


# Mixed-curvature Variational Autoencoders

**Master Thesis**

**Author(s):**

Skopek, Ondrej 

**Publication date:**

2019

**Permanent link:**

<https://doi.org/10.3929/ethz-b-000372387>

**Rights / license:**

In Copyright - Non-Commercial Use Permitted



Eidgenössische Technische Hochschule Zürich  
Swiss Federal Institute of Technology Zurich

# Mixed-curvature Variational Autoencoders

Master Thesis

Ondrej Skopek

September 4, 2019

Advisors: Prof. Dr. Thomas Hofmann, Octavian-Eugen Ganea, Gary Bécigneul

Department of Computer Science, ETH Zürich

To my family and friends — for the never-ending support.

---

## Abstract

It has previously been shown that using geometric spaces with non-zero curvature (i.e. spherical, hyperbolic, or even mixtures of these) instead of plain Euclidean spaces with zero curvature improves performance on a wide range of Machine Learning tasks for learning representations in domains ranging from Natural Language Processing to Computer Vision.

Recent work has leveraged these geometries to learn latent variable models like Variational Autoencoders (VAEs) in spherical and hyperbolic spaces with constant curvature. While these approaches work well on particular kinds of data that they were designed for (e.g. tree-like data for a hyperbolic VAE), there exists no generic approach unifying all three models. We develop a Mixed-curvature Variational Autoencoder, an efficient way to train a VAE whose latent space is a product of constant curvature Riemannian manifolds, and whose per-component curvature can also be learned. This approach presents a generalization of the standard Euclidean VAE to curved latent spaces, as the model essentially reduces to the Euclidean VAE if the curvatures of all components of the latent space go to 0. We show that this approach is more general and surpasses all baselines on a range of different tasks.

---

## Acknowledgements

A big thank you goes to Yoshihiro Nagano and Emile Mathieu, for help with reproducing their work and releasing their code, which was a helpful reference point at multiple stages of this project.

Thank you to Andreas and Gregor, for help in deriving and verifying some of the formulas for constant curvature spaces; my advisors, Octavian and Gary, for the helpful discussions; and the Data Analytics Lab, the Leonhard cluster, and ETH Zürich for GPU access and coffee.

I also want to thank Lukáš, for all the advice, discussions, and GPU access; Vignesh, for all the advice and support; and last but not least, Pandy, for always being there for me.

---

# Contents

---

<b>Contents</b>	<b>iv</b>
<b>1 Introduction</b>	<b>1</b>
<b>2 Geometry</b>	<b>4</b>
2.1 A brief introduction to Riemannian geometry . . . . .	4
2.2 Constant curvature spaces . . . . .	5
2.2.1 Euclidean space . . . . .	6
2.2.2 Hypersphere . . . . .	6
2.2.3 Hyperboloid . . . . .	7
2.3 Stereographically projected spaces . . . . .	8
2.3.1 Projected hypersphere . . . . .	10
2.3.2 Poincaré ball . . . . .	11
2.3.3 Gyrovector spaces . . . . .	11
2.4 Duality between constant curvature spaces . . . . .	13
2.5 Brief comparison of constant curvature space models . . . . .	13
2.6 Products of spaces . . . . .	15
<b>3 Probability</b>	<b>18</b>
3.1 Multivariate Normal distribution . . . . .	18
3.2 Normal-like distributions in non-Euclidean constant curvature spaces . . . . .	20
3.3 Von Mises-Fisher distribution . . . . .	21
3.4 Wrapped Normal distributions . . . . .	22
<b>4 Variational Autoencoders</b>	<b>25</b>
4.1 Autoencoders . . . . .	25
4.2 Variational Inference . . . . .	25
4.2.1 Tighter bounds on the marginal log-likelihood . . . . .	27
4.3 Variational Autoencoders . . . . .	27

4.3.1	Learning VAEs . . . . .	28
4.3.2	Riemannian manifolds as latent spaces . . . . .	29
4.3.3	Latent space as a product of constant curvature spaces . . . . .	31
4.3.4	Overview of properties . . . . .	31
<b>5</b>	<b>Learning curvature</b>	<b>32</b>
5.1	Fixed curvature VAEs . . . . .	32
5.2	Learnable curvature VAEs . . . . .	33
5.3	Universal curvature VAEs . . . . .	33
<b>6</b>	<b>Experiments</b>	<b>36</b>
6.1	Related work . . . . .	36
6.1.1	Universal models of geometry . . . . .	36
6.1.2	Concurrent VAE approaches . . . . .	37
6.1.3	Geometric deep learning . . . . .	37
6.1.4	Geometry in VAEs . . . . .	38
6.2	Experimental setup . . . . .	38
6.3	Spherical covariance matrix parametrization . . . . .	41
6.4	Diagonal covariance matrix parametrization . . . . .	43
6.4.1	Dynamically-binarized MNIST reconstruction . . . . .	43
6.4.2	Summary of experimental evaluation . . . . .	47
6.5	Future work . . . . .	49
<b>7</b>	<b>Conclusion</b>	<b>51</b>
	<b>Bibliography</b>	<b>52</b>
	<b>Notation</b>	<b>59</b>
	<b>List of Theorems</b>	<b>62</b>
	<b>List of Figures</b>	<b>64</b>
	<b>List of Tables</b>	<b>66</b>
<b>A</b>	<b>Geometrical details</b>	<b>68</b>
A.1	Euclidean geometry . . . . .	68
A.1.1	Euclidean space . . . . .	68
A.2	Hyperbolic geometry . . . . .	69
A.2.1	Hyperboloid . . . . .	69
A.2.2	Poincaré ball . . . . .	76
A.3	Spherical geometry . . . . .	81
A.3.1	Hypersphere . . . . .	81
A.3.2	Projected hypersphere . . . . .	88
A.4	Miscellaneous properties . . . . .	94

A.5 Angles in constant curvature spaces . . . . .	98
<b>B Probability details</b>	<b>100</b>
B.1 Hyperspherical uniform distribution . . . . .	100
B.2 Von Mises-Fisher distribution . . . . .	101
B.3 Wrapped Normal distributions . . . . .	101
<b>C Variational Autoencoders</b>	<b>112</b>
C.1 Why use Variational Autoencoders? . . . . .	112
<b>D Extended results</b>	<b>114</b>
D.1 Implementation remarks . . . . .	114
D.2 Spherical covariance matrix . . . . .	116
D.3 Diagonal covariance matrix . . . . .	120
D.3.1 Dynamically binarized MNIST reconstruction . . . . .	120
D.3.2 Dynamically binarized Omniglot reconstruction . . . . .	128
D.3.3 CIFAR reconstruction . . . . .	133



## Chapter 1

---

# Introduction

---

Generative models present an ever-growing area of Machine Learning, where we aim to model the data distribution  $p(\mathbf{x})$  over data points  $\mathbf{x}$  belonging to some most commonly high-dimensional space  $\mathcal{X}$  (Doersch, 2016). As is common with most Machine Learning models,  $\mathcal{X}$  is usually a subset of a high-dimensional Euclidean vector space  $\mathbb{R}^n$ , with all the associated benefits: a naturally definable scalar product, vector addition, and others. Yet, many types of data have a strongly non-Euclidean latent structure (Bronstein et al., 2017). A notorious example is the set of human-interpretable images – they are usually assumed to live on a “natural image manifold” (Zhu et al., 2016), i.e. a “lower-dimensional subset” of some high-dimensional space in which they are represented. On this continuous manifold, one finds only (and all) the images that humans can interpret using their visual system. This would mean that by moving along the manifold, we could continuously change the content and appearance of images. To illustrate, for MNIST handwritten digits (LeCun, 1998) the assumption conjectures that there exists a lower-dimensional manifold  $\mathcal{M} \subseteq \mathbb{R}^{28 \times 28}$ , which is the manifold of all possible MNIST digit images.

As mentioned in Nickel and Kiela (2017), changing the geometry of the underlying latent space enables us to represent some data better than is possible in Euclidean space. For example, a binary tree has  $2^l$  nodes at level  $l$  (counting from level 0) and  $2^{l+1} - 1$  nodes on levels less or equal to  $l$ . We can see that the number of children grows exponentially with the distance from the root node, where the distance between two nodes is the number of edges on the shortest path between them. In a hyperbolic space with just 2 dimensions, we can construct a representation of the tree that preserves the tree distance by placing points at level  $l$  onto a sphere whose radius is proportional to  $l$ . All the points in levels less than  $l$  will be inside the sphere, the ones on levels greater than  $l$  will be outside of the sphere. This is not possible in Euclidean space, because the area of a sphere grows only quadratically with respect to

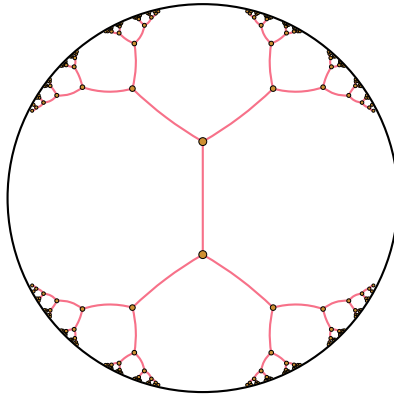


Figure 1.1: Binary tree embedded in a Poincaré ball (Mathieu et al., 2019).

its radius, as opposed to the number of children growing exponentially. For more details, see Kleinberg (2007) and Sarkar (2012), and for an illustration see Figure 1.1. Similarly to how hyperbolic spaces can be thought of as “continuous trees” (Nickel and Kiela, 2017), spherical spaces could be thought of as “continuous cycles”.

Motivated by these observations, a range of methods to learn representations in different spaces of constant curvature have recently been introduced: learning embeddings in spherical spaces (Batmanghelich et al., 2016) (positive constant curvature), hyperbolic spaces (Nickel and Kiela, 2017; Sala et al., 2018; Tifrea et al., 2019) (negative constant curvature), and even in products of spaces with constant curvature (Gu et al., 2019). The last of these approaches aims to match the underlying geometry of the data even closer than the others, by using a combination of different constant curvature spaces. How to choose the dimensionality of partial spaces and their curvatures remains an open question.

One of the most popular approaches to generative modeling recently is Variational Inference, and specifically, the Variational Autoencoder (Kingma and Welling, 2014, VAE). It provides us with a way to sidestep the intractability of marginalizing a joint probability model of the input and latent space  $p(\mathbf{x}, \mathbf{z})$  while allowing for a chosen prior probability  $p(\mathbf{z})$  on the latent space, usually a Normal distribution  $\mathcal{N}(\mathbf{0}, \mathbf{I})$ . Recently, variants of the VAE have been introduced for spherical (Davidson et al., 2018; Xu and Durrett, 2018) and hyperbolic (Mathieu et al., 2019; Nagano et al., 2019) latent spaces.

Our approach is a generalization of the different VAE variants to mixed-curvature latent spaces — more precisely, products of constant curvature spaces, similar to Gu et al. (2019). Modeling the latent space of a VAE only as a single constant curvature manifold limits the flexibility of the latent

---

space to assume a shape similar to that of the hypothetical intrinsic manifold. Therefore, we aim to learn representations in products of spaces of constant curvature, which has the advantage that we can obtain a better reduction in dimensionality while not making optimization of the model significantly more complex. The resulting latent space is then a “non-constantly” curved manifold in an ambient Euclidean space.

Our main contributions are the following:

1. We develop a framework for manipulating representations and modeling probability distributions on non-fixed constant curvature spaces<sup>1</sup>. Previously, only Mathieu et al. (2019) tried changing curvature on a per-experiment basis (as a hyperparameter).
2. We show how to generalize Variational Autoencoders to learn latent representations on products of constant curvature spaces, including a procedure to learn the structure of the product of latent spaces itself.
3. On benchmark datasets, we show that this approach is applicable to the tasks of structure reconstruction on a synthetic tree dataset (Mathieu et al., 2019) and image reconstruction on MNIST (LeCun, 1998), Omniglot (Lake et al., 2015), and CIFAR (Krizhevsky, 2009).

In Chapter 2 we discuss the necessary geometrical background, with more details in Appendix A. Chapter 3 contains definitions of probability distributions in products of constant curvatures spaces, with more details available in Appendix B. Chapter 4 briefly introduces Variational Autencoders, and formulates them for products of constant curvature spaces. Using the model formulations, Chapter 5 contains the motivation for and approaches to learning curvature in these models. A detailed description of the experiments can be found in Chapter 6, where we also elaborate on the interpretation of our results, go over related work, and propose more future work. Additional plots and experimental results can be found in Appendix D.

---

<sup>1</sup>The curvature is *constant* at all points in the space, but the value of the curvature itself is not constant (*fixed*) during model training.

## Chapter 2

---

# Geometry

---

In this chapter, we take a closer look at a few models of spaces of constant curvature, their characteristics, and common operations that will enable us to work with representations in these spaces. Numerous details, properties, and proofs are appended in Appendix A.

### 2.1 A brief introduction to Riemannian geometry

We briefly introduce the necessary differential geometry concepts, similarly to Mathieu et al. (2019). For more details, please refer to Petersen et al. (2006) or Cannon et al. (1997).

An elementary notion in Riemannian geometry is that of a real, smooth *manifold*  $\mathcal{M} \subseteq \mathbb{R}^n$ , which is a collection of real vectors  $\mathbf{x}$  that is locally similar to a linear space, and lives in the *ambient space*  $\mathbb{R}^n$ . At each point of the manifold  $\mathbf{x} \in \mathcal{M}$  a real vector space of the same dimensionality as  $\mathcal{M}$  is defined, called the *tangent space at point  $\mathbf{x}$* :  $\mathcal{T}_{\mathbf{x}}\mathcal{M}$ . Intuitively, the tangent space contains all the directions and speeds at which one can pass through  $\mathbf{x}$ . Given a matrix representation  $G(\mathbf{x}) \in \mathbb{R}^{n \times n}$  of the *Riemannian metric tensor*  $\mathfrak{g}(\mathbf{x})$ , we can define a *scalar product on the tangent space*:  $\langle \cdot, \cdot \rangle_{\mathbf{x}} : \mathcal{T}_{\mathbf{x}}\mathcal{M} \times \mathcal{T}_{\mathbf{x}}\mathcal{M} \rightarrow \mathcal{M}$ , where  $\langle \mathbf{a}, \mathbf{b} \rangle_{\mathbf{x}} = \mathfrak{g}(\mathbf{x})(\mathbf{a}, \mathbf{b}) = \mathbf{a}^T G(\mathbf{x}) \mathbf{b}$  for any  $\mathbf{a}, \mathbf{b} \in \mathcal{T}_{\mathbf{x}}\mathcal{M}$ . A *Riemannian manifold* is then the tuple  $(\mathcal{M}, \mathfrak{g})$ . The scalar product induces a norm on the tangent space  $\mathcal{T}_{\mathbf{x}}\mathcal{M}$ :  $\|\mathbf{a}\|_{\mathbf{x}} = \sqrt{\langle \mathbf{a}, \mathbf{a} \rangle_{\mathbf{x}}} \forall \mathbf{a} \in \mathcal{T}_{\mathbf{x}}\mathcal{M}$  (Petersen et al., 2006).

Although it seems like the manifold only defines a local geometry, it induces global quantities by integrating the local contributions. The metric tensor induces a local infinitesimal volume element on each tangent space  $\mathcal{T}_{\mathbf{x}}\mathcal{M}$  and hence a measure is induced as well  $d\mathcal{M}(\mathbf{x}) = \sqrt{|G(\mathbf{x})|} d\mathbf{x}$  where  $d\mathbf{x}$  is the Lebesgue measure. The *length* of a curve  $\gamma : t \mapsto \gamma(t) \in \mathcal{M}$ ,  $t \in [0, 1]$  is given by  $L(\gamma) = \int_0^1 \sqrt{\left\| \frac{d}{dt} \gamma(t) \right\|_{\gamma(t)}} dt$ .

Straight lines are generalized to constant speed curves giving the shortest path between pairs of points  $\mathbf{x}, \mathbf{y} \in \mathcal{M}$ , so called *geodesics*, for which it holds that  $\gamma^* = \arg \min_{\gamma} L(\gamma)$ , such that  $\gamma(0) = \mathbf{x}$ ,  $\gamma(1) = \mathbf{y}$ , and  $\left\| \frac{d}{dt} \gamma(t) \right\|_{\gamma(t)} = 1$ . Global distances are thus induced on  $\mathcal{M}$  by  $d_{\mathcal{M}}(\mathbf{x}, \mathbf{y}) = \inf_{\gamma} L(\gamma)$ .

Using this metric, we can go on to define a metric space  $(\mathcal{M}, d_{\mathcal{M}})$ . Moving from a point  $\mathbf{x} \in \mathcal{M}$  in a given direction  $\mathbf{v} \in \mathcal{T}_{\mathbf{x}}\mathcal{M}$  with constant velocity is formalized by the *exponential map*:  $\exp_{\mathbf{x}} : \mathcal{T}_{\mathbf{x}}\mathcal{M} \rightarrow \mathcal{M}$ . There exists a unique unit speed geodesic  $\gamma$  such that  $\gamma(0) = \mathbf{x}$  and  $\left. \frac{d\gamma(t)}{dt} \right|_{t=0} = \mathbf{v}$ , where  $\mathbf{v} \in \mathcal{T}_{\mathbf{x}}\mathcal{M}$ . The corresponding exponential map is then defined as  $\exp_{\mathbf{x}}(\mathbf{v}) = \gamma(1)$ . The logarithmic map is the inverse  $\log_{\mathbf{x}} = \exp_{\mathbf{x}}^{-1} : \mathcal{M} \rightarrow \mathcal{T}_{\mathbf{x}}\mathcal{M}$ . For geodesically complete manifolds, i.e. manifolds in which there exists a length-minimizing geodesic between every  $\mathbf{x}, \mathbf{y} \in \mathcal{M}$ , such as the Lorentz model, hypersphere, and many others,  $\exp_{\mathbf{x}}$  is well-defined on the full tangent space  $\mathcal{T}_{\mathbf{x}}\mathcal{M}$  (Figure 3.1c).

