a discrete set of measurement points on the differentiation process of DDT is transformed by some kind of black-box manipulation (a microscopic biological phenomenon at the gene level that is difficult to model directly) into observed data, gene expression levels. We model this black-box transformation by GP

The kernel hyperparameters $\theta = (\alpha, \beta, \ell)$ are discussed in the supplemental material, as they are not an essential part of this paper and can be modeled and inferred in a similar manner to conventional methods (Lawrence, 2003; Iwata et al., 2013).

3.2 Discretization and approximation

Before discussing Bayesian inference, we emphasize that above exact model is not advisable to infer it directly. This is because it involves beyond a black-box transformation with the observed data a difficult object to handle: a continuous trajectory on the latent space. Therefore, it is reasonable to use a discrete approximation model. Specifically, we make the following two assumptions: (1) Based on Theorem 2.2, we approximate the random diffusion generated from DDT by nCRP with a sufficiently tiny real variable Δt to discretize it. (2) Based on Remark 2.1, we approximate an unbounded number of partition structures of CRPs by a sufficiently large number K of partitions.

Discretized and approximated model - We first introduce a collection of *auxiliary discrete trajectories* based on nCRP for the latent trajectories τ_n $(n = 1, \ldots, N)$. Specifically, for the nth data, we shall denote each partition represented in the nCRP hierarchy by a sequence $z_n^{(1)} z_n^{(2)} \cdots, z_n^{(1/\Delta t)}$ of partition indices $\in \{1, 2, \ldots, K\}$. For example, $z_n^{(1)} z_n^{(2)} z_n^{(3)} = \mathbf{314}$ means the **3**rd block in the first hierarchy, the **1**st block in the second hierarchy, and the **4**th block in the third hierarchy. We denote the space of this sequence of partition indices as $\mathcal{Z} := \{\phi, 1, 2, \ldots, K\}^{1/\Delta t}$, including empty element ϕ. We then discretize the pseudotime space $\mathcal{T} = [0, 1]$ as $\hat{\mathcal{T}} := \{0, \Delta t, 2\Delta t, \ldots, 1 - \Delta t, 1\}$. As a result, we can replace the latent trajectory $\tau_n : \mathcal{T} \to \mathcal{X}$ $(n = 1, \ldots, N)$ as $\hat{\tau}_n : \mathcal{Z} \to \mathcal{X}$. For each $z_n^{(s+1)} = 1, 2, \ldots, K$, we have

$$\hat{\tau}_n\big(z_n^{(1)} z_n^{(2)} \cdots z_n^{(s)} z_n^{(s+1)} \phi \cdots \phi\big) \sim \mathrm{Normal}\Big(\hat{\tau}_n\big(z_n^{(1)} z_n^{(2)} \cdots z_n^{(s)} \phi\phi \cdots \phi\big), \sigma_0^2 \boldsymbol{I} \cdot \Delta t\Big).$$

Then we also discretize the latent measurement points as $\hat{t}_n \sim \hat{\mu}$, where $\hat{\mu}$ is a discrete probability measure on $\hat{\mathcal{T}}$. Finally, we obtain the observation from $\boldsymbol{Y} \sim \mathrm{GPwarp}(\hat{\boldsymbol{X}}, \theta)$, where $\hat{\boldsymbol{X}} = (\hat{\boldsymbol{x}}_1, \cdots, \hat{\boldsymbol{x}}_N)^\top$ and $\hat{\boldsymbol{x}}_n = (\hat{\tau}_n, \hat{t}_n)$. Owing to Theorem 2.2 and Remark 2.1, the this discretized and approximated model can recover the exact model described in Subsection 3.1 when we take $\Delta t \to 0$ and $K \to \infty$.

Practical tip - It should be emphasized that, for practical purposes, when Δt is taken small enough, it is reasonable for K to be somewhat smaller as well. We recall here that K, the maximum number of tables in the CRP partition in each hierarchy, implies the maximum possible number that DDT can have binary

branches during the Δt time interval. Thus, if Δt is sufficiently small, we can assume that the branching that occurs during the Δt time interval is also sufficiently small. For simplicity of implementation, we can use the setting $\Delta t = 1/N$ (where N is again the number of observation data) and $K = 2$, for example.

3.3 Bayesian inference

Owing to the discretized approximation of the model in the previous Subsection 2.3, we are able to derive Bayesian inference that, unlike standard diffusion model inference methods (Neal, 2003; Knowles and Ghahramani, 2015; Heaukulani et al., 2014; Knowles et al., 2011; Shiffman et al., 2018), avoids continuous trajectories, which are difficult to handle. Given the observation data $\boldsymbol{Y} = (\boldsymbol{y}_1, \cdots, \boldsymbol{y}_N)^\top$, we infer the posterior distribution of (1) the nCRP hierarchical partitions $z_n^{(1)} \cdots z_n^{(1/\Delta t)}$ $(n = 1, \ldots, N)$, (2) the table locations $\hat{\tau}_n$ $(n = 1, \ldots, N)$, (3) the latent landmarks $\hat{t}_n$ $(n = 1, \ldots, N)$, and (4) the kernel hyperparameters θ. We use a Markov chain Monte Carlo method that iterates Gibbs sampling and Hamiltonian Monte Carlo. We provide here a high-level intuitive sketch along with Figure 11. We specify detailed MCMC update rules in Appendix A.[1]

(1) nCRP hierarchical partitions - For each z_n^s (i.e., the index ($\in \{1, \ldots, K\}$) of the table that the nth data chooses in the sth level of the nCRP hierarchical partitioning), we can calculate the posterior probabilities of the cases $z_n^s = 1, \ldots, K$. We can then use Gibbs sampling based on a categorical distribution proportional to their posterior probabilities.

(2) Partition locations - Each discretized posterior trajectory $\hat{\tau}_n$ (i.e., a sequence of discritized points of the trajectory) is a model represented by the product of the Brownian bridge (Brownian motion under a given parent and child) and the GP warp likelihood. We shall use the Hamiltonian Monte Carlo method, since this facilitates the computation of the gradient.

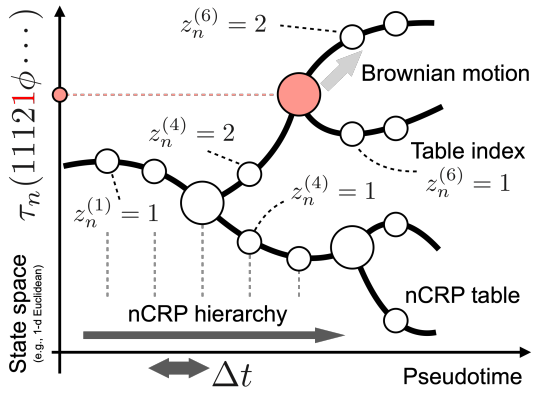

Figure 4: Discretized and approximated model.

(3) Latent landmarks - For each t_n, we can calculate the posterior probabilities of the cases $t_n = 0, \Delta t, \ldots, (1 - \Delta t), 1$. We can then use Gibbs sampling based on a categorical distribution proportional to their posterior probabilities.

