Metrics for Deep Generative Models

Nutan Chen[*] **Alexej Klushyn**[*] **Richard Kurle**[*]
Xueyan Jiang **Justin Bayer** **Patrick van der Smagt**
{first name dot last name}@volkswagen.de
AI Research, Data:Lab, Volkswagen Group, Munich, Germany

Abstract

Neural samplers such as variational autoencoders (VAEs) or generative adversarial networks (GANs) approximate distributions by transforming samples from a simple random source—the latent space—to samples from a more complex distribution represented by a dataset. While the manifold hypothesis implies that a dataset contains large regions of low density, the training criterions of VAEs and GANs will make the latent space densely covered. Consequently points that are separated by low-density regions in observation space will be pushed together in latent space, making stationary distances poor proxies for similarity. We transfer ideas from Riemannian geometry to this setting, letting the distance between two points be the shortest path on a Riemannian manifold induced by the transformation. The method yields a principled distance measure, provides a tool for visual inspection of deep generative models, and an alternative to linear interpolation in latent space. In addition, it can be applied for robot movement generalization using previously learned skills. The method is evaluated on a synthetic dataset with known ground truth; on a simulated robot arm dataset; on human motion capture data; and on a generative model of handwritten digits.

1 Introduction

Matrices of pairwise distances serve as the input for a wide range of classic machine learning algorithms, such as k-nearest neighbour, multidimensional scaling, or stationary kernels. In the case of high-dimensional spaces, obtaining a meaningful distance is challenging for two reasons. First, choosing a metric, e.g. an instance of Minkowski distance, comes with certain assumptions on the data—e.g., distances are invariant to rotation under the L2 norm. Second, these distances become increasingly meaningless for higher dimensions [Aggarwal et al., 2001]. Numerous researchers have proposed to learn distances from data, sometimes referred to as *metric learning* [Xing et al., 2003, Weinberger et al., 2006, Davis et al., 2007, Kulis et al., 2013]. For data distributed according to the multivariate normal, the Mahalanobis distance is a popular choice, making the distance measure effectively invariant to translation and scaling. The idea of a linear transformation of the data has been extended to also reflect supervised side information such as class labels [Weinberger and Saul, 2009, Goldberger et al., 2004]. Further work has pushed this to use non-linear [Salakhutdinov and Hinton, 2007] and recurrent [Bayer et al., 2012] transformations.

Probabilistic modelling of high-dimensional data has progressed enormously. Two distinct "neural sampling" approaches are those of generative adversarial networks (GANs) [Goodfellow et al., 2014] and variational autoencoders (VAEs) [Kingma and Welling, 2013, Rezende et al., 2014].

This work aims to bring a set of techniques to neural sampling that makes them powerful tools for metric learning. Pioneering work on interpolation and generation between two given points on a Riemannian manifold includes [Noakes et al., 1989] and [Crouch and Leite, 1995]. In addition, Principal Geodesic Analysis (PGA) [Fletcher et al., 2004] describing the variability of data on a manifold uses geodesics in Principal component analysis. Recent work of Tosi et al. [2014] proposed to perceive the latent space of Gaussian process latent variable models (GP-LVMs) as a Riemannian manifold, where the distance between two data points is given as the shortest path along the data manifold.

We transfer the idea of [Tosi et al., 2014] to neural samplers. We show how to represent such shortest paths in a parameterized fashion, where the calculation of the distance between two data points is effectively a minimization of the length of a curve. The method is evaluated on a range of high-dimensional datasets. Further, we provide evidence of the manifold hypothesis [Rifai et al., 2011]. We would also like to mention Arvanitidis et al. [2017] and Shao et al. [2017] who independently worked on this topic at the same time as us.

In robotic domains, our approach can be applied to path planning based on the learned skills. The demonstrations from experts enable robots to generate particular motions. Simultaneously, robots require natural motion exploration [Havoutis and Ramamoorthy, 2013]. In our method, a Riemannian metric is used to achieve such optimal motion paths along the data manifolds.

2 Using Riemannian geometry in generative latent variable models

Latent variable models are commonly defined as

$$p(\mathbf{x}) = \int p(\mathbf{x} \,|\, \mathbf{z}) \, p(\mathbf{z}) \, d\mathbf{z}, \tag{1}$$

where latent variables $\mathbf{z} \in \mathbb{R}^{N_z}$ are used to explain the data $\mathbf{x} \in \mathbb{R}^{N_x}$.

Assume we want to obtain a distance measure $\delta(\mathbf{x}, \tilde{\mathbf{x}})$ from the learned manifolds, which adequately reflects the "similarity" between data points $\mathbf{x}$ and $\tilde{\mathbf{x}}$. If we can infer the corresponding latent variables $\mathbf{z}$ and $\tilde{\mathbf{z}}$, an obvious choice is the Euclidean distance in latent space, $||\mathbf{z} - \tilde{\mathbf{z}}||_2$. This has the implicit assumption that moving a certain distance in latent spaces moves us proportionally far in observation space, $||\mathbf{z} - \tilde{\mathbf{z}}||_2 \propto ||\mathbf{x} - \tilde{\mathbf{x}}||_2$. But this is a fallacy: for latent variables to adequately model the data, stark discontinuities in the likelihood $p(\mathbf{x} \,|\, \mathbf{z})$ are virtually always present. To see this, we note that the prior can be expressed as the posterior aggregated over the data:

$$p(\mathbf{z}) = \int p(\mathbf{z} \,|\, \mathbf{x}) \, p(\mathbf{x}) \, d\mathbf{x} = \mathbb{E}_{\mathbf{x} \sim p(\mathbf{X})}[p(\mathbf{z} \,|\, \mathbf{x})].$$

A direct consequence of the discontinuities is that there are no regions of low density in the latent space. Hence, separated manifolds in the observation space (e.g., the set of points from different classes) may be placed directly next to each other in latent space—a property that can only be compensated through rapid changes in the likelihood $p(\mathbf{x} \,|\, \mathbf{z})$ at the respective "borders".