To connect vectors in tangent spaces, we use the notion of parallel transport  $\text{PT}_{\mathbf{x} \rightarrow \mathbf{y}} : \mathcal{T}_{\mathbf{x}}\mathcal{M} \rightarrow \mathcal{T}_{\mathbf{y}}\mathcal{M}$ , which is an isomorphism between the two tangent spaces, so that the transported vectors stay parallel to the connection (Figure 3.1b). It corresponds to moving tangent vectors along geodesics and defines a canonical way to connect tangent spaces.

To be able to define constantly curved spaces, we first need to define the notion of (Gaussian) *curvature* (Berger, 2012) at a point  $\mathbf{x} \in \mathcal{M}$ , denoted  $K(\mathbf{x})$ . Gaussian curvature is the product of all principal curvatures at that point. The two *principal curvatures* at  $\mathbf{x}$  are defined as the minimum and maximum curvatures of the plane curves traversing the given point  $\mathbf{x}$ . More formally, a *plane curve* is any curve  $\gamma : [0, 1] \rightarrow \mathcal{M}$ . Traversing  $\mathbf{x}$  means there exists a unique  $t \in (0, 1)$  for which  $\gamma(t) = \mathbf{x}$ . Since we deal with constant curvature spaces, we simply denote curvature as  $K$  for the whole space from now on.

## 2.2 Constant curvature spaces

Spaces/manifolds of constant curvature have the same curvature at every point in the space/manifold. We can notice that there are three fundamentally different types of manifold  $\mathcal{M}$  we can define with respect to the sign of the curvature: a positively curved space, a “flat” space, and a negatively curved space. The most common realizations of those manifolds are the hypersphere  $\mathbb{S}_K$ , the Euclidean space  $\mathbb{E}$ , and the hyperboloid  $\mathbb{H}_K$ :

$$\mathcal{M} = \begin{cases} \mathbb{S}_K^n = \{\mathbf{x} \in \mathbb{R}^{n+1} : \langle \mathbf{x}, \mathbf{x} \rangle_2 = 1/K\}, & \text{for } K > 0 \\ \mathbb{E}^n = \mathbb{R}^n, & \text{for } K = 0 \\ \mathbb{H}_K^n = \{\mathbf{x} \in \mathbb{R}^{n+1} : \langle \mathbf{x}, \mathbf{x} \rangle_{\mathcal{L}} = 1/K\}, & \text{for } K < 0 \end{cases}$$

where  $\langle \cdot, \cdot \rangle_2$  is the standard Euclidean inner product, and  $\langle \cdot, \cdot \rangle_{\mathcal{L}}$  is the Lorentz inner product,

$$\langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}} = -x_1 y_1 + \sum_{i=2}^{n+1} x_i y_i \quad \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^{n+1}.$$

Notice that  $\mathbb{E}^n$  could be represented directly in  $\mathbb{R}^n$ , whereas both  $\mathbb{S}_K^n$  and  $\mathbb{H}_K^n$  need to be represented using more dimension in the ambient space  $\mathbb{R}^{n+1}$ . To simplify the notation, we sometimes use  $\mathbb{S} = \mathbb{S}_1$  and  $\mathbb{H} = \mathbb{H}_{-1}$ . Instead of curvature  $K$ , we often use the generalized notion of a *radius*:

$$R = \frac{1}{\sqrt{|K|}}.$$

An illustrative example of a hypersphere and a hyperboloid can be found in Figure 2.2.

### 2.2.1 Euclidean space

Firstly, let us consider the  $K = 0$  case. We formally define the  $n$ -dimensional Euclidean manifold (with curvature  $K = 0$ , omitted from notation) as the set  $\mathbb{E}^n = \mathbb{R}^n$ . Along with the Euclidean distance  $d_{\mathbb{E}} = \|\mathbf{x} - \mathbf{y}\|_2$ , they form the  $n$ -dimensional Euclidean space  $(\mathbb{E}^n, d_{\mathbb{E}})$ . This space is identical to the standard Euclidean vector space  $\mathbb{R}^n$ .

We will need the following operations in the space to work with our latent representations later. Their properties and statements about correctness can be found in Section A.1.1 in Appendix A. The exponential map in Euclidean space is defined as

$$\exp_{\mathbf{x}}(\mathbf{v}) = \mathbf{x} + \mathbf{v},$$

for all  $\mathbf{x} \in \mathbb{E}^n$  and  $\mathbf{v} \in \mathcal{T}_{\mathbf{x}}\mathbb{E}^n$ . Its inverse, the logarithmic map is

$$\log_{\mathbf{x}}(\mathbf{y}) = \mathbf{y} - \mathbf{x},$$

for all  $\mathbf{x}, \mathbf{y} \in \mathbb{E}^n$ . Parallel transport in Euclidean space is simply an identity function

$$\text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}(\mathbf{v}) = \mathbf{v},$$

for all  $\mathbf{x}, \mathbf{y} \in \mathbb{E}^n$  and  $\mathbf{v} \in \mathcal{T}_{\mathbf{x}}\mathbb{E}^n$ .

### 2.2.2 Hypersphere

A hypersphere, or an  $n$ -dimensional sphere with positive curvature  $K > 0$ , is defined as the set

$$\mathbb{S}_K^n = \left\{ \mathbf{x} \in \mathbb{R}^{n+1} : \langle \mathbf{x}, \mathbf{x} \rangle_2 = R^2 = \frac{1}{K} \right\}.$$

As the curvature  $K$  increases, the radius  $R$  of the sphere decreases. Even though the definition of the sphere uses a standard Euclidean dot product  $\langle \cdot, \cdot \rangle_2$ , the distance function induced by the metric tensor is different:

$$d_{\mathbb{S}}(\mathbf{x}, \mathbf{y}) = R \cos^{-1} \left( \frac{\langle \mathbf{x}, \mathbf{y} \rangle_2}{R^2} \right) = \frac{1}{\sqrt{K}} \cos^{-1} (K \langle \mathbf{x}, \mathbf{y} \rangle_2).$$

Formally, the  $n$ -dimensional hypersphere space is a sphere with the  $d_{\mathbb{S}}$  metric ( $\mathbb{S}_K^n, d_{\mathbb{S}}$ ).

The operations we need can also be defined in the hypersphere. The exponential map is defined as

$$\exp_{\mathbf{x}}^K(\mathbf{v}) = \cos \left( \sqrt{K} \|\mathbf{v}\|_2 \right) \mathbf{x} + \sin \left( \sqrt{K} \|\mathbf{v}\|_2 \right) \frac{\mathbf{v}}{\sqrt{K} \|\mathbf{v}\|_2},$$

for all  $\mathbf{x} \in \mathbb{S}_K^n$  and  $\mathbf{v} \in \mathcal{T}_{\mathbf{x}}\mathbb{S}_K^n$ . Its inverse, the logarithmic map is

$$\log_{\mathbf{x}}^K(\mathbf{y}) = \frac{\cos^{-1}(\alpha)}{\sqrt{1 - \alpha^2}} (\mathbf{y} - \alpha \mathbf{x}),$$

where  $\alpha = K \langle \mathbf{x}, \mathbf{y} \rangle_2$ , for all  $\mathbf{x}, \mathbf{y} \in \mathbb{S}_K^n$ . Parallel transport in the hypersphere is

$$\text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}) = \mathbf{v} - \frac{K \langle \mathbf{y}, \mathbf{v} \rangle_2}{1 + K \langle \mathbf{x}, \mathbf{y} \rangle_2} (\mathbf{x} + \mathbf{y}),$$

for all  $\mathbf{x}, \mathbf{y} \in \mathbb{S}_K^n$  and  $\mathbf{v} \in \mathcal{T}_{\mathbf{x}}\mathbb{S}_K^n$ . More details and properties can be found in Section A.3.1 in Appendix A.

### 2.2.3 Hyperboloid

The hyperboloid  $\mathbb{H}_K^n$  (also called the Lorentz model) for a given curvature  $K < 0$  is defined as

$$\mathbb{H}_K^n = \{ \mathbf{x} \in \mathbb{R}^{n+1} : \langle \mathbf{x}, \mathbf{x} \rangle_{\mathcal{L}} = -R^2 = \frac{1}{K}, x_1 > 0 \},$$

where  $\langle \cdot, \cdot \rangle_{\mathcal{L}}$  is the Lorentzian inner product (or Minkowski inner product) as defined previously.

We can point out that, similarly to the spherical and Euclidean spaces, for all  $\mathbf{x}, \mathbf{y} \in \mathbb{H}_K^n$  it holds that

$$\langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}} = -R^2 \iff \mathbf{x} = \mathbf{y}.$$

Otherwise,  $\langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}} < -R^2$ . This corresponds to the hyperbolic Cauchy-Schwarz theorem (Ratcliffe, 2006, Theorem 3.1.6)

The induced distance function in the hyperboloid is

$$d_{\mathbb{H}}(\mathbf{x}, \mathbf{y}) = R \cosh^{-1} \left( -\frac{\langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}}}{R^2} \right) = \frac{1}{\sqrt{-K}} \cosh^{-1} (-K \langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}}).$$

Formally, the  $n$ -dimensional hyperboloid space is a hyperboloid with the  $d_{\mathbb{H}}$  metric ( $\mathbb{H}_K^n, d_{\mathbb{H}}$ ).

Likewise, the necessary operations can also be defined in the hyperboloid, and are dual to their hyperspherical equivalents. The exponential map is defined as

$$\exp_{\mathbf{x}}^K(\mathbf{v}) = \cosh\left(\sqrt{-K}\|\mathbf{v}\|_{\mathcal{L}}\right)\mathbf{x} + \sinh\left(\sqrt{-K}\|\mathbf{v}\|_{\mathcal{L}}\right)\frac{\mathbf{v}}{\sqrt{-K}\|\mathbf{v}\|_{\mathcal{L}}},$$

for all  $\mathbf{x} \in \mathbb{H}_K^n$  and  $\mathbf{v} \in \mathcal{T}_{\mathbf{x}}\mathbb{H}_K^n$ . Its inverse, the logarithmic map is

$$\log_{\mathbf{x}}^K(\mathbf{y}) = \frac{\cosh^{-1}(\alpha)}{\sqrt{\alpha^2 - 1}}(\mathbf{y} - \alpha\mathbf{x}),$$

where  $\alpha = K\langle\mathbf{x}, \mathbf{y}\rangle_{\mathcal{L}}$ , for all  $\mathbf{x}, \mathbf{y} \in \mathbb{H}_K^n$ . Parallel transport in the hyperboloid is

$$\text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}) = \mathbf{v} - \frac{K\langle\mathbf{y}, \mathbf{v}\rangle_{\mathcal{L}}}{1 + K\langle\mathbf{x}, \mathbf{y}\rangle_{\mathcal{L}}}(\mathbf{x} + \mathbf{y}),$$

for all  $\mathbf{x}, \mathbf{y} \in \mathbb{H}_K^n$  and  $\mathbf{v} \in \mathcal{T}_{\mathbf{x}}\mathbb{H}_K^n$ . More details and properties can be found in Section A.3.1 in Appendix A. For a summary of operations on the hyperboloid and hypersphere, see Table 2.1.

## 2.3 Stereographically projected spaces

At first sight, the above spaces are enough to cover any possible value of the curvature, and they define all the necessary operations we will need to train VAEs in them. Unfortunately, both the hypersphere (Remark A.19) and the hyperboloid (Remark A.3) have an unsuitable property, namely the non-convergence of the norm of points as the curvature goes to 0. The intuition is that both spaces grow as  $K \rightarrow 0$  and become locally “flatter”, but to do that, their points have to go away from the origin of the coordinate space  $\mathbf{0}$  to be able to satisfy their definitions. A good example of a point that diverges is the origin of the hyperboloid (equivalently, a pole of the hypersphere)  $\boldsymbol{\mu}_0^K = (1/K, 0, \dots, 0)^T = (R, 0, \dots, 0)^T$ . In general, we can easily see that  $\|\mathbf{x}\|^2 = \frac{1}{K} \xrightarrow{K \rightarrow 0} \pm\infty$ . That makes both of these spaces unsuitable for trying to learn sign-agnostic curvatures.

Luckily, there exist well-defined positively and negatively curved spaces that inherit most properties from the hyperboloid and the hypersphere, yet do not have this property — namely, the Poincaré ball and the projected sphere, respectively. We can obtain both of them using stereographic conformal projections of the hyperboloid and the hypersphere, meaning that angles are preserved by the projection. Since the distance function on the hyperboloid and hypersphere only depend on the radius and angles between points, they are isometric.



To obtain the models defined below, we first need to define the projection function. For  $(\xi; \mathbf{x}^T)^T \in \mathbb{R}^{n+1}$  where  $\xi \in \mathbb{R}$ ,  $\mathbf{x} \in \mathbb{R}^n$ , curvature  $K \in \mathbb{R}$  and the corresponding radius  $R = \frac{1}{\sqrt{|K|}}$

$$\begin{aligned}\rho_K((\xi; \mathbf{x}^T)^T) &= \frac{R\mathbf{x}}{R + \xi} = \frac{\mathbf{x}}{1 + \sqrt{|K|}\xi} \\ \rho_{K>0}^{-1}(\mathbf{y}) &= \left( R \frac{R^2 - \|\mathbf{y}\|_2^2}{R^2 + \|\mathbf{y}\|_2^2}, \frac{2R^2\mathbf{y}^T}{R^2 + \|\mathbf{y}\|_2^2} \right)^T \\ \rho_{K<0}^{-1}(\mathbf{y}) &= \left( R \frac{R^2 + \|\mathbf{y}\|_2^2}{R^2 - \|\mathbf{y}\|_2^2}, \frac{2R^2\mathbf{y}^T}{R^2 - \|\mathbf{y}\|_2^2} \right)^T \\ \rho_K^{-1}(\mathbf{y}) &= \left( \frac{1}{\sqrt{|K|}} \frac{1 - K\|\mathbf{y}\|_2^2}{1 + K\|\mathbf{y}\|_2^2}, \frac{2\mathbf{y}^T}{1 + K\|\mathbf{y}\|_2^2} \right)^T,\end{aligned}$$

where  $\mathbf{y} \in \mathbb{R}^n$ . The last formula is one that generalizes the two above for any non-zero values of  $K$ . For more details, see Sections A.3.2, A.2.2, and Theorem A.38 in Appendix A. These formulas correspond to the classical stereographic projections defined for these models (Lee, 1997, Formula 3.9). Note that both of these projections map the point  $\boldsymbol{\mu}_0 = (R, 0, \dots, 0)$  in the original space to  $\boldsymbol{\mu}_0 = \mathbf{0}$  in the projected space, and back.

Since the stereographic projection is conformal, the metric tensors of both spaces will be conformal. In this case, the metric tensors of both spaces are the same, except for the sign of  $K$

$$\mathfrak{g}_x^{\mathbb{D}^K} = \mathfrak{g}_x^{\mathbb{P}^K} = (\lambda_x^K)^2 \mathfrak{g}_x^{\mathbb{E}},$$

for all  $\mathbf{x}$  in the respective manifold (Ganea et al., 2018a, Section 2.1), and  $\mathfrak{g}_y^{\mathbb{E}} = \mathbf{I}$  for all  $\mathbf{y} \in \mathbb{E}$ . The conformal factor  $\lambda_x^K$  is defined as

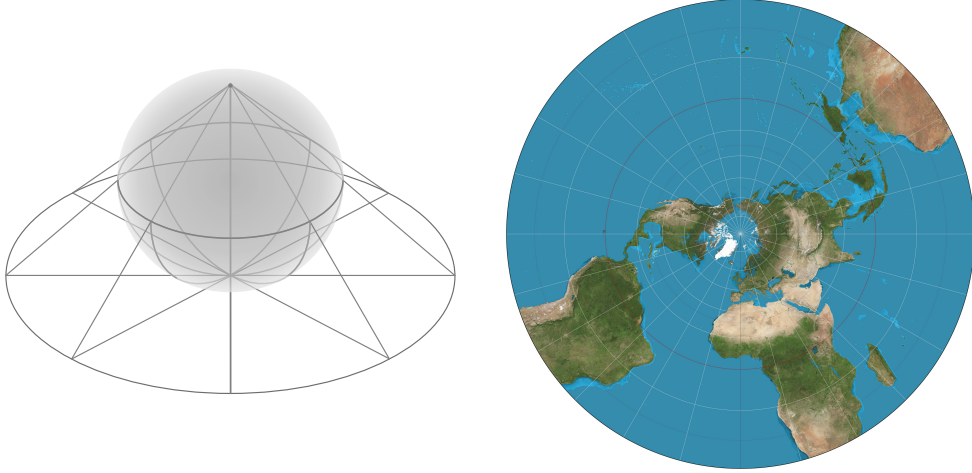
$$\lambda_x^K = \frac{2}{1 + K\|\mathbf{x}\|_2^2}.$$

Among other things, this form of the metric tensor has the consequence that we unfortunately cannot define a single unified inner product in all tangent spaces at all points. The inner product at  $\mathbf{x} \in \mathcal{M}$  has the form of

$$\langle \mathbf{u}, \mathbf{v} \rangle_x = (\lambda_x^K)^2 \langle \mathbf{u}, \mathbf{v} \rangle_2,$$

for all  $\mathbf{u}, \mathbf{v} \in \mathcal{T}_x\mathcal{M}$ .

