Only Bayes should learn a manifold

(on the estimation of differential geometric structure from data)

Søren Hauberg
Section for Cognitive Systems
Technical University of Denmark
sohau@dtu.dk

Abstract

We investigate learning of the differential geometric structure of a data manifold embedded in a high-dimensional Euclidean space. We first analyze kernel-based algorithms and show that under the usual regularizations, non-probabilistic methods cannot recover the differential geometric structure, but instead find mostly linear manifolds or spaces equipped with teleports. We repeat the analysis for probabilistic methods and show that they naturally recover the geometric structure. Fully exploiting this structure, however, requires the development of stochastic extensions to classic Riemannian geometry. We take early steps in that regard. Finally, we partly extend the analysis to models based on neural networks, thereby highlighting geometric and probabilistic shortcomings of current deep generative models.

Comments on this document are gratefully accepted at sohau@dtu.dk. This is the 2nd revision.

1 Motivation and background

Manifold learning aim to learn a low-dimensional representation of data that reflect the intrinsic structure of data. Spectral methods seek a low-dimensional embedding of high-dimensional data that preserve certain aspects of the data. This includes methods such as *Isomap* [35], *Locally linear embeddings* [29], *Laplacian eigenmaps* [2] and more [32, 9]. Probabilistic methods often view the data manifold as governed by a latent variable along with a generative model that describe how the latent manifold is to be embedded in the data space. The common theme is the quest for a low-dimensional representation that faithfully capture the data.

Ideally, we want an *operational representation*, i.e. we want to be able to make mathematically meaningful calculations with respect to the learned representation. For quantitative data analysis in the learned representation, a reasonable set of supported "operations" at least include:

- **Interpolation:** given two points, a natural unique interpolating curve that follow the manifold should exist.
- **Distances:** the distance between two points should be well-defined and reflect the amount of energy required to transform one point to another.
- **Measure:** the representation should be equipped with a measure under which integration is well-defined for all points on the manifold.

Depending on which analysis is to be performed in the new representation, one may focus on different operations. Even if the above operations should be considered elementary, most manifold learning schemes do not support any of these.

Embedding methods seek a low-dimensional embedding $\mathbf{z}_{1:N} = \{\mathbf{z}_1, \ldots, \mathbf{z}_N\}$ of the data $\mathbf{x}_{1:N}$. These methods fundamentally only describe the data manifold at the points where data is observed

Preprint. Work in progress.

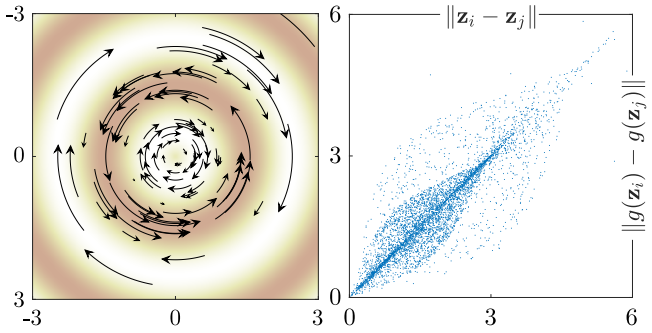

Figure 1: Reparametrizations illustrated. The left panel shows a "swirling" transformation of $\mathcal{Z}$ with the property that a Gaussian variable with zero mean and unit covariance, will have the same distribution after a reparametrization. The right panel shows pair-wise distances between points before and after reparametrization; evidently the geometry of $\mathcal{Z}$ is sensitive to reparametrizations.

and nowhere else. As such, the low-dimensional embedding space is only well-defined at $\mathbf{z}_{1:N}$. It is common to treat the low-dimensional embedding space as being Euclidean, but this is generally a *post hoc* assumption with limited grounding in the embedding method. Fundamentally, the learned representation space is a discrete space that does not lend itself to continuous interpolations. Likewise, the most natural measure will only assign mass to the points $\mathbf{z}_{1:N}$, and any associated distribution will be discrete. This is too limited to be considered an operational representation.

Generative models estimate a set of low-dimensional latent variables $\mathbf{z}_{1:N}$ along with a suitable mapping $f : \mathcal{Z} \to \mathcal{X}$ such that $f(\mathbf{z}) \approx \mathbf{x}$. It is, again, common to treat the latent space $\mathcal{Z}$ as being Euclidean. However, this assumption easily lead to arbitrariness. As an example, consider the *variational autoencoder (VAE)* [17, 27], which seek a representation in which $\mathbf{z}_{1:N}$ follow a unit Gaussian distribution. Now consider the transformation

$$g(\mathbf{z}) = \mathbf{R}_\theta \mathbf{z}, \tag{1.1}$$

where $\mathbf{R}_\theta$ is a linear transformation that rotate points by $\theta(\mathbf{z}) = \sin(\pi\|\mathbf{z}\|)$. This is a smooth invertible transformation with the property that

$$\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \quad \Rightarrow \quad g(\mathbf{z}) \sim \mathcal{N}(\mathbf{0}, \mathbf{I}). \tag{1.2}$$

Figure 1 illustrate this transformation. If the latent variables $\mathbf{z}_{1:N}$ and the generator f is an optimal VAE, then $g(\mathbf{z}_{1:N})$ and $f \circ g^{-1}$ is equally optimal. Yet, the latent spaces $\mathcal{Z}$ and $g(\mathcal{Z})$ are quite different; Fig. 1 shows the Euclidean distances between $\mathbf{z}_n$ and $g(\mathbf{z}_n)$ for samples drawn from a unit Gaussian. Clearly, the transformed latent space is significantly different from the original space. As the VAE provides no guarantees as to which latent space is recovered, we must be careful when relying on the Euclidean latent space: distances between points are effectively arbitrary and as are straight-line interpolations. Any analysis relying on vector operations in the latent space are, thus, arbitrary and positive result should be viewed either as pure luck with little mathematical grounding, or due to unspecified aspects of the model. Ideally, we want a representation space that is invariant to such transformations, but current models do not deliver.

In this paper, we consider models where the representation space $\mathcal{Z}$ is learned jointly with a smooth mapping $f : \mathcal{Z} \to \mathcal{X}$, such that $\mathcal{Z}$ is naturally endowed with a Riemannian metric (Sec. 2). This gives well-defined interpolants, distances and a natural measure. We contribute a detailed analysis of the case where f is estimated by a kernel method (Sec. 3), and show that even in the case of infinite noise-free data a non-probabilistic estimate of f cannot recover the true Riemannian structure of $\mathcal{Z}$. In contrast, we show that probabilistic estimates of f can recover the true Riemannian structure (Sec. 3.2). Fully exploiting this structure, however, require the development of Bayesian extensions to classic differential geometry (Sec. 4); we contribute elementary results in that regard, but many questions remain open. Finally, we partly extend our analysis to the case where f is a neural network and demonstrate that current deep generative models are lacking elementary properties before they can learn the Riemannian structure of data manifolds (Sec. 5). Our key finding is that uncertainty quantification is a prerequisite for learning an operational representation as the usual smoothness regularization introduce a harmful bias.

Notation. We let $\mathcal{Z}$ denote the d-dimensional *representation* or *latent space*, which is learned from data in the *observation space* $\mathcal{X} \equiv \mathbb{R}^D$. Topologically, the latent space $\mathcal{Z}$ is assumed to be Euclidean. Latent points are denoted $\mathbf{z}_n \in \mathcal{Z}$, while corresponding observations are $\mathbf{x}_n \in \mathcal{X}$. The mapping $f : \mathcal{Z} \to \mathcal{X}$ embeds $\mathcal{Z}$ in $\mathcal{X}$; we denote $\mathcal{M} = f(\mathcal{Z})$ and assume that $\mathcal{M}$ is a Riemannian manifold.

2 Riemannian manifolds

A d-dimensional manifold $\mathcal{M}$ embedded in $\mathbb{R}^D (d \leq D)$ is a topological space in which there exist a neighborhood around each point $\mathbf{x} \in \mathcal{M}$ that is homeomorphic to $\mathbb{R}^d$ [11]. Informally, $\mathcal{M}$ is a (usually nonlinear) surface in $\mathbb{R}^D$ that is locally Euclidean, i.e. it does not self-intersect or otherwise locally change dimensionality, etc. We assume that we have a d-dimensional parametrization $\mathcal{Z}$ of the manifold along with a mapping $f : \mathcal{Z} \to \mathcal{X}$, such that $\mathcal{M} = f(\mathcal{Z})$.