For the estimation of nonlinear latent variable models we use importance-weighted autoencoders (IWAE) [Burda et al., 2015] (see Section 2.1). Treating the latent space as a Riemannian manifold (see Section 2.2) provides tools to define distances between data points by taking into account changes in the likelihood.

2.1 Importance-weighted autoencoder

Inference and learning in models of the form given by Eq. (1)—based on the maximum-likelihood principle—are intractable because of the marginalization over the latent variables. Typically, approximations are used which are either based on sampling or on variational inference. In the latter case, the intractable posterior is approximated by a distribution $q(\mathbf{z}^{(i)})$. The problem of inference is then substituted by one of optimization, namely the maximization of the evidence lower bound (ELBO). Let $\mathbf{X} = \{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(N)}\}$ be observable data and $\mathbf{z}^{(i)}$ the corresponding latent variables. Further, let $p_\theta(\mathbf{x}|\mathbf{z})$ be a likelihood function parameterized by θ. Then

$$\ln p_\theta(\mathbf{X}) = \sum_{i=1}^{N} \ln p_\theta(\mathbf{x}^{(i)})$$
$$\geq \sum_{i=1}^{N} \mathbb{E}_{q(\mathbf{z}^{(i)})} \left[\ln \frac{p_\theta(\mathbf{x}^{(i)}|\mathbf{z}^{(i)}) p_\theta(\mathbf{z}^{(i)})}{q(\mathbf{z}^{(i)})} \right] = \mathcal{L}_{\mathrm{ELBO}}. \tag{2}$$

If we implement $q(\mathbf{z}^{(i)}) = q_\phi(\mathbf{z}|\mathbf{x}^{(i)})$ with a neural network parameterized by ϕ, we obtain the variational autoencoder of Kingma and Welling [2013], which jointly optimizes $\mathcal{L}_{\mathrm{ELBO}}$ with respect to θ and ϕ.

Since the inference and generative models are tightly coupled, an inflexible variational posterior has a direct impact on the generative model, causing both models to underuse their capacity.

In order to learn richer latent representations and achieve better generative performance, the importance-weighted autoencoder (IWAE) [Burda et al., 2015, Cremer et al., 2017] has been introduced. It treats $q_\phi(\mathbf{z}|\mathbf{x})$ as a proposal distribution and obtains a tighter lower bound using importance sampling:

$$\ln p_\theta(\mathbf{X}) = \sum_{i=1}^{N} \ln p_\theta(\mathbf{x}^{(i)})$$
$$\geq \sum_{i=1}^{N} \mathbb{E}_{\mathbf{z}_1^{(i)}, \ldots, \mathbf{z}_K^{(i)} \sim q_\phi(\mathbf{z}^{(i)}|\mathbf{x}^{(i)})} \left[\ln \frac{1}{K} \sum_{k=1}^{K} w_k^{(i)} \right], \tag{3}$$

where $w_k^{(i)}$ are the importance weights:

$$w_k^{(i)} = \frac{p_\theta(\mathbf{x}^{(i)}|\mathbf{z}_k^{(i)}) \, p_\theta(\mathbf{z}_k^{(i)})}{q_\phi(\mathbf{z}_k^{(i)}|\mathbf{x}^{(i)})}. \tag{4}$$

The IWAE is the basis of our approach, since it can yield an accurate generative model.

2.2 Riemannian geometry

A Riemannian manifold is a differentiable manifold M with a metric tensor $\mathbf{G}$. It assigns to each point $\mathbf{z}$ an inner product on the tangent space $T_{\mathbf{z}}M$, where the inner product is defined as:

$$\langle \mathbf{z}', \mathbf{z}' \rangle_{\mathbf{z}} := \mathbf{z}'^{T} \mathbf{G}(\mathbf{z}) \, \mathbf{z}' \tag{5}$$

with $\mathbf{z}' \in T_{\mathbf{z}}M$ and $\mathbf{z} \in M$.

Consider a curve $\gamma : [0, 1] \rightarrow \mathbb{R}^{N_z}$ in the Riemannian manifold, transformed by a continuous function $f(\gamma(t))$ to an N_x-dimensional observation space, where $\gamma(t) \in \mathbb{R}^{N_z}$. The length of the curve in the observation space is defined as

$$L(\gamma) := \int_0^1 \left\| \frac{\partial f(\gamma(t))}{\partial t} \right\| \mathrm{d}t$$

$$= \int_0^1 \left\| \frac{\partial f(\gamma(t))}{\partial \gamma(t)} \frac{\partial \gamma(t)}{\partial t} \right\| \mathrm{d}t$$

$$= \int_0^1 \left\| \mathbf{J} \frac{\partial \gamma(t)}{\partial t} \right\| \mathrm{d}t, \tag{6}$$

where $\mathbf{J}$ is the Jacobian. Eq. (6) can be expressed as

$$L(\gamma) = \int_0^1 \sqrt{\langle \gamma'(t), \gamma'(t) \rangle_{\gamma(t)}} \mathrm{d}t \tag{7}$$

with the metric tensor $\mathbf{G} = \mathbf{J}^T \mathbf{J}$.

