Hyperboloid GPLVM for Discovering Continuous Hierarchies via Nonparametric Estimation

Koshi Watanabe Hokkaido University
Keisuke Maeda Hokkaido University
Takahiro Ogawa Hokkaido University
Miki Haseyama Hokkaido University

Abstract

Dimensionality reduction (DR) offers interpretable representations of complex high-dimensional data, and recent DR methods have leveraged hyperbolic geometry to obtain faithful low-dimensional embeddings of high-dimensional hierarchical relationships. However, existing methods are dependent on neighbor embedding, which frequently ruins the continuous nature of the hierarchical structures. This paper proposes hyperboloid Gaussian process latent variable models (hGP-LVMs) to embed high-dimensional hierarchical data while preserving the implicit continuity via nonparametric estimation. We adopt generative modeling using the GP, which provides effective hierarchical embedding and executes ill-posed hyperparameter tuning. This paper presents three variants of the proposed models that employ original point, sparse point, and Bayesian estimations, and we establish their learning algorithms by incorporating the Riemannian optimization and active approximation scheme of the GP-LVM. In addition, we employ the reparameterization trick for scalable learning of the latent variables in the Bayesian estimation method. The proposed hGP-LVMs were applied to several datasets, and the results demonstrate their ability to represent high-dimensional hierarchies in low-dimensional spaces.

1 Introduction

With the emergence of large high-dimensional datasets, unsupervised dimensionality reduction (DR) techniques have gained increasing attention in discovering a faithful low-dimensional representation while preserving essential characteristics of the data. Recent studies have shifted focus from conventional toy datasets to more complicated datasets, e.g., neural activities (Jensen et al., 2020) or single-cell ribonucleic acid sequencing (scRNA-seq) (Becht et al., 2019), frequently leveraging Riemannian geometry to achieve effective data embedding. Nonlinear hierarchical relationships are frequently encountered in high-dimensional data, where *hyperbolic embedding* (Nickel and Kiela, 2017, 2018) has proven particularly effective. Utilizing curved hyperbolic geometry, hyperbolic embedding enables faithful DR of hierarchical datasets while requiring far fewer dimensions than their Euclidean counterparts (Sala et al., 2018). However, to the best of our knowledge, visualization-aided hyperbolic embedding (Klimovskaia et al., 2020; Jaquier et al., 2022) remains relatively underexplored despite its applicability in visualizing high-dimensional hierarchical data. Thus, this paper focuses on advancing hyperbolic embedding to realize improved visualization of hierarchical data structures.

First, we categorize previous DR methods into four classes based on two key axes to establish effective DR of hierarchical data, i.e., *parametric* vs. *nonparametric* and *data-embedding* vs. *neighbor-embedding*. Specifically, we classify DR methods as *parametric* or *nonparametric* based on whether they involve a parametric projection between the observed and latent spaces. Similarly, we distinguish between *data-embedding* and *neighbor-embedding* methods depending on whether they utilize the raw data or rely on neighbor relations (e.g., a k-nearest neighbor graph). For example, principal component analysis (PCA) (Hotelling, 1933) and variational autoencoders (VAE) (Kingma and Welling, 2013; Higgins et al., 2017) are parametric data-embedding methods, and t-stochastic neighbor embedding (t-SNE) (Van der Maaten and Hinton, 2008) and uniform manifold approximation and projection (UMAP) (McInnes et al., 2018) are *nonparametric neighbor-embedding* methods.

Conventionally, visualization-aided DR is based on non-

Proceedings of the 28[th] International Conference on Artificial Intelligence and Statistics (AISTATS) 2025, Mai Khao, Thailand. PMLR: Volume 258.

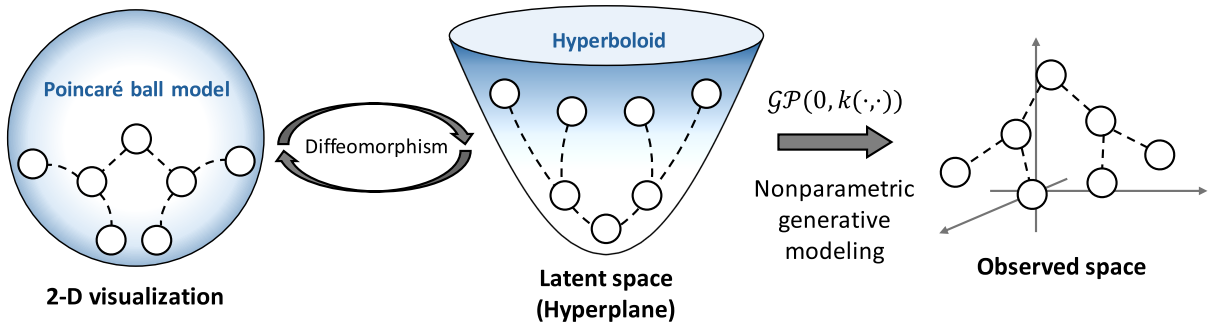

Figure 1: **Illustration of hyperboloid Gaussian process latent variable models (hGP-LVMs)**. We learn the latent variables on the Lorentz model and visualize them on the Poincaré ball model.

parametric neighbor-embedding methods because they are not dependent on ill-posed parameter tuning and better preserve local structures. However, for hierarchical data embedding, the neighbor embedding methods frequently create discrete clusters (Amid and Warmuth, 2019; Wang et al., 2021), potentially overlooking the global *implicit continuity* of the hierarchical structure. This inherent limitation of neighbor-embedding methods has motivated the development of *nonparametric data-embedding* methods, which are better suited for DR of hierarchical data.

Gaussian process (GP) latent variable models (GP-LVMs) (Lawrence, 2005; Titsias and Lawrence, 2010) are representative examples of nonparametric data-embedding methods. GP-LVMs assume the GP decoder of the observed variables, which is a widely used nonparametric model to estimate unknown functions (Rasmussen and Williams, 2006). Generative modeling between the observed and latent spaces frequently enhances the continuity of the latent variables, thereby making them suitable for preserving the continual relationships in hierarchical data. However, recent studies have explored incorporating simple Riemannian geometries in the GP (Mallasto and Feragen, 2018; Borovitskiy et al., 2020) or developing supervised kernel methods in hyperbolic spaces (Fang et al., 2021; Fan et al., 2023), and only a few methods realize the hyperbolic extension of GP-LVMs (Jaquier et al., 2022).

This paper proposes *hyperboloid GP-LVMs* (*hGP-LVMs*) to visualize the hierarchical relationships behind high-dimensional data via nonparametric estimation (Figure 1). We learn the latent variables on the Lorentz model and visualize them on the Poincaré ball model by applying the diffeomorphism among them, and we learn the hyperbolic latent variables by developing dedicated algorithms that incorporate the previous Riemannian optimization and sparse GP methods. In addition, we formulate three variants of hGP-LVMs, i.e., extension of the original point estimation (Lawrence, 2005), sparse point estimation (Lawrence, 2007; Titsias, 2009), and Bayesian estimation methods (Titsias and Lawrence, 2010; Lalchand et al., 2022) to address classical computational issues associated with the GP and realize scalable Bayesian learning of latent variables. The primary contribution of this paper is the development of the hGP-LVMs to visualize hierarchical data effectively via nonparametric estimation using generative modeling with the GP decoder. We demonstrate that the GP-based modeling and Bayesian estimation of the latent variables benefit visualization-aided DR on both synthetic and real-world datasets.

2 Background

Before introducing the proposed hGP-LVMs, we first discuss previous GP-LVMs and the basic concepts of the hyperbolic space.

2.1 Gaussian Process Latent Variable Models

Here, let $\mathbf{y}_i \in \mathbb{R}^D (i = 1, 2, \ldots, N)$ be the D-dimensional observed variables and let $\mathbf{x}_i \in \mathcal{M}^Q$ be the latent variables on a Q-dimensional smooth manifold $\mathcal{M}^Q$, i.e., the target of DR. We denote $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_N]^\top \in \mathbb{R}^{N\times D}$ and $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N]^\top \in \mathbb{R}^{N\times Q}$, and let E be an Euclidean manifold. The original GP-LVM is given as follows:

$$\mathbf{y}_{:,d} = \mathbf{f}_d(\mathbf{X}) + \boldsymbol{\epsilon}, \tag{1}$$

$$\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \beta^{-1}\mathbf{I}_n), \tag{2}$$

$$\mathbf{f}_d \sim \mathcal{GP}(\mathbf{0}, k(\cdot,\cdot)), \tag{3}$$

where $\mathbf{f}_d \in \mathbb{R}^N (d = 1, 2, \ldots, D)$ is a GP prior with a kernel function $k(\cdot,\cdot)$, and $\boldsymbol{\epsilon} \in \mathbb{R}^N$ is Gaussian noise with a precision β. The GP-LVM estimates the latent variables by maximizing the log-likelihood $\mathcal{F} = \sum_{d=1}^D \log p(\mathbf{y}_{:,d}|\mathbf{X})$ as follows:

$$\mathcal{F} = -\frac{ND}{2}\log 2\pi - \frac{D}{2}\log|\mathbf{K}_{nn} + \beta^{-1}\mathbf{I}_n| - \frac{1}{2}\mathrm{tr}\left[(\mathbf{K}_{nn} + \beta^{-1}\mathbf{I}_n)^{-1}\mathbf{Y}\mathbf{Y}^\top\right], \tag{4}$$

where $\mathbf{K}_{nn} \in \mathbb{R}^{N\times N}$ is a gram matrix whose (i, j)-th entry is $k(\mathbf{x}_i, \mathbf{x}_j)$. The maximization of Eq. (4) is performed through the gradient-based optimization; however, the evaluation of Eq. (4) requires cubic time complexity $O(N^3)$, which restricts the scalability of the GP-LVM. In addition, the nonlinearity of $(\mathbf{K}_{nn} + \beta^{-1}\mathbf{I}_n)^{-1}$ hinders the full Bayesian treatment of $\mathbf{X}$. The *inducing points method* (Titsias, 2009; Bauer et al., 2016) has been used to address these issues. This method assumes inducing point $\mathbf{u}_d$, and its positions $\mathbf{z}_k \in \mathcal{M}^Q\,(k = 1, 2, \ldots, M)$ are *sufficient statistics* for the prior $\mathbf{f}_d$, i.e., $p(\mathbf{Y}|\mathbf{X}, \mathbf{Z}) = \prod_{d=1}^{D} \int p(\mathbf{y}_{:,d}|\mathbf{u}_d, \mathbf{X}, \mathbf{Z}) p(\mathbf{u}_d|\mathbf{Z}) d\mathbf{u}_d$, and then introduces the variational inference as $\log p(\mathbf{y}_{:,d}|\mathbf{u}_d, \mathbf{X}, \mathbf{Z}) \geq \mathbb{E}_{p(\mathbf{f}_d|\mathbf{u}_d,\mathbf{X},\mathbf{Z})}\left[\log p(\mathbf{y}_{:,d}|\mathbf{f}_d)\right]$. The objective function of the sparse GP-LVM is a tight lower bound of the log-likelihood $\sum_{d=1}^{D} \log p(\mathbf{y}_{:,d}|\mathbf{X}, \mathbf{Z}) \geq \acute{\mathcal{F}}$:

$$\acute{\mathcal{F}} = -\frac{D}{2}\log\frac{(2\pi)^N|\mathbf{A}|}{\beta^N|\mathbf{K}_{mm}|} - \frac{1}{2}\mathrm{tr}\left(\mathbf{W}\mathbf{Y}\mathbf{Y}^\top\right) - \frac{\beta D}{2}\mathrm{tr}\left(\mathbf{K}_{nn}\right) + \frac{\beta D}{2}\left(\mathbf{K}_{mm}^{-1}\mathbf{K}_{mn}\mathbf{K}_{nm}\right), \quad (5)$$

where $\mathbf{A} = \mathbf{K}_{mm} + \beta\mathbf{K}_{mn}\mathbf{K}_{nm}$, $\mathbf{W} = \beta\mathbf{I}_n - \beta^2\mathbf{K}_{nm}\mathbf{A}^{-1}\mathbf{K}_{mn}$, and $\mathbf{K}_{mn} = \mathbf{K}_{nm}^\top \in \mathbb{R}^{M\times N}$ are gram matrices whose (k, j)-th entry is $k(\mathbf{z}_k, \mathbf{x}_j)$. The problems regarding the scalability and nonlinearity of Eq. (4) are solved by vanishing the inversion of $\mathbf{K}_{nn} + \beta^{-1}\mathbf{I}_n$. The Bayesian GP-LVM estimates the approximated posterior $q(\mathbf{X}) = \prod_{i=1}^{N} q(\mathbf{x}_i)$ rather than the deterministic latent variables. The objective is the evidence lower bound (ELBO) $\sum_{d=1}^{D} \log p(\mathbf{y}_{:,d}|\mathbf{Z}) \geq \acute{\mathcal{F}}_b$, which is expressed as follows:

$$\acute{\mathcal{F}}_b = -\frac{D}{2}\log\frac{(2\pi)^N|\mathbf{A}_b|}{\beta^N|\mathbf{K}_{mm}|} - \frac{1}{2}\mathrm{tr}\left(\mathbf{W}_b\mathbf{Y}\mathbf{Y}^\top\right) - \frac{\beta D}{2}\mathrm{tr}\left(\mathbf{K}_{nn}\right) + \frac{\beta D}{2}\mathrm{tr}\left(\mathbf{K}_{mm}^{-1}\mathbf{\Psi}_2\right) - \sum_{i=1}^{N}\mathrm{KL}_i, \quad (6)$$

where $\mathrm{KL}_i = \mathrm{KL}[q(\mathbf{x}_i)||p(\mathbf{x}_i)]$ is Kullback–Leibler (KL) divergence between $q(\mathbf{x}_i)$ and $p(\mathbf{x}_i)$, $p(\mathbf{x}_i)$ is a prior distribution of latent variables, $\mathbf{A}_b = \mathbf{K}_{mm} + \beta\mathbf{\Psi}_2$, $\mathbf{W}_b = \beta\mathbf{I}_n - \beta^2\mathbf{\Psi}_1^\top\mathbf{A}_b^{-1}\mathbf{\Psi}_1$, $\mathbf{\Psi}_1 = \mathbb{E}_{q(\mathbf{X})}[\mathbf{K}_{mn}]$, and $\mathbf{\Psi}_2 = \mathbb{E}_{q(\mathbf{X})}[\mathbf{K}_{mn}\mathbf{K}_{nm}]$. Note that we provide the detailed derivation from Eq. (4) to Eq. (6) in Appendix A.1. The computational intractability of Eq. (6) appears frequently in the marginalization of $\mathbf{\Psi}$ statistics, which is typically solved by sampling approximation with the reparameterization trick (Salimbeni and Deisenroth, 2017; de Souza et al., 2021; Lalchand et al., 2022).