We first define the inner product between points in $\mathbb{R}^D$ as $\langle \mathbf{x}, \mathbf{x}' \rangle = {}^1\!/_D \sum_i x_i x'_i$. The division by D ensures that the induced norm remain finite in the limit $D \to \infty$. Now, let $\mathbf{z}$ be a d-dimensional latent point and let Δ_1 and Δ_2 be infinitesimals, then we can compute their inner product around $\mathbf{z}$ in the data space using Taylor's Theorem,

$$\langle f(\mathbf{z}+\Delta_1) - f(\mathbf{z}), f(\mathbf{z}+\Delta_2) - f(\mathbf{z}) \rangle \tag{2.1}$$

$$= \langle f(\mathbf{z}) + \mathbf{J}_\mathbf{z}\Delta_1 - f(\mathbf{z}), f(\mathbf{z}) + \mathbf{J}_\mathbf{z}\Delta_2 - f(\mathbf{z}) \rangle \tag{2.2}$$

$$= \langle \mathbf{J}_\mathbf{z}\Delta_1, \mathbf{J}_\mathbf{z}\Delta_2 \rangle = {}^1\!/_D \cdot \Delta_1^\top \left(\mathbf{J}_\mathbf{z}^\top \mathbf{J}_\mathbf{z} \right) \Delta_2, \tag{2.3}$$

where $\mathbf{J}_\mathbf{z} = \partial_\mathbf{z} f \in \mathbb{R}^{D \times d}$ is the Jacobian of f at $\mathbf{z}$. The $d \times d$ symmetric positive definite matrix ${}^1\!/_D \cdot \mathbf{J}_\mathbf{z}^\top \mathbf{J}_\mathbf{z}$, thus defines a local inner product. We denote this matrix

$$\mathbf{M}_\mathbf{z} = {}^1\!/_D \cdot \mathbf{J}_\mathbf{z}^\top \mathbf{J}_\mathbf{z}, \tag{2.4}$$

and refer to it as the *(pull-back) metric* of the manifold. Note that this local inner product is invariant to reparametrizations of the manifold as it merely correspond to the inner product of $\mathcal{X}$ measured locally on the manifold. Hence it avoids the parametrization issue discussed in the opening section.

Distances & interpolants. The length of a smooth curve in latent space $\mathbf{c} : [a, b] \to \mathcal{Z}$ under the local inner product is

$$\mathcal{L}(\mathbf{c}) = \int_a^b \sqrt{\dot{\mathbf{c}}_t^\top \mathbf{M}_{\mathbf{c}_t} \dot{\mathbf{c}}_t} \mathrm{d}t, \tag{2.5}$$

where $\dot{\mathbf{c}}_t = \partial_t \mathbf{c}(t)$ is the derivative of the curve. Natural interpolants (geodesics) can then be defined as length minimizing curves connecting two points. The length of such a curve is a natural distance measure along the manifold. Unfortunately, minimizing curve length gives rise to a poorly determined optimization problem as the length of a curve is invariant to its parametrization. The following proposition provides remedy [11]:

Proposition 1. *Let* $\mathbf{c} : [a, b] \to \mathcal{Z}$ *be a smooth curve that (locally) minimizes the "curve energy"*

$$\mathcal{E}(\mathbf{c}) = \frac{1}{2} \int_a^b \dot{\mathbf{c}}_t^\top \mathbf{M}_{\mathbf{c}_t} \dot{\mathbf{c}}_t \mathrm{d}t, \tag{2.6}$$

then $\mathbf{c}$ *has constant velocity and is length-minimizing.*

This energy functional is locally uniformly convex and therefore its solution is locally unique. Standard calculus of variation shows that curves of minimal energy satisfy the following system of second order differential equations,

$$\ddot{\mathbf{c}}_t = -\frac{1}{2} \mathbf{M}_{\mathbf{c}_t}^{-1} \Big[2(\mathbf{I} \otimes \dot{\mathbf{c}}_t^\top) \partial_{\mathbf{c}_t} \mathrm{vec}[\mathbf{M}_{\mathbf{c}_t}] \, \dot{\mathbf{c}}_t - \partial_{\mathbf{c}_t} \mathrm{vec}[\mathbf{M}_{\mathbf{c}_t}]^\top (\dot{\mathbf{c}}_t \otimes \dot{\mathbf{c}}_t) \Big], \tag{2.7}$$

where $\mathrm{vec}[\cdot]$ stacks the columns of a matrix into a vector and $\otimes$ is the Kronecker product. Such systems can be solved numerically using standard techniques. Figure 2 gives an example geodesic.

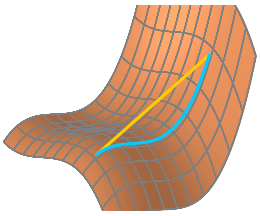

Figure 2: Geodesic interpolation along the manifold (blue) versus along a straight line (yellow).

Integration. Given a function $h : \mathcal{X} \to \mathbb{R}$ we can integrate it over a part of the manifold $f(\Omega)$, $\Omega \subseteq \mathcal{Z}$ as [24]

$$\int_{f(\Omega)} h(\mathbf{x}) \mathrm{d}\mathbf{x} = \int_\Omega h(f(\mathbf{z})) \sqrt{\det(\mathbf{M}_\mathbf{z})} \mathrm{d}\mathbf{z}. \tag{2.8}$$

The quantity $\sqrt{\det(\mathbf{M})}$ is known as the *Riemannian volume measure* and is akin to the Jacobian-determinant in the *change of variables theorem*. As before, this integration is invariant to reparametrizations of $\mathcal{Z}$ as it is performed with respect to the measure of $\mathcal{X}$.

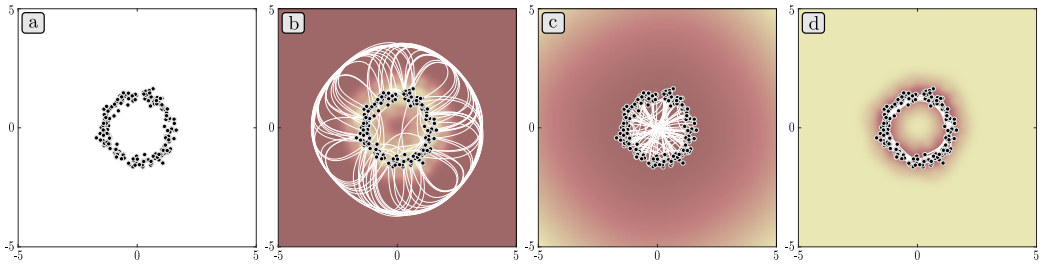

Figure 3: (a) Latent points $\mathbf{z}_n \in \mathcal{Z}$ for our guiding example. (b) Geodesics for Gaussian kernel ridge regression. These are pushed away from the data via "teleports". (c) Geodesics for kernel ridge regression with a Gaussian+linear kernel. The linear extrapolation implies (almost) linear geodesics. (d) Geodesics for Gaussian process regression. The uncertainty make geodesics move along the manifold.

3 Manifold learning with kernels

We now consider data $\mathbf{x}_{1:N}$ distributed on a compact d-dimensional Riemannian submanifold $\mathcal{M} \subset \mathbb{R}^D$ embedded in the data space. We consider a known set of d-dimensional representations $\mathbf{z}_{1:N}$ and estimate the mapping $f : \mathcal{Z} \to \mathcal{X}$ using kernel methods. Note that this manifold is only locally diffeomorphic to d-dimensional Euclidean space, and it may globally self-intersect[1]. For the sake of analysis, we assume noise-free data and consider the limit $N \to \infty$. This setting is sufficient to prove our main point, but the analysis also hold under noise.

Our key question is if we can recover the true Riemannian structure of $\mathcal{Z}$. Methods that fail at this given infinite noise-free data should be avoided. To give an answer, we will study the metric in regions that are near the training data, and in regions that are far away. We formalize this as follows.

Definition 1. *For a point* $\mathbf{z}$ *and a dataset* $\mathbf{Z}$*, the distance between them is* $\mathsf{dist}(\mathbf{z}, \mathbf{Z}) = \inf_{\tilde{\mathbf{z}} \in \mathbf{Z}} \|\mathbf{z} - \tilde{\mathbf{z}}\|$*. Note that this infimum always exist as the element-wise distance is bounded from below by 0.*

Definition 2. *For a function* $\mathbf{x} = h(\mathbf{z})$*, we define the limits*

$$\mathbf{x} \xrightarrow{\text{away}} \hat{\mathbf{x}} \quad \text{and} \quad \mathbf{x} \xrightarrow{\text{near}} \hat{\mathbf{x}} \tag{3.1}$$

if for any sequences $\hat{\mathbf{z}}_l$ *such that* $\mathsf{dist}(\hat{\mathbf{z}}_l, \mathbf{Z}) \xrightarrow{l \to \infty} \infty$*, or* $\mathsf{dist}(\hat{\mathbf{z}}_l, \mathbf{Z}) \xrightarrow{l \to \infty} 0$*, respectively, we have* $h(\hat{\mathbf{z}}_l) \xrightarrow{l \to \infty} \hat{\mathbf{x}}$*. Note that these limits are not always defined.*

A guiding example. To illustrate our main point, we draw data uniformly on a unit circle and nonlinearly embed it in $\mathcal{X} = \mathbb{R}^{1000}$ with added Gaussian noise. We project this data into $\mathcal{Z} = \mathbb{R}^2$ while keeping the circular structure of data and learn a mapping f from $\mathcal{Z}$ to $\mathcal{X}$. Details of this process are in Appendix A. Finally, we compute shortest paths under the pull-back metric; if the true metric is recovered we should see shortest paths corresponding to circular arcs in $\mathcal{Z}$. Figure 3a show the latent points in $\mathcal{Z}$.