3 Approximating the geodesic

In this work, we are primarily interested in length-minimizing curves between samples of generative models. In Riemannian geometry, locally length-minimizing curves are referred to as geodesics. We treat the latent space of generative models as a Riemannian manifold. This allows us to parametrize the curve in the latent space, while distances are measured by taking into account distortions from the generative model.

We use a neural network $g_{\omega} : \mathbb{R} \rightarrow \mathbb{R}^{N_z}$ to approximate the curve γ in the latent space, where ω are the weights and biases. The function f from Eqs. (6) corresponds to the mean of the generative model's probability distribution $h^{\mathrm{gen}} : \mathbb{R}^{N_z} \rightarrow \mathbb{R}^{N_x}$ and the components of the Jacobian are

$$J_{i,j} = \frac{\partial x_i}{\partial z_j}. \tag{8}$$

x_i and z_j denote the i-th and j-th element of the generated data points $\mathbf{x}$ and latent variables $\mathbf{z}$, respectively, with $\mathbf{x} \in \mathbb{R}^{N_x}$ and $\mathbf{z} \in \mathbb{R}^{N_z}$.

We approximate the integral of Eq. (7) with n equidistantly spaced sampling points of $t \in [0, 1]$:

$$L(g_{\omega}(t)) \approx \frac{1}{n} \sum_{i=1}^{n} \sqrt{\langle g'_{\omega}(t_i), g'_{\omega}(t_i) \rangle_{g_{\omega}(t_i)}} \tag{9}$$

$$= \frac{1}{n} \sum_{i=1}^{n} \sqrt{g'_{\omega}(t_i)^T \mathbf{J}^T \mathbf{J} g'_{\omega}(t_i)}. \tag{10}$$

The term inside the summation can be interpreted as the rate of change at point $g_{\omega}(t_i)$, induced by the generative model, and we will refer to it as velocity:

$$\phi(t) = \sqrt{\langle g'_{\omega}(t_i), g'_{\omega}(t_i) \rangle_{g_{\omega}(t_i)}}. \tag{11}$$

An approximation of the geodesic between two points in the latent space is obtained by minimizing the length in Eq. (10), where the weights and biases ω of the neural network $g_{\omega}(t)$ are subject to optimization.

With the start and end points of the curve in the latent space given as $\mathbf{z}_0$ and $\mathbf{z}_1$, we consider the following constrained optimization problem:

$$\min_{\omega} L(g_{\omega}(t))$$
$$s.t. \ g_{\omega}(0) = \mathbf{z}_0, \ g_{\omega}(1) = \mathbf{z}_1. \tag{12}$$

3.1 Dealing with boundary constraints

To satisfy the boundary constraints in Eq. (12), we shift and rescale the predicted line to get

$$\mathbf{z}(t) = \mathbf{A}\hat{\mathbf{z}}(t) - \mathbf{B}, \tag{13}$$

where

$$\mathbf{A} = \frac{\mathbf{z}_0 - \mathbf{z}_1}{\hat{\mathbf{z}}(0) - \hat{\mathbf{z}}(1)}$$

$$\mathbf{B} = \frac{\mathbf{z}_0\hat{\mathbf{z}}(1) - \mathbf{z}_1\hat{\mathbf{z}}(0)}{\hat{\mathbf{z}}(0) - \hat{\mathbf{z}}(1)}. \tag{14}$$

$\hat{\mathbf{z}}$ and $\mathbf{z}$ are the outputs of the neural network g_{ω} before cq. after normalization. The advantage of this proceed is that the optimization problem in Eq. (12) simplifies to find $\min_{\omega} L(g_{\omega}(t))$.

3.2 Smoothing the metric tensor

To ensure the geodesic is following the data manifold—which entails that the manifold distance is smaller than the Euclidean distance—a penalization term is added to smooth the metric tensor $\mathbf{G}$. It leads to the following loss function:

$$\mathcal{L} = L + \lambda_s \|\mathbf{G}\|_2, \tag{15}$$

where $\lambda_s > 0$ acts as a regularization coefficient. This optimization step is implemented as a post-processing of Eq. (12) via singular-value decomposition (SVD)

$$\mathbf{G} = \mathbf{U}\mathbf{S}\mathbf{V}^T, \tag{16}$$

where the columns of $\mathbf{U}$ are the eigenvectors of the covariance matrix $\mathbf{G}\mathbf{G}^T$, and the columns of V are the eigenvectors of $\mathbf{G}^T\mathbf{G}$. The diagonal entries in $\mathbf{S}$ contain singular values with scaling information about how a vector is stretched or shrunk when it is transformed from the column space of $\mathbf{G}$ to the row space of $\mathbf{G}$.

Minimizing the term $\lambda_s\|\mathbf{G}\|_2$ is equivalent to a low-rank reconstruction for $\mathbf{G}$

$$\hat{\mathbf{G}} = \mathbf{U}_r \text{diag}\left\{\frac{s_i^3}{s_i^2 + \lambda_s}\right\}_{i=1}^r \mathbf{V}_r^T, \tag{17}$$

where r is a pre-defined lower rank of $\mathbf{G}$. r is predefined which is equal or slightly smaller than the full rank of the metric tensor. λ_s rescales the singular values of $\mathbf{G}$ nonlinearly, which allows making the smaller singular values much smaller than the leading singular values. The smoothing therefore weakens the reconstructed off-diagonal values of $\hat{\mathbf{G}}$ which correspondingly reduces the manifold distance dramatically compared to the Euclidean distance. The smoothing effect is that a higher λ_s augments the difference between the Euclidean interpolation and the path along the manifold—and will be demonstrated experimentally in Section 4.3.

