Staying on the Manifold: Geometry-Aware Noise Injection

Anonymous Full Paper
Submission 41

Abstract

It has been shown that perturbing the input during training implicitly regularises the gradient of the learnt function, leading to smoother models and enhancing generalisation. However, previous research mostly considered the addition of ambient noise in the input space, without considering the underlying structure of the data. In this work, we propose several methods of adding geometry-aware input noise that accounts for the lower dimensional manifold the input space inhabits. We start by projecting ambient Gaussian noise onto the tangent space of the manifold. In a second step, the noise sample is mapped on the manifold via the associated geodesic curve. We also consider Brownian motion noise, which moves in random steps along the manifold. We show that geometry-aware noise leads to improved generalization and robustness to hyperparameter selection on highly curved manifolds, while performing at least as well as training without noise on simpler manifolds. Our proposed framework extends to learned data manifolds.

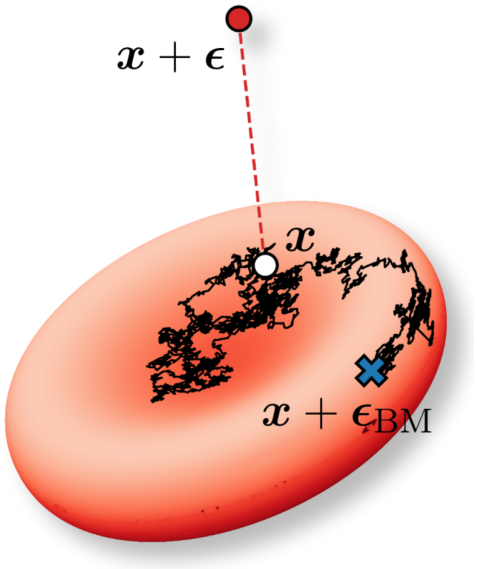

Figure 1. Noise injection is a data augmentation technique that can improve generalisation. For a data point (○) lying on a lower-dimensional manifold, sampling noise in the ambient space (●) almost surely deviates from the input manifold whereas a sample from a geometry-aware noise process (✖) *stays on the manifold* and respects the data geometry. Illustrated of the biconcave disc that resembles a red blood cell.

1 Introduction

One of the most intuitive and practical methods to improve the generalisation properties of a learnable model is to consider data augmentation techniques [1]. During training, new data samples are created from given ones, sharing the same features and labels. This approach has been extensively used with image data, for example through adjusting the illumination, changing the orientation or cropping.

Classic machine learning research has already established the influence of input noise on generalisation performance [2, 3]. One widely studied technique is adding Gaussian noise to the inputs, which leads to a smoothness penalty on the learnt function [4, 5], however, these works do not take into account the structure of the input data. A fundamental observation in modern machine learning is the manifold hypothesis: it states that high-dimensional data tends to concentrate around a lower-dimensional manifold in the ambient space [6, 7]. In the context of noise-based learning, this has the implication that, with high probability, Gaussian noise will be almost perpendicular to the manifold [8]. Hence, adding Gaussian noise to the data leads to unlikely or non-informative augmented data samples.

Additionally, many real-world problems require learning functions on a known manifold rather than the unconstrained Euclidean space. Weather and climate observations naturally live on the surface of the sphere, which approximates the shape of the Earth. In cell biology we might consider red blood cells, which can be approximated by a biconcave disc [9]. Or in brain imaging, quantities like cortical thickness and grey matter intensity are measured on the cortical surface [10]: although the cortex can be mapped onto the sphere, it is actually highly wrinkly. In such settings, applying perturbations or learning representations that ignore the intrinsic manifold structure can lead to deceptive results as Euclidean distances in the embedding space fail to capture the true distances between points: two points which might be close with respect to the Euclidean metric can be far apart when travelling along the manifold surface. This highlight the necessity of geometry-aware methods that respect the manifold structure when perturbing data as an augmentation technique.

In this paper, we propose geometry-aware noise injection strategies as a data augmentation technique and show their benefits compared to ambient space noise injection. We consider three such strategies – tangent space noise, geodesic noise and Brownian motion noise – and demonstrate their effect on manifolds embedded in $\mathbb{R}^3$, namely the Swiss roll and families of spheroids and tori.

Our contributions include:

- definitions, derivations, and implementations of geometry-aware input noise for various parametrised manifolds and their deformations,
- establishing the implicit regulariser of adding manifold-restricted input noise,
- empirical demonstration that geometry-aware noise improves generalisation and robustness over manifold-agnostic noise.

2 Preliminaries

We consider a dataset of N points $\{\boldsymbol{x}_n, y_n\}_{n=1}^N$, where the inputs $\boldsymbol{x}_n \in \mathcal{X} = \mathbb{R}^D$ are assumed to lie on an embedded d-dimensional manifold $\mathcal{M}$ with $d < D$, and the outputs $y_n \in \mathcal{Y}$ may be either continuous or discrete. Our goal is to learn a function $f_{\boldsymbol{\theta}} : \mathcal{X} \to \mathcal{Y}$, typically parametrised by a deep neural network with parameters $\boldsymbol{\theta} \in \mathbb{R}^K$. The model is trained by minimizing the empirical loss

$$\mathcal{L}(\boldsymbol{x}, \boldsymbol{\theta}) = \sum_{n=1}^{N} \ell(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n), y_n),$$

where $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}^+ \cup \{0\}$ is a loss function, often chosen as the mean squared error (MSE) in regression settings. For simplicity of notation, we write $\boldsymbol{x} = \{\boldsymbol{x}_n\}_{n=1}^N$ and $\mathcal{L}(\boldsymbol{x}) = \mathcal{L}(\boldsymbol{x}, \boldsymbol{\theta})$.

2.1 Gaussian Input Noise

As mentioned, several previous works consider Gaussian input noise [2, 4, 11, 12]. In this section, we summarise the previous analysis and show that adding Gaussian noise to the input during training is equivalent in expectation to Tikhonov regularisation [13].

Consider an input data point $\boldsymbol{x}_n \in \mathcal{X}$, which we perturb with noise following a normal distribution $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \sigma^2 \mathbb{I}_D)$ for $\sigma > 0$. Then the second-order Taylor expansion of the loss function $\mathcal{L}(\boldsymbol{x})$ is given by:

$$\mathcal{L}(\boldsymbol{x} + \boldsymbol{\epsilon}) \approx \mathcal{L}(\boldsymbol{x}) + \boldsymbol{\epsilon}^\top \nabla_x \mathcal{L}(\boldsymbol{x}) + \frac{1}{2} \boldsymbol{\epsilon}^\top \mathbf{H}_{\mathcal{L}} \boldsymbol{\epsilon}. \quad (1)$$

We take the expectation of the Gaussian noise distribution and get

$$\mathbb{E}_{\boldsymbol{\epsilon}}[\mathcal{L}(\boldsymbol{x} + \boldsymbol{\epsilon})] = \mathcal{L}(\boldsymbol{x}) + \frac{\sigma^2}{2} \Delta_x \mathcal{L}(\boldsymbol{x}), \quad (2)$$

where Δ is the Laplace operator (trace of the Hessian). Using the chain rule, this term expands to:

$$\Delta_x \mathcal{L}(\boldsymbol{x}) = \sum_{n=1}^{N} \|\nabla_x f_{\boldsymbol{\theta}}(\boldsymbol{x}_n)\|^2 \quad (3)$$

$$+ \frac{1}{2} \sum_{n=1}^{N} (f_{\boldsymbol{\theta}}(\boldsymbol{x}_n) - y_n) \Delta_x f_{\boldsymbol{\theta}}(\boldsymbol{x}_n).$$

when choosing ℓ to be the MSE. When the function interpolates the training data points, that is, $f_{\boldsymbol{\theta}}(\boldsymbol{x}_n) \approx y_n$, the second summand in Equation 3 vanishes. Thus, after plugging this back into Equation 2, we see that adding input noise is equivalent (in expectation) to optimising a regularised loss on the form $\mathcal{L}(\boldsymbol{x}) + R(\boldsymbol{x}, \boldsymbol{\theta})$, with R being the Tikohonov regulariser

$$R(\boldsymbol{x}, \boldsymbol{\theta}) = \frac{\sigma^2}{2} \sum_{n=1}^{N} \|\nabla_x f_{\boldsymbol{\theta}}(\boldsymbol{x}_n)\|^2. \quad (4)$$

Thus, a small gradient is incentivised at each training point, which implies that the optimisation process will converge to parameters $\boldsymbol{\theta}^*$ for which the function $f_{\boldsymbol{\theta}^*}$ is *flat* in the neighbourhood of the given data.