3.1 The deterministic setting

We now consider learning the mapping $f : \mathcal{Z} \to \mathcal{X}$ using *kernel ridge regression* [31], i.e.

$$f_{\text{krr}}(\mathbf{z}_*) = k_{*,\mathbf{z}} \left(k_{\mathbf{z},\mathbf{z}} + \sigma^2 \mathbf{I}\right)^{-1} \mathbf{X}, \tag{3.2}$$

where k is a suitable kernel function, $\mathbf{X} \in \mathbb{R}^{N \times D}$ is the data matrix, and we have used the short-hand notations $k_{*,\mathbf{z}} = k(\mathbf{z}_*, \mathbf{z}_{1:N}) \in \mathbb{R}^{1 \times N}$ and $k_{\mathbf{z},\mathbf{z}} = k(\mathbf{z}_{1:N}, \mathbf{z}_{1:N}) \in \mathbb{R}^{N \times N}$. Since we consider noise-free data, we have $\sigma^2 = 0$. The pull-back metric associated with this regression function is

$$\mathbf{M}_{\text{krr}}(\mathbf{z}_*) = \partial_{\mathbf{z}_*} k_{*,\mathbf{z}} k_{\mathbf{z},\mathbf{z}}^{-1} \mathbf{X}\mathbf{X}^\top k_{\mathbf{z},\mathbf{z}}^{-1} \partial_{\mathbf{z}_*} k_{*,\mathbf{z}}^\top. \tag{3.3}$$

Assuming a universal kernel [31], then f_{krr} will correspond to the true mapping where we have data when $N \to \infty$. Consequently, we recover the true metric where we have data, i.e. $\mathbf{M}_{\text{krr}} \xrightarrow{\text{near}} \mathbf{M}_{\text{true}}$.

[1] Technically, this render the manifold *immersed* rather than *embedded*; the distinction is not important for our purposes.

Teleports? The behavior away from data depend on the kernel. We first consider the Gaussian kernel

$$k_{\text{RBF}}(\mathbf{z}, \mathbf{z}') = \theta_{\text{RBF}} \cdot \exp\left(-\frac{\alpha}{2}\|\mathbf{z} - \mathbf{z}'\|^2\right), \tag{3.4}$$

but note that similar observations hold for most common stationary kernels. From this, we see that

$$f_{\text{RBF}} \xrightarrow{\text{away}} \mathbf{0} \qquad \text{and} \qquad \mathbf{M}_{\text{RBF}} \xrightarrow{\text{away}} \mathbf{0}. \tag{3.5}$$

To illustrate the geometric implication of this observation, we consider our guiding example. We compute shortest paths under the pull-back metric; if the true metric is recovered these should be circular arcs in $\mathcal{Z}$. Figure 3b show the recovered geodesics; we see that they systematically shy away from the data and generally do not resemble circular arcs. The explanation is simply that in terms of length-minimization, it is "free" to move through regions where the metric is zero. The result in Eq. 3.5, thus, implies that geodesics are encouraged to move away from the data. Intuitively, we can think of regions in $\mathcal{Z}$ without data as "teleports" that points can move freely between. This also hold true, when the manifold is densely sampled, and consequently geodesics will generally not move along the data manifold: *the manifold geometry is not recovered.*

Flat manifolds? These teleports appear because the chosen kernel cause f to extrapolate to a constant. We now consider a kernel that extrapolate linearly,

$$k_{\text{RBF+lin}}(\mathbf{z}, \mathbf{z}') = k_{\text{RBF}}(\mathbf{z}, \mathbf{z}') + \theta_{\text{lin}}\mathbf{z}^\top\mathbf{z}'. \tag{3.6}$$

Similarly to before, we see that

$$f_{\text{RBF+lin}} \xrightarrow{\text{away}} \theta_{\text{lin}}\mathbf{z}_*^\top\mathbf{Z}k_{\mathbf{z},\mathbf{z}}^{-1}\mathbf{X} = \mathbf{z}_*^\top\mathbf{B}, \tag{3.7}$$

where $\mathbf{B} = \theta_{\text{lin}}\mathbf{Z}k_{\mathbf{z},\mathbf{z}}^{-1}\mathbf{X} \in \mathbb{R}^{d\times D}$. This amounts to linear extrapolation, as expected. When we move away from the data, the metric then becomes

$$\mathbf{M}_{\text{RBF+lin}} \xrightarrow{\text{away}} \mathbf{B}\mathbf{B}^\top. \tag{3.8}$$

This is a (scaled) Euclidean metric, implying that the learned manifold is flat in regions where we do not have data. As the pull-back metric measure distances in $\mathcal{X}$, where straight lines are shortest curves, then geodesics on the learned manifold will be encouraged to go through the flat regions where data is missing. This is also evident in Fig. 3c that shows the results of our guiding example. Here we see that geodesics are almost straight lines, implying that the learned manifold did not recover the structure of the data.

A regularization perspective. The phenomenon can also be understood by considering regression functions f that minimize a local (log) likelihood function [21, 36]

$$\mathcal{L} = \sum_{n=1}^{N} w_n\mathcal{L}_n + \lambda\phi[f], \tag{3.9}$$

where $\mathcal{L}_n$ is the loss associated with the n^{th} observation, w_n is a weight that decays with the distance to the point where f is evaluated, and ϕ is the regularizer. Girosi et al. [13] show that most regularizers are low-pass filters (thereby avoiding "wiggly" regression functions). Away from the data, the regularizer dominate the above cost function, implying a strongly low-pass filtered regression function. By Parseval's theorem [34] this reduce the energy of curves that move away from the data, which bias geodesics to move away from the data.

An unstable solution? We have seen that the traditional constant and linear extrapolation schemes imply that we cannot learn the correct geometry: either we introduce teleports or we learn mostly flat manifolds. We now ask: *how should we extrapolate in order to learn the correct geometry?* For geodesics to stay on the manifold, inner products must take large values away from the data.

$$\mathbf{M} \xrightarrow{\text{away}} [\text{sufficiently large}]. \tag{3.10}$$

That is, to ensure that geodesics always stay on the manifold, length-minimization must be penalized sufficiently for leaving the manifold. We, currently, do not have a tight bound on how large the metric must be to ensure this, though a loose bound is provided by the radius of the manifold. That is, let

$$r = \sup_{\mathbf{z},\mathbf{z}' \in \mathcal{M}} \mathsf{dist}(\mathbf{z}, \mathbf{z}') \tag{3.11}$$

denote the largest distance between points on the manifold, then geodesics stay on the manifold if

$$\lambda_{\text{min}} \left(\mathbf{M}_{\text{sufficient}} \right) \xrightarrow{\text{away}} r^2. \tag{3.12}$$

Here λ_{min} denote the function returning the smallest eigenvalue of a matrix. In differential geometry, it is common to call *any* locally length-minimizing curve a geodesic. Here we mean the shortest geodesic. To ensure that *all* locally length-minimizing curves stay on the manifold, we have no tighter bound than

$$\lambda_{\text{min}} \left(\mathbf{M}_{\text{ideal}} \right) \xrightarrow{\text{away}} \infty. \tag{3.13}$$

If the metric must extrapolate to a large matrix, then the Jacobian $\mathbf{J}$ must also extrapolate to a large matrix (since $\mathbf{M} \propto \mathbf{J}^\top \mathbf{J}$). The results of Girosi et al. [13] dictate that regularizing towards functions with large derivatives imply that the solution is no longer low-pass filtered (i.e. it will "wiggle"). Regularizing towards large derivatives, thus, go against common wisdom as regression functions with large derivatives in regions of little data, will generally not be stable.

Summarizing discussion. Most common choices for estimating f will ensure that the geometry of $\mathcal{M}$ is well-estimated near the data, so the key factor to determine if we can well-estimate the manifold geometry is the behavior of f away from the data. In these regions, we depend on prior assumptions on f to determine the geometry. The most common priors are related to the smoothness of f, where we use the terminology of Girosi et al. [13], i.e. a function is considered "smooth" if its spectrum is dominated by low-frequency components. We have seen that the smoother assumptions we are willing to make, the more geodesics are drawn away from the data ("off the manifold", so to say).

As a specific example, consider a local likelihood (3.9) with regularizer

$$\phi[f] = \mathbb{E}\left[\left\| \frac{\partial f}{\partial \mathbf{z}} \right\|^2 \right] = \mathbb{E}\left[\operatorname{tr} \mathbf{J}^\top \mathbf{J} \right] = D\mathbb{E}\left[\operatorname{tr} \mathbf{M} \right]. \tag{3.14}$$

Bishop [3] has shown that this is (approximately) the regularizer implied by training under additive noise.[2] We see that this common regularizer imply a "small" metric, when moving away from the data, which in turn imply short geodesic segments away from the data. Minimization of *curve energy* (2.6) — or equivalently *curve length* (2.5) — will, thus, be biased towards regions of no data as this is where the metric is minimal.

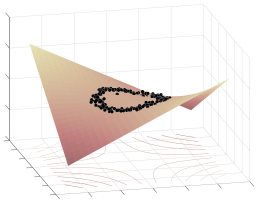

Figure 4: Smooth interpolation of data imply that shortest paths measured in data space cut across holes.

The general phenomenon is easily understood by considering a data manifold with a hole. If the applied regression function is very smooth, then the hole will be interpolated almost linearly, which imply that shortest paths along the manifold will cross over the hole. In the end, we are, thus, left with a simple choice: *either give up on learning the manifold geometry correctly* (by assuming f is very smooth) or *give up on stable learning* (by assuming f is not very smooth). Neither choice is desirable.