4 Experiments

We evaluate our approach by conducting a series of experiments on three different datasets—an artificial pendulum dataset, the binarized MNIST digit dataset [Larochelle and Murray, 2011], a simulated robot arm dataset and the human motion dataset[1].

Our goal is to enable smooth interpolations between the reconstructed images of an importance-weighted autoencoder and to differentiate between classes within the latent space. To show that the paths of geodesics can differ from Euclidean interpolations, the following experiments mainly focus on comparing geodesics with the Euclidean interpolations as well as the reconstructed data generated from points along their paths.

4.1 Training

In all experiments, we chose a Gaussian prior $p(\mathbf{z}) = \mathcal{N}(\mathbf{0}, \mathbf{I})$. The inference model and the likelihood are

---

[1]http://mocap.cs.cmu.edu/

represented by random variables of which the parameters are functions of the respective conditions.

For the inference model we consistently used a diagonal Gaussian, i.e. $q_\phi(\mathbf{z}|\mathbf{x}) = \mathcal{N}(\mu_\phi(\mathbf{x}), \text{diag}(\sigma_\phi^2(\mathbf{x})))$. Depending on the experiments, the likelihood $p_\theta(\mathbf{x}|\mathbf{z})$ either represents a Bernoulli variable $\mathcal{B}(r_\theta(\mathbf{z}))$ or a Gaussian $\mathcal{N}(\mu_\theta(\mathbf{z}), \sigma^2)$. σ is a global variable and the parameters $r_\theta, \mu_\theta, \sigma_\theta^2, \mu_\phi, \sigma_\phi^2$ are functions of the latent variables represented by neural networks parameterized by θ and ϕ respectively.

The hyperparameters of g_ω are summarized in Table 1 of the Appendix. We used sigmoid, tanh and softplus activation functions in the generative model (see App. B and C). See App. A for further details of the training procedure.

4.2 Visualization

There are several approaches to visualize the properties of the metric tensor, including Tissot's indicatrix. We use the magnification factor to visualize metric tensors during the evaluation, when we have two latent dimensions. The magnification factor [Bishop et al., 1997] is defined as

$$MF := \sqrt{\det \mathbf{G}}. \tag{18}$$

To get an intuitive understanding of the magnification factor, it is helpful to consider the rule for changing variables $d\mathbf{x} = \sqrt{\det \mathbf{J}_\mathbf{x}} d\mathbf{x}'$. This rule shows the relation between infinitesimal volumes of different equidimensional Euclidean spaces. The same rule can be applied to express the relationship between infinitesimal volumes of a Euclidean space and a Riemannian manifold—with the difference of using the MF instead of $\sqrt{\det \mathbf{J}_\mathbf{x}}$. Hence, the magnification factor visualizes the extent of change of the infinitesimal volume by mapping a point from the Riemannian manifold to the Euclidean space [Gemici et al., 2016].

4.3 Pendulum

We created an artificial dataset of 16×16 pixel images of a pendulum with a joint angle as the only degree of freedom and augmented it by adding 0.05 per-pixel Gaussian noise. We generated 15,000 pendulum images, with joint angles uniformly distributed in the range [0, 360]. The architecture of the IWAE can be found in Table 2 of the Appendix.

Fig. 2 illustrates the trained two-dimensional latent space of the IWAE. The grayscale in the background is proportional to the magnification factor, whereas the rotation angles of the pendulum are encoded by colors. The comparison of the geodesic (see Fig. 1, top row)
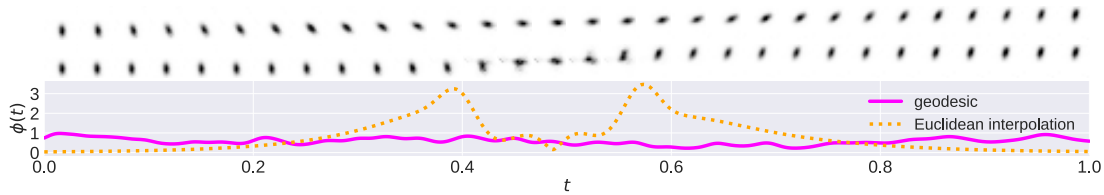
Figure 1: The reconstructions of the geodesic and the Euclidean interpolation between a single pair from the pendulum data. Top row: the mean of the reconstruction from the geodesic. Middle row: the mean of the reconstruction from the Euclidean interpolation. Bottom row: the velocity (Eq. (11)) of each sample. The distance of the Euclidean interpolation is 0.827, whereas the distance of the geodesic is 0.538.
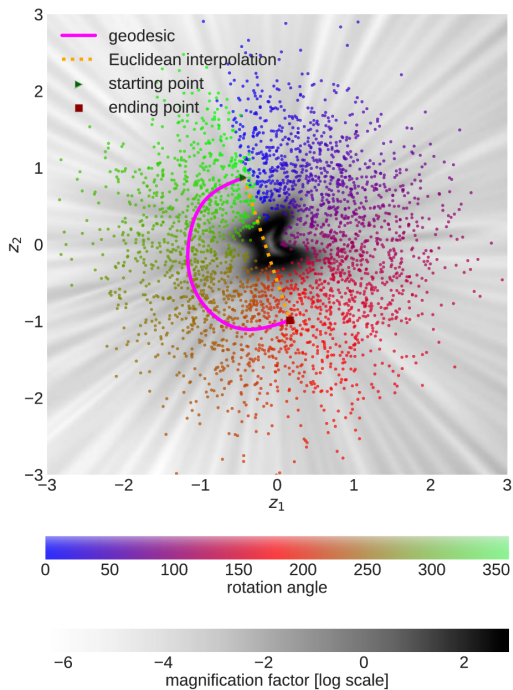

Figure 2: Geodesic and Euclidean interpolation in the latent space of the artificial pendulum dataset. The color encodes the pendulum rotation angles.