2.2 Hyperbolic Geometry

Hyperbolic spaces are smooth Riemannian manifolds and have several isometric models (Peng et al., 2021). For example, the *Poincaré ball model* (Nickel and Kiela, 2017; Ganea et al., 2018) $\mathcal{P}^Q = (\mathbb{B}^Q, g_b)$ is a representative example in machine learning literature, where $\mathbb{B}^Q = \{\mathbf{x} \in \mathbb{R}^Q : ||\mathbf{x}||_2 < 1\}$, and $g_b = \frac{2}{1-||\mathbf{x}||^2} g_e$ is the metric tensor with the Euclidean metric tensor g_e. However, the boundary $\{\mathbf{x} \in \mathbb{R}^Q : ||\mathbf{x}||_2 = 1\}$ in the Poincaré ball model causes numerical instability. The *Lorentz model* is another example that does not contain a boundary. Formally, the Lorentz model $\mathcal{L}^Q = (\mathbb{H}^Q, g_l)$ is a Q-dimensional hyperbolic space with a Q-dimensional *upper hyperboloid* $\mathbb{H}^Q = \{\mathbf{x} = [x_0, x_1, \ldots, x_Q]^\top \in \mathbb{R}^{Q+1} : \langle\mathbf{x}, \mathbf{x}\rangle_{\mathcal{L}^Q} = -1, x_0 > 0\}$ and metric tensor $g_l = \mathrm{diag}(-1, 1, \ldots, 1) \in \mathbb{R}^{(Q+1)\times(Q+1)}$, where $\langle\mathbf{x}, \mathbf{x}'\rangle_{\mathcal{L}^Q} = -x_0x_0' + \sum_{i=1}^{Q} x_ix_i'$ is the *Lorentzian inner product*. Note that $x_0 = \sqrt{1 + ||\tilde{\mathbf{x}}||_2^2}$, where $\tilde{\mathbf{x}} = [x_1, x_2, \ldots, x_Q]^\top$. The tangent space $\mathcal{T}_{\boldsymbol{\mu}}\mathcal{L}^Q$, i.e., the set of the tangent passes through $\boldsymbol{\mu} \in \mathcal{L}^Q$, is useful to extend Euclidean operations to the hyperbolic space. Furthermore, the mapping from the Lorentz model onto its tangent space at $\boldsymbol{\mu}$ is stated explicitly by the exponential map $\mathrm{Exp}_{\boldsymbol{\mu}}(\mathbf{v}) : \mathcal{T}_{\boldsymbol{\mu}}\mathcal{L}^Q \to \mathcal{L}^Q$ as follows:

$$\mathrm{Exp}_{\boldsymbol{\mu}}(\mathbf{v}) = \cosh(||\mathbf{v}||_{\mathcal{L}^Q})\boldsymbol{\mu} + \sinh(||\mathbf{v}||_{\mathcal{L}^Q})\frac{\mathbf{v}}{||\mathbf{v}||_{\mathcal{L}^Q}}, \quad (7)$$

where $||\mathbf{v}||_{\mathcal{L}^Q} = \sqrt{\langle\mathbf{v}, \mathbf{v}\rangle_{\mathcal{L}^Q}}$ is the norm of $\mathbf{v} \in \mathcal{T}_{\boldsymbol{\mu}}\mathcal{L}^Q$. The length of geodesic between two points on $\mathcal{L}^Q$ is expressed as follows:

$$d_{\mathcal{L}^Q}(\mathbf{x}, \mathbf{x}') = \cosh^{-1}\left(-\langle\mathbf{x}, \mathbf{x}'\rangle_{\mathcal{L}^Q}\right). \quad (8)$$

Finally, the diffeomorphism $p(\mathbf{x}) : \mathcal{L}^Q \to \mathcal{P}^Q$ is given as:

$$p(\mathbf{x}) = \frac{[x_1,\, x_2,\, \ldots, x_Q]^\top}{1 + x_0}. \quad (9)$$

We learn the latent variables on the Lorentz model and visualize them on the Poincaré ball model.

3 Hyperboloid GP-LVMs

The proposed hGP-LVMs, which are the primary contribution of this paper, and presented in the following. First, we establish the positive definite (PD) kernel on the Lorentz model, which is referred to as the *hyperboloid exponential kernel*. We then explain the optimization of the hGP-LVMs considering the curved geometry of the latent space. In the Bayesian estimation method, straightforward optimization cannot be realized due to the computational intractability included in the $\mathbf{\Psi}$ statistics and KL divergence. Thus, we develop a dedicated algorithm that combines the reparameterization trick (Kingma and Welling, 2013),

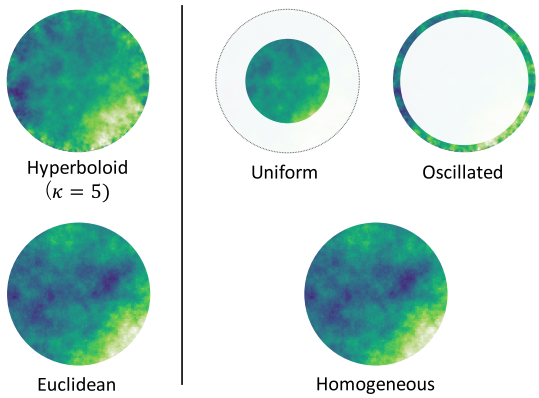

Figure 2: GP prior comparison between the hyperboloid exponential kernel (*upper*, $\kappa = 5$) and the Euclidean exponential kernel (*bottom*). The color gives the value of the sampled GP. We input the latent variables on the Poincaré ball model when $\mathcal{M} = \mathcal{L}^Q$ (*left*) and those on the unit circle when $\mathcal{M} = E$ (*right*).

Riemannian optimization (Nickel and Kiela, 2018), and active set approximation of GP-LVMs (Moreno-Muñoz et al., 2022).

3.1 Hyperboloid Exponential Kernel

Recent studies have utilized the heat kernel for the Riemannian kernel construction (Atigh et al., 2022; Niu et al., 2023; Azangulov et al., 2023); however, it has a high computational cost, which hinders the scalable optimization. Thus, we employ the result reported in the literature (Feragen et al., 2015) that shows that the *geodesic exponential kernel* is the PD kernel under a certain condition. Formally, the geodesic exponential kernel is expressed as follows:

$$k_{\mathcal{M}}(\mathbf{x}, \mathbf{x}') = \sigma \exp\left(-\frac{d_{\mathcal{M}}(\mathbf{x}, \mathbf{x}')}{\kappa}\right), \qquad (10)$$

where σ and κ are variance and *length scale* parameters,respectively. This geodesic exponential kernel can be PD if $d_{\mathcal{M}}(\mathbf{x}, \mathbf{x}')$ is a conditionally negative definite (CND) metric (Feragen et al., 2015). Here, the hyperbolic metric is CND (Istas, 2012); thus, we can apply the geodesic exponential kernel to the hyperbolic space. When $\mathcal{M} = \mathcal{L}^Q$, we refer to the kernel function as the *hyperboloid exponential (HE) kernel*. To visualize the difference between the HE and Euclidean kernels, Figure 2 compares the GP prior of the HE kernel and that of the Euclidean exponential kernel. When $\mathcal{M} = \mathcal{L}^Q$, the prior highly correlates around the origin (similar values) and oscillates around the rim (striped values), unlike the Euclidean kernel, which is homogeneous on the entire unit circle. This heterogeneity of the hyperboloid kernel contributes to embedding the hierarchical data containing the exponentially growing child nodes.

In the following, we discuss the role of the HE kernel parameters σ and κ. The variance σ is similar to the general exponential kernel, i.e., it determines the range of the kernel function. However, the *length scale* parameter κ differs from that of the Euclidean kernel. The length scale of Euclidean GP-LVMs calibrates the scale of the latent variables. In the flat Euclidean space, the scale has no structural meaning; however, it does not hold in the curved hyperbolic space. With a large length scale, the latent variables are widely spread on the curved manifold and are strongly influenced by the hyperbolic curvature. With a small length scale, we first show one result for the relation between the distance in the Lorentz model and the Euclidean space.

Lemma 1. *Here, let* $\mathbf{x}_1 \in \mathbb{R}^Q$ *and* $\mathbf{x}_2 \in \mathbb{R}^Q$ *be vectors with small scales, i.e.,* $||\mathbf{x}_1||_2 \approx 0$, $||\mathbf{x}_2||_2 \approx 0$, *and set* $\mathbf{x}'_1 = \left[\sqrt{1 + ||\mathbf{x}_1||_2^2}, \mathbf{x}_1^\top\right]^\top \in \mathcal{L}^Q$ *and* $\mathbf{x}'_2 = \left[\sqrt{1 + ||\mathbf{x}_2||_2^2}, \mathbf{x}_2^\top\right]^\top \in \mathcal{L}^Q$. *Then, we obtain the following result:*

$$d_{\mathcal{L}^Q}(\mathbf{x}'_1, \mathbf{x}'_2) \approx d_E(\mathbf{x}_1, \mathbf{x}_2) + O(d_E(\mathbf{x}_1, \mathbf{x}_2)^3), \qquad (11)$$

where $d_E(\mathbf{x}_1, \mathbf{x}_2) = ||\mathbf{x}_1 - \mathbf{x}_2||_2$.

Proof. Refer to Appendix A.2. □

Equation (11) indicates that the hyperboloid metric approaches the Euclidean metric around the origin asymptotically, and the HE kernel with a small length scale behaves like the Euclidean geodesic kernel. From the above discussion, we understand that *the meaning of the length scale κ is how much we expect the latent variables to follow the hyperbolic curvature*. However, this metamorphosis of the length scale causes difficulty with optimization. Thus, we treat the length scale as a predefined hyperparameter and determine it experimentally. In summary, the HE kernel is established by extending the geodesic exponential kernel in Eq. (10), and the length scale is treated as a hyperparameter to determine the degree to which the latent variables follow the hyperbolic curvature.

3.2 Optimization

Here, we discuss applying the objective of the GP-LVM, sparse GP-LVM, and Bayesian GP-LVM with the hyperboloid kernel to learn the latent variables on the Lorentz model. The optimization challenge is the gradient computation of the latent variables, and we must consider the curved geometry of the hyperboloid. First, we explain the optimization of the deterministic latent variables and extend it to the Bayesian case using a *wrapped Gaussian distribution* (Nagano et al., 2019; Mathieu et al., 2019; Cho et al., 2022).

Algorithm 1 Updating point latent variables.

Require: learning rate α
while $t <$ max_iter **do**
  for $i = 1, 2, \ldots, N$ **do**
    $\mathbf{g}_i^t \leftarrow g_l^{-1}[\frac{\partial \mathcal{F}}{\partial x_{i0}^t}, \frac{\partial \mathcal{F}}{\partial x_{i1}^t}, \ldots, \frac{\partial \mathcal{F}}{\partial x_{iQ}^t}]^\top$
    $\mathcal{T}\mathbf{g}_i^t \leftarrow \mathrm{proj}_{\mathbf{x}_i^t}(\mathbf{g}_i^t)$
    $\mathbf{x}_i^{t+1} \leftarrow \exp_{\mathbf{x}_i^t}(-\alpha \mathcal{T}\mathbf{g}_i^t)$
  end for
end while

hGP-LVM and Sparse hGP-LVM. We consider the curved surfaces of the latent space by employing the Riemannian gradient descent algorithm (Nickel and Kiela, 2018). This algorithm (i) computes the steepest direction, (ii) projects the *row* gradients into the tangent space, and (iii) wraps the projected vector in the surface of the Lorentz model following Eq. (7) (**Algorithm** 1). Here, the row gradients are given as $\frac{\partial \mathcal{F}}{\partial x_{iq}} = \mathrm{tr}(\frac{\partial \mathcal{F}}{\partial \mathbf{K}} \frac{\partial \mathbf{K}}{\partial x_{iq}})$. In addition, the projection from the ambient Euclidean space to the Lorentz model $\mathrm{proj}_{\boldsymbol{\mu}}(\mathbf{g}) : \mathbb{R}^{N+1} \to \mathcal{T}_{\boldsymbol{\mu}}\mathcal{L}^Q$ is expressed as follows:

$$\mathrm{proj}_{\boldsymbol{\mu}}(\mathbf{g}) = \mathbf{g} + \langle \boldsymbol{\mu}, \mathbf{g} \rangle_{\mathcal{L}^Q} \boldsymbol{\mu}. \tag{12}$$

In the sparse hGP-LVM, the positions of the inducing points $\mathbf{Z}$ must be optimized in addition to the latent variables; however, their gradient-based updates cause unstable optimization (even in the Euclidean case). Therefore, we employ the active set approximation scheme (Moreno-Muñoz et al., 2022), where we sample the inducing positions $\mathbf{Z}$ from the latent variables $\mathbf{X}$ at each multiple updates.

Bayesian hGP-LVM. To establish probabilistic latent variable models, we first require the variational distribution $q(\mathbf{x}_i)$ and prior $p(\mathbf{x}_i)$ defined on the Lorentz models. Here, for computational efficiency we employ the wrapped Gaussian distribution $\mathcal{N}^w_{\mathcal{L}^Q}(\mathbf{x}|\boldsymbol{\mu}, \mathbf{S})$ (Nagano et al., 2019) rather than the exact hyperbolic Gaussian. In addition, we assume $q(\mathbf{x}_i) = \mathcal{N}^w_{\mathcal{L}^Q}(\mathbf{x}_i|\boldsymbol{\mu}_i, \mathbf{S}_i)$ and $p(\mathbf{x}_i) = \mathcal{N}^w_{\mathcal{L}^Q}(\mathbf{x}_i|\mathbf{0}, \mathbf{I}_q)$, where $\boldsymbol{\mu}_i \in \mathcal{L}^Q$ and $\mathbf{S}_i = \mathrm{diag}(s_{i1}, s_{i2}, \ldots, s_{iQ}) \in \mathbb{R}^{Q \times Q}$ are variational parameters. The probability density function of $\mathcal{N}^w_{\mathcal{L}^Q}(\mathbf{x}_i|\boldsymbol{\mu}_i, \mathbf{S}_i)$ can be given in a closed form as follows:

$$\mathcal{N}^w_{\mathcal{L}^Q}(\mathbf{x}|\boldsymbol{\mu}, \mathbf{S}) = \left\{ \frac{\sinh(||\mathbf{u}||_{\mathcal{L}^Q})}{||\mathbf{u}||_{\mathcal{L}^Q}} \right\}^{(Q-1)} \mathcal{N}(\mathbf{v}|\mathbf{0}, \mathbf{S}). \tag{13}$$