We can now define the two models corresponding to  $K > 0$  and  $K < 0$ .



(a) Illustration of a stereographic projection from  $\mathbb{S}^2 \rightarrow \mathbb{D}^2$  (Wikimedia, 2017). (b) A stereographic projection from Earth's South pole (Wikimedia, 2012).

Figure 2.1: Illustrative visualizations of the stereographic projection  $\rho_K$ .

### 2.3.1 Projected hypersphere

An  $n$ -dimensional projected hypersphere with curvature  $K > 0$  is defined as the set

$$\mathbb{D}_K^n = \rho_K(\mathbb{S}_K^n \setminus \{-\boldsymbol{\mu}_0\}) = \mathbb{R}^n,$$

where  $\boldsymbol{\mu}_0 = (R, 0, \dots, 0)^T \in \mathbb{S}_K^n$ . For an illustration, see Figure 2.1a. The curvature of  $\mathbb{D}_K$  is identical to that of  $\mathbb{S}_K$ .

It is important to note that any point in  $\mathbb{R}^n$  can be interpreted as a point on the sphere without a ‘‘South pole’’  $\mathbb{S}_K^n \setminus \{-\boldsymbol{\mu}_0\}$  using  $\rho_K^{-1}$  and any point in  $\mathbb{R}^{n+1}$  (except points for which  $x_1 = R$ ) can be mapped to  $\mathbb{R}^n$  using  $\rho_K$ . To be able to backproject points into  $\mathbb{S}_K$  we need to, additionally, know the curvature  $K$ .

The distance function induced by the metric tensor is

$$\begin{aligned} d_{\mathbb{D}}(\mathbf{x}, \mathbf{y}) &= d_{\mathbb{S}}(\rho_K^{-1}(\mathbf{x}), \rho_K^{-1}(\mathbf{y})) \\ &= R \cos^{-1} \left( 1 - \frac{2R^2 \|\mathbf{x} - \mathbf{y}\|_2^2}{(R^2 + \|\mathbf{x}\|_2^2)(R^2 + \|\mathbf{y}\|_2^2)} \right) \\ &= \frac{1}{\sqrt{K}} \cos^{-1} \left( 1 - \frac{2K \|\mathbf{x} - \mathbf{y}\|_2^2}{(1 + K \|\mathbf{x}\|_2^2)(1 + K \|\mathbf{y}\|_2^2)} \right). \end{aligned}$$

Formally, the  $n$ -dimensional projected hypersphere space is a projected hypersphere with the  $d_{\mathbb{D}}$  metric  $(\mathbb{D}_K^n, d_{\mathbb{D}})$ .

### 2.3.2 Poincaré ball

The  $n$ -dimensional Poincaré ball  $\mathbb{P}_K^n$  (also called the Poincaré disk when  $n = 2$ ) for a given curvature  $K < 0$  is defined as

$$\mathbb{P}_K^n = \rho_K(\mathbb{H}_K^n) = \left\{ \mathbf{x} \in \mathbb{R}^n : \langle \mathbf{x}, \mathbf{x} \rangle_2 < R^2 \right\} = \left\{ \mathbf{x} \in \mathbb{R}^n : \langle \mathbf{x}, \mathbf{x} \rangle_2 < -\frac{1}{K} \right\}.$$

The induced distance function by the metric tensor is

$$\begin{aligned} d_{\mathbb{P}}(\mathbf{x}, \mathbf{y}) &= d_{\mathbb{H}}(\rho_K^{-1}(\mathbf{x}), \rho_K^{-1}(\mathbf{y})) \\ &= R \cosh^{-1} \left( 1 + \frac{2R^2 \|\mathbf{x} - \mathbf{y}\|_2^2}{(R^2 - \|\mathbf{x}\|_2^2)(R^2 - \|\mathbf{y}\|_2^2)} \right) \\ &= \frac{1}{\sqrt{-K}} \cosh^{-1} \left( 1 - \frac{2K \|\mathbf{x} - \mathbf{y}\|_2^2}{(1 + K \|\mathbf{x}\|_2^2)(1 + K \|\mathbf{y}\|_2^2)} \right). \end{aligned}$$

Formally, the  $n$ -dimensional Poincaré ball space is a Poincaré ball with the  $d_{\mathbb{P}}$  metric  $(\mathbb{P}_K^n, d_{\mathbb{P}})$ .

### 2.3.3 Gyrovector spaces

An important analogy to vector spaces (especially vector addition and scalar multiplication) in non-Euclidean geometry is the notion of gyrovector spaces (Ungar, 2008). Both of the above spaces  $\mathbb{D}_K$  and  $\mathbb{P}_K$  (jointly denoted as  $\mathcal{M}_K$ ) share the same structure, hence they also share the following definition for Möbius addition, due to Ungar (2008).

The Möbius addition  $\oplus_K$  of  $\mathbf{x}, \mathbf{y} \in \mathcal{M}_K$  is defined as

$$\mathbf{x} \oplus_K \mathbf{y} = \frac{(1 - 2K \langle \mathbf{x}, \mathbf{y} \rangle_2 - K \|\mathbf{y}\|_2^2) \mathbf{x} + (1 + K \|\mathbf{x}\|_2^2) \mathbf{y}}{1 - 2K \langle \mathbf{x}, \mathbf{y} \rangle_2 + K^2 \|\mathbf{x}\|_2^2 \|\mathbf{y}\|_2^2}. \quad (2.1)$$

We can now define “gyrospace distances” for both of the above spaces:

$$\begin{aligned} d_{\mathbb{D}_{\text{gyr}}}(\mathbf{x}, \mathbf{y}) &= \frac{2}{\sqrt{K}} \tan^{-1}(\sqrt{K} \|\mathbf{x} \oplus_K \mathbf{y}\|_2) \\ d_{\mathbb{P}_{\text{gyr}}}(\mathbf{x}, \mathbf{y}) &= \frac{2}{\sqrt{-K}} \tanh^{-1}(\sqrt{-K} \|\mathbf{x} \oplus_K \mathbf{y}\|_2) \end{aligned}$$

These two distances are equivalent to their non-gyrospace variants

$$d_{\mathcal{M}}(\mathbf{x}, \mathbf{y}) = d_{\mathcal{M}_{\text{gyr}}}(\mathbf{x}, \mathbf{y}),$$

as is shown in Theorems A.12 and A.29. Additionally, Theorems A.13 and A.30 show that

$$d_{\mathcal{M}_{\text{gyr}}}(\mathbf{x}, \mathbf{y}) \xrightarrow{K \rightarrow 0} 2 \|\mathbf{x} - \mathbf{y}\|_2 = 2d_{\mathbb{E}}(\mathbf{x}, \mathbf{y}),$$

which means that the non-gyrospace distance functions converge to the Euclidean distance function as  $K \rightarrow 0$  as well. In practice, the gyrospace distance functions are numerically more stable than the induced distance functions.

Since Ganea et al. (2018a); Tifrea et al. (2019) used the same gyrovector space formalism to define an exponential map, its inverse logarithmic map, and parallel transport in the Poincaré ball, we can define them for both manifolds. The exponential map is defined as

$$\exp_{\mathbf{x}}^K(\mathbf{v}) = \mathbf{x} \oplus_K \left( \tan \left( \sqrt{K} \frac{\lambda_{\mathbf{x}}^K \|\mathbf{v}\|_2}{2} \right) \frac{\mathbf{v}}{\sqrt{K} \|\mathbf{v}\|_2} \right)$$

in the projected hypersphere, and

$$\exp_{\mathbf{x}}^K(\mathbf{v}) = \mathbf{x} \oplus_K \left( \tanh \left( \sqrt{-K} \frac{\lambda_{\mathbf{x}}^K \|\mathbf{v}\|_2}{2} \right) \frac{\mathbf{v}}{\sqrt{-K} \|\mathbf{v}\|_2} \right)$$

in the Poincaré ball, for all  $\mathbf{x} \in \mathcal{M}_K^n$  and  $\mathbf{v} \in \mathcal{T}_{\mathbf{x}}\mathcal{M}_K^n$ . Its inverse, the logarithmic map is

$$\log_{\mathbf{x}}^K(\mathbf{y}) = \frac{2}{\sqrt{K} \lambda_{\mathbf{x}}^K} \tan^{-1} \left( \sqrt{K} \|\mathbf{z}\|_2 \right) \frac{\mathbf{z}}{\|\mathbf{z}\|_2}$$

in the projected hypersphere, and

$$\log_{\mathbf{x}}^K(\mathbf{y}) = \frac{2}{\sqrt{-K} \lambda_{\mathbf{x}}^K} \tanh^{-1} \left( \sqrt{-K} \|\mathbf{z}\|_2 \right) \frac{\mathbf{z}}{\|\mathbf{z}\|_2}$$

in the Poincaré ball, where  $\mathbf{z} = -\mathbf{x} \oplus_K \mathbf{y}$ , for all  $\mathbf{x}, \mathbf{y} \in \mathcal{M}_K^n$ .

To define parallel transport, we first need the notion of gyration (Ungar, 2008)

$$\text{gyr}[\mathbf{x}, \mathbf{y}]\mathbf{v} = \ominus_K(\mathbf{x} \oplus_K \mathbf{y}) \oplus_K (\mathbf{x} \oplus_K (\mathbf{y} \oplus_K \mathbf{v})).$$

Parallel transport in the both the projected hypersphere and the Poincaré ball then is

$$\text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}) = \frac{\lambda_{\mathbf{x}}^K}{\lambda_{\mathbf{y}}^K} \text{gyr}[\mathbf{y}, -\mathbf{x}]\mathbf{v},$$

for all  $\mathbf{x}, \mathbf{y} \in \mathcal{M}_K^n$  and  $\mathbf{v} \in \mathcal{T}_{\mathbf{x}}\mathcal{M}_K^n$ . There are simpler variants of parallel transport when we transport to or from  $\boldsymbol{\mu}_0 = \mathbf{0}$ :

$$\begin{aligned} \text{PT}_{\boldsymbol{\mu}_0 \rightarrow \mathbf{y}}^K(\mathbf{v}) &= \frac{2}{\lambda_{\mathbf{y}}^K} \mathbf{v} \\ \text{PT}_{\mathbf{x} \rightarrow \boldsymbol{\mu}_0}^K(\mathbf{v}) &= \frac{\lambda_{\mathbf{x}}^K}{2} \mathbf{v}. \end{aligned}$$

For more details on all of the above, see Sections A.3.2, A.2.2, and A.4 in Appendix A.

## 2.4 Duality between constant curvature spaces

It is very noticeable that most statements and operations in constant curvature spaces have a dual statement or operation in the corresponding space with the opposite curvature sign. For example, most theorems about the hyperboloid apply (with small adjustments) to the hypersphere, and most theorems about the Poincaré ball apply to the projected spherical space as well, and vice-versa.

The notion of duality is one which comes up very often in mathematics, and in our case is based on Euler’s formula:

$$e^{ix} = \cos(x) + i \sin(x). \quad (2.2)$$

It provides a connection between trigonometric, hyperbolic trigonometric, and exponential functions. From this, a few useful relationships can be derived, like

$$\begin{aligned} \cosh(ix) &= \cos(x) & \cosh(x) &= \cos(ix) \\ \sinh(ix) &= i \sin(x) & \sinh(x) &= -i \sin(ix) \\ \tanh(ix) &= i \tan(x) & \tanh(x) &= -i \tan(ix) \end{aligned}$$

and many more. Another important fact is the Pythagorean theorem and its hyperbolic variant

$$\cos^2(x) + \sin^2(x) = 1, \quad \cosh^2(x) - \sinh^2(x) = 1.$$

Using the above properties, along with the notion of principal square roots of complex numbers  $\sqrt{-z} = i\sqrt{z}$ , we can convert any hyperbolic formula to its spherical equivalent, and vice-versa. Using a curvature-aware definition of trigonometric functions

$$\begin{aligned} \sin_K &= \begin{cases} \sin & \text{if } K > 0 \\ \sinh & \text{if } K < 0 \end{cases} & \cos_K &= \begin{cases} \cos & \text{if } K > 0 \\ \cosh & \text{if } K < 0 \end{cases} \\ \tan_K &= \begin{cases} \tan & \text{if } K > 0 \\ \tanh & \text{if } K < 0 \end{cases} \end{aligned}$$

we can summarize all the operations for all non-zero constant curvature spaced defined above in Table 2.1 and Table 2.2 for projected spaces.

## 2.5 Brief comparison of constant curvature space models

So far, we have seen five different models of constant curvature space, each of which has advantages and disadvantages when applied to learning latent representations in them using VAEs.

2.5. Brief comparison of constant curvature space models

Distance	$d(\mathbf{x}, \mathbf{y}) = \frac{1}{\sqrt{ K }} \cos_K^{-1}( K  \langle \mathbf{x}, \mathbf{y} \rangle_K)$
Exp. map	$\exp_{\mathbf{x}}^K(\mathbf{v}) = \cos_K(\beta) \mathbf{x} + \sin_K(\beta) \frac{\mathbf{v}}{\beta}, \beta = \sqrt{ K } \ \mathbf{v}\ _K$
Log. map	$\log_{\mathbf{x}}^K(\mathbf{y}) = \frac{\cos_K^{-1}(\alpha)}{\sin_K(\cos_K^{-1}(\alpha))} (\mathbf{y} - \alpha \mathbf{x}), \alpha = K \langle \mathbf{x}, \mathbf{y} \rangle_K$
Par. transp.	$\text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}) = \mathbf{v} - \frac{K \langle \mathbf{y}, \mathbf{v} \rangle_K}{1 + K \langle \mathbf{x}, \mathbf{y} \rangle_K} (\mathbf{x} + \mathbf{y})$

Table 2.1: Summary of operations in  $\mathbb{S}_K$  and  $\mathbb{H}_K$ .

Möbius add.	$\mathbf{x} \oplus_K \mathbf{y} = \frac{(1 - 2K \langle \mathbf{x}, \mathbf{y} \rangle_2 - K \ \mathbf{y}\ _2^2) \mathbf{x} + (1 + K \ \mathbf{x}\ _2^2) \mathbf{y}}{1 - 2K \langle \mathbf{x}, \mathbf{y} \rangle_2 + K^2 \ \mathbf{x}\ _2^2 \ \mathbf{y}\ _2^2}$
Distance	$d(\mathbf{x}, \mathbf{y}) = \frac{1}{\sqrt{ K }} \cos_K^{-1} \left( 1 - \frac{2K \ \mathbf{x} - \mathbf{y}\ _2^2}{(1 + K \ \mathbf{x}\ _2^2)(1 + K \ \mathbf{y}\ _2^2)} \right)$
Gyr. dist.	$d_{\text{gyr}}(\mathbf{x}, \mathbf{y}) = \frac{2}{\sqrt{ K }} \tan_K^{-1}(\sqrt{ K } \ \mathbf{x} \oplus_K \mathbf{y}\ _2)$
Lambda	$\lambda_{\mathbf{x}}^K = \frac{2}{1 + K \ \mathbf{x}\ _2^2}$
Exp. map	$\exp_{\mathbf{x}}^K(\mathbf{v}) = \mathbf{x} \oplus_K \left( \tan_K \left( \sqrt{ K } \frac{\lambda_{\mathbf{x}}^K \ \mathbf{v}\ _2}{2} \right) \frac{\mathbf{v}}{\sqrt{ K } \ \mathbf{v}\ _2} \right)$
Log. map	$\log_{\mathbf{x}}^K(\mathbf{y}) = \frac{2}{\sqrt{ K } \lambda_{\mathbf{x}}^K} \tan_K^{-1} \left( \sqrt{ K } \ \mathbf{z}\ _2 \right) \frac{\mathbf{z}}{\ \mathbf{z}\ _2}$ where $\mathbf{z} = \mathbf{x} \oplus_K \mathbf{y}$
Gyration	$\text{gyr}[\mathbf{x}, \mathbf{y}]\mathbf{v} = \ominus_K(\mathbf{x} \oplus_K \mathbf{y}) \oplus_K(\mathbf{x} \oplus_K(\mathbf{y} \oplus_K \mathbf{v}))$
Par. transp.	$\text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}) = \frac{\lambda_{\mathbf{x}}^K}{\lambda_{\mathbf{y}}^K} \text{gyr}[\mathbf{y}, -\mathbf{x}]\mathbf{v}$ $\text{PT}_{\mu_0 \rightarrow \mathbf{y}}^K(\mathbf{v}) = \frac{2}{\lambda_{\mathbf{y}}^K} \mathbf{v}, \quad \text{PT}_{\mathbf{x} \rightarrow \mu_0}^K(\mathbf{v}) = \frac{\lambda_{\mathbf{x}}^K}{2} \mathbf{v}$

Table 2.2: Summary of operations in  $\mathbb{D}_K$  and  $\mathbb{P}_K$ .