Figure 3: λ_s. The distances, along the Euclidean interpolation and the data manifold, between 100 pairs of randomly selected starting and end points. The shadow areas represent a 95% confidence interval.

Figure 4: The horizontal axis is the angle between the starting and end points. The average of the length of the geodesics, the Euclidean interpolations, and the paths along the data manifold are 0.319, 0.820, and 0.353 respectively.

with the Euclidean interpolation (Fig. 1, middle row) shows a much more uniform rotation of the pendulum for reconstructed images of points along the geodesic.

For this dataset, an SVD regularization with large values of λ_s was necessary for the optimization, to yield a path along the data manifold. Fig. 3 illustrates the influence of λ_s on the distance metric. It is a property of this dataset, due to the generative distribution, that small values of λ_s lead to shorter distances of the Euclidean interpolation than of paths along the data manifold. Fig. 4 shows the interpolations of 100 pairs of samples. We compared the geodesic with the Euclidean interpolation and an interpolation along the data manifold. The samples are randomly chosen with
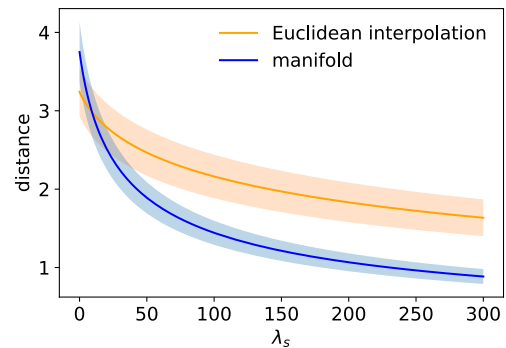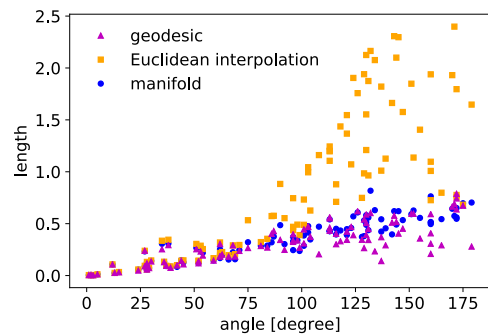
the condition to have a difference in the rotation angle of (0, 180] degrees. The distances of the geodesics and the paths along the data manifold are linearly correlated to the angles between two points in the observation space and fit to each other.
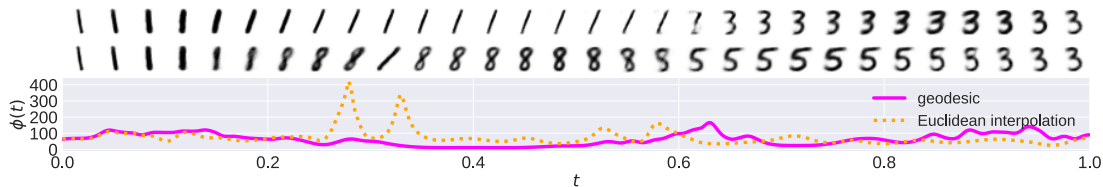
Figure 5: Reconstructions of the geodesic and the Euclidean interpolation based on the MNIST digit dataset. Top row: the mean of the reconstruction from the geodesic. Middle row: the mean of the reconstruction from the Euclidean interpolation. Bottom row: the velocity (Eq. (11)) of each sample. The distance of the Euclidean interpolation is 74.3, whereas the distance of the geodesic amounts to 62.9.
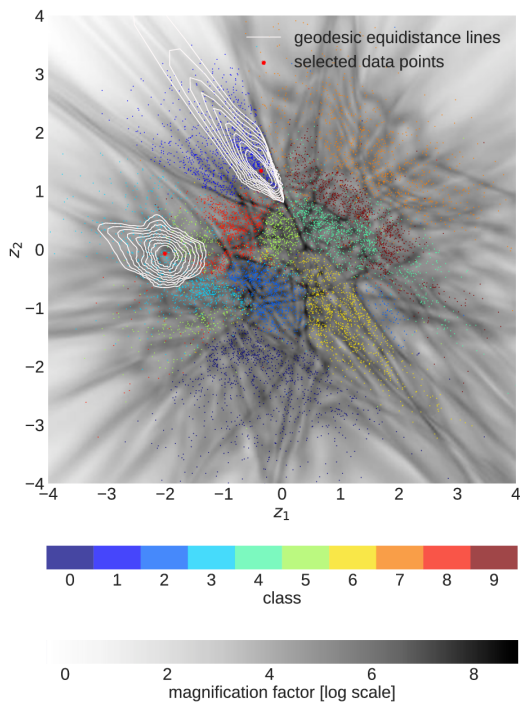