Computing the $\boldsymbol{\Psi}$ statistics and KL divergence in Eq. (6) is intractable with the HE kernel; thus, we compute them approximately using the sampling scheme and the reparameterization trick. The sampling scheme of $\mathcal{N}^w_{\mathcal{L}^Q}(\mathbf{x}|\boldsymbol{\mu}, \mathbf{S})$ involves three steps (similar to the Riemannian optimization), i.e., (i) sampling $\tilde{\mathbf{v}} \sim \mathcal{N}(\mathbf{0}, \mathbf{S})$ at the origin $\boldsymbol{\mu}_0 = [1, 0, \ldots, 0]^\top \in \mathcal{L}^Q$, (ii) carrying $\mathbf{v} = [0, \tilde{\mathbf{v}}^\top]^\top \in \mathcal{T}_{\boldsymbol{\mu}_0}\mathcal{L}^Q$ from the origin to an arbitrary point $\boldsymbol{\mu}$ as $\mathbf{u} = \mathrm{PT}_{\boldsymbol{\mu}_0 \to \boldsymbol{\mu}}(\mathbf{v})$, and (iii) wrapping the vector in the surface following Eq. (7). The parallel transportation $\mathrm{PT}_{\boldsymbol{\nu} \to \boldsymbol{\mu}}(\mathbf{v}) : \mathcal{T}_{\boldsymbol{\nu}}\mathcal{L}^Q \to \mathcal{T}_{\boldsymbol{\mu}}\mathcal{L}^Q$ is computed as follows:

$$\mathrm{PT}_{\boldsymbol{\nu} \to \boldsymbol{\mu}}(\mathbf{v}) = \mathbf{v} + \frac{\langle \boldsymbol{\mu} - \gamma \boldsymbol{\nu}, \mathbf{v} \rangle_{\mathcal{L}^Q}}{\gamma + 1}(\boldsymbol{\mu} + \boldsymbol{\nu}), \tag{14}$$

where $\gamma = \langle \boldsymbol{\mu}, \boldsymbol{\nu} \rangle_{\mathcal{L}^Q}$. Thus, the Monte Carlo approximations of $\boldsymbol{\Psi}$ and the KL divergence are obtained as follows:

$$[\boldsymbol{\Psi}_1]_{ik} = \sum_{h=1}^{H} k_{\mathcal{L}^Q}(\mathbf{x}_i^{(h)}, \mathbf{z}_k), \ \boldsymbol{\Psi}_2 = \sum_{i=1}^{N} \boldsymbol{\Psi}_2^{(i)}, \tag{15}$$

$$[\boldsymbol{\Psi}_2^{(i)}]_{kl} = \sum_{h=1}^{H} k_{\mathcal{L}^Q}(\mathbf{z}_k, \mathbf{x}_i^{(h)}) k_{\mathcal{L}^Q}(\mathbf{x}_i^{(h)}, \mathbf{z}_l), \tag{16}$$

$$\mathrm{KL}_i = \sum_{h=1}^{H} \log \frac{\mathcal{N}^w_{\mathcal{L}^Q}(\mathbf{x}_i^{(h)}|\boldsymbol{\mu}_i, \mathbf{S}_i)}{\mathcal{N}^w_{\mathcal{L}^Q}(\mathbf{x}_i^{(h)}|\mathbf{0}, \mathbf{I}_q)}. \tag{17}$$

Here, H is the number of samples, and $\mathbf{x}_i^{(h)}$ is the reparameterized latent variables:

$$\mathbf{x}_i^{(h)} = \mathrm{Exp}_{\boldsymbol{\mu}_i}(\mathbf{u}_i^{(h)}), \tag{18}$$

$$\mathbf{u}_i^{(h)} = \mathrm{PT}_{\boldsymbol{\mu}_0 \to \boldsymbol{\mu}_i}(\mathbf{v}_i^{(h)}), \tag{19}$$

$$\tilde{\mathbf{v}}_i^{(h)} = \mathbf{S}_i^{\frac{1}{2}} \boldsymbol{\zeta}_i^{(h)}, \tag{20}$$

where $\mathbf{v}_i = [0, \tilde{\mathbf{v}}_i^{(h)\top}]^\top$ and $\boldsymbol{\zeta}_i^{(h)} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_q)$. Note that we update the variational mean $\boldsymbol{\mu}_i$ following the Riemannian procedure described in **Algorithm** 1, and we determine $\mathbf{Z}$ by sampling from $\boldsymbol{\mu}_i$. The computation of the row gradient $\frac{\partial \hat{\mathcal{F}}_b}{\partial \boldsymbol{\mu}_i}$ and $\frac{\partial \hat{\mathcal{F}}_b}{\partial \mathbf{S}_i}$ is described in detail in Appendix A.3.

The time complexities of hGP-LVM, sparse hGP-LVM, and Bayesian hGP-LVM are $O(N^3)$, $O(M^2N + M^3)$, and $O(HM^2N + HM^3)$, respectively, which are equal to the costs of the previous GP-LVM baselines.

4 Related Work

Visualization-Aided DR. t-SNE (Van der Maaten and Hinton, 2008) is the gold-standard method for data visualization, and the recent UMAP (McInnes et al., 2018) approach introduces the algebraic topology concept and realizes DR with a solid theoretical foundation. Note that the neighbor embedding methods frequently overlook the global structure; thus, recent variants attempt to catch the global structure using a triplet loss function (Amid and Warmuth, 2019) or by modifying the initial embedding (Kobak and Berens, 2019; Wang et al., 2021). DR for hierarchical data has gained considerable attention, where the potential heat diffusion for

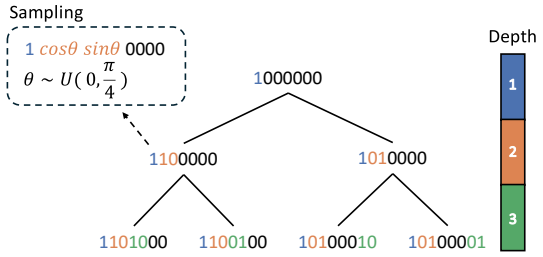

Figure 3: Illustration of the synthetic binary tree (SBT) with $d = 3$ and the sampling procedure.

affinity-based trajectory embedding (PHATE) (Moon et al., 2019) method employs the diffusion operation, and the PoincaréMap (Klimovskaia et al., 2020) develops the neighbor embedding on the Poincaré ball model. Visualization-aided DR methods have been widely applied in scRNA-seq analysis (Luecken and Theis, 2019) to support the exposition of complex cell biology.

Hyperbolic Machine Learning. Previously, the hyperbolic space has been employed to realize the hierarchical structure learning of word taxonomies (Nickel and Kiela, 2017, 2018). Recent studies have applied hyperbolic neural networks to various other tasks, including image segmentation (Atigh et al., 2022), knowledge graph embedding (Chami et al., 2020), or image-text representation learning (Desai et al., 2023). Gyrovector space equipped Möbius addition and scalar multiplication (Ganea et al., 2018) have also been investigated to define the layer-to-layer transformation of fully connected hyperbolic networks. For unsupervised learning, VAEs with a wrapped Gaussian distribution have also been studied extensively (Nagano et al., 2019; Mathieu et al., 2019; Cho et al., 2022), and hyperbolic kernel methods have been explored for support vector machines (Fan et al., 2023). In addition, several PD kernels on the Poincaré ball have been presented in the literature (Peng et al., 2021; Yang et al., 2023).

Riemannian GP-LVM. The Matérn covariance on several curved manifolds has been studied previously (Borovitskiy et al., 2020), and the heat kernel in the Lie group was developed (Azangulov et al., 2022, 2023). To the best of our knowledge, only one hyperbolic extension of GP-LVMs with the hyperbolic heat kernel exists for motion taxonomy embedding in the latent space (Jaquier et al., 2022). Its objective includes sampling computation in the hyperbolic heat kernel and $\boldsymbol{\Psi}$ computation. This involves high time complexity by double sampling, and this method is intractable for general DR with several hundreds of data points. Furthermore, the manifold GP-LVM (mGP-LVM) (Jensen et al., 2020) employs the geodesic exponential kernel to represent the neural curvature on simple manifolds, e.g., tori, Spheres, and SO(3). However, the mGP-LVM involves mathematical difficulty in the computation of the posterior distribution, which restricts the applicability of the mGP-LVM to other smooth manifolds, e.g., hyperbolic spaces.

Table 1: Details of SBT dataset.

	Depth		
	$d = 4$	$d = 5$	$d = 6$
# samples	300	620	1,260
# dimensions	15	31	63

5 Experiments

In this section, we present experimental results to validate the proposed method. First, we compare the hGP-LVMs with Euclidean and hyperbolic generative models (Nagano et al., 2019; Cho et al., 2022). We discuss visualization-aided DR on the scRNA-seq dataset and confirm the effectiveness of our methods, especially the Bayesian estimation method.

5.1 Synthetic Binary Tree Dataset

In the following, we describe the experiment conducted using the *synthetic binary tree (SBT) dataset*, which has been used to evaluate hyperbolic generative models. This dataset is based on the SBT, exhibiting a simple hierarchical structure using binary codes. Figure 3 shows the SBT and the sampling process of the data points. First, we determine the *depth* of the tree, and then we generate binary codes with lengths of $(2^d - 1)$. We obtain the code of each node by changing the Boolean values according to the depth. In this study, by oscillating the codes, we generated datasets exhibiting SBT structures. We generated 20 samples at each node with $d = 4, 5, 6$. The statistical details of these datasets are shown in Table 1. Note that we used the most basic hGP-LVM and set $\kappa = 100$ in all datasets. In this evaluation, we compared the hGP-LVM with GP-LVM (Lawrence, 2005), the hyperbolic VAE with an isotropic hyperbolic wrapped Gaussian in Eq.(13) (IsoHVAE) (Nagano et al., 2019), and the hyperbolic VAE with a rotated hyperbolic wrapped Gaussian (RotHVAE) (Cho et al., 2022). We also used the implementation and hyperparameter settings presented in the literature (Cho et al., 2022) for the hyperbolic VAEs, and we only searched the number of hidden units. The latent dimension of all methods was set to 2, and we compared the methods through a quantitative evaluation and a qualitative visualization. In the quantitative evaluation, we used the distance

Table 2: **Quantitative results obtained on the SBT dataset.** The best results are highlighted in blue. Here, we computed the mean and standard deviation over 10 runs.

	Depth		
	$d = 4$	$d = 5$	$d = 6$
hGP-LVM **(ours)**	$0.816_{\pm 0.012}$	$0.909_{\pm 0.003}$	$0.849_{\pm 0.004}$
GP-LVM (Lawrence, 2005)	$0.782_{\pm 0.000}$	$0.746_{\pm 0.000}$	$0.717_{\pm 0.000}$
IsoHVAE (Nagano et al., 2019)	$0.890_{\pm 0.022}$	$0.867_{\pm 0.025}$	$0.815_{\pm 0.031}$
RotHVAE (Cho et al., 2022)	$0.896_{\pm 0.027}$	$0.872_{\pm 0.019}$	$0.823_{\pm 0.024}$

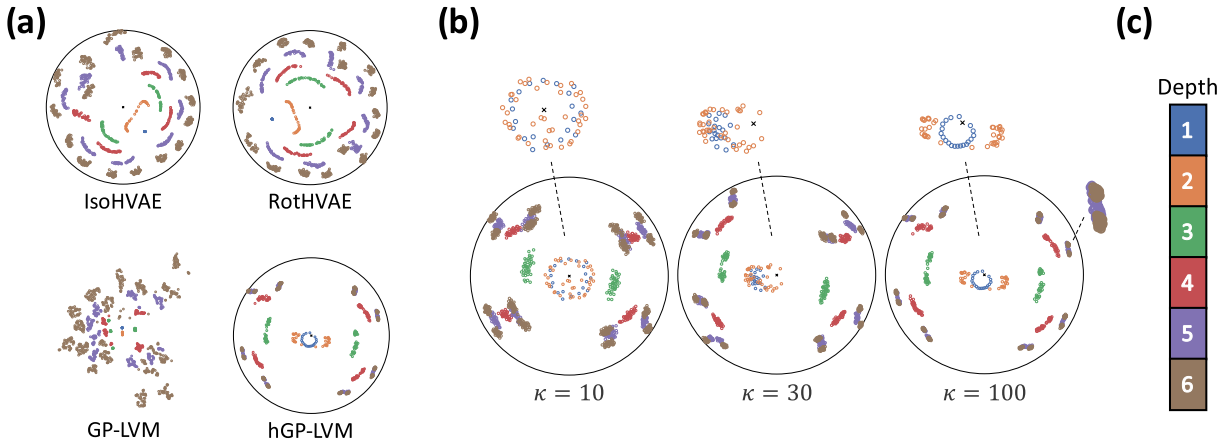

Figure 4: **Qualitative results obtained on the SBT dataset** ($d = 6$): (a) embedding comparison between generative models, (b) embedding of hGP-LVM with different length scales, and (c) color code of the embedding.

correlation score between the latent variables and the observed variables. In this study, the distance between the latent variables was computed using the hyperbolic metric for the hGP-LVM and the hyperbolic VAEs, and the Euclidean metric was computed for the GP-LVM. In addition, the distance between the observed variables was computed with the Hamming distance using the corresponding binary codes to each node.

The quantitative results are presented in Table 2. As can be seen, the results demonstrate that the hGP-LVM embedded the synthetic hierarchy with higher distance preservation quality than that of the Euclidean GP-LVM on all datasets. The priority of the hGP-LVM over the hyperbolic VAEs was also confirmed for cases where $d = 5$ and $d = 6$. Although the data size was limited, and more effective parameters may exist for the hyperbolic VAEs, we emphasize that the hGP-LVM did not require such ill-posed parameter tuning and could embed the hierarchy even with a limited amount of data. The visualization comparison is shown in Figure 4 (a). With the GP-LVM, it is difficult to find the hierarchical nature of the SBT dataset, and the embedding is spread with increasing depth. Although the embedding of the hyperbolic VAEs represented a hierarchical nature, the root nodes with depth 1 did not position around the origin, and it is difficult to interpret the tree structure in the SBT dataset. In contrast, the proposed hGP-LVM embedded the SBT's hierarchy holding internode similarity and placed the root nodes on the origin, which largely contributed to the visibility of the embedding. In summary, we validated the effectiveness of hGP-LVM toward generative models by conducting an experiment typically used for evaluating hyperbolic generative models.

Embedding Comparison with Different Length Scales As mentioned in Section 3.1, *the meaning of the length scale κ is how much we expect the latent variables to follow the hyperbolic curvature.* An experimental verification of this statement is shown in Figure 4 (b). Here, we confirm that the representation was spread and aligned as κ increased. Although the embeddings of depths 1 and 2 were mixed around the origin for $\kappa = 10$ and $\kappa = 30$, they were separated for $\kappa = 100$. The length scale parameter determined the range of the latent variables, and a large κ value effectively brought the strong hyperbolic curvature to the latent representation. Thus, the κ value must be determined according to the degree to which we expect

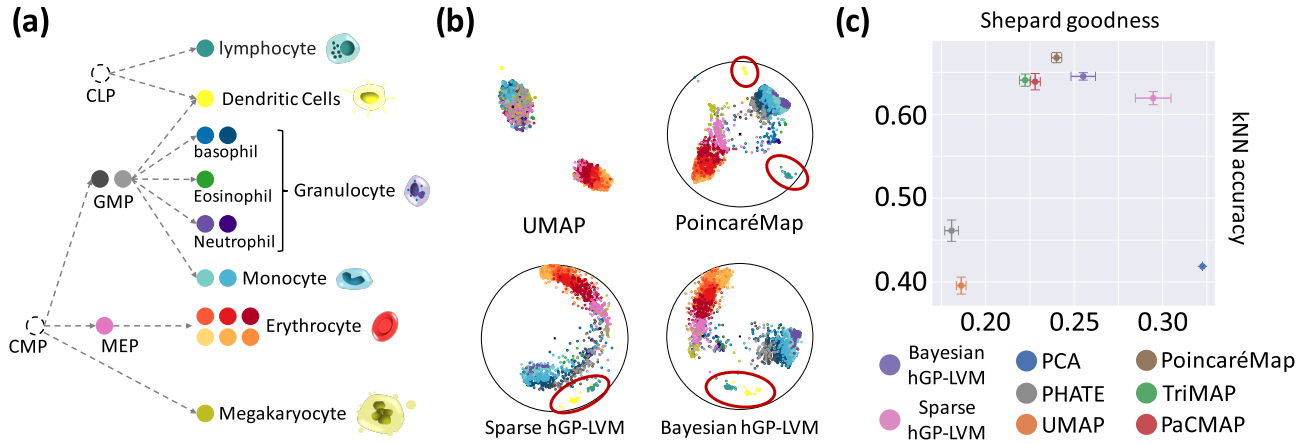

Figure 5: **Experimental results on the scRNA-seq dataset.** (a) The canonical hematopoietic cell lineage tree. (b) Two-dimensional embedding of UMAP, PoincaréMap, Sparse hGP-LVM, and Bayesian hGP-LVM. The colors correspond to those of the lineage tree. (c) The error bar plot of the compared methods. The same experiment was conducted 30 times, and we computed the mean error with the standard deviation.

the hierarchical structure in the data.