3.2 The Bayesian setting

As a natural probabilistic extension of the previous sections we now let f consist of component-wise conditionally independent Gaussian processes (GPs) [26],

$$f_i(\mathbf{z}) \sim \mathcal{GP}(m_i(\mathbf{z}), k(\mathbf{z}, \mathbf{z}')), \quad \forall i = 1, \dots, D. \tag{3.15}$$

This is a *Gaussian Process latent variable model (GP-LVM)* [20]. Here m_i and k are the mean and covariance functions of the i^{th} GP. Note that the posterior mean function coincide with the previously considered kernel ridge regression model. Like Lawrence [20], we assume the same covariance

[2] As an example, the mean decoder of a VAE is trained under this regularizer.

function across all dimensions to simplify calculations. The geometry of this model was first studied by Tosi et al. [37].

The pull-back metric $\mathbf{M} = {}^1\!/_D \mathbf{J}^\top \mathbf{J}$ is now a stochastic Riemannian metric since f is stochastic. As Gaussian variables are closed under differentiation, then $\mathbf{J}$ is Gaussian, $\mathbf{J} \sim \prod_{j=1}^{D} \mathcal{N}(\mu(j,:), \mathbf{\Sigma})$, and $\mathbf{M}$ follows a non-central Wishart distribution [37, 22]

$$D \cdot \mathbf{M} \sim \mathcal{W}_d(D, \mathbf{\Sigma}, \mathbf{\Sigma}^{-1}\mathbb{E}[\mathbf{J}]^\top \mathbb{E}[\mathbf{J}]). \tag{3.16}$$

The entire metric by definition follows a generalized Wishart process [38, 16]. A sample path from this process is smooth when the covariance function k is also smooth, and we have a proper distribution over Riemannian metrics. However, a sample from f gives a manifold that is only locally diffeomorphic to d-dimensional Euclidean space, and it may globally self-intersect.

Since the metric is a stochastic variable, we cannot apply standard Riemannian geometry to understand the space $\mathcal{Z}$ (curvature is stochastic, geodesics are stochastic, etc). We can, however, inspect the leading moments of the metric

$$\mathbb{E}[\mathbf{M}] = \frac{1}{D}\mathbb{E}[\mathbf{J}^\top \mathbf{J}] = \frac{1}{D}\mathbb{E}[\mathbf{J}]^\top \mathbb{E}[\mathbf{J}] + \mathbf{\Sigma} \tag{3.17}$$

$$\mathsf{var}\,[M_{ij}] = \frac{\Sigma_{ij}^2 + \Sigma_{ii}\Sigma_{jj}}{D} + \frac{\mu_j^\top \mathbf{\Sigma} \mu_j}{D^2} + \frac{\mu_i^\top \mathbf{\Sigma} \mu_i}{D^2}. \tag{3.18}$$

We see that $\mathbb{E}[\mathbf{M}]$ remain strictly positive, while $\mathsf{var}\,[M_{ij}] = \mathcal{O}\left({}^1\!/_D\right)$ vanishes in the limit $D \to \infty$. This can equivalently be seen from the central limit theorem. In high dimensions, the metric, thus, becomes deterministic even if the underlying manifold is stochastic. This is useful as it implies that we can well-approximate the stochastic metric with a well-understood deterministic metric.

To see if this approach can learn the geometric structure of the data manifold, we again consider the Gaussian kernel (3.4). Straight-forward calculations show that

$$\mathbf{\Sigma} \xrightarrow{\text{near}} \mathbf{0} \qquad \text{and} \qquad \mathbf{\Sigma} \xrightarrow{\text{away}} \alpha\theta_{\text{RBF}}\mathbf{I}, \tag{3.19}$$

where α and θ_{RBF} are the kernel parameters. From this we see that near the data, the expected metric (3.17) coincides with the true pull-back metric of the manifold (as in the deterministic setting),

$$\mathbb{E}[\mathbf{M}] \xrightarrow{\text{near}} \frac{1}{D}\mathbb{E}[\mathbf{J}]^\top \mathbb{E}[\mathbf{J}]. \tag{3.20}$$

In regions of $\mathcal{Z}$ where there is no data, we have

$$\mathbb{E}[\mathbf{M}] \xrightarrow{\text{away}} \frac{1}{D}\mathbb{E}[\mathbf{J}]^\top \mathbb{E}[\mathbf{J}] + \alpha\theta_{\text{RBF}}\mathbf{I}. \tag{3.21}$$

If $\alpha\theta_{\text{RBF}}$ is sufficiently large then geodesics will not go through regions of $\mathcal{Z}$ where we do not have data. When data is sampled densely on the manifold, we often estimate large values of α (corresponding to a less smooth manifold), and a large penalty will be payed for "falling off the manifold". To validate these observations, we return to our guiding example; Fig. 3d shows that geodesics under the expected metric (3.17) actually follow approximately circular arcs. This aligns with the theoretical analysis and demonstrates that, unlike a deterministic method, a probabilistic method can actually learn the differential geometric structure of a data manifold. This is, however, no guarantee: we can only accurately learn the manifold geometry when data is sampled sufficiently dense on the manifold, but now there is hope, whereas deterministic approaches are bound to fail.

Discussion. Deterministic methods can capture local geometry of the data manifold near the observed data, but they fail to capture the geometry where data is missing. This is not surprising, as we can generally only learn when we have data. What is, perhaps, more surprising is that if we can estimate the uncertainty of the manifold, then that translate directly into geometric information: if there is a hole in the manifold, then we can only see it through a lens of uncertainty. Not quantifying the uncertainty prevents us from seeing holes and boundaries of a data manifold. *Uncertainty, thus, plays the same role as topology* in classic geometry, and this must also be estimated from data.

4 Bayesian geometry

As the expected metric can capture the geometry of the data manifold, we seek a better understanding of stochastic Riemannian metrics. We here consider the case of the GP-LVM where f is a smooth GP.

4.1 Detour: Euclidean geometry

Before analyzing stochastic Riemannian metrics, consider a stochastic Euclidean metric. Let $\mathbf{u}, \mathbf{v} \in \mathbb{R}^d$ denote deterministic vectors, and define their stochastic inner product

$$\langle \mathbf{u}, \mathbf{v} \rangle = \big(\mathbf{A}\mathbf{u}\big)^\top \big(\mathbf{A}\mathbf{v}\big) = \mathbf{u}\big(\mathbf{A}^\top \mathbf{A}\big)\mathbf{v}, \tag{4.1}$$

where $\mathbf{A} \in \mathbb{R}^{D\times d}$ is a matrix-valued random variable. Under this inner product, the shortest path between two points is the straight line, so stochasticity does not change our usual intuitions. The length of this line is, however, stochastic. Its expectation is found by letting $\mathbf{\Delta} = \mathbf{A}(\mathbf{u} - \mathbf{v})$, then

$$\mathbb{E}\big[\|\mathbf{\Delta}\|^2\big] = \mathbb{E}[\mathbf{\Delta}]^\top \mathbb{E}[\mathbf{\Delta}] + \operatorname{tr}\left(\operatorname{cov}\left[\mathbf{\Delta}\right]\right). \tag{4.2}$$

The expected distance under a stochastic Euclidean metric grows with both mean and variance of the basis $\mathbf{A}$. Hence, expected distances are inherently large when the basis (or equivalently the metric) has large variance.

Example 1 (a Gaussian basis)**.** Let each row of $\mathbf{A}$ be drawn independently from $\mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$, such that the metric follows a Wishart distribution [22],

$$\mathbf{M} = \mathbf{A}^\top \mathbf{A} \sim \mathcal{W}_d(D, \mathbf{\Sigma}). \tag{4.3}$$

Then the distance from $\mathbf{u}$ to $\mathbf{v}$ is Nakagami distributed [19]

$$\mathsf{dist}(\mathbf{u}, \mathbf{v}) \sim \text{Nakagami}\left(\frac{D}{2}, D\sigma^2_{\mathbf{u},\mathbf{v}}\right), \tag{4.4}$$

and the expected distance is [15]

$$\mathbb{E}\left[\mathsf{dist}(\mathbf{u}, \mathbf{v})\right] = \frac{\Gamma\left(\frac{D+1}{2}\right)}{\Gamma\left(\frac{D}{2}\right)}\sqrt{2}\sigma_{\mathbf{u},\mathbf{v}} \propto \sigma_{\mathbf{u},\mathbf{v}}, \tag{4.5}$$

$$\text{where} \quad \sigma^2_{\mathbf{u},\mathbf{v}} = (\mathbf{u} - \mathbf{v})^\top \mathbf{\Sigma}(\mathbf{u} - \mathbf{v}). \tag{4.6}$$

We see that the expected distance correspond to Mahalanobis' distance using the inverse covariance. Also note that scaling $\mathbf{\Sigma}$ also scales the expected distance: *very uncertain metrics imply large expected distances.*

4.2 Geodesics

We now return to the geometry of the GP-LVM, and seek to understand shortest paths. Let $\mathbf{c} : [a, b] \to \mathcal{Z}$ denote a deterministic differentiable curve, and let $f(\mathbf{c})$ denote its stochastic embedding in $\mathcal{X}$. We stress that $\mathbf{c}$ is a deterministic curve in $\mathcal{Z}$, while $f(\mathbf{c})$ is a GP in $\mathcal{X}$. The energy (2.6) of $f(\mathbf{c})$ is a random quantity and it is natural to consider its expectation with respect to the random metric. Since the energy integrand is positive, Tonelli's Theorem tells us that this expected energy is

$$\bar{\mathcal{E}}(\mathbf{c}) \equiv \mathbb{E}_{\mathbf{M}}\left[\mathcal{E}(f(\mathbf{c}))\right] = \frac{1}{2}\mathbb{E}_{\mathbf{M}}\left[\int_a^b \dot{\mathbf{c}}_t^\top \mathbf{M}_{\mathbf{c}_t} \dot{\mathbf{c}}_t \mathrm{d}t\right] \tag{4.7}$$

$$= \frac{1}{2}\int_a^b \dot{\mathbf{c}}_t^\top \mathbb{E}\left[\mathbf{M}_{\mathbf{c}_t}\right] \dot{\mathbf{c}}_t \mathrm{d}t. \tag{4.8}$$

This implies that the curve $\mathbf{c}$ with minimal expected energy over the stochastic manifold, is the geodesic under the deterministic Riemannian metric $\mathbb{E}\left[\mathbf{M}\right]$. This is exactly the metric considered in Sec. 3.2.