Figure 6: Equidistance lines around two selected data points in the latent space. Equidistance refers to the observation space and illustrates how regarding the latent space as a Riemannian manifold can help to separate classes.
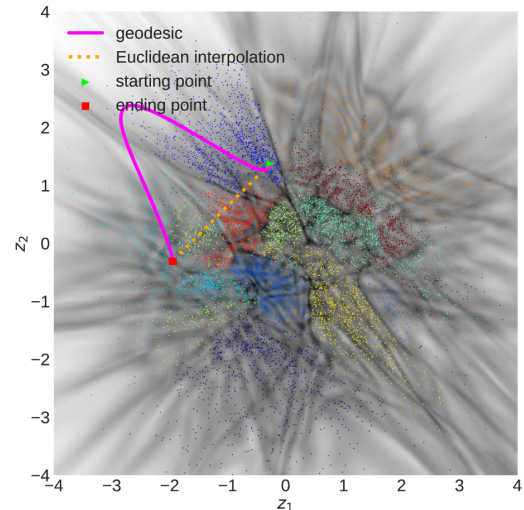

Figure 7: Geodesic and Euclidean interpolation in the latent space between two data points. The ten MNIST classes are encoded by colours, whereas the magnification factor is represented by the greyscale in the background. The *MF* and the samples are coloured the same as Fig. 6.

4.4   MNIST

To evaluate our model on a benchmark dataset, we used a fixed binarized version of the MNIST digit dataset defined by Larochelle and Murray [2011]. It consists of 50,000 training and 10,000 test images of handwritten digits (0 to 9) which are 28×28 pixels in size. The architecture of the IWAE is summarized in Table 3 of the Appendix.

Fig. 6 shows the trained two-dimensional latent space of the IWAE. Distances between the selected data point and any point on the equidistance line are equal in the observation space. The courses of the

equidistance lines demonstrate that treating the latent space as a Riemannian manifold enables to separate classes, since the geodesic between similar data points is shorter than between dissimilar ones. This is especially useful for state of the art methods that lead to very tight boundaries, like in this case—data points of different MNIST classes are almost not separable in the latent space by their Euclidean distance. Hence, the Euclidean distance cannot reflect the true similarity of two data points.

The difference between the geodesic and the Euclidean interpolation is shown in Fig. 7. The Euclidean interpolation crosses four classes, the geodesic just two. Compared to the geodesic, the Euclidean interpolation leads to less smooth transitions in the reconstructions (see Fig. 5, top and middle row). The transition between different classes is visualized by a higher velocity in this area (see Fig. 5, bottom row).

4.5 Robot arm

We simulated a movement of a KUKA robot that has six degrees of freedom (DOF). The end effector moved a circle with a 0.4 meter radius, generating a dataset with 6284 time steps. At each time step, the joint angles were obtained by inverse kinematics. The input data consisted of six-dimensional joint angles. Gaussian noise with a standard deviation of 0.03 was added to the data. The validation dataset also included a complete movement of a circle but only with 150 time steps. The architecture of the IWAE is shown in Table 4 of the Appendix.
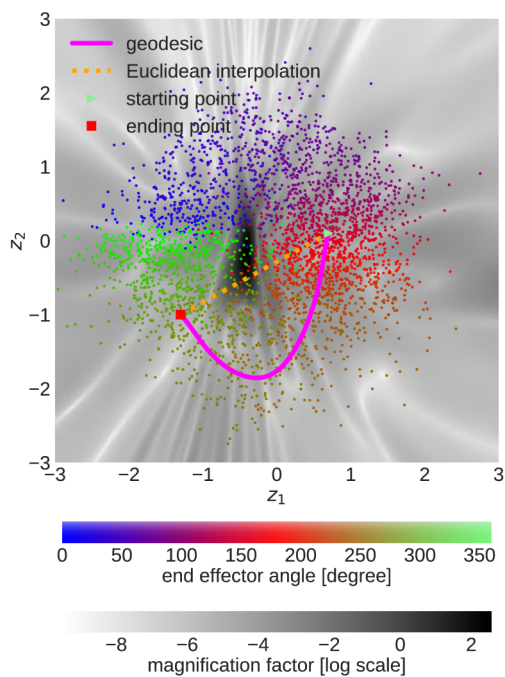

Figure 8: Geodesic and Euclidean interpolation in the latent space of the robot motions.

The geodesic interpolation outperforms the Euclidean interpolation for the robot arm movement, which is demonstrated in Fig. 8, 9 and 10. For intuitive observations, the results are shown in a two-dimensional end effector Cartesian space using forward kinematics (see Fig. 9).

To efficiently plan motions, in prior works [Berenson et al., 2009] constraints were created in the task space (e.g., constraint on the end-effector to move in a 2D instead of 3d). However, our method does not explicitly require these constraints.

The approach can be applied to movements with higher-dimensional joint angles like in case of the full-body humanoids demonstrated in Section 4.6.
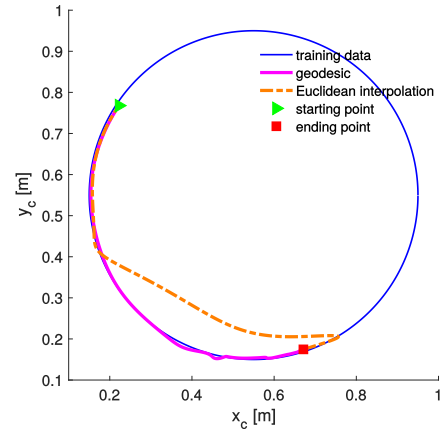

Figure 9: End effector trajectories in the Cartesian space. x_c and y_c represent the two axes of the end effector, while the third axis is not shown since the values on it are close to constant.
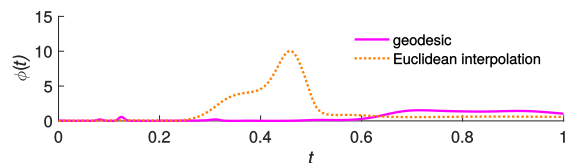

Figure 10: Velocity (Eq. (11)) of each robot sample. The distance of the Euclidean interpolation is 1.48, and the distance of the geodesic is 0.54.