5.2 Mouse Myeloid Progenitors Dataset

We also conducted an experiment on the real-world scRNA-seq dataset presented in (Paul et al., 2015). The synthesis data results are shown in Appendix A.5 to verify the effectiveness of the GP-based visualization approach. This scRNA-seq dataset exhibits multiple intermediate populations of cell populations ($N = 2,730$) from the progenitors (CMP, GMP, MEP, and CLP) to the restricted myeloid cells, e.g., erythrocytes, leukocytes, and lymphocytes, representing the hierarchical structure along the cell differentiation. Figure 5 (a) shows the canonical hematopoietic cell lineage tree. Each cell's features is a bag of gene expressions, and they have labels annotated with the clusters identified in the original study. In the current study, we preprocessed the original scRNA-seq data by selecting $1,000$ highly-variable genes ($D = 1,000$) following the literature (Klimovskaia et al., 2020) and (Zheng et al., 2017). Note that the sample size is prohibitive for hGP-LVM, and we used sparse and Bayesian hGP-LVMs with $\kappa = 100$ and $M = 50$. We compared them with the UMAP and PoincaréMap methods qualitatively and further compared with PCA (Hotelling, 1933), PHATE (Moon et al., 2019), TriMAP (Amid and Warmuth, 2019), and PacMAP (Wang et al., 2021) quantitatively. We did not take GP-LVMs as compared methods since their learning did not work under the same conditions as hGP-LVM. In this evaluation, we employed the Shepard goodness to evaluate the quality of global preservation. However, the dataset was still noisy after preprocessing, and the local structure was not trustworthy; thus, to use trustworthy label information, we adopted the k-NN classification accuracy ($k = 5$) as the local metric.

Figure 5 (b) shows the visualization of the scRNA-seq dataset. As can be seen, UMAP's embedding was torn and mixed with the population of lymphocytes, dendric cells, granulocytes, and monocytes. In addition, the PoincaréMap embedding successfully separated the cell populations; however, the continuity starting from MEP and GMP was not evident, and there was a large distance between the population of lymphocytes and dendric cells, which are developed from the same progenitors, i.e., CLP. The proposed hGP-LVMs' embedding preserved the cell population and the continuity along the development, and the inter-population similarity of the CLP progenitors was preserved. In addition, we confirmed that the megakaryocyte population of the Bayesian hGP-LVM is more evident than the sparse hGP-LVM., which indicates that the uncertainty of the latent variables introduced by the full Bayesian inference is suitable for the noisy real-world dataset and facilitates effective embedding of the scRNA-seq dataset. Finally, the quantitative results are shown in Figure 5 (c). Recall the tradeoff relationship between local and global preservation. Both preservations are required to embed the hierarchical data, and the proposed hGP-LVMs exhibit relatively high results for both metrics.

In summary, the results have confirmed the effectiveness of the proposed hGP-LVMs against hyperbolic generative models and neighbor embedding methods. Although point estimation methods work efficiently for synthetic data, Bayesian inference is required for noisy real-world datasets. Thus, in consideration of the

tradeoff between time complexity and representation power, we must utilize such methods properly.

6 Conclusion

This paper has proposed the hyperbolic extension of GP-LVMs to realize the faithful low-dimensional embedding of the hierarchical data via nonparametric estimation. We established the HE kernel and incorporated the Riemannian optimization process with the previous sparse GP inference method. We have also introduced the reparameterization trick on the Lorentz model for a fully Bayesian estimation of the latent variables. The experiments conducted in this study validated the embedding accuracy of the proposed hGP-LVMs on synthetic hierarchical and real-world scRNA-seq datasets, and the results demonstrated the effectiveness of the GP-based modeling for visualization-aided DR.

Limitation and Future Work. We assume a hierarchical structure; however, if there is no clear hierarchical relation in a dataset, the proposed method cannot embed the observed data more effectively than general GP-LVM frameworks. In addition, the computational may be an issue because it is linear to the sample size and prohibitive for several huge datasets. The embedding quality of the neighbor embedding method scales with the sample size, and they are more effective for larger datasets.

Therefore, future studies will focus on improving the Bayesian extension's time complexity and approximated inference. In addition, improving the initial embedding is also important in terms of realizing more efficient learning for the GP-based latent variable models.

Acknowledgement

This study was supported in part by JSPS KAKENHI Grant Numbers JP24K02942, JP23K21676, JP23K11211, and JP24KJ0324.

References

Amid, E. and Warmuth, M. K. (2019). TriMap: Large-scale dimensionality reduction using triplets. *arXiv preprint arXiv:1910.00204*.

Atigh, M. G., Schoep, J., Acar, E., Van Noord, N., and Mettes, P. (2022). Hyperbolic image segmentation. In *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition*, pages 4453–4462.

Azangulov, I., Smolensky, A., Terenin, A., and Borovitskiy, V. (2022). Stationary kernels and Gaussian processes on Lie groups and their homogeneous spaces i: the compact case. *arXiv preprint arXiv:2208.14960*.

Azangulov, I., Smolensky, A., Terenin, A., and Borovitskiy, V. (2023). Stationary kernels and Gaussian processes on Lie groups and their homogeneous spaces ii: non-compact symmetric spaces. *arXiv preprint arXiv:2301.13088*.

Bauer, M., Van der Wilk, M., and Rasmussen, C. E. (2016). Understanding probabilistic sparse Gaussian process approximations. *Advances in Neural Information Processing Systems*, 29:1–9.

Becht, E., McInnes, L., Healy, J., Dutertre, C.-A., Kwok, I. W., Ng, L. G., Ginhoux, F., and Newell, E. W. (2019). Dimensionality reduction for visualizing single-cell data using UMAP. *Nature Biotechnology*, 37(1):38–44.

Borovitskiy, V., Terenin, A., Mostowsky, P., et al. (2020). Matérn Gaussian processes on Riemannian manifolds. *Advances in Neural Information Processing Systems*, 33:12426–12437.

Chami, I., Wolf, A., Juan, D.-C., Sala, F., Ravi, S., and Ré, C. (2020). Low-dimensional hyperbolic knowledge graph embeddings. *arXiv preprint arXiv:2005.00545*.

Cho, S., Lee, J., Park, J., and Kim, D. (2022). A rotated hyperbolic wrapped normal distribution for hierarchical representation learning. *Advances in Neural Information Processing Systems*, 35:17831–17843.

de Souza, D., Mesquita, D., Gomes, J. P., and Mattos, C. L. (2021). Learning GPLVM with arbitrary kernels using the unscented transformation. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, pages 451–459.

Desai, K., Nickel, M., Rajpurohit, T., Johnson, J., and Vedantam, S. R. (2023). Hyperbolic image-text representations. In *Proceedings of the International Conference on Machine Learning*, pages 7694–7731.

Fan, X., Yang, C.-H., and Vemuri, B. (2023). Horospherical decision boundaries for large margin classification in hyperbolic space. *Advances in Neural Information Processing Systems*, 36:1–11.

Fang, P., Harandi, M., and Petersson, L. (2021). Kernel methods in hyperbolic spaces. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10665–10674.

Feragen, A., Lauze, F., and Hauberg, S. (2015). Geodesic exponential kernels: When curvature and linearity conflict. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3032–3042.

Ganea, O., Bécigneul, G., and Hofmann, T. (2018). Hyperbolic neural networks. *Advances in Neural Information Processing Systems*, 31:1–11.

Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., and Lerchner, A. (2017). beta-VAE: Learning basic visual concepts with a constrained variational framework. In *Proceedings of the International Conference on Learning Representations*, pages 1–22.

Hotelling, H. (1933). Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, 24(6):417–441.

Istas, J. (2012). Manifold indexed fractional fields*. *ESAIM: Probability and Statistics*, 16:222–276.

Jaquier, N., Rozo, L., González-Duque, M., Borovitskiy, V., and Asfour, T. (2022). Bringing robotics taxonomies to continuous domains via GPLVM on hyperbolic manifolds. *arXiv preprint arXiv:2210.01672*.

Jensen, K., Kao, T.-C., Tripodi, M., and Hennequin, G. (2020). Manifold GPLVMs for discovering non-Euclidean latent structure in neural data. *Advances in Neural Information Processing Systems*, 33:22580–22592.

Kingma, D. P. and Welling, M. (2013). Auto-encoding variational Bayes. *arXiv preprint arXiv:1312.6114*.

Klimovskaia, A., Lopez-Paz, D., Bottou, L., and Nickel, M. (2020). Poincaré maps for analyzing complex hierarchies in single-cell data. *Nature Communications*, 11(1):2966.

Kobak, D. and Berens, P. (2019). The art of using t-SNE for single-cell transcriptomics. *Nature Communications*, 10(1):5416.

Lalchand, V., Ravuri, A., and Lawrence, N. D. (2022). Generalised GPLVM with stochastic variational inference. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, pages 7841–7864.

Lawrence, N. (2005). Probabilistic non-linear principal component analysis with Gaussian process latent variable models. *Journal of Machine Learning Research*, 6(11):1–34.

Lawrence, N. D. (2007). Learning for larger datasets with the Gaussian process latent variable model. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, pages 243–250.

Luecken, M. D. and Theis, F. J. (2019). Current best practices in single-cell RNA-seq analysis: a tutorial. *Molecular Systems Biology*, 15(6):e8746.

Mallasto, A. and Feragen, A. (2018). Wrapped Gaussian process regression on Riemannian manifolds. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5580–5588.

Mathieu, E., Le Lan, C., Maddison, C. J., Tomioka, R., and Teh, Y. W. (2019). Continuous hierarchical representations with Poincaré variational auto-encoders. *Advances in Neural Information Processing Systems*, 32:1–12.

McInnes, L., Healy, J., and Melville, J. (2018). UMAP: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*.

Moon, K. R., Van Dijk, D., Wang, Z., Gigante, S., Burkhardt, D. B., Chen, W. S., Yim, K., Elzen, A. v. d., Hirn, M. J., Coifman, R. R., et al. (2019). Visualizing structure and transitions in high-dimensional biological data. *Nature Biotechnology*, 37(12):1482–1492.

Moreno-Muñoz, P., Feldager, C., and Hauberg, S. (2022). Revisiting active sets for Gaussian process decoders. *Advances in Neural Information Processing Systems*, 35:6603–6614.

Nagano, Y., Yamaguchi, S., Fujita, Y., and Koyama, M. (2019). A wrapped normal distribution on hyperbolic space for gradient-based learning. In *Proceedings of the International Conference on Machine Learning*, pages 4693–4702.

Nickel, M. and Kiela, D. (2017). Poincaré embeddings for learning hierarchical representations. *Advances in Neural Information Processing Systems*, 30:1–10.

Nickel, M. and Kiela, D. (2018). Learning continuous hierarchies in the Lorentz model of hyperbolic geometry. In *Proceedings of the International Conference on Machine Learning*, pages 3779–3788.

Niu, M., Dai, Z., Cheung, P., and Wang, Y. (2023). Intrinsic Gaussian process on unknown manifolds with probabilistic metrics. *Journal of Machine Learning Research*, 24(104):1–42.

Paul, F., Arkin, Y., Giladi, A., Jaitin, D. A., Kenigsberg, E., Keren-Shaul, H., Winter, D., Lara-Astiaso, D., Gury, M., Weiner, A., et al. (2015). Transcriptional heterogeneity and lineage commitment in myeloid progenitors. *Cell*, 163(7):1663–1677.

Peng, W., Varanka, T., Mostafa, A., Shi, H., and Zhao, G. (2021). Hyperbolic deep neural networks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(12):10023–10044.

Rasmussen, C. E. and Williams, C. K. (2006). *Gaussian Processes for Machine Learning*. MIT Press.

Sala, F., De Sa, C., Gu, A., and Ré, C. (2018). Representation tradeoffs for hyperbolic embeddings. In *Proceedings of the International Conference on Machine Learning*, pages 4460–4469.

Salimbeni, H. and Deisenroth, M. (2017). Doubly stochastic variational inference for deep Gaussian

processes. *Advances in Neural Information Processing Systems*, 30:1–12.

Titsias, M. (2009). Variational learning of inducing variables in sparse Gaussian processes. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, pages 567–574.

Titsias, M. and Lawrence, N. D. (2010). Bayesian Gaussian process latent variable model. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, pages 844–851.

Van der Maaten, L. and Hinton, G. (2008). Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9(11):1–25.

Wang, Y., Huang, H., Rudin, C., and Shaposhnik, Y. (2021). Understanding how dimension reduction tools work: an empirical approach to deciphering t-SNE, UMAP, TriMAP, and PaCMAP for data visualization. *Journal of Machine Learning Research*, 22(201):1–73.

Yang, M., Fang, P., and Xue, H. (2023). Expanding the hyperbolic kernels: a curvature-aware isometric embedding view. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 4469–4477.

Zheng, G. X., Terry, J. M., Belgrader, P., Ryvkin, P., Bent, Z. W., Wilson, R., Ziraldo, S. B., Wheeler, T. D., McDermott, G. P., Zhu, J., et al. (2017). Massively parallel digital transcriptional profiling of single cells. *Nature Communications*, 8(1):14049.

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] In Section 2 and Appendix A.1.
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Yes] In the last paragraph in Section 3.2.
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Yes] In the supplemental material.

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [Yes] In Section 3.2.
   (b) Complete proofs of all theoretical results. [Yes] In Appendix A.2.
   (c) Clear explanations of any assumptions. [Yes] In Section 2 and 3.2.

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] In the supplemental material.
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes] In the first paragraphs in Section 5.1, Section 5.2, and Appendix A.4.
   (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] In the experimental results in Section 5 and Appendix A.5.
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes] In Appendix A.4.

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] In Appendix A.4.
   (b) The license information of the assets, if applicable. [Yes] In Appendix A.4.
   (c) New assets either in the supplemental material or as a URL, if applicable. [Yes] In the supplemental material.
   (d) Information about consent from data providers/curators. [Yes] We use open-source datasets.
   (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]

Hyperboloid GPLVM for Discovering Continuous Hierarchies via Nonparametric Estimation: Supplementary Materials

A.7 Derivation of Objectives of GP-LVMs

We first recall the GP-LVM model definition as

$$\mathbf{y}_{:,d} = \mathbf{f}_d(\mathbf{X}) + \boldsymbol{\epsilon},\ \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \beta^{-1}\mathbf{I}_n),\ \mathbf{f}_d \sim \mathcal{GP}(\mathbf{0}, k(\cdot,\cdot)). \tag{A.1}$$

The equal expression of Eq. (A.1) as the probabilistic density function is given by

$$\begin{aligned} p(\mathbf{y}_{:,d}|\mathbf{f}_d) &= \mathcal{N}(\mathbf{y}_{:,d}|\mathbf{f}_d,\ \beta^{-1}\mathbf{I}_n), \\ p(\mathbf{f}_d|\mathbf{X}) &= \mathcal{N}(\mathbf{f}_d|\mathbf{0},\ \mathbf{K}_{nn}). \end{aligned}$$

Then, we derive the objective log-likelihood function of GP-LVM by marginalizing the GP prior $\mathbf{f}_d$ as follows:

$$\begin{aligned} p(\mathbf{Y}|\mathbf{X}) &= \prod_{d=1}^{D} \int p(\mathbf{y}_{:,d}|\mathbf{f}_d) p(\mathbf{f}_d|\mathbf{X}) d\mathbf{f}_d \\ &= -\frac{ND}{2}\log 2\pi - \frac{D}{2}\log|\mathbf{K}_{nn} + \beta^{-1}\mathbf{I}_n| - \frac{1}{2}\mathrm{tr}\left[(\mathbf{K}_{nn} + \beta^{-1}\mathbf{I}_n)^{-1}\mathbf{Y}\mathbf{Y}^\top\right]. \end{aligned} \tag{A.2}$$

The practical problems of the GP-LVM objective in Eq. (A.2) are the computation of the $N \times N$ matrix inversion of $(\mathbf{K}_{nn} + \beta^{-1}\mathbf{I}_n)$ and log-determinant $\log|\mathbf{K}_{nn} + \beta^{-1}\mathbf{I}_n|$, which are compressed into the Cholesky decomposition of $(\mathbf{K}_{nn} + \beta^{-1}\mathbf{I}_n)$ with $O(\frac{1}{3}N^3)$ time complexity.