A big advantage of the hyperboloid and hypersphere is that optimization in the spaces does not suffer from as many numerical instabilities as it does in the respective projected spaces. On the other hand, we have seen that when  $K \rightarrow 0$ , the norms of points go to infinity. As we will see in experiments, this is not a problem when optimizing curvature within these spaces in practice, except if we're trying to cross the boundary at  $K = 0$  and go from a hyperboloid to a sphere, or vice versa. Intuitively, the points are just positioned very differently in the ambient space of  $\mathbb{H}_{-\varepsilon}$  and  $\mathbb{S}_{\varepsilon}$ , for a small  $\varepsilon > 0$ .

Since points in the  $n$ -dimensional projected hypersphere and Poincaré ball models can be represented using a real vector of length  $n$ , it enables us to visualize points in these manifolds directly for  $n = 2$  or even  $n = 3$ . On the other hand, optimizing a function over these models is not very well-conditioned. In the case of the Poincaré ball, a significant amount of points lie close to the boundary of the ball (i.e. with a squared norm of almost  $\frac{1}{K}$ ), which causes numerical instabilities even when using 64-bit float precision in computations.

A similar problem occurs with the projected hypersphere with points that are far away from the origin  $\mathbf{0}$  (i.e. points that are close to the “South pole” on the backprojected sphere). Unintuitively, all points that are far away from the origin are actually very close to each other with respect to the induced distance function and very far away from each other in terms of Euclidean distance. For an illustration, see Figure 2.1b.

Both distance conversion theorems (A.13, A.30) rely on the points being *fixed* when changing curvature. If they are somehow dependent on curvature, the convergence theorem does not hold. We conjecture that if points stay close to the boundary in  $\mathbb{P}$  or far away from  $\mathbf{0}$  in  $\mathbb{D}$  as  $K \rightarrow 0$ , this is exactly the reason for numerical instabilities (apart from the standard numerical problem of representing large numbers in floating-point notation).

Because of the above reasons, we will do some of our experiments with the projected spaces and others with the hyperboloid and hypersphere, and aim to compare the performance of these empirically as well.

## 2.6 Products of spaces

In the whole chapter, we assumed our space consists of only one model of varying dimensionality  $n$  and fixed curvature  $K$ . In the spirit of Gu et al. (2019) and being able to provide a unified formulation of our geometries (Section 2.4), we propose learning VAE latent representations in products of constant curvature spaces, contrary to existing VAE approaches which are limited to a single Riemannian manifold as a latent space.

Our latent spaces, therefore, consist of several *component spaces* (or components)

$$\mathcal{M} = \times_{i=1}^k \mathcal{M}_{K_i}^{n_i},$$

where  $n_i$  is the dimensionality of the space,  $K_i$  is its curvature, and  $\mathcal{M} \in \{\mathbb{E}, \mathbb{S}, \mathbb{D}, \mathbb{H}, \mathbb{P}\}$  is the model choice. Note that, the notation is slightly loose, i.e. it permits denoting a positively-curved hyperbolic space (or vice-versa). We will assume only valid combinations of parameters.

Even though all components have constant curvature, the resulting manifold  $\mathcal{M}$  has non-constant curvature. Its distance function naturally decomposes based on its definition

$$d_{\mathcal{M}}(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^k d_{\mathcal{M}_{K_i}^{n_i}}(\mathbf{x}^{(i)}, \mathbf{y}^{(i)}),$$

where  $\mathbf{x}^{(i)}$  represents a vector in  $\mathcal{M}_{K_i}^{n_i}$ , corresponding to the part of the latent space representation of  $\mathbf{x}$  belonging to  $\mathcal{M}_{K_i}^{n_i}$ .

All other operations we defined on our manifolds are element-wise so the generalization is trivial — we simply decompose the representations into parts  $\mathbf{x}^{(i)}$  as defined before, apply the operation on that part and concatenate the resulting parts back:

$$\tilde{\mathbf{x}}^{(i)} = f_{K_i}^{(n_i)}(\mathbf{x}^{(i)}), \quad \tilde{\mathbf{x}} = \bigodot_{i=1}^k \tilde{\mathbf{x}}^{(i)}.$$

The *signature* of the product space, i.e. its parametrization, has several degrees of freedom per component:

1. the model  $\mathcal{M}$ ,
2. the dimensionality  $n_i$ , and
3. the curvature  $K_i$ .

To summarize, we need to select all of the above for every component in our product space. An example signature of total dimensionality 42 could be

$$(\mathbb{H}_{-2}^2)^6 \times (\mathbb{S}_3^5)^4 \times \mathbb{E}^{10}.$$

However, due to representations of some spaces needing more dimensions, a point in this space would be represented using a real vector of dimension  $(2+1) \cdot 6 + (5+1) \cdot 4 + 10 = 52$ .

The notation used above is to be read as

$$(\mathcal{M}_{K_i}^{n_i})^j = \times_{l=1}^j \mathcal{M}_{K_i}^{n_i}.$$



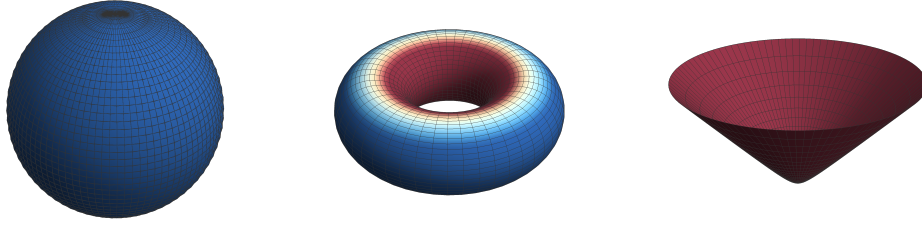
(a) Sphere  $\mathbb{S}^2$ (b) Torus  $\mathbb{S}^1 \times \mathbb{S}^1$ (c) Hyperboloid  $\mathbb{H}^2$ 

Figure 2.2: Visualization of the topological difference between a two-dimensional sphere, a torus, and a hyperboloid. Colors correspond to curvature (red is -1, white is 0, blue is +1).

**Remark (Euclidean constant curvature spaces are sub-divisible.)** *In some spaces, like Euclidean spaces, the notation is redundant. For  $n_1, \dots, n_k \in \mathbb{Z}$ , such that  $\sum_{i=1}^k n_i = n \in \mathbb{Z}$ , it holds that the Cartesian product of Euclidean spaces  $\mathbb{E}^{n_i}$  is*

$$\mathbb{E}^n = \prod_{i=1}^k \mathbb{E}^{n_i}.$$

*The proof follows directly from the definition of  $\mathbb{R}^n$ .*

However, the equality in Remark 2.1 does *not* hold for the hypersphere and hyperboloid. This is due to the additional constraints posed on the points in the definitions of individual models of curved spaces, and the remark does not hold even if all the partial spaces have an equal Gaussian curvature and are the same model.

A simple example (Figure 2.2) is that the product of two circles  $\mathbb{S}_K^1$  is not a sphere  $\mathbb{S}_K^2$ , but a torus  $\mathbb{S}_K^1 \times \mathbb{S}_K^1$ , which is well-known to be topologically different to a sphere (Gu et al., 2019). As stated above and apparent from Figure 2.2b, the resulting space also does not have constant curvature, even though both spaces were of the same type and had the same curvature.

## Chapter 3

---

# Probability

---

In this chapter, we present an overview of the different probability distributions in constant curvature spaces, in order to learn representations in latent spaces with prior probability distributions enforced on them. Some details, properties, and proofs are appended in Appendix B.

To be able to train Variational Autoencoders (Kingma and Welling, 2014), we need to choose a probability distribution  $p$  as a prior on the latent representations, and a corresponding posterior distribution family  $q$ . Both of these distributions have to be differentiable with respect to their parametrization, they need to have a differentiable Kullback-Leiber (KL) divergence

$$D_{\text{KL}}(q \parallel p) = \mathbf{E}_{\mathbf{z} \sim q} \left[ \log \left( \frac{q(\mathbf{z})}{p(\mathbf{z})} \right) \right],$$

and be “reparameterizable” (Kingma and Welling, 2014, Section 2.4). For distributions where the KL does not have a closed-form solution independent on  $\mathbf{z}$ , or where this integral is too hard to compute, we can estimate it using Monte Carlo estimation

$$D_{\text{KL}}(q \parallel p) \approx \frac{1}{L} \sum_{l=1}^L \log \left( \frac{q(\mathbf{z}^{(l)})}{p(\mathbf{z}^{(l)})} \right) \stackrel{\text{if } L=1}{=} \log \left( \frac{q(\mathbf{z}^{(1)})}{p(\mathbf{z}^{(1)})} \right),$$

where  $\mathbf{z}^{(l)} \sim q$  for all  $l = 1, \dots, L$ .

The Euclidean VAE uses a natural choice for a prior on its latent representations — the Normal distribution.

### 3.1 Multivariate Normal distribution

In Euclidean space  $\mathbb{E}^n$ , the Normal (Gaussian) distribution is defined as

$$\mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{\sqrt{(2\pi)^n \det(\boldsymbol{\Sigma})}} \exp(-(\mathbf{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu})),$$

where  $\boldsymbol{\mu} \in \mathbb{R}^n$  and  $\boldsymbol{\Sigma} \in \mathbb{R}^{n \times n}$ .

We notice that this distribution is differentiable with respect to both parameters  $\boldsymbol{\mu}$  and  $\boldsymbol{\Sigma}$ . Sampling from this distribution is usually well supported by all computational libraries, using the reparameterization trick

$$\mathbf{z}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \implies (\boldsymbol{\mu} + \boldsymbol{\Sigma}\mathbf{z}_0) \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}).$$

Importantly, this makes it possible to compute partial derivatives of the sample  $\mathbf{z}$  with respect to both parameters  $\boldsymbol{\mu}$  and  $\boldsymbol{\Sigma}$ , which is crucial for learning a VAE using this distribution as a prior on the latent space.

The only missing fact we need to train a VAE with a Normal prior on the latent representations is the Kullback-Leiber (KL) divergence. For the multivariate Normal distribution, the KL has an explicit form with respect to the parameters:

$$\begin{aligned} D_{\text{KL}}(\mathcal{N}(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0) \parallel \mathcal{N}(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)) &= \\ &= \frac{1}{2} \left( \text{trace}(\boldsymbol{\Sigma}_1^{-1}\boldsymbol{\Sigma}_0) + (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_0)^T \boldsymbol{\Sigma}_1^{-1} (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_0) - n + \ln \frac{\det(\boldsymbol{\Sigma}_1)}{\det(\boldsymbol{\Sigma}_0)} \right). \end{aligned}$$

**Remark (Diagonal covariance in multivariate Normal distributions)**

*It can be shown that a multivariate Normal distribution  $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$  with a diagonal covariance matrix*

$$\boldsymbol{\Sigma} = \text{diag}(\sigma_{11}^2, \sigma_{22}^2, \dots, \sigma_{nn}^2),$$

*is equivalent to a combination of  $n$  independent univariate Normal distributions  $\mathcal{N}(\mu_i, \sigma_{ii}^2)$ , which is a very practical property. For details, see Do (2008).*

What makes the Normal distribution in Euclidean a good choice for a prior? Apart from satisfying the requirements for a VAE prior and posterior distribution, it has additional properties.

**Remark (Maximum Likelihood characterization)** *One of the formulations of the Maximum Likelihood characterization, due to Gauss (1809), states that the Maximum Likelihood Estimator (MLE) of the location parameter of a location-scale distribution family is the sample arithmetic mean (for any sample size) if and only if the family is Normal.*

*A location-scale family is one that has a vector location/mean parameter and a scalar non-negative variance parameter.*

**Remark (Maximum Entropy principle)** *Out of all probability distributions consistent with a given set of constraints, the distribution with maximum uncertainty should be chosen (Jaynes, 1957).*

*Additionally, we know that if the solution exists, then it is unique with a given set of constraints (although it does not have to exist).*

The Normal distribution is the Maximum Entropy distribution among all distributions on  $\mathbb{R}^n$  with a specified covariance matrix  $\Sigma$ , and also satisfies the MLE-characterization.

## 3.2 Normal-like distributions in non-Euclidean constant curvature spaces

Generalizing the Normal distribution to spaces of constant non-zero curvature is not straightforward. There exist several fundamentally different approaches, with different properties. We discuss the following three generalizations based on the way they are constructed (Mathieu et al., 2019, Appendix B):

- Wrapping – This approach leverages the fact that all manifolds define a tangent vector space at every point. We simply sample from a Euclidean Normal distribution in the tangent space at  $\mu_0$  with mean  $\mathbf{0}$ , and use parallel transport and the exponential map to map the sampled point onto the manifold. The PDF can be obtained using the multivariate chain rule if we can compute the determinant of the Jacobian of the parallel transport and the exponential map. This is very computationally effective at the expense of losing some theoretical properties. An example of this category is the Wrapped Normal distribution presented in Section 3.4.
- Restriction – The “Restricted Normal” approach is conceptually antagonistic to the Wrapped Normal. Instead of expanding a point to a dimensionally larger point, we restrict a point of the ambient space sampled from a Euclidean Normal to the manifold. The consequence is that the distributions constructed this way are based on the “flat” Euclidean distance. An example of this is the von Mises-Fisher (vMF) distribution (Section 3.3).
- Maximize entropy – Assuming a known mean and covariance matrix (first and second moments), we want to maximize the entropy of the distribution (Pennec, 2006). This approach is usually called the Riemannian Normal distribution. Mathieu et al. (2019) derive it for the Poincaré ball, and Hauberg (2018) derive the Spherical Normal distribution on the hypersphere.

Riemannian Normal distributions resemble the Euclidean Normal distribution’s properties the closest, but it is usually very hard to sample from such distributions, compute their normalization constants, and even derive the specific form. Since the gains for VAE performance using this construction of Normal distributions is only marginal, as reported by Mathieu et al. (2019), we have chosen to only make limited use of them.

To the best of our knowledge, there have not been attempts to establish Restricted Normal distributions in hyperbolic spaces, neither are there any “richer” non-scalar variance generalizations of the spherical variants (vMF).

Therefore, due to their simplicity, we focus primarily on Wrapped Normal distributions.

### 3.3 Von Mises-Fisher distribution

The von-Mises Fisher (vMF) distribution is a probability distribution on  $\mathbb{S}_1^n$  in the  $(n + 1)$ -dimensional ambient space  $\mathbb{R}^{n+1}$ , parametrized by a mean  $\boldsymbol{\mu} \in \mathbb{R}^{n+1}$ , and a concentration parameter  $\kappa \in [0, \infty)$  (Davidson et al., 2018; Tanabe et al., 2007):

$$\begin{aligned} \text{vMF}(\mathbf{x}|\boldsymbol{\mu}, \kappa) &= \mathcal{C}_{n+1}(\kappa) \exp(\kappa \boldsymbol{\mu}^T \mathbf{x}), \\ \mathcal{C}_n(\kappa) &= \frac{\kappa^{n/2+1}}{(2\pi)^{n/2} \mathcal{I}_{n/2-1}(\kappa)}, \\ \mathcal{I}_r(x) &= \sum_{n=1}^{\infty} \frac{1}{n! \Gamma(n+r+1)} \left(\frac{x}{2}\right)^{2n+r}, \\ \Gamma(z) &= \int_0^{\infty} x^{z-1} \exp(-x) dx, z \in \mathbb{R}_{>0}, \end{aligned}$$

where  $\|\boldsymbol{\mu}\|_2 = 1$ , i.e.  $\boldsymbol{\mu} \in \mathbb{S}^n$ ,  $\mathcal{C}_{n+1}(\kappa)$  is the normalization constant,  $\mathcal{I}_r$  denotes the modified Bessel function of the first kind at order  $r$ , and  $\Gamma(z)$  denotes the generalized factorial function (usually called “gamma function”). An interesting property is that if  $\kappa \rightarrow 0^+$ ,  $\text{vMF}(\cdot, \kappa)$  degenerates into the hyperspherical uniform distribution and does not depend on the mean (Xu and Durrett, 2018). For details on the hyperspherical uniform distribution, see Section B.1 in Appendix B.

If we place a hyperspherical uniform prior  $U(\mathbb{S}^{n-1})$  on the sphere  $\mathbb{S}^{n-1}$ , we can derive the KL divergence (Davidson et al., 2018, Appendix B):

$$D_{\text{KL}}(\text{vMF}(\boldsymbol{\mu}, \kappa) || U(\mathbb{S}^{n-1})) = \kappa \frac{\mathcal{I}_{n/2}(\kappa)}{\mathcal{I}_{n/2-1}(\kappa)} + \log \mathcal{C}_n(\kappa) + \underbrace{\log \left( \frac{2(\pi^{n/2})}{\Gamma(n/2)} \right)}_{S_{n-1}(1)}.$$

Notice that the second and third term is a log of the ratio of normalizers of the vMF distribution and the hyperspherical uniform distribution. Xu and Durrett (2018) arrive at the same KL divergence formulation.

Davidson et al. (2018) also derive the gradient of the KL divergence with respect to  $\kappa$ , because automatic differentiation libraries have problems with

the calculation due to the occurrence of the modified Bessel function in the KL:

$$\begin{aligned} \nabla_{\kappa} D_{\text{KL}}(\text{vMF}(\boldsymbol{\mu}, \kappa) \| U(\mathbb{S}^{n-1})) &= \\ &= \frac{1}{2} \kappa \left( \frac{\mathcal{I}_{n/2+1}(\kappa)}{\mathcal{I}_{n/2-1}(\kappa)} - \frac{\mathcal{I}_{n/2}(\kappa)(\mathcal{I}_{n/2-2}(\kappa) + \mathcal{I}_{n/2}(\kappa))}{\mathcal{I}_{n/2-1}(\kappa)^2} + 1 \right). \end{aligned}$$

The modified Bessel functions in the KL divergence term and in the formula above can be computed in a numerically stable way using exponential scaling  $\mathcal{I}_r^{\text{exp}}(\kappa) = \exp(-\kappa)\mathcal{I}_r(\kappa)$ .