2.2 Riemannian Geometry

In this section, we give a brief introduction to the tools we use from Riemannian geometry [14]. The reader who is already familiar with Riemannian geometry may skip this section.

Local charts. Plainly speaking, a manifold can be seen as a d-dimensional generalisation of a surface. It locally resembles the Euclidean space $\mathbb{R}^d$, meaning that for every point $\boldsymbol{x} \in \mathcal{M}$, we can find an open neighbourhood around $\boldsymbol{x}$ which can be smoothly mapped to an open set of $\mathbb{R}^n$. For completeness, we include a more rigorous mathematical definition.

Definition 2.1 *A manifold $\mathcal{M}$ is a Hausdorff space such that for every $\boldsymbol{x} \in \mathcal{M}$ there exists a homeomorphism $X : U \to V$ from a neighbourhood $U \ni \boldsymbol{x}$ to an open set $V \subseteq \mathbb{R}^d$. Further, we require these charts to be compatible on the intersection of their domains, i.e.*

$$X_1 \circ X_2^{-1}|_{X_2(U_1 \bigcap U_2)} : X_2(U_1 \cap U_2) \subseteq \mathbb{R}^d \to \mathbb{R}^d$$

is a smooth map.

The tangent space. In $\mathbb{R}^3$, the tangent plane of a manifold is easy to picture: each point of the surface is approximated with a plane in which the tangent vectors live. In higher dimensions, we say that the tangent space $T_{\boldsymbol{x}}\mathcal{M}$ of $\mathcal{M}$ at a point $\boldsymbol{x}$ consists of the velocities of all curves on $\mathcal{M}$ passing through $\boldsymbol{x}$, that is, if γ is a smooth curve on $\mathcal{M}$ parametrised by time t with

$$\gamma(0) = \boldsymbol{x},$$

then

$$\boldsymbol{v} = \dot{\gamma}(0) \in T_{\boldsymbol{x}}\mathcal{M}.$$

Assume we have a smooth parametrisation $X : \mathbb{R}^d \to \mathbb{R}^D$. Then the Jacobian

$$\mathbf{J}_X = \left[\frac{\partial X}{\partial u_1}, \ldots, \frac{\partial X}{\partial u_d}\right]$$

is a function from $\mathbb{R}^d$ to $\mathbb{R}^{D\times d}$ and the tangent space at each point is spanned by the columns of $\mathbf{J}_X$. At every point $\boldsymbol{x} \in \mathcal{M}$, any vector $\boldsymbol{v} \in \mathbb{R}^D$ can be orthogonally decomposed into a tangential and a normal component as

$$\boldsymbol{v} = \boldsymbol{v}_\top + \boldsymbol{v}_\perp.$$

In Figure 2, we can see a manifold (the sphere) embedded in $\mathbb{R}^3$, and the tangent space at a point.

Riemannian metrics. A Riemannian manifold $(\mathcal{M}, g)$ is a smooth manifold equipped with a Riemannian metric. A metric g of $\mathcal{M}$ equips each point $\boldsymbol{x} \in \mathcal{M}$ with an inner product $g_{\boldsymbol{x}}$ on $T_{\boldsymbol{x}}\mathcal{M}$. This continuous tensor field allows us to measure distances and angles, and define geodesics along the manifold.

Assume we have a smooth parametrisation $X : \mathbb{R}^2 \to \mathbb{R}^3$. Then the matrix valued function

$$\mathbf{J}_X^\top \cdot \mathbf{J}_X : \mathbb{R}^2 \to \mathbb{R}^{2\times 2}$$

induces a metric. For $X(\boldsymbol{u}) = \boldsymbol{x} \in \mathcal{M}$ and $\boldsymbol{v}, \boldsymbol{w} \in T_{\boldsymbol{x}}\mathcal{M}$, let $\tilde{\boldsymbol{v}}, \tilde{\boldsymbol{w}} \in T_{\boldsymbol{u}}\mathbb{R}^d$ be such that $\mathbf{J}_X\tilde{\boldsymbol{v}} = \boldsymbol{v}$ and $\mathbf{J}_X\tilde{\boldsymbol{w}} = \boldsymbol{w}$. Then

$$g_{\boldsymbol{x}}(\boldsymbol{v}, \boldsymbol{w}) = \boldsymbol{v}^\top \mathbf{J}_X^\top \mathbf{J}_X \boldsymbol{w}$$

is the induced metric. By abuse of notation, we will often write g to denote the matrix $\mathbf{J}_X^\top \mathbf{J}_X$.

Geodesics. A geodesic is locally the shortest path on a manifold. We can rewrite a curve

$$\gamma : I \subseteq \mathbb{R} \to \mathcal{M}$$

on $\mathcal{M}$ as $\gamma(t) = X \circ \alpha(t)$, where $\alpha : I \to \mathbb{R}^d$ is a curve in the parameter space. Then γ is a geodesic if and only if α satisfies the following ordinary differential equation (ODE) for all $k = 1, \ldots, d$:

$$\ddot{\alpha}_k(t) = -\sum_{i,j=1}^{n} \dot{\alpha}_i(t)\dot{\alpha}_j(t) \cdot \Gamma_{ij}^k(\alpha(t)), \qquad (5)$$

where Γ_{ij}^k denote the so-called Christoffel symbols. It can be shown that if $\mathcal{M}$ is a Riemannian manifold, then for every $\boldsymbol{x} \in \mathcal{M}$ and every unit vector $\boldsymbol{e} \in T_{\boldsymbol{x}}\mathcal{M}$ there exists a unique geodesic $\gamma_{\boldsymbol{e}}$ such that

$$\gamma_{\boldsymbol{e}}(0) = \boldsymbol{x}, \quad \dot{\gamma}_{\boldsymbol{e}}(0) = \boldsymbol{e}.$$

The exponential map. Intuitively, one can imagine the exponential map like a function which wraps aluminium foil (the tangent plane) around, say, a bagel (the manifold). Despite the manifold being curved and the tangent space being flat, at any chosen point we can wrap a small part of the tangent plane around a neighbourhood of this point without folding it.

Using geodesics, for each $\boldsymbol{x} \in \mathcal{M}$ we can define a map from an open ball $B_\delta(0) \subseteq T_{\boldsymbol{x}}\mathcal{M}$ to a neighbourhood $\boldsymbol{x} \in U \subseteq \mathcal{M}$ on the manifold. We will call this map the exponential map:[1]

$$\mathrm{Exp}_{\boldsymbol{x}} : B_\delta(0) \subseteq T_{\boldsymbol{x}}\mathcal{M} \to U \subseteq \mathcal{M},$$

$$\mathrm{Exp}_{\boldsymbol{x}}(\boldsymbol{v}) = \begin{cases} \gamma_{\frac{\boldsymbol{v}}{\|\boldsymbol{v}\|}}(\|\boldsymbol{v}\|) & \text{if } \boldsymbol{v} \in B_\delta(0)\backslash\{\mathbf{0}\}, \\ \boldsymbol{x} & \text{if } \boldsymbol{v} = \mathbf{0}. \end{cases}$$

In other words, the exponential map maps a vector $\boldsymbol{v} \in T_{\boldsymbol{x}}\mathcal{M}$ from the tangent space to the endpoint of a curve on the manifold, $\gamma_{\frac{\boldsymbol{v}}{\|\boldsymbol{v}\|}}(\|\boldsymbol{v}\|)$, and the zero vector to $\boldsymbol{x}$.