Next, we adopt the inducing method to derive the sparse GP-LVM objective. Formally, we assume the inducing points $\mathbf{u}_d$, which are sufficient statistics of the prior $\mathbf{f}_d$. Therefore, the joint distribution omitted $\mathbf{X}$ and $\mathbf{Z}$ are given by

$$p(\mathbf{y}_{:,d},\ \mathbf{f}_d,\ \mathbf{u}_d) = p(\mathbf{y}_{:,d}|\mathbf{f}_d,\ \mathbf{u}_d)p(\mathbf{f}_d|\mathbf{u}_d)p(\mathbf{u}_d) \approx p(\mathbf{y}_{:,d}|\mathbf{u}_d)p(\mathbf{f}_d|\mathbf{u}_d)p(\mathbf{u}_d).$$

The marginal log-likelihood of the sparse GP model is given by

$$p(\mathbf{Y}|\mathbf{X},\ \mathbf{Z}) = \prod_{d=1}^{D} \int p(\mathbf{y}_{:,d}|\mathbf{u}_d,\ \mathbf{X},\ \mathbf{Z}) p(\mathbf{u}_d|\mathbf{Z}) d\mathbf{u}_d, \tag{A.3}$$

where

$$\begin{aligned} p(\mathbf{y}_{:,d}|\mathbf{f}_d) &= \mathcal{N}(\mathbf{y}_{:,d}|\mathbf{f}_d,\ \beta^{-1}\mathbf{I}_n), \\ p(\mathbf{f}_d|\mathbf{u}_d,\ \mathbf{X},\ \mathbf{Z}) &= \mathcal{N}(\mathbf{f}_d|\mathbf{K}_{nm}\mathbf{K}_{mm}^{-1}\mathbf{u}_d,\ \mathbf{K}_{nn} - \mathbf{K}_{nm}\mathbf{K}_{mm}^{-1}\mathbf{K}_{mn}), \\ p(\mathbf{y}_{:,d}|\mathbf{u}_d,\ \mathbf{X},\ \mathbf{Z}) &= \mathcal{N}(\mathbf{y}_{:,d}|\mathbf{K}_{nm}\mathbf{K}_{mm}^{-1}\mathbf{u}_d,\ \mathbf{K}_{nn} - \mathbf{K}_{nm}\mathbf{K}_{mm}^{-1}\mathbf{K}_{mn} + \beta^{-1}\mathbf{I}_n), \\ p(\mathbf{u}_d|\mathbf{Z}) &= \mathcal{N}(\mathbf{f}_d|\mathbf{0},\ \mathbf{K}_{mm}). \end{aligned}$$

Then, we apply the variational method with Jensen's inequality and evaluate the lower bound of the log-likelihood function as

$$\begin{aligned} \log p(\mathbf{y}_{:,d}|\mathbf{u}_d,\ \mathbf{X},\ \mathbf{Z}) &\geq \mathbb{E}_{p(\mathbf{f}_d|\mathbf{u}_d,\ \mathbf{X},\ \mathbf{Z})}\left[\log p(\mathbf{y}_{:,d}|\mathbf{f}_d)\right] \\ &\triangleq \mathcal{F}_1. \end{aligned}$$

Thus, we obtain the following lower bound of the log-likelihood as

$$\begin{aligned}\log p(\mathbf{Y}|\mathbf{X},\,\mathbf{Z}) &= \log \prod_{d=1}^{D} \int \exp\left[\log p(\mathbf{y}_{:,d}|\mathbf{u}_d,\,\mathbf{X},\,\mathbf{Z})\right] p(\mathbf{u}_{:,d}|\mathbf{Z}) d\mathbf{u}_d \\ &\geq \sum_{d=1}^{D} \log \mathcal{N}(\mathbf{y}_{:,d}|\mathbf{0},\,\mathbf{K}_{nm}\mathbf{K}_{mm}^{-1}\mathbf{K}_{mn} + \beta^{-1}\mathbf{I}_n) - \frac{D}{2}\mathrm{tr}\left(\mathbf{K}_{nn} - \mathbf{K}_{nm}\mathbf{K}_{mm}^{-1}\mathbf{K}_{mn}\right) \\ &= -\frac{D}{2}\log|\mathbf{Q}_{nn} + \beta^{-1}\mathbf{I}_n| - \frac{1}{2}\mathrm{tr}\left[(\mathbf{Q}_{nn} + \beta^{-1}\mathbf{I}_n)^{-1}\mathbf{Y}\mathbf{Y}^\top\right] - \frac{\beta D}{2}\mathrm{tr}(\mathbf{K}_{nn} - \mathbf{Q}_{nn}), \end{aligned} \tag{A.4}$$

where $\mathbf{Q}_{nn} = \mathbf{K}_{nm}\mathbf{K}_{mm}^{-1}\mathbf{K}_{mn}$. Here, we extend Eq. (A.4) by applying the matrix determinant and inversion lemmas as

$$|\mathbf{K}_{nm}\mathbf{K}_{mm}^{-1}\mathbf{K}_{mn} + \beta^{-1}\mathbf{I}_n| = |\mathbf{K}_{mm}||\mathbf{K}_{mm} + \beta\mathbf{K}_{mn}\mathbf{K}_{nm}|, \tag{A.5}$$

$$(\mathbf{K}_{nm}\mathbf{K}_{mm}^{-1}\mathbf{K}_{mn} + \beta^{-1}\mathbf{I}_n)^{-1} = \beta\mathbf{I}_n - \beta^2\mathbf{K}_{nm}(\mathbf{K}_{mm} + \beta\mathbf{K}_{mn}\mathbf{K}_{nm})^{-1}\mathbf{K}_{mn}. \tag{A.6}$$

Then, we can obtain the sparse GP-LVM objectives by substituting Eqns. (A.5) and (A.6) into Eq. (A.4) as

$$\acute{\mathcal{F}} = -\frac{D}{2}\log\frac{(2\pi)^N|\mathbf{A}|}{\beta^N|\mathbf{K}_{mm}|} - \frac{1}{2}\mathrm{tr}\left(\mathbf{W}\mathbf{Y}\mathbf{Y}^\top\right) - \frac{\beta D}{2}\mathrm{tr}\left(\mathbf{K}_{nn}\right) + \frac{\beta D}{2}\left(\mathbf{K}_{mm}^{-1}\mathbf{K}_{mn}\mathbf{K}_{nm}\right),$$

where $\mathbf{W} = \beta\mathbf{I}_n - \beta^2\mathbf{K}_{nm}\mathbf{A}^{-1}\mathbf{K}_{mn}$ and $\mathbf{A} = \mathbf{K}_{mm} + \beta\mathbf{K}_{mn}\mathbf{K}_{nm}$.

Finally, we derive the Bayesian GP-LVM objective. We first derive the lower bound straightforwardly as

$$\begin{aligned}&\log p(\mathbf{Y}|\mathbf{Z}) \\ &= \sum_{d=1}^{D} \log \int \left\{ \int p(\mathbf{y}_{:,d}|\mathbf{u}_d,\,\mathbf{X},\,\mathbf{Z}) p(\mathbf{u}_d)\mathrm{d}\mathbf{u}_d \right\} p(\mathbf{X})\mathrm{d}\mathbf{X} \\ &= \sum_{d=1}^{D} \log \int\int \exp\left[\log\left\{p(\mathbf{y}_{:,d}|\mathbf{u}_d,\,\mathbf{X},\,\mathbf{Z})\frac{p(\mathbf{X})}{q(\mathbf{X})}\right\}\right] q(\mathbf{X})p(\mathbf{u}_d)d\mathbf{X}\mathrm{d}\mathbf{u}_d \\ &\geq \sum_{d=1}^{D} \log \int \exp\left[\mathbb{E}_{q(\mathbf{X})}\left[\log p(\mathbf{y}_{:,d}|\mathbf{u}_d,\,\mathbf{X},\,\mathbf{Z})\right] + \mathbb{E}_{q(\mathbf{X})}\left[\log\frac{p(\mathbf{X})}{q(\mathbf{X})}\right]\right] p(\mathbf{u}_d)\mathrm{d}\mathbf{u}_d \\ &= \sum_{d=1}^{D} \log \int \exp\left[\mathbb{E}_{q(\mathbf{X})}\left[\log p(\mathbf{y}_{:,d}|\mathbf{u}_d,\,\mathbf{X},\,\mathbf{Z})\right]\right] p(\mathbf{u}_d)d\mathbf{u}_d - \mathrm{KL}[q(\mathbf{X})||p(\mathbf{X})] \\ &\geq \sum_{d=1}^{D} \log \int \exp\left[\mathbb{E}_{q(\mathbf{X})}\left[\mathcal{F}_1\right]\right] p(\mathbf{u}_d)d\mathbf{u}_d - \mathrm{KL}[q(\mathbf{X})||p(\mathbf{X})]. \end{aligned} \tag{A.7}$$

The expectation of $\mathcal{F}_1$ in Eq. (A.7) can be computed in the closed form as

$$\begin{aligned}\mathbb{E}_{q(\mathbf{X})}\left[\mathcal{F}_1\right] = &-\frac{\beta}{2}\mathbf{y}_{:,d}^\top\mathbf{y}_{:,d} + \beta\mathbf{y}_{:,d}^\top\mathbf{\Psi}_1\mathbf{K}_{mm}^{-1}\mathbf{u}_d - \frac{\beta}{2}\mathbf{u}_d^\top\mathbf{K}_{mm}^{-1}\mathbf{\Psi}_2\mathbf{K}_{mm}^{-1}\mathbf{u}_d \\ &- \frac{\beta}{2}\psi_0 + \frac{\beta}{2}\mathrm{tr}\left(\mathbf{K}_{mm}^{-1}\mathbf{\Psi}_2\right) - \frac{N}{2}\log 2\pi\beta^{-1}. \end{aligned} \tag{A.8}$$

By substituting Eq. (A.8) into the first term of Eq. (A.7), we obtain the following equation:

$$\begin{aligned}&\sum_{d=1}^{D} \log \int \exp\left[\mathbb{E}_{q(\mathbf{X})}\left[\mathcal{F}_1\right]\right] p(\mathbf{u}_d)d\mathbf{u}_d \\ &= \sum_{d=1}^{D} \log\left[\frac{\sqrt{\beta^N|\mathbf{K}_{MM}|}}{\sqrt{(2\pi)^N|\beta\mathbf{\Psi}_2 + \mathbf{K}_{MM}|}} \exp\left\{-\frac{1}{2}\mathbf{y}_{:,d}^\top\mathbf{W}_b\mathbf{y}_{:,d} - \frac{\beta}{2}\psi_0 + \frac{\beta}{2}\mathrm{tr}\left(\mathbf{K}_{mm}^{-1}\mathbf{\Psi}_2\right)\right\}\right], \end{aligned} \tag{A.9}$$

where $\psi_0 = \mathrm{tr}\left[\mathbf{K}_{nn}\right]$, $\mathbf{W}_b = \beta\mathbf{I}_n - \beta^2\mathbf{\Psi}_1^\top\mathbf{A}_b\mathbf{\Psi}_1$ and $\mathbf{A}_b^{-1} = \mathbf{K}_{mm} + \beta\mathbf{\Psi}_2$. Finally, we expand Eq. (A.7) as

$$\acute{\mathcal{F}}_b = -\frac{D}{2}\log\frac{(2\pi)^N|\mathbf{A}_b|}{\beta^N|\mathbf{K}_{mm}|} - \frac{1}{2}\mathrm{tr}\left(\mathbf{W}_b\mathbf{Y}\mathbf{Y}^\top\right) - \frac{\beta D}{2}\psi_0 + \frac{\beta D}{2}\mathrm{tr}\left(\mathbf{K}_{mm}^{-1}\mathbf{\Psi}_2\right) - \sum_{i=1}^{N}\mathrm{KL}_i. \tag{A.10}$$

A.8 Proof of Lemma 1

Recall that the Lemma is $d_{\mathcal{L}^Q}(\mathbf{x}'_1, \mathbf{x}'_2) \approx d_E(\mathbf{x}_1, \mathbf{x}_2) + O(d_E(\mathbf{x}_1, \mathbf{x}_2)^3)$ if $||\mathbf{x}_1||_2 \approx 0$ and $||\mathbf{x}_2||_2 \approx 0$.