We can understand the curve minimizing expected energy in more explicit terms as follows. Let $u_t = \mathbb{E}[\|\dot{\mathbf{c}}_t\|]$ and $v_t = 1$ denote two functions over the interval $[a, b]$; here we use the short-hand

notation $\|\dot{\mathbf{c}}_t\| = \sqrt{\dot{\mathbf{c}}_t^\top \mathbf{M}_{\mathbf{c}_t} \dot{\mathbf{c}}_t}$. Then Cauchy-Scwartz's inequality tells us that

$$|\langle u, v \rangle|^2 \leq \|u\|^2 \cdot \|v\|^2 \tag{4.9}$$

$$\left(\int_a^b \mathbb{E}[\|\dot{\mathbf{c}}_t\|] \mathrm{d}t \right)^2 \leq \int_a^b \mathbb{E}[\|\dot{\mathbf{c}}_t\|]^2 \mathrm{d}t \cdot \int_a^b \mathrm{d}t \tag{4.10}$$

$$= (b-a) \int_a^b \mathbb{E}[\|\dot{\mathbf{c}}_t\|]^2 \mathrm{d}t. \tag{4.11}$$

Let

$$\bar{\mathcal{L}}(\mathbf{c}) = \mathbb{E}\left[\int_a^b \|\dot{\mathbf{c}}_t\| \mathrm{d}t \right] = \int_a^b \mathbb{E}[\|\dot{\mathbf{c}}_t\|] \mathrm{d}t \tag{4.12}$$

denote the expected length of $\mathbf{c}$ then

$$\int_a^b \mathbb{E}[\|\dot{\mathbf{c}}_t\|]^2 \mathrm{d}t \geq \frac{\bar{\mathcal{L}}^2(\mathbf{c})}{b-a}. \tag{4.13}$$

Equality is achieved when u_t and v_t are parallel, that is when $\mathbb{E}[\|\dot{\mathbf{c}}_t\|]$ is constant. We can always reparametrize $\mathbf{c}_t$ to have constant expected speed and achieve equality. Since $\mathsf{var}\,[x] = \mathbb{E}[x^2] - \mathbb{E}[x]^2$, we see that

$$\begin{aligned} \int_a^b \mathbb{E}[\|\dot{\mathbf{c}}_t\|]^2 \mathrm{d}t &= \int_a^b \mathbb{E}[\|\dot{\mathbf{c}}_t\|^2] \mathrm{d}t - \int_a^b \mathsf{var}\left[\|\dot{\mathbf{c}}_t\|\right] \mathrm{d}t \\ &= 2\bar{\mathcal{E}}(\mathbf{c}) - \int_a^b \mathsf{var}\left[\|\dot{\mathbf{c}}_t\|\right] \mathrm{d}t. \end{aligned} \tag{4.14}$$

Assuming that the curve has been parametrized to have constant expected speed, we then get

$$\bar{\mathcal{E}}(\mathbf{c}) = \frac{\bar{\mathcal{L}}^2(\mathbf{c})}{2(b-a)} + \frac{1}{2} \int_a^b \mathsf{var}\left[\|\dot{\mathbf{c}}_t\|\right] \mathrm{d}t. \tag{4.15}$$

Minimizing expected curve energy, thus, does not always minimize the expected curve length. Rather, this balances the minimization of expected curve length and the minimization of curve variance.

Implications and interpretation. On a deterministic manifold, minimizing curve energy results in a curve of minimal length (by Proposition 1). When the manifold is stochastic, we see that minimizing expected curve energy does not imply a minimization of expected curve length. In some sense, this is disappointing. Yet, it is intriguing that minimizing expected energy corresponds to minimizing a combination of length and variance. Since the expected energy minimizing curve is the geodesic under the expected Riemannian metric, this also lend itself to easy computation.

Example 2 (a GP prior manifold)**.** There are cases where expected energy and length are strongly related. Let each dimension of f be a zero-mean GP with a sufficiently smooth covariance function. Such processes are common for specifying priors. Let the Jacobian of f at $\mathbf{z}$ be

$$\mathbf{J}_\mathbf{z} \sim \prod_{j=1}^{D} \mathcal{N}(\mathbf{0}, \mathbf{\Sigma}_\mathbf{z}), \tag{4.16}$$

such that

$$D \cdot \mathbf{M}_\mathbf{z} = \mathbf{J}_\mathbf{z}^\top \mathbf{J}_\mathbf{z} \sim \mathcal{W}_d(D, \mathbf{\Sigma}_\mathbf{z}). \tag{4.17}$$

The expected energy of a curve $\mathbf{c} : [a, b] \to \mathcal{Z}$ is then

$$\bar{\mathcal{E}}(\mathbf{c}) = \frac{1}{2} \int_a^b \mathbb{E}\left[\dot{\mathbf{c}}_t^\top \mathbf{M}_{\mathbf{c}_t} \dot{\mathbf{c}}_t\right] \mathrm{d}t = \frac{D}{2} \int_a^b \dot{\mathbf{c}}_t^\top \mathbf{\Sigma}_{\mathbf{c}_t} \dot{\mathbf{c}}_t \mathrm{d}t \tag{4.18}$$

since $\dot{\mathbf{c}}_t^\top \mathbf{M}_{\mathbf{c}_t} \dot{\mathbf{c}}_t \sim \mathcal{W}_1(D, \dot{\mathbf{c}}_t^\top \boldsymbol{\Sigma}_{\mathbf{c}_t} \dot{\mathbf{c}}_t)$. Following Example 1 we see that the expected length of $\mathbf{c}$ is

$$\bar{\mathcal{L}}(\mathbf{c}) \propto \int_a^b \sqrt{\dot{\mathbf{c}}_t^\top \boldsymbol{\Sigma}_{\mathbf{c}_t} \dot{\mathbf{c}}_t} \mathrm{d}t. \tag{4.19}$$

We can then interpret $\boldsymbol{\Sigma}_{\mathbf{z}}$ as a Riemannian metric and note that the expressions for curve length and energy under this metric correspond to Eqs. 4.19 and 4.18. By Proposition 1, we then have that minimizing expected curve energy also minimize expected curve length.

4.3 Integration

Let $\Omega \subseteq \mathcal{Z}$ such that $f(\Omega) \subseteq \mathcal{M}$, and let $h : f(\Omega) \to \mathbb{R}$ denote a real-valued integratable function over the specified part of the manifold. We now seek its integral over the entire domain under a random Riemannian metric $\mathbf{M}$. That is,

$$\bar{h} = \int_{f(\Omega)} h(\mathbf{x}) \mathrm{d}\mathbf{x} = \int_\Omega h(f(\mathbf{z})) \sqrt{\det \mathbf{M}_{\mathbf{z}}} \mathrm{d}\mathbf{z}. \tag{4.20}$$

Since the metric is stochastic, then so is $\bar{h}$. We evaluate its expectation as

$$\mathbb{E}_{\mathbf{M}}[\bar{h}] = \mathbb{E}_{\mathbf{M}} \left[\int_\Omega h(f(\mathbf{z})) \sqrt{\det \mathbf{M}_{\mathbf{z}}} \right] \mathrm{d}\mathbf{z} \tag{4.21}$$

$$= \int_\Omega h(f(\mathbf{z})) \mathbb{E} \left[\sqrt{\det \mathbf{M}_{\mathbf{z}}} \right] \mathrm{d}\mathbf{z}. \tag{4.22}$$

That is, the integration is simply performed under the expected volume measure (as expected).

The variance of the integral can be expressed as

$$\mathsf{var}\left[\bar{h}\right] = \mathbb{E}\left[\bar{h}^2\right] - \mathbb{E}\left[\bar{h}\right]^2, \tag{4.23}$$

where the last term easily is computed from Eq. 4.22. The missing term is

$$\mathbb{E}\left[\bar{h}^2\right] = \mathbb{E}_{\mathbf{M}} \left[\left(\int_\Omega h(f(\mathbf{z})) \sqrt{\det \mathbf{M}_{\mathbf{z}}} \mathrm{d}\mathbf{z} \right)^2 \right], \tag{4.24}$$

which generally does not permit a closed-form expression.