3 Noise Injection Strategies

We consider three strategies of increasing complexity for geometry-aware input noise: tangential noise, geodesic noise and Brownian motion noise. The goal was to create noise injection techniques which either stay close to the manifold or, better, stay on the manifold.

3.1 Projected Tangent Space Noise

The simplest method we will try is Gaussian noise projected to the tangent space. Intuitively, this takes a noise sample in the ambient space, $\boldsymbol{\epsilon} \sim \mathcal{N}\left(\mathbf{0}, \sigma^2\mathbb{I}_D\right)$, and pulls it closer to the manifold.

To isolate the tangential component $\boldsymbol{\epsilon}_\top$, we subtracting the component $\boldsymbol{\epsilon}_\perp$:

$$\boldsymbol{\epsilon}_\top = \boldsymbol{\epsilon} - \sum_i \langle \boldsymbol{\epsilon}, \boldsymbol{n}_i \rangle \cdot \boldsymbol{n}_i,$$

where $\{\boldsymbol{n}_i\}$ is a set of unit vectors spanning the normal space of $\mathcal{M}$. For more details we recommend

[1] Here, $\delta \in \mathbb{R}^+$ ensures that the exponential map is a well defined diffeomorphism. Loosely speaking, it is the largest radius we can choose while guaranteeing that the geodesics are well defined and do not overlap.

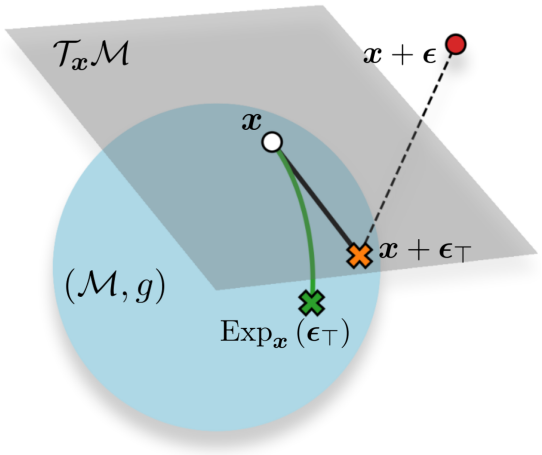

Figure 2. Noise injection strategies with increasing level of conceptual complexity, i.e. ambient space noise (●), tangent space noise (✖) and geodesic noise (✖). The Brownian motion strategy is visualised in Figure 3.

the classic textbook [15]. Equivalently, the tangential noise can be defined with projection matrix, $\mathbf{P} = \mathbb{I}_D - \sum_i \boldsymbol{n}_i \boldsymbol{n}_i^\top$, as

$$\boldsymbol{\epsilon}_\top = \mathbf{P}\boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}\left(\mathbf{0}, \sigma^2 \mathbb{I}_D\right).$$

This allows for directly sampling tangent noise as

$$\boldsymbol{\epsilon}_\top \sim \mathcal{N}\left(\mathbf{0}, \sigma^2 \mathbf{P}\right).$$

Regularisation perspective: We now analyse how adding tangential noise $\boldsymbol{\epsilon}_\top$ affects the model $f_{\boldsymbol{\theta}}$. We proceed as in Subsection 2.1.

$$\begin{aligned}\mathbb{E}[\boldsymbol{\epsilon}_\top^\top \mathbf{H}_{\mathcal{L}} \boldsymbol{\epsilon}_\top] &= \sum_{n=1}^{N} \mathbb{E}\left[\boldsymbol{\epsilon}_\top^\top \nabla_x f_{\boldsymbol{\theta}}(\boldsymbol{x}_n) \nabla_x f_{\boldsymbol{\theta}}(\boldsymbol{x}_n)^\top \boldsymbol{\epsilon}_\top\right] \\ &\quad + \frac{1}{2}\sum_{n=1}^{N} \mathbb{E}\left[\boldsymbol{\epsilon}_\top^\top \left(f_{\boldsymbol{\theta}}(\boldsymbol{x}_n) - y_n\right) \Delta_x f_{\boldsymbol{\theta}}(\boldsymbol{x}_n) \boldsymbol{\epsilon}_\top\right] \\ &= \sum_{n=1}^{N} \|\nabla_x f_{\boldsymbol{\theta}}(\boldsymbol{x})_\top\|^2.\end{aligned}$$

The second summand again vanishes if we assume that the model $f_{\boldsymbol{\theta}}$ interpolates the target values perfectly, that is, $f_{\boldsymbol{\theta}}(\boldsymbol{x}_n) = y_n$ for all $n = 1, \ldots, N$. When evaluating the first summand, we see that

$$\boldsymbol{\epsilon}_\top^\top \nabla_x f_{\boldsymbol{\theta}}(\boldsymbol{x}_n) = \boldsymbol{\epsilon}_\top^\top \nabla_x f_{\boldsymbol{\theta}}(\boldsymbol{x}_n)_\top + \underbrace{\boldsymbol{\epsilon}_\top^\top \nabla_x f_{\boldsymbol{\theta}}(\boldsymbol{x}_n)_\perp}_{=0}. \quad (6)$$

Combining our results, we obtain the regulariser

$$R(\boldsymbol{x}, \boldsymbol{\theta}) = \frac{\sigma^2}{2} \sum_{n=1}^{N} ||\nabla_x f_{\boldsymbol{\theta}}(\boldsymbol{x}_n)_\top||^2.$$

This shows that the addition of tangential noise only regularises the tangential component of $f_{\boldsymbol{\theta}}$.

3.2 Geodesic Noise

As explained in Subsection 2.2, at every $\boldsymbol{x} \in \mathcal{M}$, and for every $\boldsymbol{v} \in T_{\boldsymbol{x}}\mathcal{M}$ there exists a geodesic $\gamma : I \to \mathcal{M}$ such that $\gamma(0) = \boldsymbol{x}$, and $\dot{\gamma}(0) = \boldsymbol{v}$. All manifolds in our paper are complete, and hence $I = \mathbb{R}$, and γ can be extended to the whole of $\mathbb{R}$. This allows us to generate points $\tilde{\boldsymbol{x}}$ near $\boldsymbol{x}$ by sampling initial velocities and mapping them to the manifold via the exponential map. We proceed as follows:

1. sample $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \sigma^2 \mathbb{I}_D)$,
2. project $\boldsymbol{\epsilon}$ into the tangent space $T_{\boldsymbol{x}}\mathcal{M}$ as explained in Subsection 3.1, let
$$\boldsymbol{\epsilon}_\top = \mathbf{P}\boldsymbol{\epsilon},$$
3. evaluate γ at $||\boldsymbol{\epsilon}_\top||$, the new point is now
$$\tilde{\boldsymbol{x}} = \mathrm{Exp}_{\boldsymbol{x}}(\boldsymbol{\epsilon}_\top) = \gamma(||\boldsymbol{\epsilon}_\top||).$$

For a small step size σ, we expect this to have a similar effect as the tangential noise but may improve robustness for increased step sizes. Details about the implementation of this method can be found in Appendix B.