Proof. We use the following expansion:

$$\sqrt{1+z} = 1 + \frac{1}{2}z - \frac{1}{8}z^2 + \cdots,$$
$$\cosh^{-1}(z) = \sqrt{2(z-1)}\{1 - \frac{1}{12}(z-1) + \frac{3}{160}(z-1)^2 + \cdots\}.$$

The first equation is the Taylor expansion of $\sqrt{1+z}$ around $z=0$, and the second equation is derived from the Puiseux expansion of $\frac{\cosh^{-1}(z)}{\sqrt{2(z-1)}}$ around $z=1$ (Sloane, 2007). We first expand the Lorentzian inner product by substituting the Taylor expansion as

$$\begin{aligned}
-\langle \mathbf{x}_1, \mathbf{x}_2 \rangle_{\mathcal{L}^Q} &= \sqrt{1+||\mathbf{x}_1||_2^2}\sqrt{1+||\mathbf{x}_2||_2^2} - \mathbf{x}_1^\top \mathbf{x}_2 \\
&= \left(1 + \frac{1}{2}||\mathbf{x}_1||_2^2 - \frac{1}{8}||\mathbf{x}_1||_2^4 + \cdots\right)\left(1 + \frac{1}{2}||\mathbf{x}_2||_2^2 - \frac{1}{8}||\mathbf{x}_2||_4^2 + \cdots\right) - \mathbf{x}_1^\top \mathbf{x}_2 \\
&\approx 1 + \frac{1}{2}||\mathbf{x}_1||_2^2 + \frac{1}{2}||\mathbf{x}_2||_2^2 - \mathbf{x}_1^\top \mathbf{x}_2 \\
&= 1 + \frac{1}{2}||\mathbf{x}_1 - \mathbf{x}_2||_2^2 \\
&= 1 + \frac{1}{2}d_E(\mathbf{x}_1, \mathbf{x}_2)^2.
\end{aligned}$$

and then we obtain

$$\begin{aligned}
d_{\mathcal{L}^Q}(\mathbf{x}_1, \mathbf{x}_2) &= \cosh^{-1}(-\langle \mathbf{x}_1, \mathbf{x}_2 \rangle_{\mathcal{L}^Q}) \\
&\approx \cosh^{-1}\left(1 + \frac{1}{2}d_E(\mathbf{x}_1, \mathbf{x}_2)^2\right) \\
&= d_E(\mathbf{x}_1, \mathbf{x}_2)\left\{1 - \frac{1}{24}d_E(\mathbf{x}_1, \mathbf{x}_2)^2 + \frac{3}{640}d_E(\mathbf{x}_1, \mathbf{x}_2)^4 + \cdots\right\} \\
&= d_E(\mathbf{x}_1, \mathbf{x}_2) + O\left(d_E(\mathbf{x}_1, \mathbf{x}_2)^3\right).
\end{aligned}$$

□

A.9 Differentiation of hGP-LVMs

We differentiate objectives to realize the gradient-based optimization. We use the chain rules similar to the previous GP-LVM and derive the differentiation w.r.t. gram matrices. Recall the objectives of GP-LVM $\mathcal{F}$, Sparse GP-LVM $\acute{\mathcal{F}}$, and Bayesian GP-LVM $\acute{\mathcal{F}}_b$ as

$$\begin{aligned}
\mathcal{F} &= -\frac{ND}{2}\log 2\pi - \frac{D}{2}\log|\mathbf{K}_{nn} + \beta^{-1}\mathbf{I}_n| - \frac{1}{2}\mathrm{tr}\left[(\mathbf{K}_{nn} + \beta^{-1}\mathbf{I}_n)^{-1}\mathbf{Y}\mathbf{Y}^\top\right], \\
\acute{\mathcal{F}} &= -\frac{D}{2}\log\frac{(2\pi)^N|\mathbf{A}|}{\beta^N|\mathbf{K}_{mm}|} - \frac{1}{2}\mathrm{tr}\left(\mathbf{W}\mathbf{Y}\mathbf{Y}^\top\right) - \frac{\beta D}{2}\mathrm{tr}\left(\mathbf{K}_{nn}\right) + \frac{\beta D}{2}\left(\mathbf{K}_{mm}^{-1}\mathbf{K}_{mn}\mathbf{K}_{nm}\right), \\
\acute{\mathcal{F}}_b &= -\frac{D}{2}\log\frac{(2\pi)^N|\mathbf{A}_b|}{\beta^N|\mathbf{K}_{mm}|} - \frac{1}{2}\mathrm{tr}\left(\mathbf{W}_b\mathbf{Y}\mathbf{Y}^\top\right) - \frac{\beta D}{2}\mathrm{tr}\left(\mathbf{K}_{nn}\right) + \frac{\beta D}{2}\mathrm{tr}\left(\mathbf{K}_{mm}^{-1}\mathbf{\Psi}_2\right) - \sum_{i=1}^{N}\mathrm{KL}_i.
\end{aligned}$$

where $\mathbf{A} = \mathbf{K}_{mm} + \beta\mathbf{K}_{mn}\mathbf{K}_{nm}$, $\mathbf{W} = \beta\mathbf{I}_n - \beta^2\mathbf{K}_{nm}\mathbf{A}^{-1}\mathbf{K}_{mn}$, $\mathbf{A}_b = \mathbf{K}_{mm} + \beta\mathbf{\Psi}_2$, $\mathbf{W}_b = \beta\mathbf{I}_n - \beta^2\mathbf{\Psi}_1^\top\mathbf{A}_b^{-1}\mathbf{\Psi}_1$, $\mathbf{\Psi}_1 = \mathbb{E}_{q(\mathbf{X})}[\mathbf{K}_{mn}]$, $\mathbf{\Psi}_2 = \mathbb{E}_{q(\mathbf{X})}[\mathbf{K}_{mn}\mathbf{K}_{nm}]$, and $\mathrm{KL}_i = \mathrm{KL}[q(\mathbf{x}_i)||p(\mathbf{x}_i)]$.

A.9.1 Differentiation w.r.t. Gram Matrices

GP-LVM The gram matrix in GP-LVM in Eq. (A.2) is $\mathbf{K}_{nn}$. The differentiation w.r.t. $\mathbf{K}_{nn}$ is given as follows:

$$\frac{\partial \mathcal{F}}{\partial \mathbf{K}_{nn}} = -\frac{D}{2}(\mathbf{K}_{nn} + \beta^{-1}\mathbf{I}_n)^{-1} + \frac{1}{2}(\mathbf{K}_{nn} + \beta^{-1}\mathbf{I}_n)^{-1}\mathbf{Y}\mathbf{Y}^\top(\mathbf{K}_{nn} + \beta^{-1}\mathbf{I}_n)^{-1}. \tag{A.11}$$

Sparse GP-LVM The gram matrices in sparse GP-LVM are $\mathbf{K}_{mn}$ and $\mathbf{K}_{mm}$. The differentiation w.r.t. $\mathbf{K}_{mn}$ and $\mathbf{K}_{mm}$ is given as follows:

$$\frac{\partial \acute{\mathcal{F}}}{\partial \mathbf{K}_{mn}} = -\beta D\mathbf{A}^{-1}\mathbf{K}_{mn} + \beta^2\mathbf{A}^{-1}\mathbf{K}_{mn}\mathbf{Y}\mathbf{Y}^\top - \beta^3\mathbf{A}^{-1}\mathbf{K}_{mn}\mathbf{Y}\mathbf{Y}^\top\mathbf{K}_{nm}\mathbf{A}^{-1}\mathbf{K}_{mn} + \beta D\mathbf{K}_{mm}^{-1}\mathbf{K}_{mn}, \tag{A.12}$$

$$\frac{\partial \acute{\mathcal{F}}}{\partial \mathbf{K}_{mm}} = \frac{D}{2}\mathbf{K}_{mm}^{-1} - \frac{D}{2}\mathbf{A}^{-1} - \frac{\beta^2}{2}\mathbf{A}^{-1}\mathbf{K}_{mn}\mathbf{Y}\mathbf{Y}^\top\mathbf{K}_{nm}\mathbf{A}^{-1} - \frac{\beta D}{2}\mathbf{K}_{mm}^{-1}\mathbf{K}_{mn}\mathbf{K}_{nm}\mathbf{K}_{mm}^{-1}. \tag{A.13}$$

Bayesian GP-LVM The gram matrices in Bayesian GP-LVM is $\mathbf{\Psi}_1$, $\mathbf{\Psi}_2$, and $\mathbf{K}_{mm}$. The differentiation w.r.t. $\mathbf{\Psi}_1$, $\mathbf{\Psi}_2$, and $\mathbf{K}_{mm}$ is given as follows:

$$\frac{\partial \acute{\mathcal{F}}_b}{\partial \mathbf{\Psi}_1} = \beta^2\mathbf{A}_b\mathbf{\Psi}_1\mathbf{Y}\mathbf{Y}^\top, \tag{A.14}$$

$$\frac{\partial \acute{\mathcal{F}}_b}{\partial \mathbf{\Psi}_2} = -\frac{\beta D}{2}\mathbf{A}_b^{-1} - \frac{\beta^3}{2}\mathbf{A}_b^{-1}\mathbf{\Psi}_1\mathbf{Y}\mathbf{Y}^\top\mathbf{\Psi}_1^\top\mathbf{A}_b^{-1} + \frac{\beta D}{2}\mathbf{K}_{mm}^{-1}, \tag{A.15}$$

$$\frac{\partial \acute{\mathcal{F}}_b}{\partial \mathbf{K}_{mm}} = \frac{D}{2}\mathbf{K}_{mm}^{-1} - \frac{D}{2}\mathbf{A}_b^{-1} - \frac{\beta^2}{2}\mathbf{A}_b^{-1}\mathbf{\Psi}_1\mathbf{Y}\mathbf{Y}^\top\mathbf{\Psi}_1^\top\mathbf{A}_b^{-1} - \frac{\beta D}{2}\mathbf{K}_{mm}^{-1}\mathbf{\Psi}_2\mathbf{K}_{mm}^{-1}. \tag{A.16}$$

We can compute the gradient w.r.t. kernel parameters as

$$\frac{\partial \mathcal{F}}{\partial \sigma} = \frac{1}{\sigma}\mathrm{tr}\left(\frac{\partial \mathcal{F}}{\partial \mathbf{K}_{nn}}\right), \tag{A.17}$$

$$\frac{\partial \acute{\mathcal{F}}}{\partial \sigma} = \frac{1}{\sigma}\mathrm{tr}\left(\frac{\partial \acute{\mathcal{F}}}{\partial \mathbf{K}_{mn}} + \frac{\partial \acute{\mathcal{F}}}{\partial \mathbf{K}_{mm}}\right), \tag{A.18}$$

$$\frac{\partial \acute{\mathcal{F}}_b}{\partial \sigma} = \frac{1}{\sigma}\mathrm{tr}\left(\frac{\partial \acute{\mathcal{F}}_b}{\partial \mathbf{\Psi}_1} + 2\frac{\partial \acute{\mathcal{F}}_b}{\partial \mathbf{\Psi}_2}\right). \tag{A.19}$$

A.9.2 Differentiation w.r.t. Latent Variables

Next, we derive the differentiation w.r.t. latent variables. In GP-LVM and sparse GP-LVM, the differentiation can be derived by using the chain rule as

$$\frac{\partial \mathcal{F}}{\partial x_{iq}} = \mathrm{tr}\left(\frac{\partial \mathcal{F}}{\partial \mathbf{K}_{nn}}\frac{\partial \mathbf{K}_{nn}}{\partial x_{iq}}\right), \ \frac{\partial \acute{\mathcal{F}}}{\partial x_{iq}} = \mathrm{tr}\left(\frac{\partial \mathcal{F}}{\partial \mathbf{K}_{mn}}\frac{\partial \mathbf{K}_{mn}}{\partial x_{iq}}\right). \tag{A.20}$$

We derive the differentiation of the gram matrix w.r.t. latent variables by entry-wise computation. The differentiation of $[\mathbf{K}_{nn}]_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$ and $[\mathbf{K}_{mn}]_{kj} = k(\mathbf{z}_k, \mathbf{x}_j)$ w.r.t. latent variables is given as follows:

$$\frac{\partial k_{\mathcal{L}^Q}(\mathbf{x}_i, \mathbf{x}_j)}{\partial x_{iq}} = \frac{k_{\mathcal{L}^Q}(\mathbf{x}_i, \mathbf{x}_j)}{\kappa\sqrt{\langle \mathbf{x}_i, \mathbf{x}_j \rangle_{\mathcal{L}^Q} - 1}}\left(x_{jq} - \frac{x_{j0}}{x_{i0}}x_{iq}\right), \tag{A.21}$$

$$\frac{\partial k_{\mathcal{L}^Q}(\mathbf{z}_k, \mathbf{x}_j)}{\partial x_{iq}} = \frac{k_{\mathcal{L}^Q}(\mathbf{z}_k, \mathbf{x}_j)}{\kappa\sqrt{\langle \mathbf{z}_k, \mathbf{x}_j \rangle_{\mathcal{L}^Q} - 1}}\left(x_{jq} - \frac{x_{j0}}{z_{k0}}z_{kq}\right). \tag{A.22}$$

However, the differentiation of the Bayesian hGP-LVM objective is challenging due to the reparameterization as

$$\mathbf{x}_i^{(h)} = \mathrm{Exp}_{\boldsymbol{\mu}_i}(\mathbf{u}_i^{(h)}), \ \mathbf{u}_i^{(h)} = \mathrm{PT}_{\boldsymbol{\mu}_0 \to \boldsymbol{\mu}_i}(\mathbf{v}_i^{(h)}), \ \tilde{\mathbf{v}}_i^{(h)} = \mathbf{S}_i^{\frac{1}{2}}\boldsymbol{\zeta}_i^{(h)}.$$

We first differentiate $\acute{\mathcal{F}}_b$ without the KL term. In our implementation, we use $\frac{\partial \acute{\mathcal{F}}_b}{\partial \mathbf{\Psi}_1}$ and $\frac{\partial \acute{\mathcal{F}}_b}{\partial \mathbf{\Psi}_2}$ and recall $\mathbf{\Psi}$ statistics as

$$\mathbf{\Psi}_1 = \sum_{h=1}^{H} \mathbf{\Psi}_1^{(h)}, \quad [\mathbf{\Psi}_1^{(h)}]_{kj} = k(\mathbf{z}_k, \mathbf{x}_j^{(h)}),$$
$$\mathbf{\Psi}_2 = \sum_{h=1}^{H} \mathbf{\Psi}_2^{(h)}, \quad \mathbf{\Psi}_2^{(h)} = \sum_{i=1}^{N} \mathbf{\Psi}_2^{(h,i)}, \quad [\mathbf{\Psi}_2^{(h,i)}]_{kl} = k(\mathbf{z}_k, \mathbf{x}_i^{(h)}) k(\mathbf{z}_l, \mathbf{x}_i^{(h)}).$$

The chain rules were used to compute the differentiation w.r.t. variational parameters as

$$\frac{\partial \acute{\mathcal{F}}_b}{\partial \mu_{iq}} = \sum_{h=1}^{H} \mathrm{tr}\left(\frac{\partial \acute{\mathcal{F}}_b}{\partial \mathbf{\Psi}_1^{(h)}} \frac{\partial \mathbf{\Psi}_1^{(h)}}{\partial \mu_{iq}} + \frac{\partial \acute{\mathcal{F}}_b}{\partial \mathbf{\Psi}_2^{(h)}} \frac{\partial \mathbf{\Psi}_2^{(h)}}{\partial \mu_{iq}} \right), \tag{A.23}$$
$$\frac{\partial \acute{\mathcal{F}}_b}{\partial s_{iq}} = \sum_{h=1}^{H} \mathrm{tr}\left(\frac{\partial \acute{\mathcal{F}}_b}{\partial \mathbf{\Psi}_1^{(h)}} \frac{\partial \mathbf{\Psi}_1^{(h)}}{\partial s_{iq}} + \frac{\partial \acute{\mathcal{F}}_b}{\partial \mathbf{\Psi}_2^{(h)}} \frac{\partial \mathbf{\Psi}_2^{(h)}}{\partial s_{iq}} \right). \tag{A.24}$$