We are still missing a proper sampling scheme to be able to use the vMF distribution in a VAE latent space. To date, the only way to sample from a vMF distribution for  $n > 3$  is a rejection-sampling scheme as described by Davidson et al. (2018, Section 3.3 and Appendix A), Naesseth et al. (2017), and Ulrich (1984). Algorithm 1 in Davidson et al. (2018) briefly summarizes the result and the sampling process. It is important to note that the rejection-sampling part of the procedure only happens on a one-dimensional random variable, so it does not suffer from the ‘‘curse of dimensionality’’. In the case of  $n = 3$ , we can substitute rejection sampling with a direct sampling as outlined in the algorithm.

However, as pointed out by Davidson et al. (2018), as dimensionality increases, even a simple diagonal multivariate Normal distribution starts to approximate a hypersphere, while its posterior becomes more expressive due to a different variance parametrization (a diagonal matrix versus a single scalar concentration parameter). As noted in Section B.1, the sphere surface area starts to collapse around  $n = 7$  and vanishes as  $n \rightarrow \infty$ . Additionally, we also observed that in practice the numerical properties of the rejection sampling procedure (Algorithm 2, Davidson et al. (2018)) are insufficient and the procedure takes significantly longer to converge as dimensionality increases, probably due to numerical errors.

To the best of our knowledge, the von-Mises Fischer distribution was never defined on a hypersphere of different radius than 1. In Remark B.1 in Appendix B we briefly elaborate on why sampling  $\mathbf{z} \sim \text{vMF}(\boldsymbol{\mu}, \kappa \cdot R^{-n})$  on  $\mathbb{S}_1^n$  and then orthogonally projecting

$$\mathbf{z}' = \frac{R\mathbf{z}}{\|\mathbf{z}\|_2} = R\mathbf{z}$$

the sampled point onto  $\mathbb{S}_K^n$  is roughly equivalent to defining vMF on  $\mathbb{S}_K^n$  directly.

It is worth noting that for a given  $\boldsymbol{\mu} \in \mathbb{S}_1^n$ , Mardia (1975) has proven that the vMF distribution is the Maximum Entropy distribution on the sphere, and also satisfies the Maximum Likelihood characterization, hence it is as close

as we can get to a Normal distribution with a scalar scale parameter on the hypersphere.

### 3.4 Wrapped Normal distributions

The following distribution can be applied to all manifolds that we have introduced. The only differences are the specific forms of the operations and the log-determinant in the PDF.

First of all, we need to define an “origin” point on the manifold, which we will denote as  $\boldsymbol{\mu}_0 \in \mathcal{M}_K$ . What this point corresponds to is manifold-specific: in the hyperboloid and hypersphere it corresponds to the point

$$\boldsymbol{\mu}_0 = (R, 0, \dots, 0)^T,$$

and in the Poincaré ball, projected sphere, and Euclidean space it is simply  $\boldsymbol{\mu}_0 = \mathbf{0}$ , the origin of the coordinate system.

Sampling from the distribution  $\mathcal{WN}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$  has been described in detail in Nagano et al. (2019) and Mathieu et al. (2019). Essentially, this corresponds to:

1.  $\boldsymbol{v} \sim \mathcal{N}(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}) \in \mathcal{T}_{\boldsymbol{\mu}_0} \mathcal{M}_K$ ,
2.  $\boldsymbol{u} = \text{PT}_{\boldsymbol{\mu}_0 \rightarrow \boldsymbol{\mu}}^K(\boldsymbol{v}) \in \mathcal{T}_{\boldsymbol{\mu}} \mathcal{M}_K$ ,
3.  $\boldsymbol{z} = \exp_{\boldsymbol{\mu}}^K(\boldsymbol{u}) \in \mathcal{M}_K$ .

The process is illustrated for  $\mathbb{H}^1$  in Figure 3.1. The log-probability density function can be computed by the reverse procedure:

1.  $\boldsymbol{u} = \log_{\boldsymbol{\mu}}^K(\boldsymbol{z}) \in \mathcal{T}_{\boldsymbol{\mu}} \mathcal{M}_K$ ,
2.  $\boldsymbol{v} = \text{PT}_{\boldsymbol{\mu} \rightarrow \boldsymbol{\mu}_0}^K(\boldsymbol{u}) \in \mathcal{T}_{\boldsymbol{\mu}_0} \mathcal{M}_K$ ,
3.  $\log \mathcal{WN}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \log \mathcal{N}(\boldsymbol{v}; \boldsymbol{\mu}_0, \boldsymbol{\Sigma}) - \log \det \left( \frac{\partial f}{\partial \boldsymbol{v}} \right)$ ,

where  $f = \exp_{\boldsymbol{\mu}}^K \circ \text{PT}_{\boldsymbol{\mu}_0 \rightarrow \boldsymbol{\mu}}^K$ . We notice that the intermediate results for the log-PDF  $\boldsymbol{u}$  and  $\boldsymbol{v}$  are the same as in the sampling procedure for a given  $\boldsymbol{x}$  and  $\boldsymbol{z}$ . This allows us to not do redundant computations and also helps numerical stability.

The specific forms of the log-PDF for the four spaces  $\mathbb{H}$ ,  $\mathbb{S}$ ,  $\mathbb{D}$ , and  $\mathbb{P}$  are derived in Section B.3 of Appendix B. All the variants of this distribution are reparameterizable, differentiable, and the KL can be computed using Monte Carlo estimation. As a consequence of the distance function and operations convergence theorems A.13, A.30, A.40, A.41, and A.43, we can see that the Wrapped Normal distribution converges to the Euclidean Normal distribution as  $K \rightarrow 0$ .

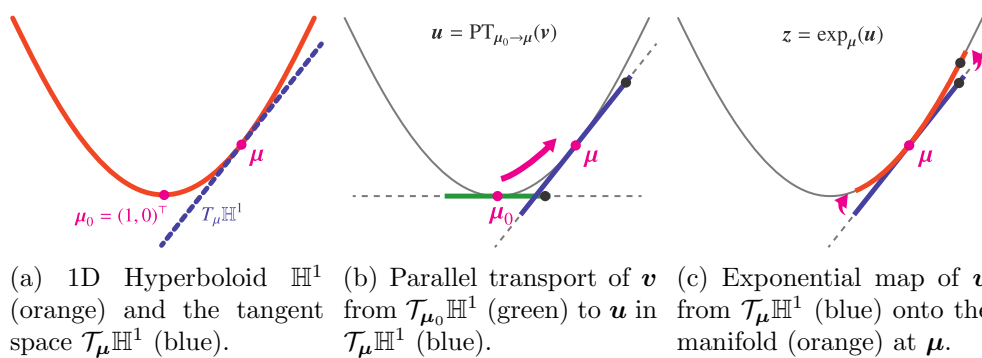


Figure 3.1: Three-step transformation of a sampled point  $v$  in  $\mathcal{T}_{\mu_0} \mathbb{H}^1$  for the hyperbolic Wrapped Normal distribution (Nagano et al., 2019).



## Variational Autoencoders

---

In this chapter, we briefly look at variational inference and introduce Variational Autoencoders (VAEs), originally defined by Kingma and Welling (2014) and Rezende et al. (2014). Lastly, we re-formulate VAEs for latent spaces that are products of constant curvature spaces.

### 4.1 Autoencoders

Autoencoders are neural networks used for reconstruction task (Goodfellow et al., 2016). They consist of an *encoder*  $\mathbf{h} = f(\mathbf{x})$  and a *decoder*  $\hat{\mathbf{x}} = g(\mathbf{h})$ , where  $\mathbf{x}$  represents the input and  $\hat{\mathbf{x}}$  a reconstruction of the input. The simplest choice for  $f$  and  $g$  would be identity functions — instead, we constrain both functions so that they cannot reproduce the input  $\mathbf{x}$  perfectly, hence introducing a “bottleneck” into the model. The simplest variant of linear autoencoders with mean squared loss directly correspond to a well-known dimensionality reduction procedure, Principal Component Analysis (PCA).

### 4.2 Variational Inference

Casting our problem into a probabilistic view (as summarized by Blei et al. (2017)): let  $\mathbf{x}$  be a set of observed variables and  $\mathbf{z}$  a set of latent (unobserved) variables. Our model is the joint density  $p(\mathbf{z}, \mathbf{x})$ . We aim to find the “hidden” representation of our observed variables  $\mathbf{x}$ , i.e. compute the conditional distribution

$$p(\mathbf{z}|\mathbf{x}) = \frac{p(\mathbf{z}, \mathbf{x})}{p(\mathbf{x})}.$$

The denominator (or the normalization constant), in Bayesian literature usually called the *evidence*, can also be computed from our model

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

which is effectively intractable to compute for most models.

Since we are therefore unable to do exact inference, we aim to do approximate inference by introducing a distribution  $q$  from a variational distribution family  $\mathcal{F}$  and use optimization to find such a  $q \in \mathcal{F}$  that approximates  $p(\mathbf{z}|\mathbf{x})$  best. This approach is called variational inference (Jordan et al., 1999).

Do note that there are other approaches to approximate inference. A natural one is estimating  $p(\mathbf{z}|\mathbf{x})$  pointwise, i.e. doing maximum a posteriori (MAP) estimation. This is fast and simple to compute, but very biased. Another standard approach is sampling using Markov Chain Monte Carlo (MCMC). Whilst this is (asymptotically) unbiased and easy to set up, the speed of convergence might be too slow, and convergence is hard to assess (Kingma and Welling, 2014).

Formally, we attempt to find the best probability distribution  $q(\mathbf{z})$  from a given family of distributions  $\mathcal{F}$ , such that it approximates the exact conditional distribution  $p(\mathbf{z}|\mathbf{x})$  best (Blei et al., 2017)

$$q^*(\mathbf{z}) = \arg \min_{q(\mathbf{z}) \in \mathcal{F}} D_{\text{KL}}(q(\mathbf{z}) || p(\mathbf{z}|\mathbf{x})),$$

where

$$D_{\text{KL}}(q || p) = \mathbf{E}_{\mathbf{z} \sim q} \left[ \log \left( \frac{q(\mathbf{z})}{p(\mathbf{z})} \right) \right],$$

for any probability distributions  $q$  and  $p$  (Kullback and Leibler, 1951).

However, computing the KL divergence above is intractable in practice, as we would have to evaluate  $\log p(\mathbf{x})$

$$\begin{aligned} D_{\text{KL}}(q(\mathbf{z}) || p(\mathbf{z}|\mathbf{x})) &= \mathbf{E}_{\mathbf{z} \sim q} [\log q(\mathbf{z})] - \mathbf{E}_{\mathbf{z} \sim q} [\log p(\mathbf{z}|\mathbf{x})] \\ &= \mathbf{E}_{\mathbf{z} \sim q} [\log q(\mathbf{z})] - \mathbf{E}_{\mathbf{z} \sim q} [\log p(\mathbf{z}, \mathbf{x})] + \log p(\mathbf{x}). \end{aligned}$$

From this, we can derive a lower-bound on  $\log p(\mathbf{x})$

$$\begin{aligned} \log p(\mathbf{x}) &= \mathbf{E}_{\mathbf{z} \sim q} [\log p(\mathbf{z}, \mathbf{x})] - \mathbf{E}_{\mathbf{z} \sim q} [\log q(\mathbf{z})] + D_{\text{KL}}(q(\mathbf{z}) || p(\mathbf{z}|\mathbf{x})) \\ &= \mathbf{E}_{\mathbf{z} \sim q} [\log(p(\mathbf{x}|\mathbf{z})p(\mathbf{z}))] - \mathbf{E}_{\mathbf{z} \sim q} [\log q(\mathbf{z})] + D_{\text{KL}}(q(\mathbf{z}) || p(\mathbf{z}|\mathbf{x})) \\ &= \mathbf{E}_{\mathbf{z} \sim q} [\log p(\mathbf{x}|\mathbf{z})] - D_{\text{KL}}(q(\mathbf{z}) || p(\mathbf{z})) + \underbrace{D_{\text{KL}}(q(\mathbf{z}) || p(\mathbf{z}|\mathbf{x}))}_{\geq 0} \\ &\geq \underbrace{\mathbf{E}_{\mathbf{z} \sim q} [\log p(\mathbf{x}|\mathbf{z})]}_{\text{reconstruction term}} - \underbrace{D_{\text{KL}}(q(\mathbf{z}) || p(\mathbf{z}))}_{\text{regularization term}} =: \text{ELBO}(q). \end{aligned}$$

The name ELBO is a shortcut of *evidence lower bound*. Its first term is usually called the reconstruction term, as it is proportional to the “difference” of the input and the reconstruction. The KL term can be interpreted as a regularizer to not let the variational distribution  $q(\mathbf{z})$  be “far” (in distribution space) from the prior  $p(\mathbf{z})$ .

Since  $\log p(\mathbf{x})$  is a constant with respect to  $q(\mathbf{z})$ , we can see that maximizing the ELBO is equivalent to minimizing  $D_{\text{KL}}(q(\mathbf{z}) \parallel p(\mathbf{z}|\mathbf{x}))$ , which was our original intention

$$q^*(\mathbf{z}) = \arg \min_{q(\mathbf{z}) \in \mathcal{F}} D_{\text{KL}}(q(\mathbf{z}) \parallel p(\mathbf{z}|\mathbf{x})) = \arg \max_{q(\mathbf{z}) \in \mathcal{F}} \text{ELBO}(q).$$

For more details on Variational Inference, see Blei et al. (2017).

### 4.2.1 Tighter bounds on the marginal log-likelihood

As proven by Burda et al. (2016, IWAE), we can obtain a tighter bound on the evidence than the ELBO

$$\text{ELBO}_L(q) = \mathbf{E}_{\mathbf{z}^{(l)} \sim q} \left[ \log \left( \frac{1}{L} \sum_{l=1}^L \frac{p(\mathbf{z}^{(l)}, \mathbf{x})}{q(\mathbf{z}^{(l)})} \right) \right].$$

As is apparent,

$$\begin{aligned} \text{ELBO}_1(q) &= \mathbf{E}_{\mathbf{z}^{(1)} \sim q} \left[ \log \left( \frac{1}{1} \sum_{l=1}^1 \frac{p(\mathbf{z}^{(1)}, \mathbf{x})}{q(\mathbf{z}^{(1)})} \right) \right] \\ &= \mathbf{E}_{\mathbf{z} \sim q} \left[ \log \left( \frac{p(\mathbf{z}, \mathbf{x})}{q(\mathbf{z})} \right) \right] = \text{ELBO}(q). \end{aligned}$$

Additionally, the property

$$\log p(\mathbf{x}) \geq \text{ELBO}_{L+1} \geq \text{ELBO}_L,$$

means that the bound  $\text{ELBO}_L$  imposes on the log-evidence  $\log p(\mathbf{x})$  becomes tighter as  $L$  grows. Even though it is computationally expensive to calculate due to the sampling, we use it as an evaluation metric. For more details, see (Burda et al., 2016).

## 4.3 Variational Autoencoders

We shortly present a generalized family of VAE models, so that we can exchange individual parts of the models and use our previously defined probability distributions. More about the motivation for using variational autoencoders compared to classical autoencoders can be found in Section C.1 of Appendix C.

A variational autoencoder  $\mathcal{V}$  consists of several parts: a latent space prior distribution, a posterior distribution family, and an encoder and decoder maps (and their parameters)

$$\mathcal{V} = \{p(\mathbf{z}), \mathcal{F}_q, f_\theta, g_\varphi\}.$$

Similarly to the equivalent in a classical autoencoder, the encoder map  $f_\theta(\mathbf{x}) : \mathcal{X} \rightarrow \mathcal{Z}$  is a differentiable function parametrized by  $\theta$ , that (deterministically) maps samples from the sample space  $\mathcal{X}$  to the latent space  $\mathcal{Z}$ . The encoder is then defined as a probability distribution  $q_{f_\theta(\mathbf{x})}(\mathbf{z})$ , usually shortened to  $q_\theta(\mathbf{z}|\mathbf{x})$ .

The decoder map  $g_\varphi(\mathbf{z}) : \mathcal{Z} \rightarrow \mathcal{X}$  is a differentiable function parameterized by  $\varphi$ , that (deterministically) maps samples back from the latent space  $\mathcal{Z}$  into the sample space  $\mathcal{X}$ . The decoder is then defined as a probability distribution  $p_{g_\varphi(\mathbf{z})}(\mathbf{x})$ , usually shortened to  $p_\varphi(\mathbf{x}|\mathbf{z})$ .

### 4.3.1 Learning VAEs

To learn a VAE given a specific dataset  $\mathbf{X} = (\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(N)})$ , we can choose an appropriate posterior family, prior distribution, and an encoder and decoder map structure. The most common choice of posterior family is the Normal distribution with the corresponding prior  $\mathcal{N}(\mathbf{0}, \mathbf{I})$ , and the encoder and decoder are usually two separate neural networks, where  $\theta$  and  $\varphi$  correspond to their respective weights and biases.