3.3 Intrinsic Brownian Motion

Brownian motion (BM) is a stochastic process, which has first been used to describe the random movement of particles suspended in a fluid. Due to its occurrence in nature, this provides a realistic way of modelling how data points might move along a manifold. In the parameter space of a Riemannian manifold, the Brownian motion is defined by the following stochastic process [16]:

$$\begin{aligned}du_k(t) = &\frac{1}{2}\frac{1}{\sqrt{\det g}}\sum_{l=1}^{d}\frac{\partial}{\partial u_l}\left(\sqrt{\det g}\cdot g^{kl}\right)dt \quad (7)\\ &+ \left(\sqrt{g^{-1}}dB(t)\right)_k.\end{aligned}$$

where $dB(t)$ is the standard Euclidean BM and t is the time. The summands are referred to as the *drift term* and *noise term*, respectively. Since the Brownian motion on a manifold is generated by the Laplace-Beltrami operator [17], which is intrinsic, it is independent of the local charts [18].

3.4 Example: the Swiss roll

In this subsection, we will go through the computations for one example manifold, namely the Swiss roll. The derivations for the other manifolds follow a similarly approach. The Swiss roll is parametrised as follows:

$$X : \mathbb{R}^2 \to \mathbb{R}^3, \;\; X(u_1, u_2) = (au_1 \sin u_1, au_1 \cos u_1, u_2).$$

Here, $a \in \mathbb{R}^+$ is a positive coefficient which determines how tightly the manifold is rolled. The metric is then given by

$$\begin{pmatrix} a^2(1+u_1^2) & 0 \\ 0 & 1 \end{pmatrix}.$$

Tangent space noise. The unit normal vector at each point (u, v) is given by

$$\boldsymbol{n} = \frac{1}{\sqrt{1+u_1^2}} \cdot \begin{bmatrix} \cos u_1 - u_1 \sin u_1 \\ -\sin u_1 - u_1 \cos u_1 \\ 0 \end{bmatrix}.$$

To get tangential noise, we subtract the normal component from the Gaussian ambient noise, i.e.

$$\boldsymbol{\epsilon}_\top = \boldsymbol{\epsilon} - \langle \boldsymbol{\epsilon}, \boldsymbol{n} \rangle \cdot \boldsymbol{n}.$$

Geodesic noise. A curve on the manifold can be obtained by taking a curve $\alpha : I \to \mathbb{R}^2$ in $\mathbb{R}^2$ and mapping it on the manifold through X. Let $\gamma : I \to \mathcal{M}$ be given by $\gamma(t) = X \circ \alpha(t)$.

$$\ddot{\alpha}_1(t) = -\frac{\alpha_1(t)}{1+\alpha_1(t)^2}\dot{\alpha}_1(t)^2,$$
$$\alpha_2(t) = \alpha_2(0) + t\dot{\alpha}_2(0).$$

Brownian motion. We start with a few precomputations. It is clear that

$$\det(g) = a^2(1+u^2),$$
$$g^{-1} = \begin{pmatrix} \frac{1}{a^2(1+u_1^2)} & 0 \\ 0 & 1 \end{pmatrix}.$$

Plugging it into Equation 7, we get:

$$\begin{bmatrix} du_1 \\ du_2 \end{bmatrix} = -\frac{dt}{2}\begin{pmatrix} \frac{u}{(1+u^2)^2} \\ 0 \end{pmatrix} + \sqrt{dt}\begin{pmatrix} \frac{1}{\sqrt{a^2(1+u^2)}} & 0 \\ 0 & 1 \end{pmatrix}\tilde{\boldsymbol{\epsilon}}.$$

We remark that $dB(t) = \sqrt{dt} \cdot \tilde{\boldsymbol{\epsilon}}$ where $\tilde{\boldsymbol{\epsilon}} \sim \mathcal{N}(\mathbf{0}, \mathbb{I}_d)$ is a noise sample in the parameter space.

4 Deformation of a Manifold

In the following, we briefly elaborate on the underlying approach to constructing deformations of manifolds which we will use in Section 5.

We consider a vector field v for defining a time-dependent diffeomorphism, $\phi : \mathcal{M} \times [0, T] \to \mathbb{R}^D$ that maps points from a parametrised manifold to a deformed version of the manifold, $\widetilde{\mathcal{M}}$. This is also known as a *flow*. The vector field induces the flow through an ordinary differential equation:

$$\frac{d}{dt}\phi_t(\boldsymbol{x}) = v_t(\phi_t(\boldsymbol{x})), \quad (8)$$
$$\phi_0(\mathbf{0}) = \boldsymbol{x},$$

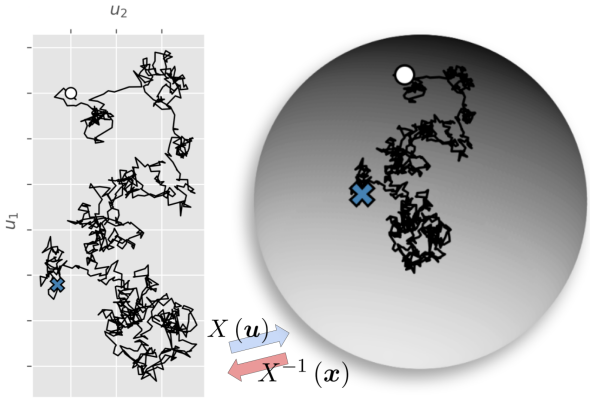

Figure 3. Brownian motion is computed from an initial point (○) in the parameter space (*left*), then mapped to the manifold (*right*) via the chart X. The endpoint of the Brownian motion on the manifold (✖) acts as the noisy observation.

where $\boldsymbol{x} \in \mathcal{M}$ is a point on the parametrised manifold. As our manifold is defined through local coordinates, we can express points on the deformed manifold as

$$\tilde{\boldsymbol{x}} = \phi_T(X(\boldsymbol{u})) \in \widetilde{\mathcal{M}},$$

which is obtained by integrating the ODE (8) up to time T. The Jacobian of ϕ_t with respect to u at $\boldsymbol{u} = X^{-1}(\boldsymbol{x})$ is given by

$$\mathbf{J}_{\boldsymbol{u}}(t) := \frac{\partial \phi_t(X(\boldsymbol{u}))}{\partial u} = \frac{\partial \phi_t(\boldsymbol{x})}{\partial x}\frac{\partial X(\boldsymbol{u})}{\partial u}.$$

It can be computed by solving another ODE:

$$\frac{d}{dt}\mathbf{J}_{\boldsymbol{u}}(t) = \mathbf{J}_v(t)\,\mathbf{J}_{\boldsymbol{u}}(t),$$
$$\mathbf{J}_{\boldsymbol{u}}(0) = \frac{\partial X(\boldsymbol{u})}{\partial u},$$

where $\mathbf{J}_v(t) = \frac{\partial v_t(\phi_t(\boldsymbol{x}))}{\partial \phi_t}$ is the Jacobian of the velocity field function. Thus, the metric $\tilde{g}$ of $\widetilde{\mathcal{M}}$ is

$$\tilde{g} = \mathbf{J}_{\boldsymbol{u}}(T)^\top \mathbf{J}_{\boldsymbol{u}}(T).$$

We can now sample initial velocities on the tangent space $T_{\tilde{\boldsymbol{x}}}\widetilde{M}$ at $\tilde{\boldsymbol{x}}$, using (9) and generate geodesics on the deformed manifold $\widetilde{M}$.

This framework allows for highly expressive and flexible deformations of any parametrised manifold while ensuring invertibility. Previous research [19, 20] parametrise $v_{t,\boldsymbol{\theta}}$ with a neural network. Though we in practice only consider a fixed parametrisation of such a network to generate diffeomorphic deformations, our framework works for any map v_t. This opens new pathways to neural network settings where a learned flow approximates the underlying data manifold from which we can compute intrinsic geometric quantities, which we leave for future work.