We chain the differentiation to the variational parameters via latent representation $\mathbf{x}_i^{(h)}$ as

$$\frac{\partial [\mathbf{\Psi}_1^{(h)}]_{ki}}{\partial \mu_{iq}} = \left(\frac{\partial [\mathbf{\Psi}_1^{(h)}]_{ki}}{\partial \mathbf{x}_i^{(h)}} \right)^\top \frac{\partial \mathbf{x}_i^{(h)}}{\partial \mu_{iq}}, \qquad \frac{\partial [\mathbf{\Psi}_2^{(h)}]_{kl}}{\partial \mu_{iq}} = \left(\frac{\partial [\mathbf{\Psi}_2^{(h)}]_{kl}^{(h,i)}}{\partial \mathbf{x}_i^{(h)}} \right)^\top \frac{\partial \mathbf{x}_i^{(h)}}{\partial \mu_{iq}},$$
$$\frac{\partial [\mathbf{\Psi}_1^{(h)}]_{ki}}{\partial s_{iq}} = \left(\frac{\partial [\mathbf{\Psi}_1^{(h)}]_{ki}}{\partial \mathbf{x}_i^{(h)}} \right)^\top \frac{\partial \mathbf{x}_i^{(h)}}{\partial s_{iq}}, \qquad \frac{\partial [\mathbf{\Psi}_2^{(h)}]_{kl}}{\partial s_{iq}} = \left(\frac{\partial [\mathbf{\Psi}_2^{(h)}]_{kl}^{(h,i)}}{\partial \mathbf{x}_i^{(h)}} \right)^\top \frac{\partial \mathbf{x}_i^{(h)}}{\partial s_{iq}},$$

where the differentiation of the gram matrices is given as

$$\frac{\partial [\mathbf{\Psi}_1^{(h)}]_{ki}}{\partial \mathbf{x}_i^{(h)}} = \frac{k_{\mathcal{L}^Q}(\mathbf{z}_k, \mathbf{x}_i^{(h)})}{\kappa \sqrt{\left\langle \mathbf{z}_k, \mathbf{x}_i^{(h)} \right\rangle_{\mathcal{L}^Q} - 1}} \left(x_{iq}^{(h)} - \frac{x_{i0}^{(h)}}{z_{k0}} z_{kq} \right), \tag{A.25}$$
$$\begin{aligned} \frac{\partial [\mathbf{\Psi}_2^{(h,i)}]_{kl}}{\partial \mathbf{x}_i^{(h)}} = & \frac{k_{\mathcal{L}^Q}(\mathbf{z}_k, \mathbf{x}_i^{(h)}) k_{\mathcal{L}^Q}(\mathbf{z}_l, \mathbf{x}_i^{(h)})}{\kappa \sqrt{\left\langle \mathbf{z}_k, \mathbf{x}_i^{(h)} \right\rangle_{\mathcal{L}^Q} - 1}} \left(x_{jq}^{(h)} - \frac{x_{j0}^{(h)}}{z_{k0}} z_{kq} \right) \\ & + \frac{k_{\mathcal{L}^Q}(\mathbf{z}_k, \mathbf{x}_i^{(h)}) k_{\mathcal{L}^Q}(\mathbf{z}_l, \mathbf{x}_i^{(h)})}{\kappa \sqrt{\left\langle \mathbf{z}_l, \mathbf{x}_i^{(h)} \right\rangle_{\mathcal{L}^Q} - 1}} \left(x_{jq}^{(h)} - \frac{x_{j0}^{(h)}}{z_{l0}} z_{lq} \right). \end{aligned} \tag{A.26}$$

Finally, we derive the differentiation of $\mathbf{x}_i^{(h)}$ w.r.t. variational parameters by chaining the reparameterization. Let $\mathbb{1}_q \in \mathbb{R}^{Q+1}$ be a one-hot vector whose $(q+1)$-th element is 1. We first derive w.r.t. variational mean μ_{iq} as

$$\begin{aligned} \frac{\partial \mathbf{x}_i^{(h)}}{\partial \mu_{iq}} = & \cosh(||\mathbf{u}_i^{(h)}||_{\mathcal{L}^Q}) \mathbb{1}_q + \frac{\partial}{\partial \mu_{iq}} \left\{ \cosh(||\mathbf{u}_i^{(h)}||_{\mathcal{L}^Q}) \right\} \boldsymbol{\mu}_i \\ & + \frac{\partial}{\partial \mu_{iq}} \left\{ \frac{\sinh(||\mathbf{u}_i^{(h)}||_{\mathcal{L}^Q})}{||\mathbf{u}_i^{(h)}||_{\mathcal{L}^Q}} \right\} \mathbf{u}_i^{(h)} + \frac{\sinh(||\mathbf{u}_i^{(h)}||_{\mathcal{L}^Q})}{||\mathbf{u}_i^{(h)}||_{\mathcal{L}^Q}} \frac{\partial \mathbf{u}_i^{(h)}}{\partial \mu_{iq}}. \end{aligned} \tag{A.27}$$

We then compute the differentiation of the second and third terms in Eq. (A.27) as

$$\frac{\partial}{\partial \mu_{iq}} \left\{ \cosh(||\mathbf{u}_i^{(h)}||_{\mathcal{L}^Q}) \right\} = \frac{\sinh(||\mathbf{u}_i^{(h)}||_{\mathcal{L}^Q})}{||\mathbf{u}_i^{(h)}||_{\mathcal{L}^Q}} \cdot \hat{\mathbf{u}}_i^{(h)\top} \frac{\partial \mathbf{u}_i^{(h)}}{\partial \mu_{iq}}, \tag{A.28}$$
$$\frac{\partial}{\partial \mu_{iq}} \left\{ \frac{\sinh(||\mathbf{u}_i^{(h)}||_{\mathcal{L}^Q})}{||\mathbf{u}_i^{(h)}||_{\mathcal{L}^Q}} \right\} = \frac{\cosh(||\mathbf{u}_i^{(h)}||_{\mathcal{L}^Q}) ||\mathbf{u}_i^{(h)}||_{\mathcal{L}^Q} - \sinh(||\mathbf{u}_i^{(h)}||_{\mathcal{L}^Q})}{||\mathbf{u}_i^{(h)}||_{\mathcal{L}^Q}^3} \cdot \hat{\mathbf{u}}_i^{(h)\top} \frac{\partial \mathbf{u}_i^{(h)}}{\partial \mu_{iq}}, \tag{A.29}$$

where $\hat{\mathbf{u}}_i = [-u_{i0}, u_{i1}, \ldots, u_{iQ}]^\top$. The differentiation of $\mathbf{u}_i^{(h)}$ given as

$$\frac{\partial \mathbf{u}_i^{(h)}}{\partial \mu_{iq}} = \begin{cases} -\frac{\tilde{\boldsymbol{\mu}}_i^\top \tilde{\mathbf{v}}_i^{(h)}}{(\mu_{i0}+1)^2}(\boldsymbol{\mu}_0 + \boldsymbol{\mu}_i) + \frac{\tilde{\boldsymbol{\mu}}_i^\top \tilde{\mathbf{v}}_i^{(h)}}{\mu_{i0}+1}\mathbb{1}_0 & (q=0), \\ \frac{\tilde{v}_{iq}}{\mu_{i0}+1}(\boldsymbol{\mu}_0 + \boldsymbol{\mu}_i) + \frac{\tilde{\boldsymbol{\mu}}_i^\top \tilde{\mathbf{v}}_i^{(h)}}{\mu_{i0}+1}\mathbb{1}_q & (q \neq 0). \end{cases} \tag{A.30}$$

Next, we derive the differentiation w.r.t. variational variance s_{iq} as

$$\frac{\partial \mathbf{x}_i}{\partial s_{iq}} = \frac{\partial}{\partial s_{iq}}\left\{\cosh(||\mathbf{u}_i^{(h)}||_{\mathcal{L}^Q})\right\}\boldsymbol{\mu}_i + \frac{\partial}{\partial s_{iq}}\left\{\frac{\sinh(||\mathbf{u}_i^{(h)}||_{\mathcal{L}^Q})}{||\mathbf{u}_i^{(h)}||_{\mathcal{L}^Q}}\right\}\mathbf{u}_i^{(h)} + \frac{\sinh(||\mathbf{u}_i^{(h)}||_{\mathcal{L}^Q})}{||\mathbf{u}_i^{(h)}||_{\mathcal{L}^Q}}\frac{\partial \mathbf{u}_i^{(h)}}{\partial s_{iq}}. \tag{A.31}$$

We compute the first and second terms in Eq. (A.31) as

$$\frac{\partial}{\partial s_{iq}}\left\{\cosh(||\mathbf{u}_i^{(h)}||_{\mathcal{L}^Q})\right\} = \frac{\sinh(||\mathbf{u}_i^{(h)}||_{\mathcal{L}^Q})}{||\mathbf{u}_i^{(h)}||_{\mathcal{L}^Q}} \cdot \hat{\mathbf{u}}_i^{(h)\top}\frac{\partial \mathbf{u}_i^{(h)}}{\partial s_{iq}}, \tag{A.32}$$

$$\frac{\partial}{\partial s_{iq}}\left\{\frac{\sinh(||\mathbf{u}_i^{(h)}||_{\mathcal{L}^Q})}{||\mathbf{u}_{iq}^{(h)}||_{\mathcal{L}^Q}}\right\} = \frac{\cosh(||\mathbf{u}_i^{(h)}||_{\mathcal{L}^Q})||\mathbf{u}_i^{(h)}||_{\mathcal{L}^Q} - \sinh(||\mathbf{u}_i^{(h)}||_{\mathcal{L}^Q})}{||\mathbf{u}_i^{(h)}||_{\mathcal{L}^Q}^3} \cdot \hat{\mathbf{u}}_i^{(h)\top}\frac{\partial \mathbf{u}_i^{(h)}}{\partial s_{iq}}. \tag{A.33}$$

Then, the differentiation of $\mathbf{u}_i$ w.r.t. s_{iq} is derived as

$$\frac{\partial \mathbf{u}_i^{(h)}}{\partial s_{iq}} = \zeta_{iq}^{(h)}\mathbb{1}_q + \frac{\mu_{iq}\zeta_{iq}^{(h)}}{\mu_{i0}+1}(\boldsymbol{\mu}_0 + \boldsymbol{\mu}_i). \tag{A.34}$$

Finally, we compute the differentiation of the KL term. We first extend the KL term with Monte Carlo approximations as

$$\begin{aligned} \mathrm{KL}_i &= \sum_{h=1}^{H}\left\{\log \mathcal{N}_{\mathcal{L}^Q}^w(\mathbf{x}_i^{(h)}|\boldsymbol{\mu}_i, \mathbf{S}_i) - \log \mathcal{N}_{\mathcal{L}^Q}^w(\mathbf{x}_i^{(h)}|\mathbf{0}, \mathbf{I}_n)\right\} \\ &= \sum_{h=1}^{H}\left\{\log \mathcal{N}(\mathbf{v}_i^{(h)}|\mathbf{0}, \mathbf{S}_i) - (Q-1)\log\frac{\sinh(||\mathbf{u}_i^{(h)}||_{\mathcal{L}^Q})}{||\mathbf{u}_i^{(h)}||_{\mathcal{L}^Q}}\right. \\ &\qquad \left. - \log \mathcal{N}(\mathbf{v}_i^{(h)}|\mathbf{0}, \mathbf{I}_n) + (Q-1)\log\frac{\sinh(||\mathbf{v}_i^{(h)}||_{\mathcal{L}^Q})}{||\mathbf{v}_i^{(h)}||_{\mathcal{L}^Q}}\right\}, \end{aligned} \tag{A.35}$$

where

$$\begin{aligned} \log \mathcal{N}(\mathbf{v}_i^{(h)}|\mathbf{0}, \mathbf{S}_i) &= -\frac{Q}{2}\log 2\pi - \frac{1}{2}\log|\mathbf{S}_i| - \frac{1}{2}\boldsymbol{\zeta}_i^{(h)\top}\boldsymbol{\zeta}_i^{(h)}, \\ \log \mathcal{N}(\mathbf{v}_i^{(h)}|\mathbf{0}, \mathbf{I}_n) &= -\frac{Q}{2}\log 2\pi - \frac{1}{2}\boldsymbol{\zeta}_i^{(h)\top}\mathbf{S}_i\boldsymbol{\zeta}_i^{(h)}. \end{aligned}$$

Its differentiation can be computed using Eqns. (A.30) and (A.34). The differentiation of the second and fourth terms is given as

$$\begin{aligned} \frac{\partial}{\partial \mu_{iq}}\left\{\log\frac{\sinh(||\mathbf{u}_i^{(h)}||_{\mathcal{L}^Q})}{||\mathbf{u}_i^{(h)}||_{\mathcal{L}^Q}}\right\} &= \left\{\frac{1}{\tanh(||\mathbf{u}_i^{(h)}||_{\mathcal{L}^Q})||\mathbf{u}_i^{(h)}||_{\mathcal{L}^Q}} - \frac{1}{||\mathbf{u}_i^{(h)}||_{\mathcal{L}^Q}}\right\} \cdot \hat{\mathbf{u}}_i^{(h)\top}\frac{\partial \mathbf{u}_i^{(h)}}{\partial \mu_{iq}}, \\ \frac{\partial}{\partial s_{iq}}\left\{\log\frac{\sinh(||\mathbf{u}_i^{(h)}||_{\mathcal{L}^Q})}{||\mathbf{u}_i^{(h)}||_{\mathcal{L}^Q}}\right\} &= \left\{\frac{1}{\tanh(||\mathbf{u}_i^{(h)}||_{\mathcal{L}^Q})||\mathbf{u}_i^{(h)}||_{\mathcal{L}^Q}} - \frac{1}{||\mathbf{u}_i^{(h)}||_{\mathcal{L}^Q}}\right\} \cdot \hat{\mathbf{u}}_i^{(h)\top}\frac{\partial \mathbf{u}_i^{(h)}}{\partial s_{iq}}, \\ \frac{\partial}{\partial s_{iq}}\left\{\log\frac{\sinh(||\mathbf{v}_i^{(h)}||_{\mathcal{L}^Q})}{||\mathbf{v}_i^{(h)}||_{\mathcal{L}^Q}}\right\} &= \left\{\frac{1}{\tanh(||\mathbf{u}_i^{(h)}||_{\mathcal{L}^Q})||\mathbf{u}_i^{(h)}||_{\mathcal{L}^Q}} - \frac{1}{||\mathbf{u}_i^{(h)}||_{\mathcal{L}^Q}}\right\} \frac{\zeta_{iq}^{(h)}}{\sqrt{s_{iq}}}. \end{aligned}$$

A.10 Experimental Details

We implemented our methods partially using the GPy (GPy, 2012) source code (BSD-3-Clause license) and experimented partially using PoincaréMap (CC BY-NC 4.0 license) (Klimovskaia et al., 2020) and RotHVAE (Cho et al., 2022), sincerely appreciating their contribution. Our code ran on a single Intel Core i7-10700 CPU without GPU. Before inference, the latent variables of hGP-LVM and sparse hGP-LVM and the variational mean of Bayesian hGP-LVM were initialized by generating two-dimensional random variables following $U(-10^{-3}, 10^{-3})$ and then mapping them into the Lorentz model as $[\sqrt{1+x_1^2+x_2^2}, x_1, x_2]^\top$. We initialized the variance parameter $U(-10^{-5}, 10^{-5})$ in the Bayesian hGP-LVM. During optimization, we set a large learning rate at the beginning of optimization and fixed variance parameters in the first 100 epochs for Bayesian hGP-LVM. The position of inducing variables $\mathbf{Z}$ was updated every 10 epoch by sampling from latent variables. Our implementation is available in https://github.com/koshi-lmd/hyperboloid_gplvm.git.

We next notified the details of the quality metrics we used. Trustworthiness $T(k) \in [0, 1]$ is a local one and defined as follows:

$$T(k) = 1 - \frac{2}{Nk(2N - 3k - 1)} \sum_{i=1}^{N} \sum_{j \in U_i^k} \max(0, r(i, j) - k), \tag{A.36}$$

where U_i^k denotes a set of the k-nearest neighbors of the sample i in the *latent* space, and j is an $r(i, j)$-th neighbor of i. Continuity $C(k)$ is the converse of trustworthiness computed by changing U_i^k in Eq. (A.36) into V_i^k, a set of the k-nearest neighbors of the sample i in the *observed* space. The Shepard goodness is the Spearman rank correlation in the Shepard diagram. The Shepard diagram is the scatterplot of the pointwise distance between two variables, and there are $\frac{N(N-1)}{2}$ points corresponding to the distance value between any two variables. The quality metrics for DR are compared in (Espadoto et al., 2019).