We can learn the weights  $\theta$  and  $\varphi$  using gradient-based optimization of the evidence lower bound, as defined in Section 4.2. A “forward pass” given a specific sample  $\mathbf{x}^{(n)}$  using this model includes the following steps:

1. Encode the sample:  $\tilde{\mathbf{x}}^{(n)} = f_\theta(\mathbf{x}^{(n)})$ .
2. Estimate the parameters of our posterior family:

$$\begin{aligned}\boldsymbol{\mu}(\tilde{\mathbf{x}}^{(n)}) &= h_\mu(\tilde{\mathbf{x}}^{(n)}) \\ \boldsymbol{\sigma}^2(\tilde{\mathbf{x}}^{(n)}) &= \exp(h_\sigma(\tilde{\mathbf{x}}^{(n)}/2)) \\ \boldsymbol{\Sigma}(\tilde{\mathbf{x}}^{(n)}) &= \boldsymbol{\sigma}^2 \mathbf{I}\end{aligned}$$

Above, we computed parameters of our sample-specific Normal distribution posterior. Using a single-layer neural network  $h_\mu$ , we made sure the resulting mean has the proper dimensions. Using a different single-layer neural network  $h_\sigma$ , we compute the logarithm of the standard deviation and transform it to the squared standard deviation. Finally, we transform  $\boldsymbol{\sigma}$  to a diagonal covariance matrix. Note that both neural networks’ weights and biases are part of  $\theta$ .

There is an important choice to be made here, that changes the performance of VAEs in practice: the specific form of the transformation functions  $\boldsymbol{\mu}(\cdot)$  and  $\boldsymbol{\Sigma}(\cdot)$ . By changing these, we can influence the number of parameters of the encoder, but more importantly of the posterior distribution — therefore significantly influence the degrees of freedom the samples from this distribution will have.

3. Construct the sample-specific posterior:

$$q_\theta(\mathbf{z}|\mathbf{x}^{(n)}) = \mathcal{N}(\mathbf{z}; \boldsymbol{\mu}(\tilde{\mathbf{x}}^{(n)}), \boldsymbol{\Sigma}(\tilde{\mathbf{x}}^{(n)})).$$

4. Sample from the *reparameterized* posterior:  $\mathbf{z}^{(n)} \sim q_\theta(\mathbf{z}|\mathbf{x}^{(n)})$ .

For example,  $\mathbf{z}^{(n)} = \boldsymbol{\mu}(\tilde{\mathbf{x}}^{(n)}) + \boldsymbol{\Sigma}(\tilde{\mathbf{x}}^{(n)}) \mathbf{v}$ , where  $\mathbf{v} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ .

5. Decode back into the sample space:  $\hat{\mathbf{x}}^{(n)} = g_\varphi(\mathbf{z}^{(n)})$ .

6. Compute the loss for  $\hat{\mathbf{x}}^{(n)}$ :  $L_n(\theta, \varphi) = \text{ELBO}(q_\theta(\cdot|\hat{\mathbf{x}}^{(n)}))$ .

Note that

$$\text{ELBO}(q_\theta(\cdot|\hat{\mathbf{x}}^{(n)})) = \underbrace{\mathbf{E}_{\mathbf{z} \sim q} [\log p_\varphi(\hat{\mathbf{x}}^{(n)}|\mathbf{z})]}_{\text{decoder reconstruction error}} - \underbrace{D_{\text{KL}}(q_\theta(\mathbf{z}|\hat{\mathbf{x}}^{(n)}) || p(\mathbf{z}))}_{\text{encoder regularization term}}.$$

In the above, we used a Euclidean Normal distribution as  $q$  and  $p$  only as an example. See Figure 4.1 for an illustration of the above algorithm. To get an estimate of the marginal log-likelihood  $\log p(\hat{\mathbf{x}})$  for evaluation, we repeat steps 4–5 in the process  $L$  times and then compute  $\text{ELBO}_L(q)$  instead (denoted as LL).

For inference, we skip steps 1–3, sample from the prior  $\mathbf{z} \sim p(\mathbf{z})$  instead of from the posterior in step 4, and decode exactly like in step 5.

### 4.3.2 Riemannian manifolds as latent spaces

To be able to learn latent representations in Riemannian manifolds instead of in  $\mathbb{R}^d$  as above, we only need to change step 2 of the VAE forward pass, and the choice of prior and posterior distributions.

The prior and posterior have to be chosen depending on the chosen manifold and are essentially treated as hyperparameters of our VAE. All the distributions we have mentioned so far are reparameterizable. Since we have defined the Wrapped Normal family of distributions for all spaces, we can use  $\mathcal{WN}(\boldsymbol{\mu}_0, \boldsymbol{\sigma}^2 \mathbf{I})$  as the posterior family, and  $\mathcal{WN}(\boldsymbol{\mu}_0, \mathbf{I})$  as the prior distribution. The actual forms of the distributions depend on the chosen constant curvature space type.

In experiments, we sometimes use vMF( $\boldsymbol{\mu}, \kappa$ ) for the hypersphere  $\mathbb{S}_K^n$  (or a backprojected variant of vMF for  $\mathbb{D}_K^n$ ) with the associated hyperspherical uniform distribution  $U(\mathbb{S}_K^n)$  as a prior, or the Riemannian Normal distribution  $\mathcal{RN}(\boldsymbol{\mu}, \sigma^2)$  and the associated prior  $\mathcal{RN}\boldsymbol{\mu}_0, 1$  for the Poincare ball  $\mathbb{P}_K^n$ , which are the corresponding maximum entropy distributions in those spaces.

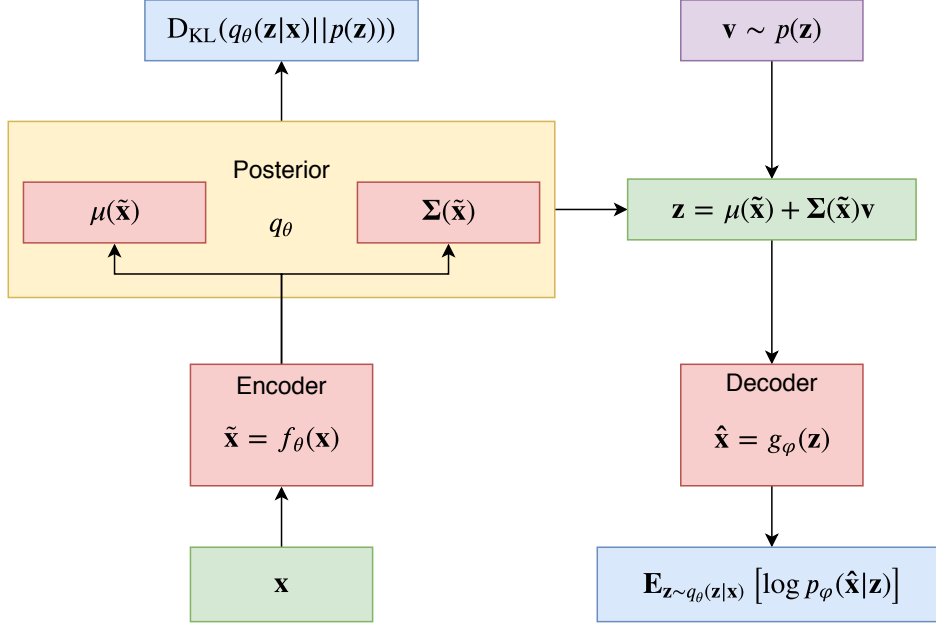


Figure 4.1: Illustration of a VAE model. Green boxes represent sample and latent space representations, red boxes represent parameterized functions (neural networks), the purple box represents a reparameterized source of randomness, yellow denotes a probability distribution, and blue boxes are loss terms.

For most distributions, the parametrization is the following:

$$\begin{aligned}\boldsymbol{\mu}(\tilde{\boldsymbol{x}}^{(n)}) &= \exp_{\boldsymbol{\mu}_0}^K(h_{\boldsymbol{\mu}}(\tilde{\boldsymbol{x}}^{(n)})) \\ \boldsymbol{\sigma}^2(\tilde{\boldsymbol{x}}^{(n)}) &= \exp\left(1 + \log(h_{\boldsymbol{\sigma}}(\tilde{\boldsymbol{x}}^{(n)}))\right) + \varepsilon \\ \boldsymbol{\Sigma}(\tilde{\boldsymbol{x}}^{(n)}) &= \boldsymbol{\sigma}^2 \boldsymbol{I},\end{aligned}$$

where  $\varepsilon = 10^{-5}$  is a small constant to prevent complete collapse. The exceptions to the above parametrization are the von Mises-Fischer distribution and the Riemannian Normal distribution, where the scale parameter is scalar, and we adjust  $\boldsymbol{\sigma}$  and the dimensions of  $h_{\boldsymbol{\sigma}}$  accordingly (this is important when comparing latent space degrees of freedom). Note that we can also use a scalar parameterization with the Wrapped Normal and Gaussian Normal distributions.

It is important to mention that all of the parameters of our model live in Euclidean space, not on the manifolds. We just implicitly parametrize the distributions in latent space using the exponential map at the origin of the space  $\exp_{\boldsymbol{\mu}_0}^K$ .

### 4.3.3 Latent space as a product of constant curvature spaces

Given Remark 2.1, dividing the latent space of a VAE into different “components” does not make topological sense. However, in spaces of non-zero constant curvature, the remark does not hold. Therefore, we need a way to define VAEs where  $\mathcal{Z}$  is a Cartesian product of different subspaces, which are each one of the defined models we have introduced previously.

Using the geometrical properties from Section 2.6, we have shown that all operations defined we have defined in our manifolds “generalize” to products of such spaces. Therefore, we simply define our latent space as some product of  $I$  spaces

$$\mathcal{Z} = \prod_{i=1}^I \mathcal{Z}_{K_i}^{n_i},$$

where each component  $\mathcal{Z}_{K_i}^{n_i} \in \{\mathbb{S}_{K_i}^{n_i}, \mathbb{D}_{K_i}^{n_i}, \mathbb{E}^{n_i}, \mathbb{H}_{K_i}^{n_i}, \mathbb{P}_{K_i}^{n_i}\}$  is one of our defined models. Then, we do steps 2–4 of the VAE forward pass per-component of the latent space, and concatenate all samples  $\mathbf{z}^{(n)} = \bigodot_{i=1}^I \mathbf{z}_{(i)}^{(n)}$ , where  $\mathbf{z}_{(i)}^{(n)} \sim q_{\theta}^{(i)}(\mathbf{z}_{(i)}|\mathbf{x}^{(n)})$  from step 3. Every component has its own  $\boldsymbol{\mu}(\tilde{\mathbf{x}}^{(n)})$  and  $\boldsymbol{\Sigma}(\tilde{\mathbf{x}}^{(n)})$  functions, with the appropriate dimensionality-adjusting maps  $h_{\boldsymbol{\mu}}$  and  $h_{\boldsymbol{\Sigma}}$  inside of them.

### 4.3.4 Overview of properties

To summarize, we have defined a fully specified VAE model with the latent space lying on a product of Riemannian manifolds of constant curvature. The motivation for this was that the latent space would, therefore, have different curvature in different “components”, which should make it more suitable for uncovering an even more compact hidden structure in a given dataset than VAEs with a single constant curvature manifold as a latent space. What is more, we can take derivatives with respect to the curvature in every component of our space, and hence attempt to learn it from data as well.

Additionally, if we define the latent space to only consist of Euclidean, Poincaré ball, and projected hypersphere models with the Wrapped Normal distributions, the resulting VAE will be a generalization of the Euclidean VAE (Kingma and Welling, 2014) as well as its spherical (Davidson et al., 2018; Xu and Durrett, 2018) and hyperbolic (Mathieu et al., 2019; Nagano et al., 2019) variants.

Finally, our VAE shows good asymptotic behavior as the curvatures of our latent space components  $K_i$  go to 0 — essentially, the VAE degenerates into the Euclidean VAE (Kingma and Welling, 2014).

## Chapter 5

---

# Learning curvature

---

We have already seen approaches to learning VAEs in products of spaces of constant curvature. However, we can also change the curvature constant in each of the spaces during training. The individual spaces will still have constant curvature at each point, we just allow changing the constant in between training steps. To differentiate between these training procedures, we will call them *fixed* curvature and *learnable* curvature VAEs respectively.

The motivation behind changing curvature of non-Euclidean constant curvature spaces might not be clear, since it is apparent from the definition of the distance function in the hypersphere and hyperboloid

$$d(\mathbf{x}, \mathbf{y}) = R \cdot \theta_{\mathbf{x}, \mathbf{y}},$$

that the distances between two points that stay at the same angle only get rescaled when changing the radius of the space. Same applies for the Poincaré ball and the projected spherical space since they are stereographic conformal projections of the hyperboloid and the hypersphere, hence they preserve angles between points.

The motivation for learning curvature in our model is that the decoder does not only depend on pairwise distances, but rather on the specific positions of points in the space. It can be conjectured that the KL term of the ELBO indeed is only “rescaled” when we change the curvature, however, the reconstruction process is influenced in non-trivial ways. Since that is hard to quantify and prove, we devise a series of practical experiments to show overall model performance is enhanced when learning curvature.

### 5.1 Fixed curvature VAEs

In *fixed* curvature VAEs, all component latent spaces have a fixed curvature that is selected a priori and fixed for the whole duration of the training procedure, as well as during evaluation. For Euclidean components it is 0, for



positively or negatively curved spaces any positive or negative number can be chosen, respectively. For stability reasons, we select curvature values from the range  $[0.25, 1.0]$ , which corresponds to radii in  $[1.0, 2.0]$ . The exact curvature value does not have a too significant impact on performance when training a fixed curvature VAE, as motivated by the distance rescaling remark above, although some constant might perform better. In the following, we refer to fixed curvature components with a constant subscript, e.g.  $\mathbb{H}_1^n$ .

## 5.2 Learnable curvature VAEs

In all non-Euclidean manifolds, we can differentiate the ELBO with respect to the curvature  $K$ . This enables us to treat  $K$  as a parameter of the model and learn it using gradient-based optimization, exactly like we learn the encoder/decoder maps in a VAE.

There are several problems with this trivial approach. First of all, we have no gradient with respect to  $K$  in a Euclidean model, as the curvature is 0. Secondly, learning curvature directly is badly conditioned — we are trying to learn one scalar parameter that influences the resulting decoder and hence the ELBO quite heavily. It comes up in the PDF of the latent space distribution, in the exponential mapping of the mean to the manifold, and in many more places in the model. Hence, parts of the gradient for  $K$  come from the reconstruction term and parts from the KL term. Lastly, when doing a curvature update, we should adjust all the points so that their pairwise distances remain the same. However, this is very hard to do since the points on our manifold are parametrized by a neural network, therefore we resort to skipping this step, and learning curvature “jointly” with the representations.

Empirically, we have found that Stochastic Gradient Descent works well to optimize the radius of a component. We constrain the radius to be strictly positive in all non-Euclidean spaces by applying a ReLU activation function before we use it in operations

$$\text{relu}(R) = \max(0, R).$$

To increase stability, we pre-train the model by first fixing the curvatures of components for a few epochs and only later letting the model adjust the radii.

## 5.3 Universal curvature VAEs

In the previous two sections, we have explained how to train VAEs in latent spaces that are products of constant curvature spaces. Additionally, we have shown how to adjust the curvature of the component spaces during training. However, we must still a priori select the “partitioning” of our latent space — we must choose the number of components and for each of them select the

dimension and at least the sign of the curvature of that component. We call this the *signature* of the latent space. Hence, we have improved the flexibility of our latent space geometry by introducing a product of spaces, but have introduced the problem of signature estimation.

The simplest approach would be to just try all combinations and compare the results on the specific dataset. This procedure would most likely be optimal, but does not scale. With a latent space of dimension  $n$ , three types of components (spherical, Euclidean, and hyperbolic), and a minimum component dimensionality of 2, we would have to run

$$\sum_{m=1}^{\lfloor n/2 \rfloor} \sum_{k_1+k_2+\dots+k_m=n} \binom{n}{k_1, k_2, \dots, k_m} 3^m = \sum_{m=1}^{\lfloor n/2 \rfloor} 3^m m^n$$

experiments, possibly multiple times to obtain confidence bounds, which is infeasible in practice.

To eliminate this, we propose a method that approximates the resulting model of the proposed grid search. We partition our space into 2-dimensional components (if the number of dimensions is odd, one component will have 3 dimensions). We initialize all of them as Euclidean components and train for half the number of maximal epochs we are allowed. Then, we split the components into 3 approximately equal-sized groups and make one group into hyperbolic components, one into spherical, and the last remains Euclidean. We do this by changing the curvature of a component by a very small  $\varepsilon$  — a reasonable choice would be around  $10^{-2}$  to  $10^{-5}$ . We then train just the encoder/decoder maps for a few (e.g. 10) epochs to stabilize the representations after changing the curvatures. Finally, we allow learning the curvatures of all non-Euclidean components and train for the rest of the allowed epochs.

Note that the method is not completely general, as it never uses components bigger than dimension 2, but the approximation has empirically performed satisfactorily. The approach is general and we can select any dimensions of components, but that re-introduces some of the complexity of doing a grid search, as described above.

We also do not constrain the curvature of the components to a specific sign in the last stage of training. Therefore, components may change their type of space from a positively curved to a negatively curved one, or vice-versa. Even though the gradient is not defined at  $K = 0$ , we can approximate it from both sides using a Taylor expansion at 0. In practice, that is not necessary, as it is very unlikely that the value of  $K$  will be set to exact  $K = 0$  when optimized using gradient descent.