Since geodesics under the expected metric are well-behaved, it is tempting to treat the manifold as having this metric. From an integration point-of-view this implies working with the measure $\sqrt{\det(\mathbb{E}[\mathbf{M}_{\mathbf{z}}])}$ as suggested by Arvanitidis et al. [1]. The above analysis, however, indicate that it is perhaps more suitable to use the expected measure (4.22).

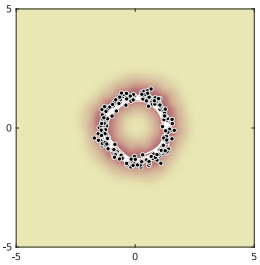

Figure 5: The expected measure of the metric associated with the GP-LVM. Notice that the plot is practically identical to Fig. 3d.

Example 3 (a GP prior manifold)**.** There are cases where the choice of measure is of less importance. To see this, we return to the Gaussian process prior manifold of Example 2. As before, the metric at $\mathbf{z}$ follows a (scaled) Wishart distribution, and Theorem 3.2.15 of Muirhead's book [22] tells us that

$$\mathbb{E}\left[\sqrt{\det(D\mathbf{M}_{\mathbf{z}})}\right] = \sqrt{2^d \det \boldsymbol{\Sigma}_{\mathbf{z}}} \frac{\Gamma\left(\frac{D+1}{2}\right)}{\Gamma\left(\frac{D-d+1}{2}\right)} \Leftrightarrow \tag{4.25}$$

$$\mathbb{E}\left[\sqrt{\det(\mathbf{M}_{\mathbf{z}})}\right] \propto \sqrt{\det \boldsymbol{\Sigma}_{\mathbf{z}}}. \tag{4.26}$$

The measure associated with the expected metric is $\sqrt{\det \mathbb{E}\left[\mathbf{M}_{\mathbf{z}}\right]} = \sqrt{\det \boldsymbol{\Sigma}_{\mathbf{z}}}$ and we conclude that for this prior manifold, the two measures are proportional.

Example 4 (GP-LVM)**.** Things are not as simple when considering the posterior GP-LVM manifold. Here the (scaled) metric at $\mathbf{z}$ follows a non-central Wishart distribution

$$D \cdot \mathbf{M} \sim \mathcal{W}_d(D, \mathbf{\Sigma}_\mathbf{z}, \mathbf{\Sigma}_\mathbf{z}^{-1}\mathbb{E}[\mathbf{J}]^\top\mathbb{E}[\mathbf{J}]). \tag{4.27}$$

By Theorem 10.3.7 of Muirhead's book [22] we get that

$$\begin{aligned}\mathbb{E}\Big[\sqrt{\det(D\mathbf{M}_\mathbf{z})}\Big] = \frac{2^{d/2}}{\pi D^{d/2}} \frac{\Gamma\left(\frac{D+2}{4}\right)}{\Gamma\left(\frac{D-d+2}{4}\right)} \sqrt{\det \mathbf{\Sigma}_\mathbf{z}} \\ \cdot {}_1F_1\left(-1/2, D/2, -1/2\mathbf{\Sigma}_\mathbf{z}^{-1}\mathbb{E}[\mathbf{J}]^\top\mathbb{E}[\mathbf{J}]\right),\end{aligned} \tag{4.28}$$

where ${}_1F_1$ is the confluent hypergeometric function of the first kind. On the other hand, the measure associated with the expected metric is

$$\sqrt{\det \mathbb{E}\left[\mathbf{M}_\mathbf{z}\right]} = \sqrt{\det\left(1/D\, \mathbb{E}[\mathbf{J}_\mathbf{z}]^\top\mathbb{E}[\mathbf{J}_\mathbf{z}] + \mathbf{\Sigma}_\mathbf{z}\right)} \tag{4.29}$$

and we see that the two measures appear quite different. To understand this difference in practice, we show the volume measure of the expected metric (4.29) in the background of Fig. 3d. Similarly, we show the expected volume measure (4.28) in Fig. 5. Somewhat surprisingly, there is no visual difference between the two different measures. This indicates that for the GP-LVM, the more simple volume measure associated with the expected metric may be a good approximation to the expected volume measure.

5 Deep generative models

So far, we have studied kernel based methods due to their ease. The key observations, however, generally hold true, and we now consider neural networks.

5.1 The deterministic setting

The natural case is to estimate f with a (potentially deep) feed-forward neural network. We call this an *autoencoder* as these classic methods are the prime example of such an architecture [30], though we note that other models such as *generative adversarial networks* [14] also fall within this category. When we consider the associated pull-back metric, then the same considerations hold true as in the kernel-based setting (Sec. 3.1). That is, if we regularize towards a smooth f, then geodesics will naturally cross through holes in the data manifold. To validate this, Fig. 6a shows that the geodesics of our guiding example are almost straight lines. As before, *the lack of uncertainty prevent us from learning the manifold topology and geometry.*

5.2 The Bayesian setting

In the neural networks literature, (Gaussian) probabilistic mappings f are commonly represented as [23]

$$f(\mathbf{z}) = \boldsymbol{\mu}(\mathbf{z}) + \text{diag}(\epsilon)\boldsymbol{\sigma}(\mathbf{z}), \qquad \epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_D), \tag{5.1}$$

where $\boldsymbol{\mu}, \boldsymbol{\sigma} : \mathbb{R}^d \to \mathbb{R}^D$ are neural networks, $\text{diag}(\cdot)$ constructs a diagonal matrix from a vector, and $\mathbf{I}_D$ is the $D \times D$ identity matrix. This is a key part of variational autoencoders (VAEs) [17, 27] that realize the generative model $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, $\mathbf{x} = f(\mathbf{z})$. From a geometric perspective it is worth noting that the noise ϵ does not form a smooth process. As such, sample paths from f are not smooth, and one can question the validity of pull-back metrics of this model. If we disregard any such concerns, then it is easy to show that [1]

$$D \cdot \mathbb{E}[\mathbf{M}_\mathbf{z}] = \left(\mathbf{J}_\mathbf{z}^{(\boldsymbol{\mu})}\right)^\top\left(\mathbf{J}_\mathbf{z}^{(\boldsymbol{\mu})}\right) + \left(\mathbf{J}_\mathbf{z}^{(\boldsymbol{\sigma})}\right)^\top\left(\mathbf{J}_\mathbf{z}^{(\boldsymbol{\sigma})}\right), \tag{5.2}$$

where $\mathbf{J}_\mathbf{z}^{(\boldsymbol{\mu})}$ and $\mathbf{J}_\mathbf{z}^{(\boldsymbol{\sigma})}$ are the Jacobians of $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$, respectively. As before, the variance of the metric also goes to zero when $D \to \infty$, due to the central limit theorem. This can be taken as a hint that the expected metric is a reasonable geometric structure for $\mathcal{Z}$.

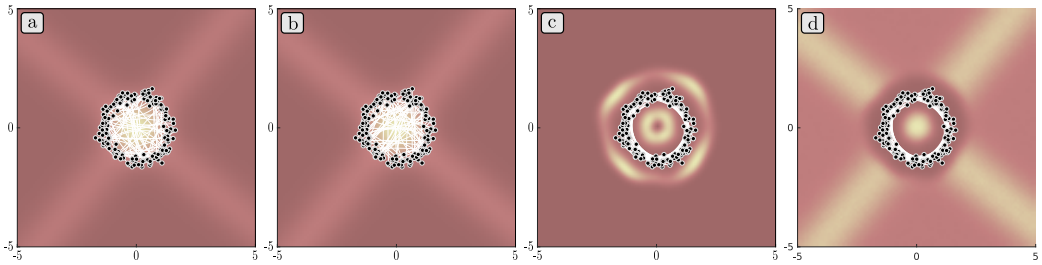

Figure 6: Geodesics for various autoencoders. (a) Here f is a smooth feed-forward network. (b) A naive VAE where both mean and standard deviation of f are smooth feed-forward networks. (c) A VAE with decaying precision, i.e. the inverse standard deviation of f is a positive RBF network. (d) Same network as (c) but with background color proportional to the expected measure.

A perhaps more sensible way to arrive at Eq. 5.2 is to view Eq. 5.1 as a random projection of the deterministic manifold spanned by

$$f : \mathbb{R}^d \to \mathbb{R}^{2D}, \qquad f(\mathbf{z}) = \begin{pmatrix} \boldsymbol{\mu}(\mathbf{z}) \\ \boldsymbol{\sigma}(\mathbf{z}) \end{pmatrix}. \tag{5.3}$$

Stacking $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$ imply that their Jacobians stack as well, such that Eq. 5.2 is the pull-back metric associated with Eq. 5.3.

Returning to our guiding example, we let $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$ be smooth feed-forward neural networks. Figure 6b shows that recovered geodesics are almost straight lines, i.e. the model failed to capture the data geometry. This is because σ is a poor proxy for uncertainty [1]. When σ is a feed-forward neural network, we assume that we can smoothly interpolate the uncertainty estimates recovered at $\mathbf{z}_{1:N}$. But smooth interpolation of uncertainty is nonsensical.[3] To counter this, Arvanitidis et al. [1] model $\boldsymbol{\sigma}^{-1}$ with a positive RBF network [25], which ensure that uncertainty grows away from the data, and Fig. 6c shows that it allows us to recover the geometry of the data manifold.