(4) Kernel hyperparameters - We can straightforwardly apply the Hamilton Monte Carlo method, similar to the GPLVM variants (Iwata et al., 2013).

Practical tip - To empirically avoid local modes with worse posterior probabilities, it is recommended that each parameter should be initialized by sampling from each prior model.

4 Applications

4.1 Data visualization

As a first application, we show that our method (referred to as DDTGP) can be easily used to represent latent diffusion structures for a variety of multimedia.

Synthetic dataset - First, we demonstrate the application of our DDTGP to the type of data that has often been discussed in extensions of GPLVM. First, we demonstrate the application of our DDTGP to the type of data that has often been discussed in extensions of GPLVM. Here the observation data is on a two-dimensional Euclidean space for visualization intuition, but we imagine a situation where the data is embedded in some black-box manifold. Figures 5,6, and 7 show the results of applying our method to three synthetic data sets provided by iWMM (Iwata et al., 2013).[2] Each figure show the observation data points (marked with $\times$) and the reconstructed/predicted

[1] **Computational complexity** - In this paper, we do not promote the computational efficiency of DDTGP as an advantage. As one can immediately see from construction, our DDTGP requires additional processing for conventional GPLVM and iWMM. Roughly speaking, we can regard DDTGP as replacing the single parametric distribution in GPLVM and the infinite mixture portion in iWMM with DDT approximated by nCRP. Therefore, it is a little less efficient than GPLVM and iWMM. On the other hand, it should be emphasized that the computational bottleneck of the extensions of GPLVM, including our DDTGP, is the MCMC update with respect to GP (especially the computation of the inverse of the kernel matrix). Therefore, in situations where GPLVM and iWMM can be utilized, our DDTGP can also be used with little stress on computational efficiency. Various innovations (Matthews, 2016; Lázaro-Gredilla et al., 2012; Candela and Rasmussen, 2005; Snelson and Ghahramani, 2005) that increase the efficiency of GP inference are expected to be applicable to our DDTGP, which in itself must be an important research topic in the near future, but is not the current main focus.

[2] `http://github.com/duvenaud/warped-mixtures`

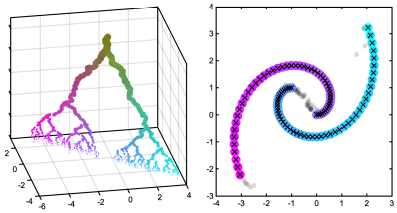

Figure 5: Spiral

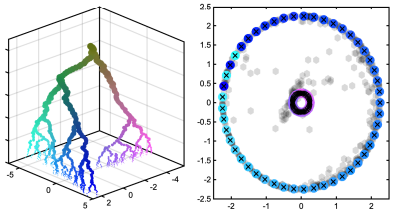

Figure 6: Circles

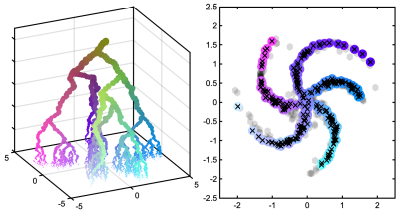

Figure 7: Pinwheel

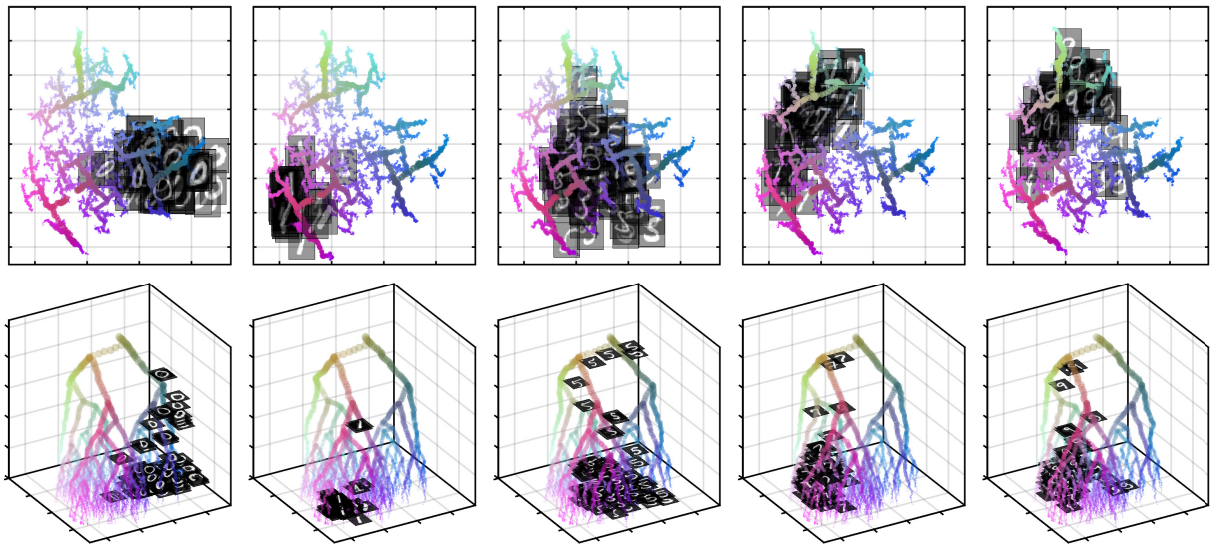

Figure 8: Latent diffusion structure of MNIST from top viewpoint (**top**) and oblique viewpoint (**bottom**). Our inference does not use any true number labels, but for visualization purposes we extract and display the numbers $0, 1, 5, 7$, and 9, which correspond to the true labels, from left to right.

point density distribution (approximated by samples circled in gray) simultaneously. We note that predictive density distribution can be typically approximated by sampling in the same way as in the usual GPLVMs (e.g., Section 4.1 of Iwata et al. (2013)). For example, Figure 5 shows two intertwining spiral structures in the observation data, whereas our DDTGP is able to show a diffusion structure divided into two large parts by DDT and an intrinsically one-dimensional (linear) state space $\mathcal{X}$ by GP warping. For each of the three examples, the left figure shows a sample of diffuse structure $\tau_1, \ldots, \tau_N$ at 2000 MCMC iterations, when the MCMC has completed burn-in and has converged sufficiently (to a posterior local mode), while the right shows the observation data.

Image dataset - Next we will demonstrate the application of our DDTGP to Modified National Institute of Standards and Technology database (MNIST)[3] and fashion MNIST.[4] As is well known, MNIST consists of 28×28 pixel images representing handwritten numbers from 0 to 9, and fashion MNIST similarly consists of 28×28 pixel images of clothing. For each we use the 10000 images originally provided for testing. Figure 12 shows the results of applying our DDTGP to MNIST. The latent diffusion structure samples from iterations 2001 to 2005 of MCMC are shown, each with latent landmarks for a single numerically labeled group of images. We can see how the group of images of the numbers of each label is actively utilizing only a portion of the overall diffuse structure. It can

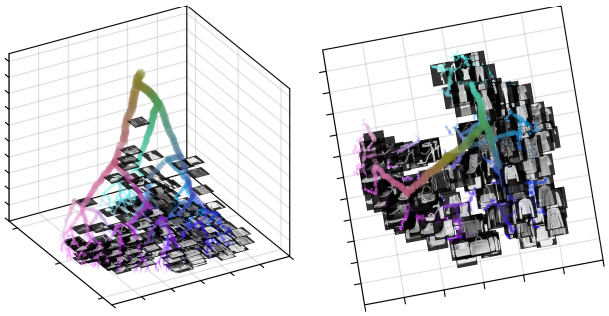

Figure 9: Latent differentiation of fashion MNIST from oblique (**left**) and top (**right**) viewpoints.