Because of the divergence of points as  $K \rightarrow 0$  for the hyperboloid and hypersphere (Remark A.3 and A.19, Figure 5.1) and the equal ambient space

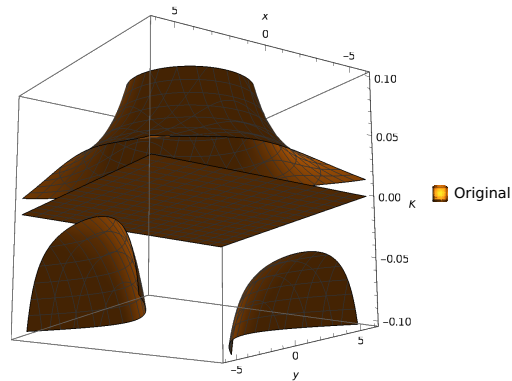


Figure 5.1: Visualization of a one-dimensional hypersphere and hyperboloid around  $K = 0$ .

dimensionality, the universal curvature VAE assumes the positively curved space is a projected hypersphere  $\mathbb{D}$  and the negatively curved space is the Poincaré ball  $\mathbb{P}$ . In all experiments, this universal approach is denoted as  $\mathbb{U}^n$ .

## Chapter 6

---

# Experiments

---

In this chapter, we briefly review work related to ours. Then, we compare our model to other current VAE approaches on the synthetic tree dataset of Mathieu et al. (2019), on image reconstruction of dynamically binarized MNIST (LeCun, 1998) and dynamically binarized Omniglot (Lake et al., 2015), and image reconstruction of natural images of CIFAR-10 (Krizhevsky, 2009). Our PyTorch (Paszke et al., 2017) code will be made available on GitHub<sup>1</sup>. Additional implementation details, experimental results, tables, and plots are appended in Appendix D.

## 6.1 Related work

### 6.1.1 Universal models of geometry

Duality between spaces of constant curvature (Section 2.4) was first noticed by Lambert (1770), and later gave rise to various theorems that have the same or similar forms in all three geometries, like the law of sines (Bolyai, 1832)

$$\frac{\sin A}{p_K(a)} = \frac{\sin B}{p_K(b)} = \frac{\sin C}{p_K(c)},$$

where  $p_K(r) = 2\pi \sin_K(r)$  denotes the circumference of a circle of radius  $r$  in a space of constant curvature  $K$ , and

$$\sin_K(x) = x - \frac{Kx^3}{3!} + \frac{K^2x^5}{5!} - \dots = \sum_{i=0}^{\infty} \frac{(-1)^i K^i x^{2i+1}}{(2i+1)!}.$$

Other unified formulas for the law of cosines, or recently, a unified Pythagorean theorem has also been proposed (Foote, 2017):

$$A(c) = A(a) + A(b) - \frac{K}{2\pi} A(a)A(b),$$

---

<sup>1</sup><https://github.com/oskopek/mvae>

where  $A(r)$  is the area of a circle of radius  $r$  in a space of constant curvature  $K$ . Unfortunately, in this formulation  $A(r)$  still depends on the sign of  $K$  w.r.t. the choice of trigonometric functions in its definition.

There also exist approaches defining a universal geometry of constant curvature spaces. Li et al. (2001, Chapter 4) define a unified model of all three geometries using the null cone (light cone) of a Minkowski space. The term “Minkowski space” comes from special relativity and is usually denoted as  $\mathbb{R}^{1,n}$ , similar to the ambient space of what we defined as  $\mathbb{H}^n$ , with the Lorentz scalar product  $\langle \cdot, \cdot \rangle_{\mathcal{L}}$ . The hyperboloid  $\mathbb{H}^n$  corresponds to the positive (upper, future) null cone of  $\mathbb{R}^{1,n}$ . All the other models can be defined in this space using the appropriate stereographic projections and pulling back the metric onto the specific sub-manifold. Unfortunately, we found the formalism not useful for our application, apart from providing a very interesting theoretical connection among the models.

### 6.1.2 Concurrent VAE approaches

The variational autoencoder was originally proposed in Kingma and Welling (2014) and concurrently in Rezende et al. (2014). One of the most common improvements on the VAE in practice is the choice of the encoder and decoder maps, ranging from linear parametrizations of the posterior to feed-forward neural networks, convolutional neural networks, etc. For different data domains, extensions like the GraphVAE (Simonovsky and Komodakis, 2018) using graph convolutional neural networks for the encoder and decoder were proposed.

The basic VAE framework was mostly improved upon by using autoregressive flows (Chen et al., 2014) or small changes to the ELBO loss function (Burda et al., 2016; Matthey et al., 2017). An important work in this area is  $\beta$ -VAE, which adds a simple scalar multiplicative constant to the KL divergence term in the ELBO, and has shown to improve both sample quality and (if  $\beta > 1$ ) disentanglement of different dimensions in the latent representation. For more information on disentanglement, see Locatello et al. (2018).

### 6.1.3 Geometric deep learning

One of the emerging trends in deep learning has been to leverage non-Euclidean geometry<sup>2</sup> to learn representations, originally emerging from knowledge-base and graph representation learning (Bronstein et al., 2017).

Recently, several approaches to learning representations in Euclidean spaces have been generalized to non-Euclidean spaces (Dhingra et al., 2018; Ganea et al., 2018b; Nickel and Kiela, 2017). Since then, this research direction has

<sup>2</sup><http://geometricdeeplearning.com/>

grown immensely and accumulated more approaches, mostly for hyperbolic spaces, like Ganea et al. (2018a); Law et al. (2019); Nickel and Kiela (2018); Tifrea et al. (2019). Similarly, spherical spaces have also been leveraged for learning non-Euclidean representations (Batmanghelich et al., 2016; Wilson and Hancock, 2010).

To be able to learn representations in these spaces, new Riemannian optimization methods were required as well (Bécigneul and Ganea, 2019; Bonnabel, 2013; Wilson and Leimeister, 2018).

The generalization to products of constant curvature Riemannian manifolds is only natural and has been proposed by Gu et al. (2019). They evaluated their approach by directly optimizing a distance-based loss function using Riemannian optimization in products of spaces on graph reconstruction and word analogy tasks, in both cases reaping the benefits of non-Euclidean geometry, especially when learning lower-dimensional representations.

#### 6.1.4 Geometry in VAEs

One of the first attempts at leveraging geometry in VAEs was Arvanitidis et al. (2018). They examine how a Euclidean VAE benefits both in sample quality and latent representation distribution quality when employing a non-Euclidean Riemannian metric in the latent space using kernel transformations.

Hence, a potential improvement area of VAEs could be the choice of the posterior family and prior distribution. However, the Gaussian (Normal) distribution works very well in practice, as it is the maximum entropy probability distribution with a known variance, and imposes no constraints on higher-order moments (skewness, kurtosis, etc.) of the distribution. Recently, non-Euclidean geometry has been used in learning variational autoencoders as well. Generalizing Normal distributions to these spaces is in general non-trivial (see Section 3.2).

Two similar approaches, Davidson et al. (2018) and Xu and Durrett (2018), used the von Mises-Fischer distribution on the unit hypersphere to generalize VAEs to spherical spaces. The von Mises-Fischer distribution is again a maximum entropy probability distribution on the unit hypersphere, but only has a spherical covariance parameter, which makes it less general than a Gaussian distribution.

Conversely, two approaches, Mathieu et al. (2019) and Nagano et al. (2019), have generalized VAEs to hyperbolic spaces — both the Poincaré ball and the hyperboloid, respectively. They both adopt a non-maximum entropy probability distribution called the Wrapped Normal. Additionally, Mathieu et al. (2019) also derive the Riemannian Normal, which is a maximum entropy distribution on the Poincaré disk, but in practice performs similar to the Wrapped Normal, especially in higher dimensions.

Our approach generalizes on the afore-mentioned geometrical VAE work, by employing a “products of spaces” approach similar to Gu et al. (2019) and unifying the different approaches into a single framework for all spaces of constant curvature.

## 6.2 Experimental setup

For our experiments, we use four datasets:

1. Branching diffusion process (Mathieu et al., 2019, BDP) — A synthetic tree-like dataset with injected noise. Hence, a priori, hyperbolic VAEs should have an advantage.
2. Dynamically-binarized MNIST digits (LeCun, 1998) — MNIST is a dataset of 60 000 training samples and 10 000 testing samples, all of which consist of grayscale  $28 \times 28$  pixel handwritten digits 0–9. We binarize the images similarly to Burda et al. (2016); Salakhutdinov and Murray (2008). The training set is binarized dynamically (uniformly sampled threshold per-sample:  $\text{bin}(\mathbf{x}) \in \{0, 1\}^D = \mathbf{x} > \mathcal{U}[0, 1], \mathbf{x} \in \mathbb{R}^D \subseteq [0, 1]^D$ ), and the evaluation set is done with a fixed binarization ( $\mathbf{x} > 0.5$ ).
3. Dynamically-binarized Omniglot characters (Lake et al., 2015) — Omniglot is a dataset of thousands of handwritten characters in 50 different alphabets. Each image is grayscale  $150 \times 150$  pixels, drawn by 20 different people using Amazon’s Mechanical Turk. Before training and evaluating, we bilinearly downsample the image to  $28 \times 28$  pixels and invert the colors so that the background is black as in MNIST. Binarization is the same as for MNIST.
4. CIFAR-10 (Krizhevsky, 2009) — CIFAR-10 is a dataset of thousands of real-life  $32 \times 32$  pixel RGB color images in 10 classes (airplane, automobile, bird, cat, deer, dog, frog, horse, ship, and truck).

All models in all datasets are trained with early stopping on training ELBO with a lookahead of 50 epochs and a warmup of 100 epochs (Bowman et al., 2016). All BDP models are trained for a 1000 epochs, MNIST and Omniglot models are trained for 300 epochs, and CIFAR for 200 epochs.

We compare models with a given latent space dimension using marginal log-likelihood with importance sampling (Burda et al., 2016) with 500 samples, except for CIFAR, which uses 50 due to memory constraints. Log-likelihood is by definition a negative number, and larger values are better. In all tables, we denote it as LL. We run all experiments at least 3 times to get an estimate of variance when using different initial values.

In all the BDP, MNIST, and Omniglot experiments below, we use a simple feed-forward encoder and decoder architecture consisting of a single dense

layer with 400 neurons and element-wise ReLU activation

$$\text{relu}(x) = \max(0, x).$$

Since all the VAE parameters  $\{\theta, \varphi\}$  live in Euclidean manifolds, we can use standard gradient-based optimization methods. Specifically, we use the Adam (Kingma and Ba, 2015) optimizer with a learning rate of  $10^{-3}$  and standard settings for  $\beta_1 = 0.9$ ,  $\beta_2 = 0.999$ , and  $\varepsilon = 10^{-8}$ .

For the CIFAR encoder map, we use a simple convolutional neural networks with three convolutional layers with 64, 128, and 512 channels respectively. For the decoder map, we first use a dense layer of dimension 2048, and then three consecutive transposed convolutional layers with 256, 64, and 3 channels. All layers are followed by a ReLU activation function, except for the last one. All convolutions have  $4 \times 4$  kernels with stride 2, and padding of size 1. Note that this architecture is known to not produce state of the art images (Chen et al., 2014), but we use it due to limited computation time.

The first 10 epochs for all models are trained with a fixed radius (and hence curvature) of  $11 - e$  where  $e$  denotes the epoch number, starting at 0. For example, in epoch 9, the radius is therefore set to 2 (curvature  $\pm 0.25$ ). This corresponds to a burn-in period similarly to Nickel and Kiela (2017). For learnable curvature approaches we then use Stochastic Gradient Descent with learning rate  $10^{-4}$  and let the optimizers adjust the value freely, for fixed curvature approaches it stays at the last burn-in value.

All our models use the Wrapped Normal distribution, or equivalently Euclidean Normal in Euclidean components, unless specified otherwise. We run a few experiments with the Riemannian Normal and the von Mises-Fischer distribution as well, clearly marked in plots and tables.

All fixed curvature components are denoted with a  $\mathcal{M}_1$  or  $\mathcal{M}_{-1}$  subscript, or with “-fixed” in plots. Learnable curvature components do not have a subscript. This notation is omitted for Euclidean components, as they are always fixed. For example,  $\mathbb{H}_{-1}^n$  is a fixed curvature hyperboloid of dimension  $n$  and  $\mathbb{H}^n$  is the equivalent but with a learnable curvature. For an overview of the different components, see Table 6.1.

As baselines, we train VAEs with spaces that have a fixed constant curvature, i.e. assume a single Riemannian manifold (potentially a product of them) as their latent space. It is apparent that our models with a single component, like  $\mathbb{S}_1^n$  correspond to Davidson et al. (2018) and Xu and Durrett (2018),  $\mathbb{H}_{-1}^n$  is equivalent to the Hyperbolic VAE of Nagano et al. (2019),  $\mathbb{P}_{-c}^n$  corresponds to the  $\mathcal{P}^c$ -VAE of Mathieu et al. (2019), and  $\mathbb{E}^n$  is equivalent to the Euclidean VAE of Kingma and Welling (2014).

In the following, we present a selection of all the obtained results. For more information and plots, see Appendix D. Do note that bold font numbers simply



### 6.3. Spherical covariance matrix parametrization

Curvature	Notation	Ambient dimension	Description
$K > 0$	$\mathbb{S}^n$	$n + 1$	$n$ -dim. hypersphere
	$\mathbb{D}^n$	$n$	$n$ -dim. projected hypersphere
$K = 0$	$\mathbb{E}^n$	$n$	$n$ -dim. Euclidean space
$K < 0$	$\mathbb{H}^n$	$n + 1$	$n$ -dim. hyperboloid
	$\mathbb{P}^n$	$n$	$n$ -dim. Poincaré disk

Table 6.1: Brief overview of components and their properties.

represent values that are particularly interesting, not necessarily best performers.

### 6.3 Spherical covariance matrix parametrization

Since the Riemannian Normal and the von Mises-Fischer distribution only have a spherical covariance matrix, i.e. a single scalar variance parameter per component, we briefly evaluate all our approaches with a spherical covariance parametrization. The complete results can be found in Section D.2 of Appendix D.

**Binary diffusion process** For the BDP dataset and latent dimension 6 (Table 6.2), we observe that all VAEs that only use the von Mises-Fischer distribution perform worse than a Wrapped Normal. However, when a VMF spherical component was paired with other component types, it performed better than if a Wrapped Normal spherical component was used instead.

Riemannian Normal VAEs did very well on their own — the fixed Poincaré VAE ( $\mathcal{RN} \mathbb{P}_{-1}^2$ )<sup>3</sup> obtains the best score. It did not fare as well when we tried to learn curvature with it, however. Another thing to consider is that we, unfortunately, were not able to run  $\mathcal{RN} \mathbb{P}^6$  due to issues with the rejection sampling code not working well.

An interesting observation is that all single-component VAEs  $\mathcal{M}^6$  performed worse than product VAEs ( $\mathcal{M}^2$ )<sup>3</sup> when curvature was learned, across all component types. Our universal curvature VAE ( $\mathbb{U}^2$ )<sup>3</sup> managed to get better results than all other approaches except for the Riemannian Normal baseline, but it is within the margin of error of some other models. It also outperformed its single-component variant  $\mathbb{U}^6$ . However, we did not find that it converged to specific curvature values, only that they were in the approximate range of  $(-0.1, +0.1)$ .

As expected, spherical models did worse than hyperbolic ones in general.