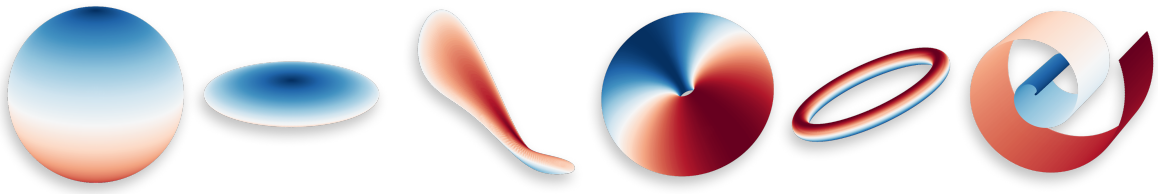

Figure 4. Illustrations of the manifolds and functions on manifolds that we consider. From left: `Sphere`, `SqueezedSphere`, `DeformedSphere`, `Bead`, `OnionRing` and `SwissRoll`. Specifically, we use the deformation approach described in Section 4 to construct the `DeformedSphere` from a parametrised unit sphere in $\mathbb{R}^3$.

Implementation details. The Jacobian of the vector field, v_t, rarely has a closed form, however we can compute it efficiently using *automatic differentiation*. In practice, this allows us to evaluate derivatives of deformed manifolds with respect to the local coordinates of points on the manifold, without manually deriving the expressions. This algorithmic framework allows us to apply the technique to any manifold as long as some parametrisation is available and we have a differentiable ODE solver. In practice, we solve the flow equation numerically using an explicit Euler scheme and compute Jacobians and induced metrics with automatic differentiation. We remark that higher-order ODE solvers could be used for improved accuracy, yet the choice of the Euler scheme is based on practical challenges with existing toolboxes, for instance current incompatibility issues between existing libraries.

5 Experimental validation

We test our hypothesis on a range of manifolds visualised in Figure 4. We generate $N = 200$ training points on each manifold and train an overparametrised 3-layer neural network with 64 nodes per layer to learn a specific function for each manifold. We train for 500 epochs using a learning rate of 10^{-3} with a MSE objective. For each training step, we add either ambient space noise, tangential noise, geodesic noise or Brownian motion noise and compare to a baseline network trained without adding input noise. We treat the noise covariance σ^2 as a hyperparameter, and, in the Brownian motion setting, interpret it as the total time of the process, i.e. $T = \sigma^2$. We provide the average error per method relative to the baseline's MSE in Table 1 with uncertainties given by the standard error of the mean computed from 5 independent runs. We provide computations for the geodesic equations and Brownian motion along with the target functions for the parametrised manifolds in Appendix A.

Our results show that geometry-aware noise injection provides advantages to ambient space noise on complex manifolds. In particular, geodesic and Brownian motion noise not only yield lower errors on "wigglier" geometries - such as the `SwissRoll` and `DeformedSphere` - but they also exhibit greater robustness to the noise intensity hyperparameter (Figure 5). This indicates that geometric approaches can both improve generalisation and reduce sensitivity to hyperparameter choices.

At the same time, performance never significantly deteriorates when using any noise strategy, compared to the baseline trained without noise (Table 1). For some manifolds simple ambient Gaussian noise can suffice: we observe this particularly for manifolds of which only a small part is problematic, such as the `Bead` (the fat torus). Here, Gaussian noise only leads to misleading samples near the genus. Since the surface area of the genus is proportionally small, the overall error remains low. The `SwissRoll`, on the other hand, is sensitive to Gaussian noise everywhere, and hence our methods work better. For completeness, we report results across all tested manifolds, even when geometry-aware strategies do not provide measurable gains.

6 Related Work

A recent work [3] surveys classical perspectives and modern advances for how noise injection influences learning . Bishop's analysis [4] of Gaussian input noise is restricted to considering the Gauss-Newton part of the Hessian with the argument that the non-linear modelling error (NME) vanishes at an optimum [4]. The implications of this were recently discussed in the paper [21], with the conclusion that the neglected Hessian component can play a crucial role in shaping the geometry of the loss surface.

Instead of assuming that the input points live on a manifold, we can also enforce that the parameters of the model belong to a manifold. A previous work [5] analyses the impact of adding Gaussian noise to weights of a parametric model . Other works [22, 23] study orthogonal regularisers on the weight matrices, promoting the columns to be orthonormal.

Table 1. Relative mean squared error to the baseline (B) model trained without adding noise. We report results for the optimal hyperparameter σ^2 for each strategy and manifold. We compare with ambient noise (A), tangent noise (T), geodesic noise (G) and Brownian motion noise (BM). We highlight the best strategy per manifold in **bold**. Adding noise does not improve performance for some manifolds, but results are included for completeness.

	Sphere	SqueezedSphere	DeformedSphere	Bead	OnionRing	SwissRoll
B	1.00 ± 0.20	1.00 ± 0.14	1.00 ± 0.16	1.00 ± 0.12	1.00 ± 0.16	1.00 ± 0.05
A	1.00 ± 0.19	0.99 ± 0.13	0.33 ± 0.07	**0.70 ± 0.13**	1.00 ± 0.17	0.93 ± 0.09
T	0.98 ± 0.19	1.00 ± 0.14	0.29 ± 0.06	0.77 ± 0.15	1.00 ± 0.16	0.93 ± 0.09
G	**0.93 ± 0.16**	0.99 ± 0.14	0.27 ± 0.06	0.81 ± 0.13	1.01 ± 0.17	**0.43 ± 0.01**
BM	**0.93 ± 0.16**	**0.98 ± 0.13**	**0.26 ± 0.05**	0.81 ± 0.13	**0.98 ± 0.16**	0.44 ± 0.02

These constraints restrict the parameter space to the Stiefel or Grassmann manifolds, which improves numerical stability. This line of work highlights that geometry can be injected not only through noise in the input space but also by shaping the structure of the model's parameters. Another approach to noise injection considers adding structured noise to the gradient during training with gradient-based optimisers for improved generalisation [24, 25].

In the context of Riemannian representation learning, adding noise according to the structure of the manifold stabilises results in the recent paper [26]. This approach replaces the traditional encoder–decoder setup with a Riemannian generative decoder. It directly optimises manifold-valued latent variables via a Riemannian optimiser, thereby avoiding the difficulties of approximating densities on complex manifolds. By enforcing the manifold structure during training, the learnt latent representations remain aligned with the intrinsic geometry of the data, leading to more interpretable models and stable training dynamics.

The tangent plane of a data manifold is approximated through singular value decomposition and used for sampling points in alignment with the data's structure in a recent work [27]. This resembles our tangent space projected noise. For the methodology of the geodesic noise, a closely related idea has been explored in the context of Riemannian Laplace approximations for Bayesian inference in deep neural networks [28, 29].

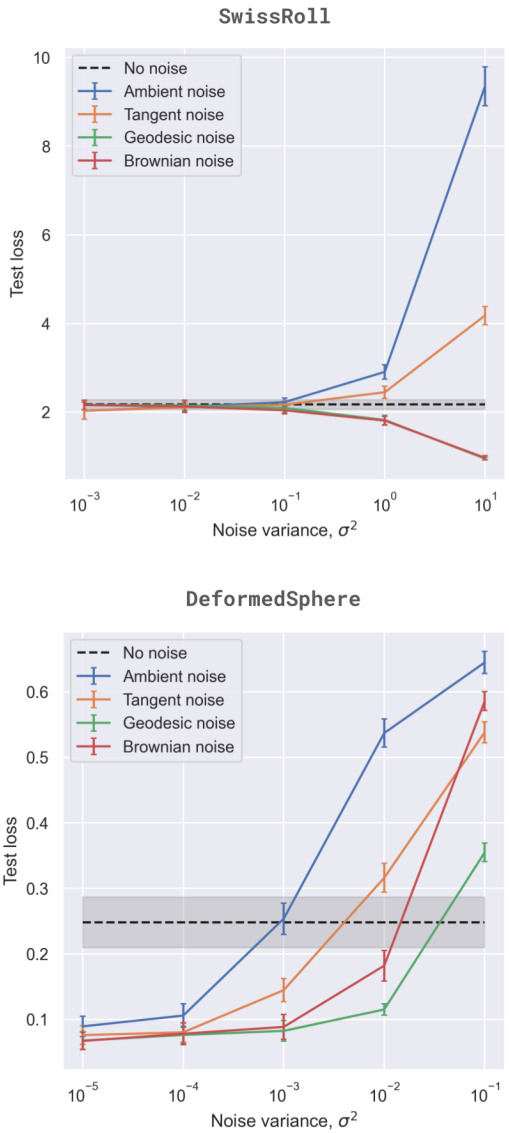

Figure 5. Test loss as a function of noise intensity σ^2 for different noise injection strategies. The geometry-aware noise strategies that stay on the manifold, i.e. geodesic noise and Brownian motion noise, show greater robustness to the noise intensity compared to ambient or tangential noise. All methods perform at least as well as training without noise (dashed line).