[3] `http://yann.lecun.com/exdb/mnist/`

[4] `https://github.com/zalandoresearch/fashion-mnist`

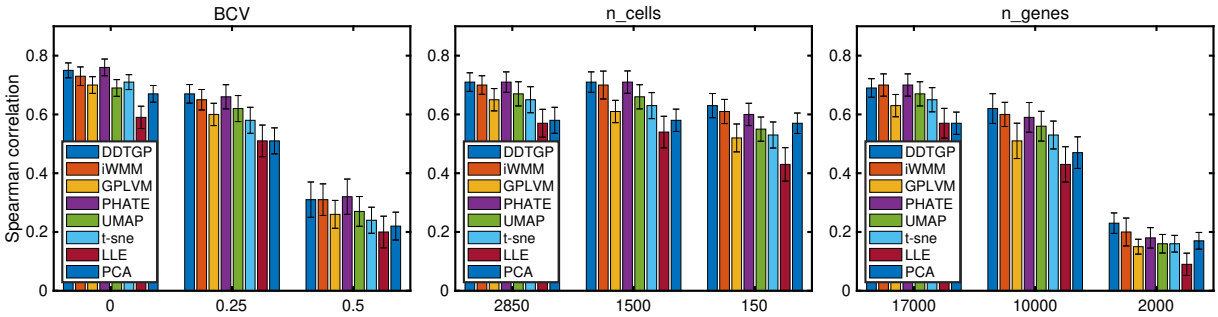

Figure 10: Pseudotime reconstruction performance (mean ± standard error) based on DEMaP metric.

also be seen that early in the pseudo-time (on the upper side of the vertical axis), some characters appear that are difficult to distinguish from other characters in appearance. Figure 9 shows the result pf applying our DDTGP to fashion MNIST. We can see how the clothing items expand into a structure similar to the subcategories interpreted by humans as they diffuse.

4.2 Pseudotime reconstruction

In order to perform a qualitative evaluation comparing the proposed method to other methods, we evaluate it using the denoised embedding manifold preservation (DEMaP) metric (Moon et al., 2019) on a pseudotime estimation task from gene expression matrices on cells × genes, which is a standard evaluation method for differentiation inference. DEMaP compares geodesic distances on noisy data with Euclidean distances of embeddings extracted from noisy data using *Spearman correlation*. This is intended to evaluate (1) that the relationships in the data hold such that cells that are close in the latent space are also close in the embedding space, and cells that are far apart in the latent space are also far apart in the embedding space, and (2) that the low-dimensional latent space accurately represents ground truth data and is as invariant as possible against biological and technical noise.

In order to have known true diffusion structures for objective evaluation, we followed the benchmark evaluation method and generated single cell Ribonucleic acid (RNA) sequence data using Splatter[5] (Zappia et al., 2017), a biosystem simulator. We simulate 100 datasets, including branches, and add biological and technical noise to the reference data to replicate the following three scenarios: (1) Using Splatter's biological coefficient of variation (BCV) parameter, BCV = $\{0, 0.25, 0.5\}$, to simulate stochastic gene expression. (2) Undersampling from the true counts, n_cells = $\{150, 1500, 2850\}$, using the default BCV to simulate inefficient capture of RNA in single cells. (3) Randomly remove genes from the data matrix, n_genes = $\{2000, 10000, 17000\}$ to demonstrate robustness to variation in all measured genes. Simulation details are provided in the Supplementary Material. We compare our proposed method, DDTGP, with (a) the closely related GPLVM (Lawrence, 2003), (b) its extension, iWMM (Iwata et al., 2013), (c) one of the standards, PHATE (Moon et al., 2019), and (d) UMAP (McInnes et al., 2018), t-sne (van der Maaten and Hinton, 2008), LLE (Polito and Perona, 2001), and PCA, which are responsible for essentially low-dimensional embedding modules in many pseudotime analysis methods. Figure 10 shows the comparison results based on the DEMaP metric (the closer to 1, the more accurate the pseudotime estimation is). We can see that our DDTGP performs as well as or better than many methods on objective metrics. In particular, we can see that the performance reduction is reduced even when n_cells and n_genes become smaller (i.e., when the number of cues from the data decreases). This may be due to the characteristics of BNP methods, in which the prior model induces the analysis results to be reasonable even in situations with few clues. DDTGP also shows better performance than other BNP models, iWMM and GPLVM. This may reflect the fact that DDTGP is a generalization of iWMM and GPLVM. Indeed, our model is attributed to iWMM when the hierarchical structure of DDT represented by nCRPs is one layer, and furthermore, our model is attributed to GPLVM when the hierarchical structure of nCRPs is replaced by standard Gaussian distribution.

5 Conclusion

This paper has proposed a new Bayesian nonparametric model with Dirichlet diffusion trees in the latent variable space as an extension of the Gaussian process latent variable model and derived a Bayesian inference algorithm with a reasonable approximation representation using the nested Chinese restaurant process. We expect that this fits very well to situations where (1) the diffusion structure is hidden in the data generation mechanism and (2) the transformation from diffusion structure to observed data is based on some black box transformation, as in the case of biological cell differentiation inference from gene expression data.

[5] `https://github.com/Oshlack/splatter`

References

Ahmed, S., Rattray, M., and Boukouvalas, A. (2018). GrandPrix: scaling up the Bayesian GPLVM for single-cell data. *Bioinformatics*, 35(1):47–54.

Akmaev, V. R., Kelley, S. T., and Stormo, G. D. (2000). Phylogenetically enhanced statistical tools for RNA structure prediction. *Bioinformatics*, 16(6):501–512.

Blei, D., Griffiths, T., and Jordan, M. (2010a). The nested Chinese restaurant process and Bayesian nonparametric inference of topic hierarchies. *Journal of the ACM*, 2(57):1–30.

Blei, D. M., Griffiths, T. L., and Jordan, M. I. (2010b). The nested Chinese restaurant process and Bayesian nonparametric inference of topic hierarchies. *Journal of the ACM*, 57(2):7:1–7:30.

Boukouvalas, A., Hensman, J., and Rattray, M. (2018). BGP: identifying gene-specific branching dynamics from single-cell data with a branching Gaussian process. *Genome Biology*, 19.

Braccini, M. (2023). differentes: An R package for computing cell differentiation trees from boolean networks. *Software Impacts*, 15:100470.

Cai, T. T. and Ma, R. (2022). Theoretical foundations of t-SNE for visualizing high-dimensional clustered data. *Journal of Machine Learning Research*, 23:301:1–301:54.