A.11 Additional Experimental Results

In this section, we show the additional experimental results to compare our methods with the visualization-aided DR methods using synthesis datasets.

Spiral. First, we validate the effectiveness of the GP to preserve the continuity with the *Spiral* dataset. The Spiral dataset contains 10 spirals starting from the origin with random oscillation in proportion to the norm (Figure A.6 (a)), synthesizing easy hierarchies. We generated 10 spirals with 80 points in a two-dimensional space ($N = 800$) and mapped them in a 20-dimensional space through a random linear transformation ($D = 20$). We embedded the Spiral into the two-dimensional space and compared the quality with quantitative metrics and qualitative visualization. For quantitative evaluation, we used the three metrics following (Espadoto et al., 2019; Zu and Tao, 2022) to evaluate the preservation quality of the local and global structure: trustworthiness (Venna and Kaski, 2001), continuity, and Shepard goodness (Joia et al., 2011). Trustworthiness tells the reliability of the embedding, and continuity measures the preservation quality of the local continuity. The local metrics require the number of neighbors, and we set $k = 3$ in both metrics. The Shepard goodness is a global metric and the Spearman rank correlation of the pointwise distances between observed and latent variables. The Shepard goodness measures the match of global coordinates between variables. Since we expected the visualization purposes, we computed each metric using Euclidean coordinates. We compared hGP-LVMs with UMAP (McInnes et al., 2018) as a benchmark for DR, and PoincaréMap (Klimovskaia et al., 2020) as a DR method on the hyperbolic model to confirm the effectiveness of the GP-based modeling. We set $\kappa = 100$ and $M = 30$ in all hGP-LVMs and conducted the same experiment ten times to confirm the reproducibility.

We first show the quantitative results in Figure A.6 (b). hGP-LVM and sparse hGP-LVM performed the highest results with low variances in all metrics, indicating that they preserved Spirals' global and local structure. However, Bayesian hGP-LVM contains high variance in the local metrics caused by the approximated inference. Figure A.6 (c) shows that UMAP and PoincaréMap preferred the neighbor relations of Spiral, unlike that of hGP-LVM, which preferred their global coordinates. The results in Figures A.6 (b) and (c) imply that local and global structure preservation is needed to embed the continuity of structured data. Figure A.6 (d) shows the detailed results to confirm the effect of the length scale parameter κ. The range of the latent variables spread with large κ, and the continuity score increased, which comes in the hyperbolic curvature of the latent space. However, the

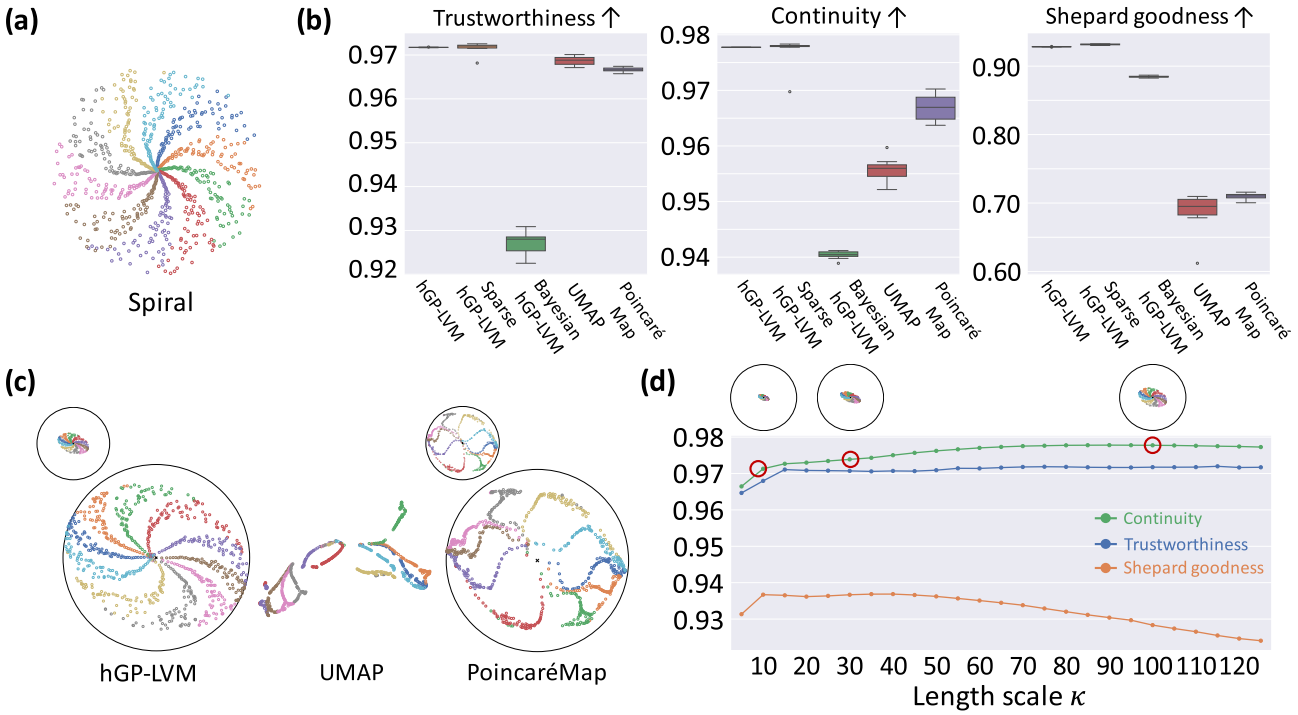

Figure A.6: **Experimental results on the Spiral dataset.** (a) Shape of the spirals before linear transformation. (b) Error bar plot of trustworthiness (left), continuity (center), and Shepard goodness (right). (c) Scatter plot of two-dimensional embeddings of hGP-LVM, UMAP, and PoincaréMap. We zoomed in on latent variables and showed the original latent space in the upper left corner. (d) Quality metric scores of hGP-LVM with different length scales κ. (e) Comparison of reconstruction error of sparse hGP-LVM and sparse GP-LVM.

Shepard goodness score decreased with large κ, and we observed the tradeoff relationship between global and local preservation quality. From the above, we confirm that hGP-LVMs have higher results than comparatives in the Spiral dataset.

Synthetic Myeloid Progenitors (Klimovskaia et al., 2020). We show another synthetic experimental results on the Myeloid Progenitors (MP) dataset, synthesizing cell differentiation from the progenitors into four cells: erythrocyte, neutrophil, monocyte, and megakaryocyte ($N = 640$, $D = 11$) (Figure A.7 (a)). We set $\kappa = 100$ and $M = 50$.

Figure A.7 (b) shows the quantitative results. hGP-LVM outperformed the comparative methods except for the local metrics. However, the absolute value was still high, and hGP-LVM produced high accuracy in both local and global metrics. Figure A.7 (c) shows the visualization results of hGP-LVM, GP-LVM, and PoincaréMap. Although the embeddings of GP-LVM and PoincaréMap are torn or wiggling, the hGP-LVM well preserved the continuity behind the hierarchical data, contributing to the visibility of the low-dimensional embeddings. The effect of the length scale in Figures A.6 (d) is similar to the result on the Spiral dataset in Figure A.7 (d). From the above, we confirm that hGP-LVMs produced better results than previous visualization-aided DR methods in the synthetic setting.

A.11.1 Embedding Comparison with Different Lengthscale

This section presents the visualization comparison with different length scales, κ. In Section 3.1 of the main paper, we state that *the meaning of the length scale κ is how much we expect the latent variables to follow the hyperbolic curvature*. We show the experimental verification of this statement in Figure A.8 on all synthesis datasets. In all results, we can confirm that the representation was spread and aligned as growing κ. In the SBT results (bottom), although the embeddings of depths 1 and 2 were mixed around the origin, they were separated with $\kappa = 100$. The length scale parameter determined the range of the latent variables, and large κ brought the hyperbolic curvature to the latent representation. We must determine κ according to the degree to which we expected the hierarchical structure in data.

References

Cho, S., Lee, J., Park, J., and Kim, D. (2022). A rotated hyperbolic wrapped normal distribution for hierarchical representation learning. *Advances in Neural Information Processing Systems*, 35:17831–17843.

Espadoto, M., Martins, R. M., Kerren, A., Hirata, N. S., and Telea, A. C. (2019). Toward a quantitative survey of dimension reduction techniques. *IEEE Transactions on Visualization and Computer Graphics*, 27(3):2153–2173.

GPy (2012). GPy: A Gaussian process framework in Python. Available: http://github.com/SheffieldML/GPy.

Joia, P., Coimbra, D., Cuminato, J. A., Paulovich, F. V., and Nonato, L. G. (2011). Local affine multidimensional projection. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2563–2571.

Klimovskaia, A., Lopez-Paz, D., Bottou, L., and Nickel, M. (2020). Poincaré maps for analyzing complex hierarchies in single-cell data. *Nature Communications*, 11(1):2966.

McInnes, L., Healy, J., and Melville, J. (2018). UMAP: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*.

Sloane, N. J. (2007). The on-line encyclopedia of integer sequences. In *Proceedings of the International Conference on Towards Mechanized Mathematical Assistants*, pages 130–130.

Venna, J. and Kaski, S. (2001). Neighborhood preservation in nonlinear projection methods: An experimental study. In *Proceedings of the International Conference on Artificial Neural Networks*, pages 485–491.

Zu, X. and Tao, Q. (2022). SpaceMAP: Visualizing high-dimensional data by space expansion. In *Proceedings of the International Conference on Machine Learning*, pages 27707–27723.

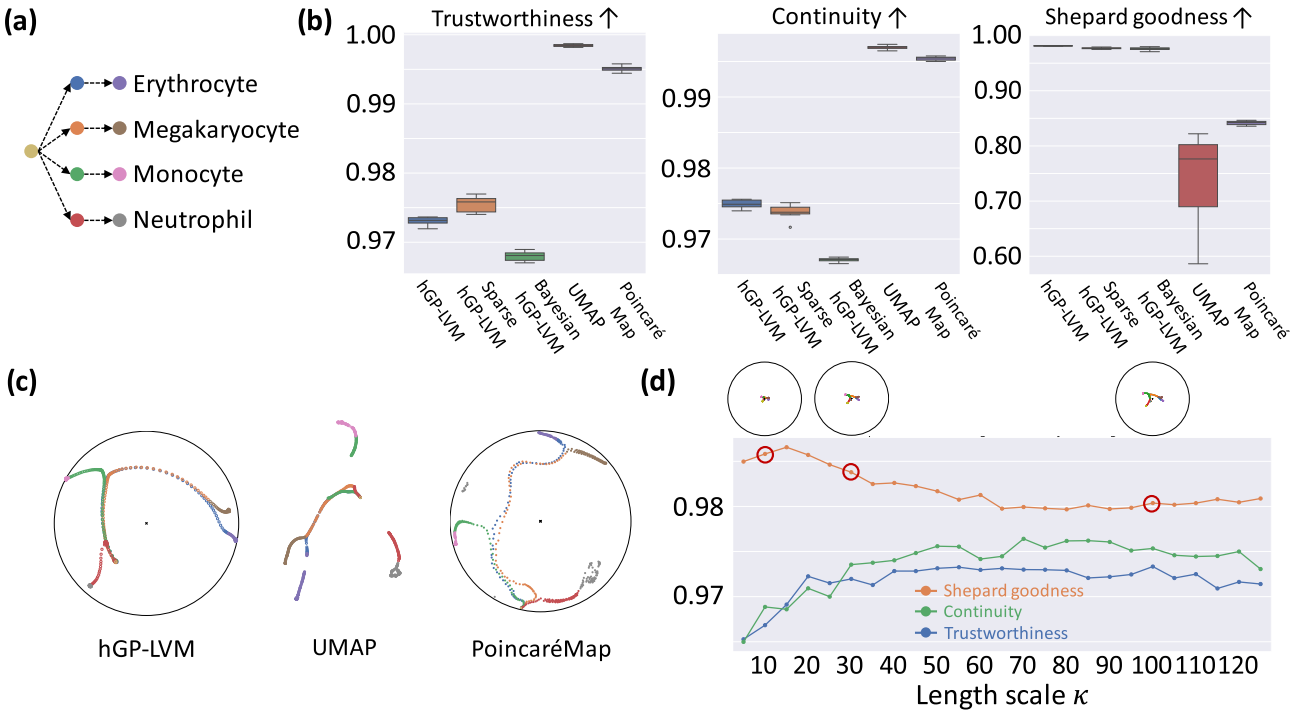

Figure A.7: **Experimental results on the synthetic myeloid progenitors dataset.** (a) Hierarchical relation and color code of visualization. (b) Error bar plot of trustworthiness (left), continuity (center), and Shepard goodness (right). (c) Scatter plot of two-dimensional embeddings of hGP-LVM, UMAP, and PoincaréMap. We zoomed in on latent variables and showed the original latent space in the upper left corner. (d) Quality metric scores of hGP-LVM with different length scales κ. (e) Comparison of reconstruction error of sparse hGP-LVM and sparse GP-LVM.

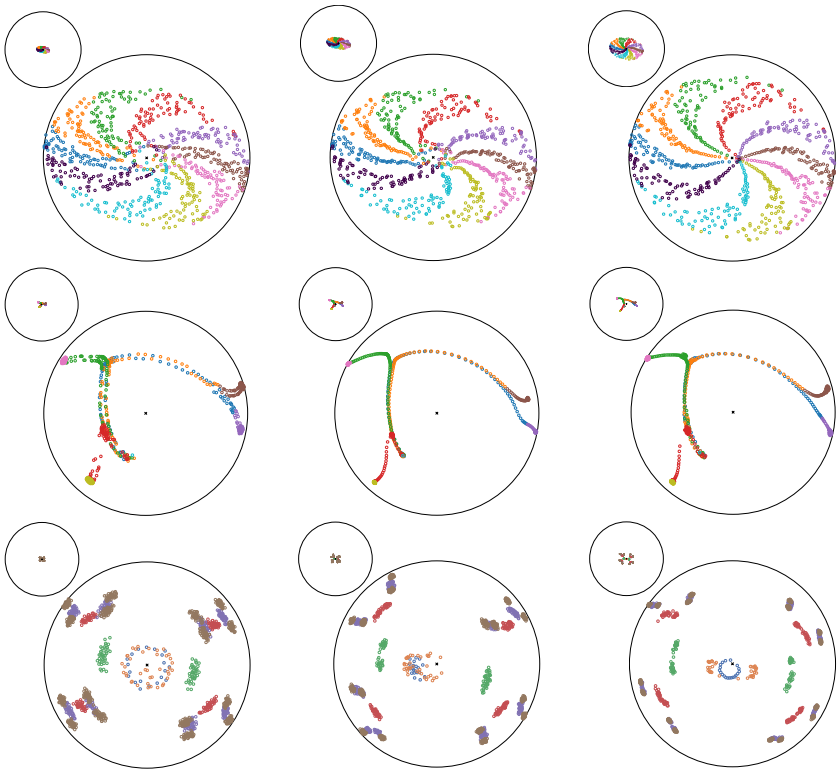

Figure A.8: **Embedding comparison of hGP-LVM with different length scales on synthetic datasets, Spiral (top), synthetic myeloid progenitors (middle), and SBT (bottom)).** We zoomed in on the latent variables.