7 Conclusion

We have established several geometry-aware noise injection strategies and demonstrated their need through theoretical and experimental contributions. Further, we have shown their qualities and shortcomings. In particular, we find that while ambient Gaussian noise is simple and may improve performance on nearly Euclidean manifolds, it falls short on more curved or "wiggly" manifolds, where geodesic and Brownian motion noise provide clear

advantages. These geometry-aware strategies not only improve generalisation, but are also more robust to the noise intensity. The latter reduces the burden of hyperparameter tuning. Additionally, we proposed a framework for deforming parametrised manifolds to arbitrary manifolds, which extends the use of our methods beyond standard benchmark geometries. However, we remark that this added flexibility currently comes with increased computational cost.

Limitations and future work. We expect our approach to extend to higher dimensions. As explained in the introduction, a large difference between the dimensions of the ambient space and the data manifold could lead to more dramatic results as Gaussian noise samples will with high probability be normal to the manifold. However, this increases the computational complexity.

One direction we hope to explore is to consider manifolds of which we have no explicit parametrisation. One approach is to approximate the data manifold with a generative model, e.g. as previously done with variational autoencoders [30] or explore flow matching techniques as established in Section 4. Another promising direction for geometry-aware noise for point clouds is topological data analysis [31]. Lastly, an idea is to approximate the tangent plane of a data manifold as in [27] and use it for generating geodesics.

References

[1] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. "Gradient-based learning applied to document recognition". In: *Proceedings of the IEEE* 86.11 (2002), pp. 2278–2324.

[2] J. Sietsma and R. J. Dow. "Creating artificial neural networks that generalize". In: *Neural networks* 4.1 (1991), pp. 67–79.

[3] M. Ferianc, O. Bohdal, T. Hospedales, and M. Rodrigues. *Navigating Noise: A Study of How Noise Influences Generalisation and Calibration of Neural Networks*. 2024. arXiv: 2306.17630 [cs.LG]. URL: https://arxiv.org/abs/2306.17630.

[4] C. M. Bishop. "Training with Noise is Equivalent to Tikhonov Regularization". In: *Neural Computation* 7.1 (Jan. 1995), pp. 108–116. ISSN: 0899-7667. DOI: 10.1162/neco.1995.7.1.108. eprint: https://direct.mit.edu/neco/article-pdf/7/1/108/812990/neco.1995.7.1.108.pdf. URL: https://doi.org/10.1162/neco.1995.7.1.108.

[5] G. An. "The effects of adding noise during backpropagation training on a generalization performance". In: *Neural computation* 8.3 (1996), pp. 643–674.

[6] M. Belkin and P. Niyogi. "Laplacian eigenmaps and spectral techniques for embedding and clustering". In: *Advances in neural information processing systems* 14 (2001).

[7] C. Fefferman, S. Mitter, and H. Narayanan. "Testing the manifold hypothesis". In: *Journal of the American Mathematical Society* 29.4 (2016), pp. 983–1049.

[8] C. Fefferman, S. Ivanov, M. Lassas, and H. Narayanan. "Fitting a manifold to data in the presence of large noise". In: *arXiv preprint arXiv:2312.10598* (2023).

[9] P. W. Kuchel, C. D. Cox, D. Daners, et al. "Surface model of the human red blood cell simulating changes in membrane curvature under strain". In: *Scientific Reports* 11.1 (2021), p. 13712. DOI: 10.1038/s41598-021-92699-7. URL: https://doi.org/10.1038/s41598-021-92699-7.

[10] M. R. et al. "Detection of Epileptogenic Focal Cortical Dysplasia Using Graph Neural Networks: A MELD Study". English. In: *JAMA Neurology* 82.4 (Feb. 2025), pp. 397–406. ISSN: 2168-6149. DOI: 10.1001/jamaneurol.2024.5406.

[11] K. Matsuoka. "Noise injection into inputs in back-propagation learning". In: *IEEE Transactions on Systems, Man, and Cybernetics* 22.3 (1992), pp. 436–440.

[12] S. Rifai, X. Glorot, Y. Bengio, and P. Vincent. "Adding noise to the input of a model trained with a regularized objective". In: *arXiv preprint arXiv:1104.3250* (2011).

[13] A. N. Tikhonov. "Solutions of ill posed problems". In: (1977).

[14] J. M. Lee. *Introduction to Smooth Manifolds*. 2000.

[15] P.-A. Absil, R. Mahony, and R. Sepulchre. *Optimization Algorithms on Matrix Manifolds*. USA: Princeton University Press, 2007. ISBN: 0691132984.

[16] E. P. Hsu. "A brief introduction to Brownian motion on a Riemannian manifold". In: *lecture notes* (2008).

[17] E. P. Hsu. *Stochastic Analysis on Manifolds*. Vol. 38. Graduate Studies in Mathematics. American Mathematical Society, 2002, p. 281. ISBN: 9780821808023.

[18] N. Ikeda and S. Watanabe. *Stochastic differential equations and diffusion processes*. Vol. 24. Elsevier, 2014.

[19] R. T. Chen, Y. Rubanova, J. Bettencourt, and D. K. Duvenaud. "Neural ordinary differential equations". In: *Advances in neural information processing systems* 31 (2018).

[20] Y. Lipman, R. T. Chen, H. Ben-Hamu, M. Nickel, and M. Le. "Flow matching for generative modeling". In: *arXiv preprint arXiv:2210.02747* (2022).

[21] Y. N. Dauphin, A. Agarwala, and H. Mobahi. *Neglected Hessian Component Explains Mysteries in Sharpness Regularization*. 2024. arXiv: 2401.10809 [cs.LG]. URL: https://arxiv.org/abs/2401.10809.

[22] E. Massart. "Orthogonal regularizers in deep learning: how to handle rectangular matrices?" In: *2022 26th International Conference on Pattern Recognition (ICPR)*. IEEE. 2022, pp. 1294–1299.

[23] B. Massion and E. Massart. "Minimizers of Deficient Orthogonal Regularizers". In: *preprint* (2024). URL: https://www.esat.kuleuven.be/stadius/E/DEEPK2024/9_minimizers_of_deficient_orthog.pdf.

[24] A. Orvieto, H. Kersting, F. Proske, F. Bach, and A. Lucchi. "Anticorrelated noise injection for improved generalization". In: *International Conference on Machine Learning*. PMLR. 2022, pp. 17094–17116.

[25] A. Orvieto, A. Raj, H. Kersting, and F. Bach. "Explicit regularization in overparametrized models via noise injection". In: *International Conference on Artificial Intelligence and Statistics*. PMLR. 2023, pp. 7265–7287.

[26] A. Bjerregaard, S. Hauberg, and A. Krogh. "Riemannian generative decoder". In: *arXiv preprint arXiv:2506.19133* (2025).