In Example 4, we saw that for the GP-LVM the measure associated with the expected metric was practically identical to the expected measure, even if their mathematical expressions are quite different. This does not appear to be the case for the variational autoencoder. Figure 6c shows the measure associated with the expected metric, Fig. 6d shows the expected measure (computed using sampling). Unlike for the GP-LVM, we now see significant differences between the two measures. At this stage, it is unclear which measure is to be preferred from a practical perspective. We see that both measures exhibit a somewhat arbitrary behavior, which we take as a hint that the RBF network for inverse variance does not provide an excellent fit to the data.

6 Quantitative summary

Throughout the previous section we have used a simple example to illustrate the fundamental bias in deterministic geometry estimation. For completeness, we here quantify these results, but emphasize that our main objective is to understand this bias, rather than to perform empirical studies. Since we know that ground truth geodesics are circular arcs, we can compare these to estimated geodesics. As a first measure of quality, we compute the correlation between the estimated geodesic lengths and the length of the ground truth geodesics. Table 1 show this correlation for each considered model, and individual correlation plots are found in Appendix B. We observe that the GP-LVM achieve an almost perfect correlation, while deterministic kernel methods fare significantly worse. Individual correlation plots show that for deterministic models short curves provide a better

Method	corr	Hausdorff
GP-LVM	0.996819	0.858773
KRM	0.843259	5.443746
KRM+ridge	0.892995	3.467452
AE	0.974833	2.825305
VAE	0.975200	2.826954
RBF VAE	0.982504	0.788519

Table 1: Correlation between length of true and estimated geodesics, and Hausdorff distances between these curves.

[3] Consider two low-variance temperature readings at the poles of our planet. If we, from this data, interpolate the temperature at the equator, then a smooth interpolation of the uncertainty would imply a very certain prediction.

correlation than long curves, which is in line with the Riemannian assumption of a locally Euclidean model. All autoencoder-based models achieve a strong correlation, which is somewhat surprising given the almost straight geodesics found by deterministic methods.

As a second quality measure, we report the average Hausdorff distance [28] between estimated and ground truth geodesics (Table 1). We see that the GP-LVM and the VAE with RBF variance estimation significantly outperform the other methods. This match the visual observations made throughout the paper.

7 Previous work

Pull-back metrics have been studied in mathematics at least since the seminal work of Gauss [12], and formed the initial foundation of Riemannian geometry. In machine learning, these metrics have only been studied in few instances. Tosi et al. [37] was the first to give the latent space of the GP-LVM [20] a geometric foundation. Bishop et al. [4] used the deterministic volume measure of a *Generative Topographic Map (GTM)* [5] as a visualization tool; our analysis imply that incorporating uncertainty should improve such a tool. Recently, several authors have studied the geometry of deep generative models [33, 7, 18, 1]. Shao et al. [33] and Chen et al. [7] consider pull-back metrics of VAEs, but only consider the mean of f; in our terminology they therefore consider autoencoders rather than variational autoencoders. Shao et al. note that most geodesics in their model are straight lines and speculate that this is because most data manifold are actually flat. Our analysis shows that this conclusion is most likely incorrect, and that flatness is an artifact of disregarding uncertainty. Arvanitidis et al. [1] also considered pull-back metrics of variational autoencoders by taking the expected metric. Here significant curvature is reported, which coincides with intuition.

As an alternative to estimating a geometry that also reflect topology, one can bias the estimated geodesics such that they are attracted to data. Chen et al. [7] initialize geodesics by a density maximizing curve before optimizing curve energy, and in later work Chen et al. [6] force geodesics to be polygonal curves that interpolate the data. From our perspective, these approaches work around a deeper more fundamental problem, as it is not clear if these biased curves correspond to geodesics under any metric (they most surely do not minimize the energy associated with the deterministic pull-back metric).

8 Concluding remarks

The driving motivation for introducing pull-back metrics in the latent space of a generative model is to avoid arbitrariness in parametrizing the latent space. This is an important issue if we are to interpret the latent variables of a fitted model. We have argued that geometry provides *a* solution to the issue, but emphasize that this need not be the only solution. We have demonstrated that methods that do not quantify their uncertainty cannot, in a meaningful way, capture the geometry of a data manifold. The key issue is that the usual smoothness assumptions imply that holes in the data manifold are interpolated so smoothly that geodesics are encouraged to pass through the holes rather than stay on the manifold. Methods that provide reasonable estimates of the uncertainty of the estimated manifold naturally avoid this issue as the uncertainty directly alters the estimated geometry. We find that uncertainty quantification in manifold learning ends up playing the role of topology in classic geometry: uncertainty informs us about holes and boundaries in the manifold and provides us with a global notion of connectivity. Disregarding uncertainty, thus, imply disregarding the most fundamental aspects manifold learning.

We have provided an extensive analysis of the geometry of Gaussian latent variable models, and have developed an elementary theory for stochastic Riemannian manifolds. This appear to be the first of its kind. We have seen that minimizing expected curve energy on a stochastic manifold does not imply minimization of expected curve length, which forces us to reconsider which measure define the most natural interpolants. A more formal treatment of this material alongside approximation bounds when using expected metrics have recently appeared [10].

Parts of our analysis can be extended to models based on neural networks. This raises two key issues for future research: 1) since sample paths from deep generative models are not continuous it is perhaps not a good idea to enforce a geometric analysis. It then becomes interesting to determine if such smoothness can be introduced in deep generative models without sacrificing the computational

efficiency of the models. 2) Uncertainty is essential for estimating the geometric structure of a data manifold, but current deep generative models provide rather poor estimators of uncertainty. A heuristic from Arvanitidis et al. [1] seems to work, but more principled methods would be valuable; Detlefsen et al. [8] provide some early work, but the question largely remain open.

Finally, we repeat the key point of the paper: *without uncertainty quantification, we cannot learn the geometric structure of a data manifold, and any attempt to do so is bound to fail beyond the most simple examples.*

Acknowledgments

The author is grateful to Vagn Lundsgaard Hansen, Martin Jørgensen, Georgios Arvanitidis, Lars Kai Hansen, David Eklund and Aasa Feragen for enlightening discussions. SH was supported by a research grant (15334) from VILLUM FONDEN. This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement n^{o} 757360).

References

[1] G. Arvanitidis, L. K. Hansen, and S. Hauberg. Latent space oddity: on the curvature of deep generative models. In *International Conference on Learning Representations (ICLR)*, 2018.

[2] M. Belkin and P. Niyogi. Laplacian Eigenmaps for Dimensionality Reduction and Data Representation. *Neural Computation*, 15(6):1373–1396, June 2003.

[3] C. M. Bishop. Training with noise is equivalent to tikhonov regularization. *Neural computation*, 7(1): 108–116, 1995.

[4] C. M. Bishop, M. Svensen, and C. K. Williams. Magnification factors for the gtm algorithm. 1997.

[5] C. M. Bishop, M. Svensén, and C. K. Williams. Gtm: The generative topographic mapping. *Neural computation*, 10(1):215–234, 1998.

[6] N. Chen, F. Ferroni, A. Klushyn, A. Paraschos, J. Bayer, and P. van der Smagt. Fast approximate geodesics for deep generative models. *arXiv preprint arXiv:1812.08284*, 2018.

[7] N. Chen, A. Klushyn, R. Kurle, X. Jiang, J. Bayer, and P. Smagt. Metrics for deep generative models. In *International Conference on Artificial Intelligence and Statistics*, pages 1540–1550, 2018.

[8] N. S. Detlefsen, M. Jørgensen, and S. Hauberg. Reliable training and estimation of variance networks. *arXiv preprint arXiv:1906.03260*, 2019.

[9] D. L. Donoho and C. Grimes. Hessian eigenmaps: Locally linear embedding techniques for high-dimensional data. *Proceedings of the National Academy of Sciences*, 100(10):5591–5596, 2003. ISSN 0027-8424.

[10] D. Eklund and S. Hauberg. Expected path length on random manifolds, 2019.

[11] S. Gallot, D. Hulin, and J. Lafontaine. *Riemannian geometry*, volume 3. Springer, 1990.

[12] C. F. Gauss. Disquisitiones generales circa superficies curvas. *Commentationes Societatis Regiae Scientiarum Gottingesis Recentiores*, VI:99–146, 1827.

[13] F. Girosi, M. Jones, and T. Poggio. Regularization theory and neural networks architectures. *Neural computation*, 7(2):219–269, 1995.

[14] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[15] S. Hauberg. The non-central nakagami distribution. 2018.

[16] S. Hauberg. On the geometry of latent variable models. *Oberwolfach reports : OWR*, (3), 2018.

[17] D. P. Kingma and M. Welling. Auto-Encoding Variational Bayes. In *Proceedings of the 2nd International Conference on Learning Representations (ICLR)*, 2014.

[18] S. Laine. Feature-based metrics for exploring the latent space of generative models, 2018. URL `https://openreview.net/forum?id=BJslDBkwG`.

[19] D. Laurenson. Nakagami distribution. *Indoor Radio Channel Propagation Modelling by Ray Tracing Techniques*, 1994.

[20] N. D. Lawrence. Probabilistic non-linear principal component analysis with gaussian process latent variable models. *Journal of machine learning research*, 6(Nov):1783–1816, 2005.