Candela, J. Q. and Rasmussen, C. E. (2005). A unifying view of sparse approximate Gaussian process regression. *Journal of Machine Learning Research*, 6:1939–1959.

Dias, Z., Goldenstein, S., and Rocha, A. (2013). Large-scale image phylogeny: Tracing image ancestral relationships. *IEEE MultiMedia*, 20(3):58–70.

Dias, Z., Rocha, A., and Goldenstein, S. (2012). Image phylogeny by minimal spanning trees. *IEEE Transactions on Information Forensics and Security*, 7(2):774–788.

Diaz, A., Liu, S. J., Sandoval, C., Pollen, A., Nowakowski, T. J., Lim, D. A., and Kriegstein, A. (2016). SCell: integrated analysis of single-cell RNA-seq data. *Bioinformatics*, 32(14):2219–2220.

Enright, J. A. and Kondrak, G. (2011). The application of chordal graphs to inferring phylogenetic trees of languages. In *Fifth International Joint Conference on Natural Language Processing*, pages 545–552.

Ferguson, T. (1973). Bayesian analysis of some nonparametric problems. *Annals of Statistics*, 2(1):209–230.

Ghojogh, B., Crowley, M., Karray, F., and Ghodsi, A. (2023). *Elements of Dimensionality Reduction and Manifold Learning*. Springer.

Ghojogh, B., Ghodsi, A., Karray, F., and Crowley, M. (2021). Laplacian-based dimensionality reduction including spectral clustering, Laplacian eigenmap, locality preserving projection, graph embedding, and diffusion map: Tutorial and survey. *CoRR*, abs/2106.02154.

He, X. and Niyogi, P. (2003). Locality preserving projections. In *Advances in Neural Information Processing Systems*, pages 153–160.

Heaukulani, C., Knowles, D. A., and Ghahramani, Z. (2014). Beta diffusion trees. In *Proceedings of the 31th International Conference on Machine Learning*, volume 32, pages 1809–1817.

Herring, C. A., Banerjee, A., McKinley, E. T., Simmons, A. J., Ping, J., Roland, J. T., Franklin, J. L., Liu, Q., Gerdes, M. J., Coffey, R. J., and Lau, K. S. (2018). Unsupervised trajectory analysis of single-cell RNA-Seq and imaging data reveals alternative tuft cell origins in the gut. *Cell Systems*, 6(1):37–51.

Iwata, T., Duvenaud, D., and Ghahramani, Z. (2013). Warped mixtures for nonparametric cluster shapes. In *Proceedings of the Twenty-Ninth Conference on Uncertainty in Artificial Intelligence*. AUAI Press.

Jin, S., MacLean, A. L., Peng, T., and Nie, Q. (2018). scEpath: energy landscape-based inference of transition probabilities and cellular trajectories from single-cell transcriptomic data. *Bioinformatics*, 34(12):2077–2086.

Kanojia, D., Kulkarni, M., Bhattacharyya, P., and Haffari, G. (2019). Cognate identification to improve phylogenetic trees for Indian languages. In *Proceedings of the ACM India Joint International Conference on Data Science and Management of Data*, pages 297–300.

Kim, K., Yang, S., Ha, S., and Lee, I. (2020). Virtualcytometry: a webserver for evaluating immune cell differentiation using single-cell RNA sequencing data. *Bioinformatics*, 36(2):546–551.

Knowles, D. A., Gael, J. V., and Ghahramani, Z. (2011). Message passing algorithms for the Dirichlet diffusion tree. In *International Conference on Machine Learning*, pages 721–728.

Knowles, D. A. and Ghahramani, Z. (2015). Pitman Yor diffusion trees for Bayesian hierarchical clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(2):271–289.

Laubach, T., von Haeseler, A., and Lercher, M. J. (2012). Treesnatcher plus: capturing phylogenetic trees from images. *BMC Bioinformatics*, 13:110.

Lawrence, N. D. (2003). Gaussian process latent variable models for visualisation of high dimensional data. In *Advances in Neural Information Processing Systems*, pages 329–336. MIT Press.

Lázaro-Gredilla, M., Vaerenbergh, S. V., and Lawrence, N. D. (2012). Overlapping mixtures of Gaussian processes for the data association problem. *Pattern Recognition*, 45(4):1386–1395.

Martin, G. R. and Evans, M. J. (1975). Differentiation of clonal lines of teratocarcinoma cells: formation of embryoid bodies in vitro. *National Academy of Sciences of the United States of America*, 72(4).

Matsumoto, H., Kiryu, H., Furusawa, C., Ko, M. S. H., Ko, S. B. H., Gouda, N., Hayashi, T., and Nikaido, I. (2017). SCODE: an efficient regulatory network inference algorithm from single-cell rna-seq during differentiation. *Bioinformatics*, 33(15):2314–2321.

Matthews, A. G. d. G. (2016). Scalable Gaussian process inference using variational methods. *Department of Engineering, University of Cambridge*.

McInnes, L., Healy, J., Saul, N., and Großberger, L. (2018). UMAP: uniform manifold approximation and projection. *Journal of Open Source Software*, 3(29):861.

Milani, S., Fontana, M., Bestagini, P., and Tubaro, S. (2016). Phylogenetic analysis of near-duplicate images using processing age metrics. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 2054–2058.

Moon, K. R., van Dijk, D., Wang, Z., Gigante, S., Burkhardt, D. B., Chen, W. S., Yim, K., van den Elzen, A., Hirn, M. J., Coifman, R. R., Ivanova, N. B., Wolf, G., and Krishnaswamy, S. (2019). Visualizing structure and transitions in high-dimensional biological data. *Nature Biotechnology*, 37:1482–1492.

Moravec, J. C., Lanfear, R., Spector, D. L., Diermeier, S. D., and Gavryushkin, A. (2023). Testing for phylogenetic signal in single-cell RNA-seq data. *Journal of Computational Biology*, 30(4):518–537.

Neal, R. (2003). Density modeling and clustering using Dirichlet diffusion trees. *Bayesian Statistics*, 7:619–629.

Orbanz, P. (2009). Construction of nonparametric Bayesian models from parametric Bayes equations. In *Advances in Neural Information Processing Systems*.

Orbanz, P. (2011). Conjugate projective limits. *arXiv:1012.0363*.

Parra, R. G., Papadopoulos, N., Ahumada-Arranz, L., Kholtei, J. E., Mottelson, N., Horokhovsky, Y., Treutlein, B., and Soeding, J. (2019). Reconstructing complex lineage trees from scRNA-seq data using MERLoT. *Nucleic Acids Research*, 47(17):8961–8974.

Polito, M. and Perona, P. (2001). Grouping and dimensionality reduction by locally linear embedding. In *Advances in Neural Information Processing Systems*, pages 1255–1262.

Qiu, X., Mao, Q., Tang, Y., Wang, L., Chawla, R., Pliner, H. A., and Trapnell, C. (2017). Reversed graph embedding resolves complex single-cell trajectories. *Nature Methods*, 14(10):979–982.

Rasmussen, C. E. (2000). The infinite Gaussian mixture model. In *Advances in Neural Information Processing Systems*.