4.6 Human motion

The CMU Graphics Lab Motion Capture Database consists of a large number of human motion recordings, which were recorded by using a Vicon motion capture system. Human subjects wear 41 markers while walking, dancing, etc. The data consists of 62-dimensional feature vectors, rendered using Vicon Bodybuilder. We pre-process the 62-dimensional data to 50-dimensional vectors as described in [Chen et al., 2015]. To evaluate the metric on this dataset, we used the walking movements (viz. trial 1 to 16) of subject 35, since it is very stable and widely used for algorithm evaluation, e.g. [Schölkopf et al., 2007, Bitzer et al., 2008, Chen et al., 2016]. The total of 6616 frames in the dataset were augmented with Gaussian noise with a standard deviation of 0.03, resulting in four times the size of the original dataset. The noises smooth the latent space which is observed through the magnification factor and the interpolation reconstructions. The architecture of the IWAE ca be found in Table 5 of the Appendix.

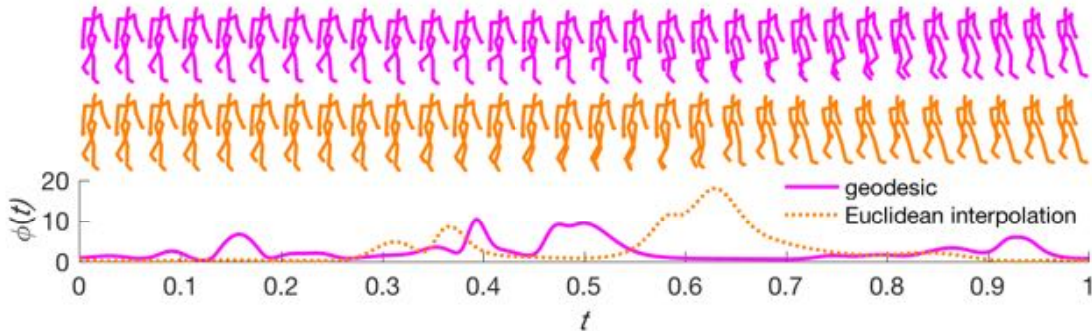Fig. 11 and 12 show the geodesic and the Euclidean

Figure 11: The reconstructions of the geodesic and Euclidean interpolation of the human motion. Top row: mean of the reconstruction from the geodesic. Middle row: mean of the reconstruction from the Euclidean interpolation. Bottom row: velocity (Eq. (11)) of each sample. The distance of the Euclidean interpolation is 2.89, and the distance of the geodesic is 2.57.
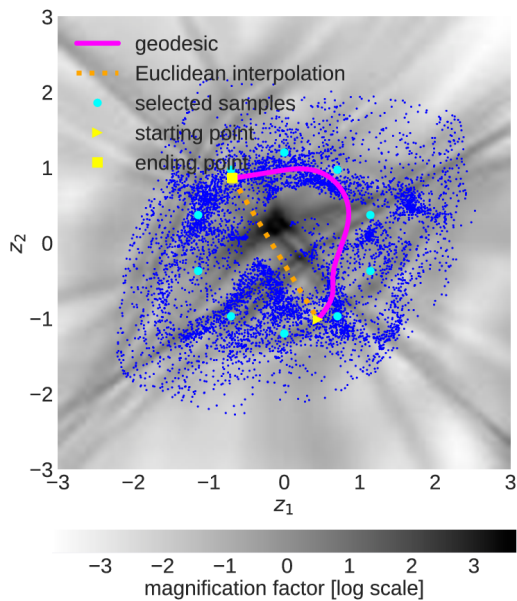

Figure 12: Geodesic and Euclidean interpolation in the latent space of the human motions. The blue dots are samples inferred from the training set.

interpolation[2]. The geodesic follows the path along the data manifold and generates a natural and smooth walking movement. In contrast, the Euclidean interpolation traverses two high MF areas which cause large jumps of the movement, while the body poses hardly change in other areas.

5 Conclusion and future work

The distance between points in the latent space in general does not reflect the true similarity of corresponding points in the observation space. We gave insight

---

[2]https://github.com/lawrennd/mocap is used to visualize the movement in the observation space.

Figure 13: Selected samples of the human motion dataset. The subfigures from left to right are the reconstructions of the cyan colored points in Fig. 12. From some center position $(0, 0)$, the starting point in the latent space is chosen with a radius of 1.5 in the vertical up direction. The direction of movement in the latent space is counterclockwise with a step length of 40 degree. One circle in the latent space corresponds to two steps of a walking movement in the observation space (until the body reaches the same pose as in the beginning).

into these issues using techniques from Riemannian geometry, applied to probabilistic latent variable models using neural sampling.

In our approach, the Riemannian distance metric has been successfully applied as an alternative metric that takes into account the underlying manifold. In order to produce shorter distances along the manifold compared to the Euclidean distance, we applied SVD to the metric tensor. As a secondary effect, the metric can be used for smoother interpolations in the latent space.

For two-dimensional latent spaces, the MF serves as a powerful tool to visualize the magnitude of the generative model's distortion of infinitesimal areas in the latent space.

Future work includes facilitating the use of this distance metric and applying it to models with dynamics such as [Chen et al., 2016] and [Karl et al., 2017].

References

C. C. Aggarwal, A. Hinneburg, and D. A. Keim. On the surprising behavior of distance metrics in high dimensional spaces. In *The International Conference on Database Theory (ICDT)*, volume 1, pages 420–434. Springer, 2001.