[27] I. Kaufman and O. Azencot. "First-order manifold data augmentation for regression learning". In: *arXiv preprint arXiv:2406.10914* (2024).

[28] F. Bergamin, P. Moreno-Muñoz, S. Hauberg, and G. Arvanitidis. *Riemannian Laplace Approximations for Bayesian Neural Networks*. 2023. arXiv: 2306.07158 [stat.ML]. URL: https://arxiv.org/abs/2306.07158.

[29] N. Da Costa, B. Mucsányi, and P. Hennig. "Geometric Gaussian Approximations of Probability Distributions". In: *arXiv preprint arXiv:2507.00616* (2025).

[30] G. Arvanitidis, L. K. Hansen, and S. Hauberg. "Latent space oddity: on the curvature of deep generative models". In: *arXiv preprint arXiv:1710.11379* (2017).

[31] F. Chazal and B. Michel. *An introduction to Topological Data Analysis: fundamental and practical aspects for data scientists*. 2021. arXiv: 1710.04019 [math.ST]. URL: https://arxiv.org/abs/1710.04019.

A Computations on the example manifolds

A.1 Biconcave disc

The biconcave disc yields an approximation of human erythrocytes, as shown in [9]. Letting $r = \sqrt{u^2+v^2}$, and let a, b, c, d be parameters, then the height function for the upper half is given by

$$z(r) = d\sqrt{1-\frac{4r^2}{d^2}} \cdot \left(a + \frac{br^2}{d^2} + \frac{cr^4}{d^4}\right).$$

Here, d describes the diameter, a the height at the centre, b the height of the highest point, and c the flatness in the centre. A parametrisation of the upper half of this surface of rotation is given by

$$X(r,\theta) = (r\cos\theta, r\sin\theta, z(r)).$$

A.1.1 Tangential noise on the biconcave disc

The tangent space is then spanned by

$$\begin{aligned} X_r &= \left[\cos\theta, \sin\theta, \frac{\partial z}{\partial r}\right], \\ X_\theta &= [-r\sin\theta, r\cos\theta, 0]. \end{aligned}$$

An standard computation shows that

$$\begin{aligned} \frac{\partial z}{\partial r} &= \frac{-8r}{d\sqrt{1-\frac{4r^2}{d^2}}} \cdot \left(a + \frac{br^2}{d^2} + \frac{cr^4}{d^4}\right) \\ &\quad + \sqrt{1-\frac{4r^2}{d^2}} \cdot \left(\frac{2br}{d} + \frac{4cr^3}{d^3}\right), \end{aligned}$$

and clearly

$$\frac{\partial r}{\partial u} = \frac{2u}{r}, \quad \frac{\partial r}{\partial v} = \frac{2v}{r}.$$

The unit normal vector is now given by

$$\boldsymbol{n} = \frac{\left[\frac{\partial z}{\partial r} r\cos\theta, -\frac{\partial z}{\partial r} r\sin\theta, r\right]}{r\left(\frac{\partial z}{\partial r}^2 + 1\right)}.$$

A.1.2 Geodesics on the biconcave disc

We obtain

$$g(r,\theta) = \begin{pmatrix} 1+\frac{\partial z}{\partial r}^2 & 0 \\ 0 & r^2 \end{pmatrix}.$$

A computation shows that

$$\begin{aligned} \ddot{r}(t) &= -\frac{\frac{\partial z}{\partial r}\frac{\partial^2 z}{\partial r^2}}{1+\frac{\partial z}{\partial r}^2} \cdot \dot{r}(t)^2 + \frac{r(t)}{1+\frac{\partial z}{\partial r}^2} \cdot \dot{\theta}(t)^2, \\ \ddot{\theta}(t) &= -\frac{2\dot{r}(t)}{r(t)} \cdot \dot{\theta}(t). \end{aligned}$$

The second derivative of z is given by the following:

$$\begin{aligned} \frac{\partial^2 z}{\partial r^2} &= \frac{-4}{d}\left(1-\frac{4r^2}{d^2}\right)^{-\frac{3}{2}}\left(a+\frac{br^2}{d^2}+\frac{cr^4}{d^4}\right) \\ &\quad -\frac{16r}{d}\cdot\left(1-\frac{4r^2}{d^2}\right)^{-\frac{1}{2}}\cdot\left(\frac{br}{d^2}+\frac{2cr^3}{d^4}\right) \\ &\quad +2\sqrt{1-\frac{4r^2}{d^2}}\cdot\left(\frac{b}{d^2}+\frac{6cr^2}{d^4}\right). \end{aligned}$$

A.1.3 Brownian motion on the biconcave disc

For the Brownian motion, we yield

$$\begin{aligned} dr(t) &= \frac{1}{2}\cdot\left(\frac{1+\frac{\partial z}{\partial r}^2 - \frac{\partial z}{\partial r}\frac{\partial^2 z}{\partial r^2}}{(1+\frac{\partial z}{\partial r}^2)^2}\right)dt \\ &\quad + \frac{1}{\sqrt{1+\frac{\partial z}{\partial r}^2}} dB(t)_1 \\ d\theta(t) &= \frac{1}{2}\cdot\left(\frac{r\frac{\partial z}{\partial r}\frac{\partial^2 z}{\partial r^2} - 1 - \frac{\partial z}{\partial r}^2}{r^3(1+\frac{\partial z}{\partial r}^2)}\right)dt + \frac{1}{r}dB(t)_2, \end{aligned}$$

for all $r > 0$.

A.2 Spheroids

We consider manifolds which are squeezed spheres. For $a, c \in \mathbb{R}^+$, consider the parametrisation $X : \mathbb{R}^2 \to \mathbb{R}^3$ given by

$$X(u,v) = (a\sin u\sin v, a\sin u\cos v, c\cos u).$$

If $a = c$, then this gives the usual sphere. If $a > c$, then the manifold is a sphere squished along the z-axis. The tangent plane is spanned by

$$\begin{aligned} X_u &= [a\cos u\sin v, a\cos u\cos v, -c\sin u], \\ X_v &= [a\sin u\cos v, -a\sin u\sin v, 0]. \end{aligned}$$

We then obtain the metric

$$\begin{pmatrix} a^2\cos^2 u + c^2\sin^2 u & 0 \\ 0 & a^2\sin^2 u \end{pmatrix}.$$

A.2.1 Tangential noise on the spheroid

To obtain tangential noise, we note that the unit normal is given by

$$\boldsymbol{n} = \frac{[c\sin u\sin v, c\sin u\cos v, a\cos u]}{\sqrt{c^2\sin^2 u + a^2\cos^2 u}}.$$

A.2.2 Geodesics on the spheroid

A curve $\gamma = X \circ \alpha$ is a geodesic on the spheroid if and only if $\alpha : I \to \mathbb{R}^2$ satisfies

$$\begin{aligned} \ddot{\alpha}_1(t) &= \frac{(a^2-c^2)\sin\alpha_1(t)\cos\alpha_1(t)}{a^2\cos^2\alpha_1(t) + c^2\sin^2\alpha_1(t)} \cdot \dot{\alpha}_1(t)^2 \\ &\quad + \frac{a^2\sin\alpha_1(t)\cos\alpha_1(t)}{a^2\cos^2\alpha_1(t) + c^2\sin^2\alpha_1(t)} \cdot \dot{\alpha}_2(t)^2, \\ \ddot{\alpha}_2(t) &= -2\cdot\frac{\cos\alpha_1(t)}{\sin\alpha_1(t)}\dot{\alpha}_1(t)\dot{\alpha}_2(t). \end{aligned}$$

A.2.3 Brownian motion on the spheroid

A computation yields the following result for the Brownian motion.

$$\begin{aligned} du_k(t) &= \left[\begin{matrix} \frac{a^2\cos u}{2\sin u(a^2\cos^2 u + c^2\sin^2 u)^2} \\ 0 \end{matrix}\right]_k dt \\ &+ \left[\begin{pmatrix} \frac{1}{\sqrt{a^2\cos^2 u + c^2\sin^2 u}} & 0 \\ 0 & \frac{1}{a\sin u} \end{pmatrix} dB(t)\right]_k . \end{aligned}$$

A.3 Tori

We also investigate different tori, some more like onion rings, others more like beads. For coefficients $a, c \in \mathbb{R}^+$, they can be parametrised by $X : \mathbb{R}^2 \to \mathbb{R}^3$,

$$X(u,v) = ((a + c\sin u)\sin v, (a + c\sin u)\cos v, c\cos u).$$

Here, c describes the thickness of the handle and a the size of the torus. To avoid self-intersection, c is bounded by a. Further, if $c << a$, we have an onion ring, and if $c \uparrow a$ we have a rounded torus with a very thin hole.