Rasmussen, C. E. and Williams, C. K. I. (2005). *Gaussian Processes for Machine Learning*. The MIT Press.

Saelens, W., Cannoodt, R., Todorov, H., and Saeys, Y. (2019). A comparison of single-cell trajectory inference methods. *Nature Biotechnology volume*, 37:547—-554.

Schiebinger, G., Shu, J., Tabaka, M., Cleary, B., Subramanian, V., Solomon, A., Gould, J., Liu, S., Lin, S., Berube, P., Lee, L., Chen, J., Brumbaugh, J., Rigollet, P., Hochedlinger, K., Jaenisch, R., Regev, A., and Lander, E. S. (2019). Optimal-transport analysis of single-cell gene expression identifies developmental trajectories in reprogramming. *Cell*, 176(6):928–943.

Shiffman, M., Stephenson, W. T., Schiebinger, G., Huggins, J. H., Campbell, T., Regev, A., and Broderick, T. (2018). Reconstructing probabilistic trees of cellular differentiation from single-cell RNA-seq data. *CoRR*, abs/1811.11790.

Shu, K., Ortegaray, A., Berwick, R. C., and Marcolli, M. (2017). Phylogenetics of indo-european language families via an algebro-geometric analysis of their syntactic structures. *CoRR*, abs/1712.01719.

Snelson, E. L. and Ghahramani, Z. (2005). Sparse Gaussian processes using pseudo-inputs. In *Advances in Neural Information Processing Systems*, pages 1257–1264.

van der Maaten, L. (2014). Accelerating t-SNE using tree-based algorithms. *Journal of Machine Learning Research*, 15(1):3221–3245.

van der Maaten, L. and Hinton, G. (2008). Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9(86):2579–2605.

Wang, J. M., Fleet, D. J., and Hertzmann, A. (2008). Gaussian process dynamical models for human mo-

tion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(2):283–298.

Xiang, G., Giardine, B., An, L., Sun, C., Keller, C. A., Heuston, E. F., Anderson, S. M., Kirby, M., Bodine, D. M., Zhang, Y., and Hardison, R. C. (2023). Snapshot: a package for clustering and visualizing epigenetic history during cell differentiation. *BMC Bioinformatics*, 24(1):102.

Yuan, L., Lu, H., Li, F., Nielsen, J., and Kerkhoven, E. J. (2023). Hgtphylodetect: facilitating the identification and phylogenetic analysis of horizontal gene transfer. *Briefings in Bioinformatics*, 24(2).

Zappia, L., Phipson, B., and Oshlack, A. (2017). Splatter: simulation of single-cell RNA sequencing data. *Genome Biology*, 18.

Zou, Q., Wan, S., and Zeng, X. (2016). Hptree: Reconstructing phylogenetic trees for ultra-large unaligned DNA sequences via NJ model and hadoop. In *IEEE International Conference on Bioinformatics and Biomedicine*, pages 53–58. IEEE Computer Society.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes]
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Yes]
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Yes]

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [Yes]
   (b) Complete proofs of all theoretical results. [Not Applicable]
   (c) Clear explanations of any assumptions. [Yes]

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Yes]
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes]
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes]
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. [Yes]
   (b) The license information of the assets, if applicable. [Yes]
   (c) New assets either in the supplemental material or as a URL, if applicable. [Not Applicable]
   (d) Information about consent from data providers/curators. [Not Applicable]
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]

5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. [Not Applicable]
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]

A Details of Bayesian Inference Algorithm

We begin by restating our generative probabilistic model described in the text, just in case the details of the Bayesian inference algorithm need to be presented. We utilize the discretized and approximated model of the Dirichlet diffusion tree (DDT) and the Gaussian process (GP) for Bayesian inference. Figure 11 provides a visual illustration of the generative probabilistic model we use for inference.

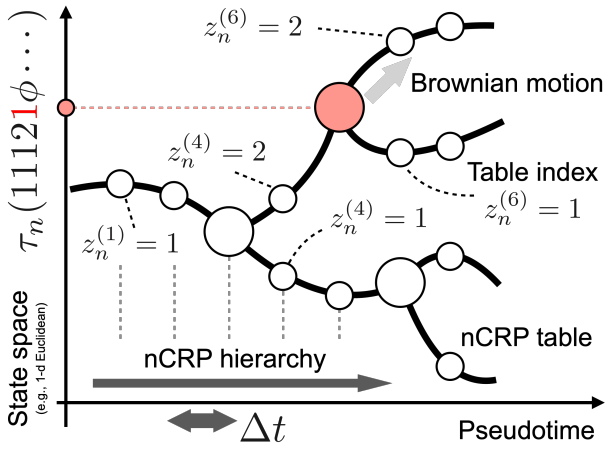

Figure 11: (Reprinted from the text.) Discretized and approximated model.

Discretized and approximated model - We introduce two approximations to the hierarchical Bayesian model of an exact Dirichlet diffusion tree (DDT)-Gaussian process (GP) to simplify the inference algorithm. Specifically, we make the following two assumptions:

- (1) Owing to Theorem 2.2 (in the main text), we approximate the random diffusion generated from DDT by nested Chinese restaurant process (nCRP) with a sufficiently tiny real variable Δt to discretize it.
- (2) Owing to Remark 2.1 (in the main text), we approximate an unbounded number of partition structures of CRPs by a sufficiently large number K of partitions.

We suppose that Δt is a small nonnegative real number and that, for simplicity, $(1/\Delta t)$ is a natural number. Let $\mathrm{nCRP}(a(t))$ be the law of nCRP with the divergence function $a(t) : [0,1] \to \mathbb{R}_+$ and the concentration parameter $a((s-1)\Delta t)/S$ at level $s \in \{1, 2, \ldots, S\}$. It should be emphasized that taking the limit $\Delta t \to 0$ recovers DDT with the divergence function $a(t)$. We first introduce a collection of *auxiliary discrete trajectories* based on the nCRP with the divergence function $a(t) = \gamma/(1-t)$ for the latent trajectories. Specifically, for the nth data, we shall denote each partition represented in the nCRP hierarchy by a sequence $z_n^{(1)} z_n^{(2)} \cdots, z_n^{(1/\Delta t)}$ of partition indices $\in \{1, 2, \ldots, K\}$. For example, $z_n^{(1)} z_n^{(2)} z_n^{(3)} = \mathbf{314}$ means the **3**rd block in the first hierarchy, the **1**st block in the second hierarchy, and the **4**th block in the third hierarchy. We denote the space of this sequence of partition indices as $\mathcal{Z} := \{\phi, 1, 2, \ldots, K\}^{1/\Delta t}$, including empty element ϕ. We also denote the nth trajectory as $\boldsymbol{z}_n := z_n^{(1)} z_n^{(2)} \cdots z_n^{(1/\Delta t)}$. For notational convenience, we shall use the following expression:

$$\boldsymbol{z}_{1:N} := \{\boldsymbol{z}_1, \ldots, \boldsymbol{z}_N\} \quad \sim \quad \mathrm{nCRP}\big(\gamma/(1-t)\big). \tag{8}$$