G. Arvanitidis, L. K. Hansen, and S. Hauberg. Latent space oddity: on the curvature of deep generative models. *arXiv*, abs/1710.11379, 2017.

J. Bayer, C. Osendorfer, and P. van der Smagt. Learning sequence neighbourhood metrics. In *Artificial Neural Networks and Machine Learning (ICANN)*, pages 531–538, 2012.

D. Berenson, S. S. Srinivasa, D. Ferguson, and J. J. Kuffner. Manipulation planning on constraint manifolds. In *International Conference on Robotics and Automation (ICRA)*, pages 625–632, 2009.

C. M. Bishop, M. Svensén, and C. K. Williams. Magnification factors for the SOM and GTM algorithms. In *Proceedings Workshop on Self-Organizing Maps*, 1997.

S. Bitzer, I. Havoutis, and S. Vijayakumar. Synthesising novel movements through latent space modulatoin of scalable control policies. In *International Conference on Simulation of Adaptive Behaviour (SAB)*, 2008.

Y. Burda, R. B. Grosse, and R. Salakhutdinov. Importance weighted autoencoders. *CoRR*, abs/1509.00519, 2015.

N. Chen, J. Bayer, S. Urban, and P. van der Smagt. Efficient movement representation by embedding dynamic movement primitives in deep autoencoders. In *International Conference on Humanoid Robots (HUMANOIDS)*, pages 434–440, 2015.

N. Chen, M. Karl, and P. van der Smagt. Dynamic movement primitives in latent space of time-dependent variational autoencoders. In *International Conference on Humanoid Robots (HU-MANOIDS)*, 2016.

C. Cremer, Q. Morris, and D. Duvenaud. Reinterpreting importance-weighted autoencoders. *International Conference on Learning Represenations Workshop Track*, 2017.

P. Crouch and F. S. Leite. The dynamic interpolation problem: on Riemannian manifolds, Lie groups, and symmetric spaces. *Journal of Dynamical and control systems*, 1(2):177–202, 1995.

J. V. Davis, B. Kulis, P. Jain, S. Sra, and I. S. Dhillon. Information-theoretic metric learning. In *Proceedings of the 24th international conference on Machine learning (ICML)*, pages 209–216, 2007.

P. T. Fletcher, C. Lu, S. M. Pizer, and S. Joshi. Principal geodesic analysis for the study of nonlinear statistics of shape. *IEEE transactions on medical imaging*, 23(8):995–1005, 2004.

M. C. Gemici, D. J. Rezende, and S. Mohamed. Normalizing flows on Riemannian manifolds. *CoRR*, abs/1611.02304, 2016.

J. Goldberger, S. T. Roweis, G. E. Hinton, and R. Salakhutdinov. Neighbourhood components analysis. In *Advances in Neural Information Processing Systems (NIPS)*, pages 513–520, 2004.

I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. C. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems (NIPS)*, pages 2672–2680, 2014.

I. Havoutis and S. Ramamoorthy. Motion generation with geodesic paths on learnt skill manifolds. *Modeling, Simulation and Optimization of Bipedal Walking*, 18:43, 2013.

M. Karl, M. Soelch, J. Bayer, and P. van der Smagt. Deep variational Bayes filters: Unsupervised learning of state space models from raw data. *International Conference on Learning Representations (ICLR)*, 2017.

D. P. Kingma and M. Welling. Auto-encoding variational Bayes. *CoRR*, abs/1312.6114, 2013.

B. Kulis et al. Metric learning: A survey. *Foundations and Trends® in Machine Learning*, 5(4):287–364, 2013.

H. Larochelle and I. Murray. The neural autoregressive distribution estimator. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 29–37, 2011.

L. Noakes, G. Heinzinger, and B. Paden. Cubic splines on curved spaces. *IMA Journal of Mathematical Control and Information*, 6(4):465–473, 1989.

D. J. Rezende, S. Mohamed, and D. Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *Proceedings of the 31th International Conference on Machine Learning (ICML)*, pages 1278–1286, 2014.

S. Rifai, Y. Dauphin, P. Vincent, Y. Bengio, and X. Muller. The manifold tangent classifier. In *Advances in Neural Information Processing Systems (NIPS)*, pages 2294–2302, 2011.

R. Salakhutdinov and G. E. Hinton. Learning a nonlinear embedding by preserving class neighbourhood structure. In *Proceedings of the Eleventh International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 412–419, 2007.

B. Schölkopf, J. Platt, and T. Hofmann. Modeling human motion using binary latent variables. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1345–1352, 2007.

H. Shao, A. Kumar, and P. T. Fletcher. The Riemannian geometry of deep generative models. *arXiv*, abs/1711.08014, 2017.

A. Tosi, S. Hauberg, A. Vellido, and N. D. Lawrence. Metrics for probabilistic geometries. *Proceedings of the Thirtieth Conference on Uncertainty in Artificial Intelligence*, pages 800–808, 2014.

K. Q. Weinberger and L. K. Saul. Distance metric learning for large margin nearest neighbor classification. *Journal of Machine Learning Research*, 10:207–244, 2009.

K. Q. Weinberger, J. Blitzer, and L. K. Saul. Distance metric learning for large margin nearest neighbor classification. In *Advances in neural information processing systems (NIPS)*, pages 1473–1480, 2006.

E. P. Xing, M. I. Jordan, S. J. Russell, and A. Y. Ng. Distance metric learning with application to clustering with side-information. In *Advances in neural information processing systems (NIPS)*, pages 521–528, 2003.