A.3.1 Tangential noise on tori

The tangent plane is spanned by

$$\begin{aligned} X_u &= [c\cos u\sin v, c\cos u\cos v, -c\sin u], \\ X_v &= [(a + c\sin u)\cos v, -(a + c\sin u)\sin v, 0]. \end{aligned}$$

This yields the metric

$$g = \begin{pmatrix} c^2 & 0 \\ 0 & (a + c\sin u)^2 \end{pmatrix}.$$

The unit normal is given by

$$\boldsymbol{n} = \frac{[\sin u\sin v, \sin u\cos v, \cos u\sin^2 v]}{\sqrt{\sin^2 u + \cos^2 u\sin^2 u}}.$$

A.3.2 Geodesic noise on tori

A curve $\gamma = X \circ \alpha$ on the torus is a geodesic if and only if α satisfies

$$\begin{aligned} \ddot{\alpha}_1(t) &= \frac{(a + c\sin\alpha_1(t))\cos\alpha_1(t)}{c} \cdot \dot{\alpha}_2(t)^2 \\ \ddot{\alpha}_2(t) &= 2\frac{c\cos\alpha_1(t)}{a + c\sin\alpha_1(t)} \cdot \dot{\alpha}_1(t)\dot{\alpha}_2(t). \end{aligned}$$

A.3.3 Brownian motion on tori

We obtain the Brownian motion terms

$$\begin{aligned} du_k(t) &= \left[\begin{matrix} \frac{\cos u}{2c(a + c\sin u)} \\ 0 \end{matrix}\right]_k dt, \\ &+ \left[\begin{pmatrix} \frac{1}{c} & 0 \\ 0 & \frac{1}{a + c\sin u} \end{pmatrix} dB(t)\right]_k \end{aligned}$$

B Implementation details

B.1 Geodesic noise

To simplify our computations, instead of sampling a vector $\boldsymbol{\epsilon}_\top \sim \mathcal{N}\left(0, \sigma^2\mathbf{P}\right)$ in the tangent space, we can also sample a vector $\tilde{\boldsymbol{\epsilon}}$ in the parameter space $\mathbb{R}^d$ from an adjusted distribution. In the following assume that $\boldsymbol{u} \in \mathbb{R}^d$, $X(\boldsymbol{u}) = \boldsymbol{x} \in \mathcal{M}$, where X is a smooth parametrisation of a regular manifold $\mathcal{M}$. As previously described, the Jacobian transforms vectors in the parameter space to the tangent space, i.e. for a vector $\boldsymbol{\epsilon} \in T_{\boldsymbol{u}}\mathbb{R}^d$, we have that

$$\boldsymbol{\epsilon} = \mathbf{J}_X\tilde{\boldsymbol{\epsilon}} \in T_{\boldsymbol{x}}\mathcal{M}$$

For the inverse relation, we obtain

$$\tilde{\boldsymbol{\epsilon}} = g^{-1}\mathbf{J}_X^\top\boldsymbol{\epsilon}.$$

Consequently, if

$$\boldsymbol{\epsilon} \sim \mathcal{N}(0, \sigma^2\mathbb{I}_D),$$

then for its tangential component it holds that

$$\boldsymbol{\epsilon}_\top \sim \mathcal{N}(0, \sigma^2\mathbf{P}),$$

and for the pull-back it holds that

$$\tilde{\boldsymbol{\epsilon}}_\top \sim \mathcal{N}\left(\mathbf{0}, \sigma^2 g^{-1}\mathbf{J}_X^\top\mathbf{P}\mathbf{J}_X g^{-1}\right), \tag{9}$$

which follows from affine transformation properties of the multivariate Gaussian distribution.

We now can find the curve $\alpha : \mathbb{R} \to \mathbb{R}^d$ such that

$$\alpha(0) = X^{-1}(\boldsymbol{x}), \quad \dot{\alpha}(0) = \tilde{\boldsymbol{\epsilon}}_\top.$$

Our new sample point is then

$$\tilde{\boldsymbol{x}} = X\left(\alpha(\|\tilde{\boldsymbol{\epsilon}}_\top\|)\right).$$

This method is equivalent to the one described in Subsection 3.2. For simplicity, we ignore the injectivity radius of the domain of the exponential map. This is not a problem since we do not require injectivity for our purposes and the manifolds we consider are complete.

B.2 Functions on the manifolds

For the `Sphere`, `SqueezedSphere` and `DeformedSphere`, we select the target function as:

$$y = v$$

i.e. the second local coordinate.

For the `Bead` we select the target function as:

$$y = \sin v$$

i.e. a periodic function of the second local coordinate.

For the `OnionRing` we select the target function as:

$$y = 100 \cdot c \cdot \cos u = 100 \cdot z$$

i.e. the scaled height of the manifold.

For the `SwissRoll` we select the target function as:

$$y = u$$

i.e. the first local coordinate, which is a linearly increasing function along the roll.

C Details on manifold deformations

Recall the definition of the flow field from Equation 8:

$$\frac{d}{dt}\phi_t\left(X\left(\boldsymbol{u}\right)\right) = v_t\left(\phi_t\left(X\left(\boldsymbol{u}\right)\right)\right).$$

We take the derivative with respect to the local coordinates $\boldsymbol{u}$ and get

$$\frac{\partial}{\partial u}\left(\frac{d}{dt}\phi_t\left(X\left(\boldsymbol{u}\right)\right)\right) = \frac{\partial}{\partial u}v_t\left(\phi_t\left(X\left(\boldsymbol{u}\right)\right)\right),$$

which is equivalent to

$$\frac{d}{dt}\frac{\partial}{\partial u}\phi_t\left(X\left(\boldsymbol{u}\right)\right) = \frac{\partial}{\partial u}v_t\left(\phi_t\left(X\left(u\right)\right)\right).$$

By using the chain rule on the right hand side, we get

$$\frac{d}{dt}\frac{\partial}{\partial u}\phi_t\left(X\left(\boldsymbol{u}\right)\right) = \left.\frac{\partial v_t\left(\phi_t\left(\boldsymbol{x}\right)\right)}{\partial \phi_t}\right|_{\boldsymbol{x}=X\left(\boldsymbol{u}\right)}\frac{\partial}{\partial u}\phi_t\left(X\left(\boldsymbol{u}\right)\right).$$

We get the Jacobian ODE by setting

$$\mathbf{J}_{\boldsymbol{u}}\left(t\right) := \frac{\partial \phi_t\left(X\left(\boldsymbol{u}\right)\right)}{\partial u},$$

$$\mathbf{J}_{v}\left(t\right) := \left.\frac{\partial v_t\left(\phi_t\left(\boldsymbol{x}\right)\right)}{\partial \phi_t}\right|_{\boldsymbol{x}=X\left(\boldsymbol{u}\right)}.$$