Specifically, we place an noninformative gamma prior $\mathrm{Gamma}(\epsilon, \epsilon)$ on γ. We then discretize the pseudotime space $\mathcal{T} = [0,1]$ as $\hat{\mathcal{T}} := \{0, \Delta t, 2\Delta t, \ldots, 1-\Delta t, 1\}$. As a result, we can replace the latent trajectory $\tau_n : \mathcal{T} \to \mathcal{X}$ $(n = 1, \ldots, N)$ as $\hat{\tau}_n : \mathcal{Z} \to \mathcal{X}$. For each $z_n^{(s+1)} = 1, 2, \ldots, K$, we have

$$\hat{\tau}_n\big(z_n^{(1)} z_n^{(2)} \cdots z_n^{(s)} z_n^{(s+1)} \phi \cdots \phi\big) \quad \sim \quad \mathrm{Normal}\Big(\hat{\tau}_n\big(z_n^{(1)} z_n^{(2)} \cdots z_n^{(s)} \phi\phi \cdots \phi\big), \sigma_0^2 \boldsymbol{I} \cdot \Delta t\Big), \tag{9}$$

where $\sigma_0 > 0$ is the base variance. Specifically, we place an noninformative gamma prior $\mathrm{Gamma}(\epsilon, \epsilon)$ on σ_0, similar to Shiffman et al. (2018). For notational convenience, we use $\hat{\tau}_{1:N} = \{\hat{\tau}_1, \ldots, \hat{\tau}_N\}$. We will simply rewrite this discritized Brownian motion (DBM) as

$$\hat{\tau}_n \quad \sim \quad \mathrm{DBM}(\sigma_0) \qquad (n = 1, 2, \ldots, N). \tag{10}$$

Then we also discretize the latent measurement points as $\hat{t}_n \sim \hat{\mu}$, where $\hat{\mu}$ is a discrete probability measure on $\hat{\mathcal{T}}$. In this paper, we use the discritization of Beta$(1, 5)$. Finally, we obtain the observation as follows:

$$\boldsymbol{Y} \quad \sim \quad \mathrm{GPwarp}(\hat{\boldsymbol{X}}(\hat{\tau}_{1:N}, \hat{t}_{1:N}), \theta), \tag{11}$$

where $\hat{\boldsymbol{X}}(\hat{\tau}_{1:N}, \hat{t}_{1:N}) = (\hat{\boldsymbol{x}}_1, \cdots, \hat{\boldsymbol{x}}_N)^\top$ and $\hat{\boldsymbol{x}}_n = (\hat{\tau}_n(z_n^{(1)} \cdots z_n^{(\hat{t}_n)} \phi \cdots \phi), \hat{t}_n)$. In summary, the whole generative probabilistic model can be expressed as follows:

$$\begin{aligned}
\gamma &\sim \mathrm{Gamma}(\epsilon, \epsilon) && && : \textit{nCRP concentration parameter} && (12)\\
\boldsymbol{z}_{1:N} &\sim \mathrm{nCRP}\big(\gamma/(1-t)\big) && && : \textit{nCRP hierarchical partition} && (13)\\
\sigma_0 &\sim \mathrm{Gamma}(\epsilon, \epsilon) && && : \textit{DBM base variance} && (14)\\
\hat{\tau}_n &\sim \mathrm{DBM}(\sigma_0) && (n = 1, 2, \ldots, N) && : \textit{Latent trajectory of } n\textit{th observation} && (15)\\
\boldsymbol{Y} &\sim \mathrm{GPwarp}(\hat{\boldsymbol{X}}(\hat{\tau}_{1:N}, \hat{t}_{1:N}), \theta) && && : \textit{Observation data} && (16)
\end{aligned}$$

As a result, we can obtain the following posterior probability density of the model parameters:

$$\begin{aligned}
&p(\hat{\tau}_{1:N}, \hat{t}_{1:N}, \boldsymbol{z}_{1:N}, \gamma, \sigma_0, \theta \mid \boldsymbol{Y}) \propto p(\hat{\tau}_{1:N}, \hat{t}_{1:N}, \boldsymbol{z}_{1:N}, \gamma, \varphi, \theta, \boldsymbol{Y})\\
&\quad = p_{\mathrm{nCRP}}(\boldsymbol{z}_{1:N}; \gamma) \cdot \prod_{n=1}^{N} p_{\mathrm{DBM}}(\hat{\tau}_n; \sigma_0) \cdot \prod_{n=1}^{N} p_{\mathrm{Cat}}(\hat{t}_n; \hat{\mu}) \cdot p_{\mathrm{GP}}(\boldsymbol{Y}; \boldsymbol{z}_{1:N}, \hat{\tau}_{1:N}, \hat{t}_{1:N}, \theta)\\
&\qquad \cdot p_{\mathrm{Gamma}}(\gamma; \epsilon, \epsilon) \cdot p_{\mathrm{Gamma}}(\sigma_0; \epsilon, \epsilon),
\end{aligned} \tag{17}$$

where $p_{\mathrm{nCRP}}(\cdot; \gamma)$ is the probability of nCRP with the divergence function $\gamma/(1-t)$, $p_{\mathrm{DBM}}(\cdot; \sigma_0)$ is the probability of DBM with the base variance σ_0, $p_{\mathrm{Cat}}(\cdot; \hat{\mu})$ is the probability of the categorical distribution with the weights $\hat{\mu}$, p_{GP} is the probability density of the Gaussian process, $p_{\mathrm{Gamma}}(\cdot; \epsilon, \epsilon)$ is the probability density of the gamma distribution with the shape parameter ϵ and the rate parameter ϵ.

Bayesian inference - Given the observation data $\boldsymbol{Y} = (\boldsymbol{y}_1, \cdots, \boldsymbol{y}_N)^\top$, we infer the posterior distribution of (1) the nCRP hierarchical partitions $\boldsymbol{z}_n = z_n^{(1)} \cdots z_n^{(1/\Delta t)}$ $(n = 1, \ldots, N)$, (2) the table locations $\hat{\tau}_n$ $(n = 1, \ldots, N)$, (3) the latent landmarks $\hat{t}_n$ $(n = 1, \ldots, N)$, and (4) the kernel hyperparameters θ and the noninformative gamma variables γ (i.e., the nCRP concentration parameter) and σ_0 (i.e., the base variance for DBM). We use a Markov chain Monte Carlo method that iterates Gibbs sampling and Hamiltonian Monte Carlo.