[21] C. Loader. *Local Regression and Likelihood*. Springer, New York, 1999.

[22] R. J. Muirhead. *Aspects of Multivariate Statistical Theory*. John Wiley & Sons, 2005.

[23] D. A. Nix and A. S. Weigend. Estimating the mean and variance of the target probability distribution. In *Proceedings of 1994 IEEE International Conference on Neural Networks (ICNN'94)*, volume 1, 1994.

[24] X. Pennec. Intrinsic Statistics on Riemannian Manifolds: Basic Tools for Geometric Measurements. *Journal of Mathematical Imaging and Vision*, 25(1):127–154, July 2006.

[25] Q. Que and M. Belkin. Back to the future: Radial basis function networks revisited. In *Artificial Intelligence and Statistics (AISTATS)*, 2016.

[26] C. E. Rasmussen and C. Williams. *Gaussian Processes for Machine Learning*. University Press Group Limited, 2006.

[27] D. J. Rezende, S. Mohamed, and D. Wierstra. Stochastic backpropagation and variational inference in deep latent gaussian models. In *International Conference on Machine Learning*, volume 2, 2014.

[28] R. T. Rockafellar and R. J.-B. Wets. *Variational analysis*, volume 317. Springer Science & Business Media, 2009.

[29] S. T. Roweis and L. K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *science*, 290(5500):2323–2326, 2000.

[30] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning internal representations by error propagation. Technical report, California Univ San Diego La Jolla Inst for Cognitive Science, 1985.

[31] B. Schölkopf and A. J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, 2001.

[32] B. Schölkopf, A. Smola, and K.-R. Müller. Kernel principal component analysis. In *Advances in Kernel Methods - Support Vector Learning*, pages 327–352, 1999.

[33] H. Shao, A. Kumar, and P. T. Fletcher. The riemannian geometry of deep generative models. *arXiv preprint arXiv:1711.08014*, 2017.

[34] J. Y. Stein. *Digital signal processing: A computer science perspective*. John Wiley & Sons, Inc., 2000.

[35] J. B. Tenenbaum, V. De Silva, and J. C. Langford. A global geometric framework for nonlinear dimensionality reduction. *science*, 290(5500):2319–2323, 2000.

[36] R. Tibshirani and T. Hastie. Local likelihood estimation. *Journal of the American Statistical Association*, 82(398):559–567, 1987.

[37] A. Tosi, S. Hauberg, A. Vellido, and N. D. Lawrence. Metrics for Probabilistic Geometries. In *The Conference on Uncertainty in Artificial Intelligence (UAI)*, July 2014.

[38] A. G. Wilson and Z. Ghahramani. Generalised wishart processes. In *Uncertainty in Artificial Intelligence (UAI)*, 2011.

A Experimental details

Data creation: We sample $N = 200$ numbers $\{t_n\}_{n=1}^N$ uniformly over the interval $[0, 2\pi]$, and embed these first in $\mathbb{R}^3$ as

$$\mathbf{v}_n = \begin{bmatrix} \cos(t_n) \\ \sin(t_n) \\ \cos(t_n)\sin(t_n) \end{bmatrix} \tag{A.1}$$

The data is then generated as

$$\mathbf{x}_n = \begin{bmatrix} \mathbf{v}_n \\ \mathbf{0} \end{bmatrix} + \sigma\boldsymbol{\epsilon}, \tag{A.2}$$

where $\mathbf{0}$ is a $D-3$ dimensional vector of zeros, $\boldsymbol{\epsilon}$ is a D dimensional vector drawn from a unit Gaussian, and $\sigma = 0.1$ captures "off manifold"-noise. Here we embed into a $D = 1000$ dimensional vector space.

Since the data fundamentally live on a unit circle (which we then nonlinearly embed in $\mathbb{R}^D$), we fix the latent variables $\mathbf{z}_n$ to be the first two dimensions of $\mathbf{x}_n$; this correspond to points on the unit circle with added Gaussian noise. In order to compare different models, we fix the latent variables to have the same values in all models.

Kernel methods: We first fit a model using Gaussian process (GP) regression from $\mathbf{z}_{1:N}$ to $\mathbf{x}_{1:N}$ with hyperparameters estimated using maximum likelihood. We do not update the latent variables $\mathbf{z}_{1:N}$ as is commonly done for the GP-LVM as this would complicate a direct comparison between different models. The methods based on kernel ridge regression are simple taken as the GP mean function. This ensure that the exact same hyperparameters are used in the GP and kernel ridge regression experiments. Again, this choice was made to simplify the comparison of different methods.

Neural network methods: We first train a multilayer perceptron from $\mathbf{z}_{1:N}$ to $\mathbf{x}_{1:N}$ according to the usual autoencoding criterion. We use this to form our autoencoder models. We keep this mapping as the mean function μ of the variational autoencoders (VAE), and fit the uncertainty σ according to the usual VAE criterion. As before, we take this restricted approach to ensure that models are as comparable as possible.

B Correlation plots

In the main text we quantitatively compare models in two ways. The key idea is to take advantage of the fact that we know that ground truth geodesics should be circular arcs. Let $\mathbf{c}_{\text{gt}}$ denote a ground truth geodesic, and let $\mathbf{c}_{\text{est}}$ denote an estimated geodesic. As a first measure of quality, we compute the length of each curve under the model-specific metric

$$\mathcal{L}_{\text{gt}} = \mathcal{L}(\mathbf{c}_{\text{gt}}) = \int_a^b \|\partial_t \mathbf{c}_{\text{gt}}\|_{\mathbf{M}} \mathrm{d}t \tag{B.1}$$

$$\mathcal{L}_{\text{est}} = \mathcal{L}(\mathbf{c}_{\text{est}}) = \int_a^b \|\partial_t \mathbf{c}_{\text{est}}\|_{\mathbf{M}} \mathrm{d}t. \tag{B.2}$$

Both integrals are evaluated using standard quadrature. In the main text, we report the correlation between $\mathcal{L}_{\text{gt}}$ and $\mathcal{L}_{\text{est}}$ for randomly sampled points. More insight into the empirical behavior of the different models can be found be directly plotting $\mathcal{L}_{\text{gt}}$ and $\mathcal{L}_{\text{est}}$ against each other, which we do in Fig. 7. Here we see a clear trend that methods with no or meaningless uncertainty quantification all exhibit a trend that short curves correlate well with the ground truth, but long curves do not. As the Riemannian model is that we work with locally Euclidean models, this behavior is not too surprising. We see that with methods with reasonably sensible uncertainty quantification the two lengths have a more well-behaved correlation. It is worth pointing out that that VAE with RBF precision is not as well behaved as the GP-LVM, which indicate that the variance estimation leaves something to be desired.

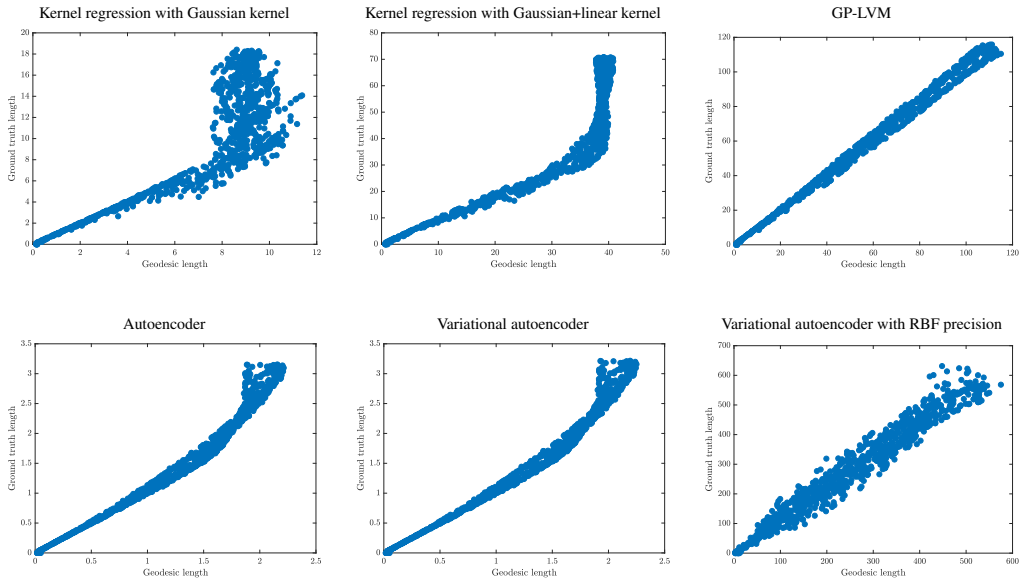

Figure 7: Correlation between length of ground truth geodesics and estimated geodesics for different models.

As a second measure of quality we consider the Haussdorf distance between $\mathbf{c}_{\text{gt}}$ and $\mathbf{c}_{\text{est}}$. This is defined as

$$\text{dist}_H(\mathbf{c}_{\text{gt}}, \mathbf{c}_{\text{est}}) = \max\{\sup_{x \in \mathbf{c}_{\text{gt}}} \inf_{y \in \mathbf{c}_{\text{est}}} \|x - y\|, \sup_{y \in \mathbf{c}_{\text{est}}} \inf_{x \in \mathbf{c}_{\text{gt}}} \|x - y\|\}. \tag{B.3}$$

We refer to the main text for results and discussion.