(1) nCRP hierarchical partitions - For each z_n^s $(n = 1, \ldots, N,\ s = 1, \ldots, 1/\Delta t)$, i.e., the index $(\in \{1, \ldots, K\})$ of the table that the nth data chooses in the sth level of the nCRP hierarchical partitions, we now consider an update law for the target variable z_n^s under which all parameters except the target variable are conditioned. We can condition on the pseudotime $\hat{t}_n$ at the latent landmark and update the nCRP partition index at the level $s \in \{1, \ldots, 1/\Delta t\}$ from $s = 1$ to $\hat{t}_n/\Delta t$ and from $s = (\hat{t}_n/\Delta t) + 1$ to $1/\Delta t$ as follows:

- $s = 1$ to $\hat{t}_n/\Delta t$ - For all $a_1 a_2 \ldots a_{\hat{t}_n/\Delta t}$ where $a_\cdot \in \{1, \ldots, K\}$, we calculate the corresponding posterior probability proportional to Equation 17, and set it as $q_{a_1 a_2 \ldots a_{\hat{t}_n/\Delta t}}$. Then, we can apply the Gibbs sampling method to sample $z_n^{(1)} z_n^{(2)} \ldots z_n^{(\hat{t}_n/\Delta t)}$ from the categorical distribution:

$$z_n^{(1)} z_n^{(2)} \ldots z_n^{(\hat{t}_n/\Delta t)} \quad \sim \quad \mathrm{Categorical}\left(\left\{q_{a_1 a_2 \ldots a_{\hat{t}_n/\Delta t}} | a_i \in \{1, \ldots, K\}\ (i = 1, \ldots, N)\right\}\right). \tag{18}$$

  This seems naively to require calculating posterior probabilities for an exponentially increasing number of candidates, but it is important to emphasize that there are only at most $2N$ possible candidates for N number of observations. This is because, due to the exchangeability of CRPs, there is no need to distinguish between tables that do not have latent landmarks for each nCRP hierarchy.

- $s = (\hat{t}_n/\Delta t) + 1$ to $1/\Delta t$ - Under conditioning on the latent landmark's pseudotime $\hat{t}_n$, the trajectory $z_n^{((\hat{t}_n/\Delta t)+1)} \cdots z_n^{1/\Delta t}$ do not contribute to the likelihood to the observation data. Therefore, we can directly sample them from the prior model. Specifically, for each nCRP hierarchy $s = (\hat{t}_n/\Delta t) + 1, \ldots, 1/\Delta t$, we can sequentially apply the following Gibbs sampling:

$$z_n^{(s)} \quad \sim \quad \mathrm{Categorical}\left(\left\{\frac{\mathcal{N}_{k,n}^{(s)} + \gamma/K}{N - 1 + \gamma} | k \in \{1, \ldots, K\}\right\}\right), \tag{19}$$

where $\mathcal{N}_{k,n}^{(s)}$ is the number of customers sitting at the kth table in the sth nCRP level, other than the nth customer, who is the current target.

(2) Partition locations - For each $\hat{\tau}_n(z_n^{(1)} \cdots z_n^{(\hat{t}_n)} \phi \cdots \phi)$ $(n = 1, \ldots, N,\ s = 1, \ldots, 1/\Delta t)$, i.e., the nCRP table location that the nth data chooses in the sth level, we now consider an update law for the target variable under which all parameters except the target variable are conditioned. In this situation, it is difficult to derive the conditional posterior probability (for Gibbs sampling) of the target variable analytically. But on the other hand we have no difficulty in computing the gradient of the log posterior probability with respect to the target variable. Thus, we can use a Hamiltonian Monte Carlo method that uses the gradient to produce an MCMC update rule. For convenience, we introduce a set v as

$$v := \left\{ i \in \{1, \ldots, N\} | \hat{\tau}_i(z_n^{(1)} \cdots z_n^{(\hat{t}_n)} \phi \cdots \phi) = \hat{\tau}_n(z_n^{(1)} \cdots z_n^{(\hat{t}_n)} \phi \cdots \phi) \right\}. \tag{20}$$

Using the chain rule for composite functions, we can compute the gradient of the log posterior probability with respect to the target variable as follows:

$$\begin{aligned}
&\frac{\partial}{\partial \hat{\tau}_n(z_n^{(1)} \cdots z_n^{(\hat{t}_n)} \phi \cdots \phi)} \log p\left(\hat{\tau}_n(z_n^{(1)} \cdots z_n^{(\hat{t}_n)} \phi \cdots \phi) \mid \hat{\tau}_{1:N} \setminus \hat{\tau}_n(z_n^{(1)} \cdots z_n^{(\hat{t}_n)} \phi \cdots \phi), \hat{t}_{1:N}, \boldsymbol{z}_{1:N}, \gamma, \sigma_0, \theta, \boldsymbol{Y} \right) \\
&\quad = \frac{\partial}{\partial \hat{\tau}_n(s\Delta t)} \left\{ \sum_{n=1}^{N} \log p_{\mathrm{DBM}}(\hat{\tau}_n; \sigma_0) + \log p_{\mathrm{GP}}(\boldsymbol{Y}; \boldsymbol{z}_{1:N}, \hat{\tau}_{1:N}, \hat{t}_{1:N}, \theta) \right\} \\
&\quad = \sum_{i \in v} \boldsymbol{F}(i, 1:N) \sum_{m=1}^{N} \left\{ -\frac{\alpha}{\ell^2} \exp\left(-\frac{1}{2\ell^2} (\boldsymbol{x}_i - \boldsymbol{x}_m)^\top (\boldsymbol{x}_i - \boldsymbol{x}_m) \right) (\boldsymbol{x}_i - \boldsymbol{x}_m) \right\} \\
&\qquad + \sum_{i \in v} \frac{1}{\sigma_0^2} \left\{ \boldsymbol{x}_i - \hat{\tau}_n(z_n^{(1)} \cdots z_n^{(\hat{t}_n - 1)} \phi \cdots \phi) \right\} + \sum_{i \in v} \sum_{k=1}^{K} \frac{1}{\sigma_0^2} \left\{ \boldsymbol{x}_i - \hat{\tau}_n(z_n^{(1)} \cdots z_n^{(\hat{t}_n)} k\phi \cdots \phi) \right\}
\end{aligned} \tag{21}$$

where $\boldsymbol{F} = -\frac{1}{2} D \boldsymbol{K}^{-1} + \frac{1}{2} \boldsymbol{K}^{-1} \boldsymbol{Y} \boldsymbol{Y}^T \boldsymbol{K}^{-1}$. Finally, we apply the Hamilton Monte Carlo framework, similar to the infinite warped mixture model (Iwata et al., 2013), to update the target variable $\hat{\tau}_n(z_n^{(1)} \cdots z_n^{(\hat{t}_n)} \phi \cdots \phi)$.

(3) Latent landmarks - For each t_n $(n = 1, \ldots, N)$, we can calculate the posterior probability of the case $t_n = s$ $(s = 1, \ldots, 1/\Delta t)$ using Equation 17, and set it as r_s. We can then use Gibbs sampling based on a categorical distribution proportional to their posterior probabilities, that is,

$$t_n \quad \sim \quad \mathrm{Categorical}(r_1, \ldots, r_{1/\Delta t}). \tag{22}$$

(4) Hyperparameters - For the kernel hyperparameters θ, we can straightforwardly apply the Hamilton Monte Carlo method, similar to the GPLVM variants (Iwata et al., 2013). For the noninformatice gamma variables γ and σ_0, we can simply use the Metropolis-Hastings algorithm. Specifically, we draw a sample candidate from the prior distribution (i.e., the noninformative gamma prior $\mathrm{Gamma}(\epsilon, \epsilon)$), and then apply the accept/reject scheme.

B Experiment details

B.1 Denoised embedding manifold preservation (used in Section 4.2 of main body)

We wish to calculate the degree to which each method preserves the basic structure of the reference data set and removes noise as a measure for quantitatively comparing each pseudotime estimation method. In general, single-cell RNA sequences and other biological data are very noisy, so data visualization methods that present a latent space that reveals the underlying structure of the data should be highly valued. Accordingly, metrics have been proposed to quantify the correspondence between distances in low-dimensional embeddings and manifold distances in ground-truth references (Moon et al., 2019). One standard way to define a quantitative notion of manifold distance is to use geodesic distance. The geodesic distance is the shortest path distance on the nearest graph of the data, weighted by the Euclidean distance between connected points. Importantly, the geodesic distance converges exactly to the distance along the manifold of data when points are noiselessly sampled from some manifold, as in our ground-truth reference. Thus, if the geodesic distances between points on a noise-free

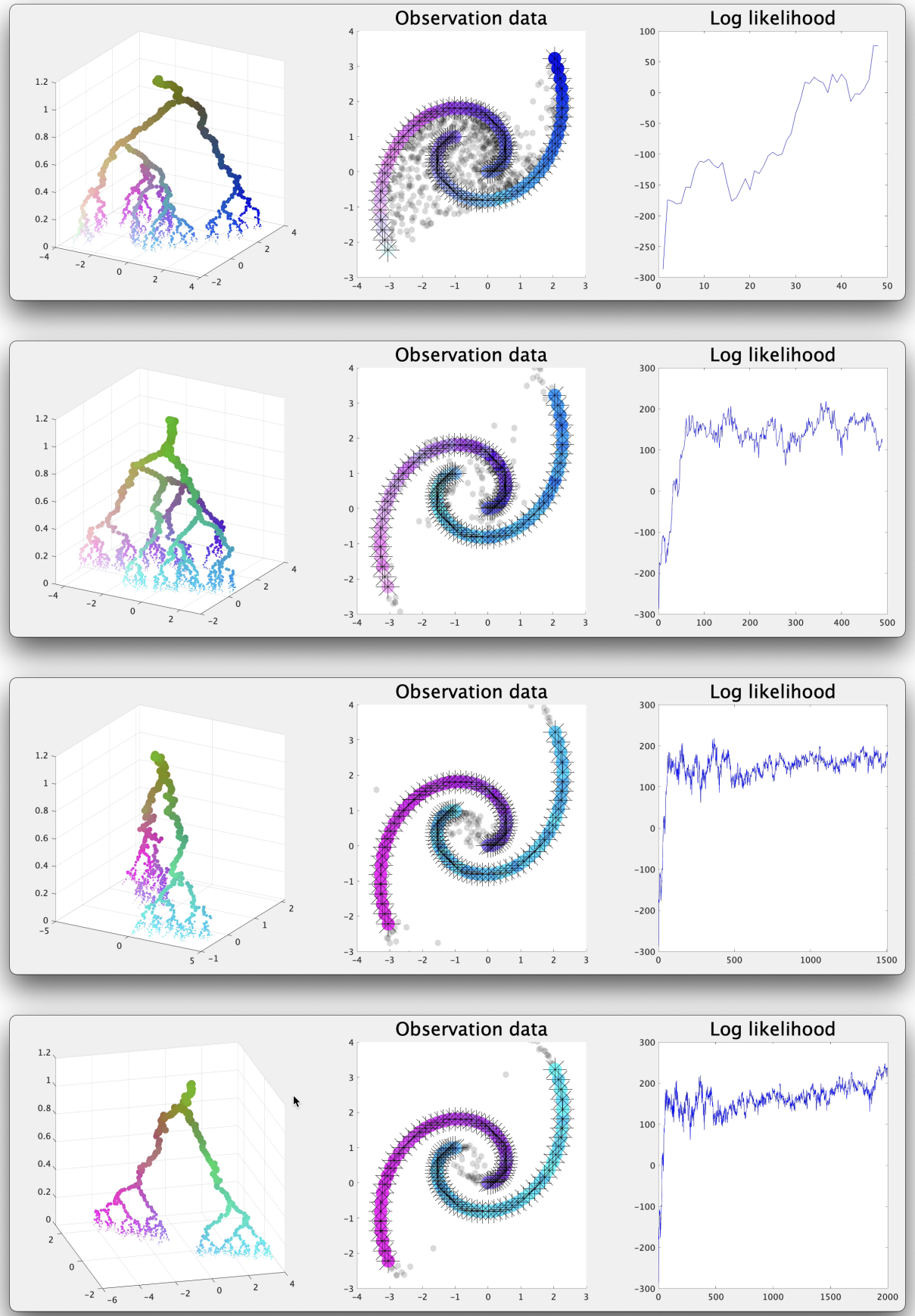

Figure 12: Screenshots of MCMC evolution for Spiral data.

manifold are preserved by an embedding computed for noisy data, we can expect the data to be adequately denoised and the true data structure to be adequately preserved. Thus, a measure called denoised embedding manifold preservation (DEMaP) quantifies the preservation of manifold distance as the correlation between the geodesic distance in a noiseless reference data set and the Euclidean distance in the embedding space, and is the standard measure of structure preservation utilized.

B.2 Splatter simulation (used in Section 4.2 of main body)

Splatter is a single cell Ribonucleic acid (RNA) sequence simulation package that uses a parametric model to generate data with various structures, such as branches or clusters (Zappia et al., 2017). We use Splatter to simulate multiple ground truth datasets for multiple experiments. The parameters of the simulation were set following previous studies. They were chosen to fit the Embryoid Body data (Martin and Evans, 1975) from the Splatter `paths` (for latent diffusion structures). The default parameters used in the simulation are the following:

n_cells	n_genes	mean.shape	mean.rate	lib.loc	lib.scale	out.prob	out.facLoc	out.facScale	bcv.common	bcv.df	de.prob
3000	17580	6.6	0.45	9.1	0.33	0.016	5.4	0.90	0.18	21.6	0.2

For more information on the parameters, we like to direct the reader to the Splatter project page.[6] We simulate 100 datasets, including branches, and add biological and technical noise to the reference data to replicate the following three scenarios: (1) Using Splatter's biological coefficient of variation (BCV) parameter, `bcv.common` $= \{0, 0.25, 0.5\}$, to simulate stochastic gene expression. (2) Undersampling from the true counts, `n_cells` $= \{150, 1500, 2850\}$, using the default BCV to simulate inefficient capture of RNA in single cells. (3) Randomly remove genes from the data matrix, `n_genes` $= \{2000, 10000, 17000\}$ to demonstrate robustness to variation in all measured genes. Additionally, for the `paths` simulation, we draw the number of groups from a Poisson distribution with rate 10, and then draw the `group.prob` parameter from a Dirichlet distribution with n categories and a uniform concentration $(1, 1, \ldots, 1)$. Finally, we set the ith entry in the parameter `path.from` as a random integer between 0 and $i-1$, draw the parameter `path.nonlinearProb` from a uniform distribution on the interval $(0, 1)$, and draw the parameter `path.skew` from a beta distribution with shape $(10, 10)$.

[6] `https://github.com/Oshlack/splatter`