### 6.3. Spherical covariance matrix parametrization

Model	LL	ELBO	BCE	KL
$\mathbb{S}_1^6$	$-55.81 \pm 0.35$	$-56.57 \pm 0.44$	$51.16 \pm 0.78$	$5.41 \pm 0.42$
$\mathbb{D}_1^6$	$-55.78 \pm 0.07$	$-56.38 \pm 0.06$	$50.85 \pm 0.20$	$5.53 \pm 0.24$
$\mathbb{E}^6$	$-56.28 \pm 0.56$	$-56.99 \pm 0.59$	$51.58 \pm 0.69$	$5.41 \pm 0.29$
$(\mathbb{H}_{-1}^2)^3$	$-56.08 \pm 0.52$	$-56.80 \pm 0.54$	$50.94 \pm 0.38$	$5.86 \pm 0.25$
$\mathbb{H}_{-1}^6$	$-56.18 \pm 0.32$	$-57.10 \pm 0.21$	$51.48 \pm 0.47$	$5.62 \pm 0.31$
$(\mathbb{P}_{-1}^2)^3$	$-55.98 \pm 0.62$	$-56.49 \pm 0.62$	$50.96 \pm 0.61$	$5.52 \pm 0.31$
$\mathbb{P}_{-1}^6$	$-56.74 \pm 0.55$	$-57.61 \pm 0.74$	$52.01 \pm 0.71$	$5.60 \pm 0.24$
$(\mathcal{RN} \mathbb{P}_{-1}^2)^3$	<b><math>-54.99 \pm 0.12</math></b>	$-55.90 \pm 0.13$	$52.42 \pm 0.71$	$3.48 \pm 0.60$
$(\mathbb{S}^2)^3$	$-56.05 \pm 0.21$	$-56.69 \pm 0.36$	$51.07 \pm 0.21$	$5.61 \pm 0.22$
$(\mathbb{D}^2)^3$	$-56.06 \pm 0.36$	$-56.69 \pm 0.54$	$50.95 \pm 0.40$	$5.74 \pm 0.17$
$(\mathbb{H}^2)^3$	<b><math>-55.80 \pm 0.32</math></b>	$-56.72 \pm 0.16$	$51.14 \pm 0.39$	$5.58 \pm 0.28$
$(\mathbb{P}^2)^3$	$-56.29 \pm 0.05$	$-57.11 \pm 0.22$	$51.41 \pm 0.19$	$5.69 \pm 0.30$
$(\mathcal{RN} \mathbb{P}^2)^3$	$-56.25 \pm 0.56$	$-57.26 \pm 0.45$	$53.16 \pm 1.07$	$4.11 \pm 0.64$
$\mathbb{D}^2 \times \mathbb{E}^2 \times \mathbb{P}^2$	$-55.87 \pm 0.22$	$-56.35 \pm 0.22$	$50.67 \pm 0.57$	$5.69 \pm 0.43$
$\mathbb{E}^2 \times \mathbb{H}^2 \times \mathbb{S}^2$	$-55.92 \pm 0.42$	$-56.54 \pm 0.45$	$51.13 \pm 0.74$	$5.41 \pm 0.40$
$\mathbb{E}^2 \times \mathbb{H}^2 \times (\text{vMF } \mathbb{S}^2)$	$-55.82 \pm 0.43$	$-56.32 \pm 0.47$	$51.10 \pm 0.67$	$5.21 \pm 0.20$
$\mathbb{E}^2 \times \mathbb{H}_{-1}^2 \times (\text{vMF } \mathbb{S}_1^2)$	<b><math>-55.77 \pm 0.51</math></b>	$-56.34 \pm 0.65$	$51.33 \pm 0.57$	$5.01 \pm 0.17$
$(\mathbb{U}^2)^3$	<b><math>-55.56 \pm 0.15</math></b>	$-56.05 \pm 0.32$	$50.68 \pm 0.23$	$5.37 \pm 0.10$

Table 6.2: Summary of results (mean and standard deviation), latent space dimension 6, spherical covariance, on the BDP dataset.

**Dynamically-binarized MNIST reconstruction** We also tested our approach on binarized MNIST (Table 6.3). With spherical covariance, we noticed that VMF again rather under-performed Wrapped Normal, except when it was part of a product like  $\mathbb{E}^2 \times \mathbb{H}^2 \times (\text{vMF } \mathbb{S}^2)$ .

The projected spherical space had big problems handling MNIST, especially on its own. When paired with another Euclidean and a Riemannian Normal Poincaré disk component, it performed well, but that might be because the  $\mathcal{RN} \mathbb{P}_{-1}$  component achieved best results across the board on MNIST, both when curvature was learned and especially if it was fixed. As we will see later, it achieved the best results even when compared to diagonal covariance matrix VAEs on 6-dimensional MNIST.

Several approaches seem to be better than the Euclidean baseline. That applies mainly to the above mentioned Riemannian Normal Poincaré ball components, but also  $\mathbb{S}^6$  both with Wrapped Normal and VMF, as well as most product space VAEs with different curvatures (third section of the table). Our  $(\mathbb{U}^2)^3$  performed similarly to the Euclidean baseline VAE.

## 6.4. Diagonal covariance matrix parametrization

Model	LL	ELBO	BCE	KL
$\mathbb{S}_1^6$	-96.71±0.17	-101.55±0.30	86.90±0.30	14.65±0.10
vMF $\mathbb{S}_1^6$	-97.03±0.14	-102.12±0.26	87.42±0.28	14.69±0.03
$\mathbb{D}_1^6$	-98.21±0.23	-103.02±0.14	88.44±0.05	14.58±0.11
$\mathbb{E}^6$	-97.16±0.15	-101.67±0.14	87.17±0.26	14.50±0.20
$\mathbb{H}_{-1}^6$	-97.10±0.44	-101.89±0.33	87.32±0.22	14.56±0.20
$(\mathbb{P}_{-1}^2)^3$	-97.56±0.04	-102.33±0.22	87.93±0.32	14.40±0.10
$(\mathcal{RN} \mathbb{P}_{-1}^2)^3$	<b>-92.54±0.19</b>	-97.19±0.21	88.42±0.20	8.76±0.04
$(\mathbb{S}^2)^3$	-96.46±0.12	-101.30±0.17	86.79±0.25	14.51±0.09
$\mathbb{S}^6$	-96.72±0.15	-101.39±0.16	86.69±0.15	14.70±0.13
vMF $\mathbb{S}^6$	-96.72±0.18	-101.55±0.21	86.82±0.23	14.73±0.02
$\mathbb{D}^6$	-97.72±0.15	-102.31±0.16	87.70±0.22	14.61±0.06
$(\mathbb{H}^2)^3$	-97.37±0.13	-102.07±0.24	87.56±0.30	14.51±0.11
$(\mathcal{RN} \mathbb{P}^2)^3$	<b>-94.16±0.68</b>	-98.65±0.66	89.27±0.79	9.38±0.15
$\mathbb{D}^2 \times \mathbb{E}^2 \times \mathbb{P}^2$	-97.48±0.18	-102.22±0.29	87.85±0.17	14.37±0.13
$\mathbb{D}^2 \times \mathbb{E}^2 \times (\mathcal{RN} \mathbb{P}^2)$	-96.43±0.47	-101.31±0.51	88.82±0.50	12.50±0.03
$\mathbb{D}_1^2 \times \mathbb{E}^2 \times (\mathcal{RN} \mathbb{P}_{-1}^2)$	<b>-96.18±0.21</b>	-100.91±0.31	88.58±0.47	12.33±0.19
$\mathbb{E}^2 \times \mathbb{H}^2 \times \mathbb{S}^2$	-96.80±0.20	-101.60±0.33	87.13±0.19	14.47±0.17
$\mathbb{E}^2 \times \mathbb{H}_{-1}^2 \times \mathbb{S}_1^2$	-96.76±0.09	-101.48±0.13	86.99±0.17	14.49±0.05
$\mathbb{E}^2 \times \mathbb{H}^2 \times (\text{vMF } \mathbb{S}^2)$	-96.56±0.27	-101.49±0.28	86.58±0.36	14.91±0.14
$(\mathbb{U}^2)^3$	<b>-97.12±0.04</b>	-101.68±0.06	87.13±0.14	14.55±0.16

Table 6.3: Summary of selected models (mean and standard deviation), latent space dimension 6, spherical covariance, on the MNIST dataset.

## 6.4 Diagonal covariance matrix parametrization

All the following models are trained with a diagonal covariance matrix, i.e. a vector of variance parameters per component. This corresponds to the most common covariance matrix parametrization of VAEs (Kingma and Welling, 2014).

### 6.4.1 Dynamically-binarized MNIST reconstruction

The complete results can be found in Section D.3.1 of Appendix D.

First, we look at latent dimension 6 (Table 6.4). These models can directly be compared to the spherical covariance MNIST 6 models, even though they have more parameters (more covariance parameters). Interestingly, the Riemannian Normal Poincaré ball VAE is still the best performer. The Euclidean baseline VAE achieved better results than its spherical covariance counterpart. Overall, the best result is achieved by the single-component spherical model, with

## 6.4. Diagonal covariance matrix parametrization

Model	LL	ELBO	BCE	KL
$\mathbb{S}_1^6$	$-96.51 \pm 0.09$	$-101.29 \pm 0.18$	$86.71 \pm 0.20$	$14.58 \pm 0.13$
$\mathbb{D}_1^6$	$-97.89 \pm 0.10$	$-102.65 \pm 0.10$	$88.39 \pm 0.16$	$14.26 \pm 0.08$
$\mathbb{E}^6$	$-96.88 \pm 0.16$	$-101.36 \pm 0.08$	$86.90 \pm 0.14$	$14.46 \pm 0.07$
$\mathbb{H}_{-1}^6$	$-97.38 \pm 0.73$	$-102.22 \pm 0.95$	$87.75 \pm 0.59$	$14.47 \pm 0.37$
$\mathbb{P}_{-1}^6$	$-97.33 \pm 0.15$	$-102.02 \pm 0.35$	$87.71 \pm 0.36$	$14.31 \pm 0.04$
$\mathbb{S}^6$	$-96.44 \pm 0.20$	$-101.18 \pm 0.36$	$86.74 \pm 0.38$	$14.44 \pm 0.05$
$\mathbb{D}^6$	$-97.53 \pm 0.22$	$-102.31 \pm 0.38$	$87.97 \pm 0.37$	$14.34 \pm 0.08$
$(\mathbb{H}^2)^3$	$-96.86 \pm 0.31$	$-101.61 \pm 0.30$	$87.13 \pm 0.30$	$14.48 \pm 0.08$
$\mathbb{H}^6$	$-96.90 \pm 0.26$	$-101.48 \pm 0.35$	$87.18 \pm 0.48$	$14.30 \pm 0.15$
$\mathbb{P}^6$	$-97.26 \pm 0.16$	$-102.00 \pm 0.17$	$87.58 \pm 0.16$	$14.42 \pm 0.08$
$\mathbb{D}^2 \times \mathbb{E}^2 \times \mathbb{P}^2$	$-97.37 \pm 0.14$	$-102.12 \pm 0.19$	$87.78 \pm 0.23$	$14.34 \pm 0.12$
$\mathbb{D}_1^2 \times \mathbb{E}^2 \times \mathbb{P}_{-1}^2$	$-97.29 \pm 0.16$	$-101.86 \pm 0.16$	$87.54 \pm 0.17$	$14.32 \pm 0.04$
$\mathbb{E}^2 \times \mathbb{H}^2 \times \mathbb{S}^2$	$-96.71 \pm 0.19$	$-101.34 \pm 0.16$	$86.91 \pm 0.17$	$14.43 \pm 0.06$
$\mathbb{E}^2 \times \mathbb{H}_{-1}^2 \times \mathbb{S}_1^2$	$-96.66 \pm 0.27$	$-101.46 \pm 0.44$	$87.02 \pm 0.38$	$14.44 \pm 0.08$
$(\mathbb{U}^2)^3$	$-97.06 \pm 0.13$	$-101.66 \pm 0.19$	$87.22 \pm 0.12$	$14.44 \pm 0.07$
$\mathbb{U}^6$	$-96.90 \pm 0.10$	$-101.68 \pm 0.07$	$87.27 \pm 0.11$	$14.42 \pm 0.12$

Table 6.4: Summary of selected models (mean and standard deviation), latent space dimension 6, diagonal covariance, on the MNIST dataset.

learnable curvature  $\mathbb{S}_6$ . Interestingly, all single-component VAEs performed better than their  $(\mathcal{M}^2)^3$  counterparts, except for the  $\mathbb{H}^6$  hyperboloid, but only by a tiny margin. Products of different component types also achieve good results. Noteworthy is that their fixed curvature variants seem to perform marginally better than learnable curvature ones. Our universal VAEs perform at around the Euclidean baseline VAE performance. Interestingly, all of them end up with negative curvatures  $-0.3 < K < 0$ .

Secondly, we run our models with a latent space dimension of 12 (Table 6.5). We immediately notice, that not many models can beat the Euclidean VAE baselines  $(\mathbb{E}^{12})$  and  $(\mathbb{E}^2)^6$  consistently, but several are within the margin of error. Notably, the product VAEs of  $\mathbb{H}$ ,  $\mathbb{S}$ , and  $\mathbb{E}$ , fixed and learnable  $\mathbb{H}^{12}$ , and our universal VAE  $(\mathbb{U}^2)^6$ . Interestingly, products of small components perform better when curvature is fixed (but only by a very tiny margin), whereas single big component VAEs are better when curvature is learned, but again within the margin of error.

Thirdly, the experiments are repeated for a latent space dimension of 72 (Table 6.6). At this dimension, the Euclidean single-component VAE performs better than all other models.  $\mathbb{E}^{72}$  performs best, in a close second  $(\mathbb{E}^2)^{36}$  and our universal VAE  $(\mathbb{U}^2)^{36}$ . Other well-performing models are learnable cur-

## 6.4. Diagonal covariance matrix parametrization

Model	LL	ELBO	BCE	KL
$(\mathbb{S}_1^2)^6$	−79.92±0.21	−84.88±0.14	62.83±0.21	22.06±0.07
$(\mathbb{D}_1^2)^6$	−80.53±0.10	−85.59±0.08	63.62±0.12	21.97±0.16
$(\mathbb{E}^2)^6$	− <b>79.51</b> ±0.10	−83.91±0.12	61.84±0.06	22.07±0.13
$\mathbb{E}^{12}$	− <b>79.51</b> ±0.09	−83.95±0.06	61.66±0.10	22.29±0.04
$(\mathbb{H}_{-1}^2)^6$	−80.54±0.23	−86.05±0.52	63.78±0.26	22.27±0.26
$\mathbb{H}_{-1}^{12}$	− <b>79.37</b> ±0.14	−84.76±0.08	62.32±0.05	22.44±0.10
$(\mathbb{P}_{-1}^2)^6$	−80.39±0.07	−85.46±0.15	63.48±0.22	21.98±0.17
$\mathbb{S}^{12}$	−79.99±0.27	−84.78±0.26	62.89±0.29	21.89±0.18
$\mathbb{D}^{12}$	−80.37±0.16	−85.26±0.19	63.24±0.15	22.02±0.13
$\mathbb{H}^{12}$	−79.77±0.10	−84.58±0.15	62.49±0.10	22.09±0.20
$(\mathbb{P}^2)^6$	−80.31±0.08	−85.35±0.10	63.57±0.17	21.79±0.07
$(\mathbb{D}_1^2)^2 \times (\mathbb{E}^2)^2 \times (\mathbb{P}_{-1}^2)^2$	−80.14±0.11	−85.00±0.08	62.99±0.16	22.01±0.24
$\mathbb{D}_1^4 \times \mathbb{E}^4 \times \mathbb{P}_{-1}^4$	−80.14±0.20	−84.99±0.26	63.06±0.26	21.92±0.08
$(\mathbb{E}^2)^2 \times (\mathbb{H}^2)^2 \times (\mathbb{S}^2)^2$	− <b>79.59</b> ±0.25	−84.43±0.20	62.68±0.20	21.75±0.20
$\mathbb{E}^4 \times \mathbb{H}^4 \times \mathbb{S}^4$	− <b>79.69</b> ±0.14	−84.45±0.12	62.64±0.28	21.81±0.21
$(\mathbb{U}^2)^6$	− <b>79.61</b> ±0.06	−84.13±0.04	61.92±0.22	22.21±0.23
$\mathbb{U}^{12}$	−80.01±0.30	−84.86±0.51	62.90±0.63	21.96±0.16

Table 6.5: Summary of selected models (mean and standard deviation), latent space dimension 12, diagonal covariance, on the MNIST dataset.

vature single-component hyperboloid  $\mathbb{H}^{72}$  and hypersphere  $\mathbb{S}^{72}$ , and learnable curvature product VAEs  $\mathbb{E}^{24} \times \mathbb{H}^{24} \times \mathbb{S}^{24}$  and  $\mathbb{D}^{24} \times \mathbb{E}^{24} \times \mathbb{P}^{24}$ .

Lastly, reconstruction of a few MNIST test digits from some of these models can be visually compared in Figure D.7, and a small interpolation visualization is available in Figure D.8. We also present an illustrative latent space visualization in Figure D.9. All of the figures are attached in Appendix D.

**Dynamically-binarized Omniglot reconstruction** The complete results can be found in Section D.3.2 of Appendix D.

For a latent space of dimension 6 (Table 6.7), the best of the baseline models is the Poincaré VAE of (Mathieu et al., 2019). Our models that come very close to the average estimated marginal log-likelihood, and are definitely within the margin of error, are mainly  $(\mathbb{S}^2)^3$ ,  $\mathbb{D}^2 \times \mathbb{E}^2 \times \mathbb{P}^2$ , and  $\mathbb{U}^6$ . However, with the variance of performance across different runs, we cannot draw a clear conclusion (as is apparent from Figure D.10a in Appendix D). In general, hyperbolic VAEs seem to be doing a bit better on this dataset than spherical VAEs, which is also confirmed by the fact that almost all universal curvature models finished with negative curvature components.

## 6.4. Diagonal covariance matrix parametrization

Model	LL	ELBO	BCE	KL
$(\mathbb{S}_1^2)^{36}$	$-78.43 \pm 0.44$	$-84.99 \pm 0.49$	$56.88 \pm 0.28$	$28.11 \pm 0.56$
$(\mathbb{D}_1^2)^{36}$	$-76.03 \pm 0.17$	$-83.04 \pm 0.25$	$54.35 \pm 0.15$	$28.69 \pm 0.17$
$\mathbb{E}^{72}$	<b><math>-74.42 \pm 0.06</math></b>	$-80.09 \pm 0.12$	$51.45 \pm 0.30$	$28.63 \pm 0.20$
$\mathbb{H}_{-1}^{72}$	$-77.30 \pm 0.12$	$-86.98 \pm 0.09$	$58.04 \pm 0.29$	$28.94 \pm 0.25$
$(\mathbb{P}_{-1}^2)^{36}$	$-76.11 \pm 0.08$	$-82.63 \pm 0.19$	$53.89 \pm 0.36$	$28.74 \pm 0.30$
$\mathbb{P}_{-1}^{72}$	$-77.50 \pm 0.05$	$-84.53 \pm 0.13$	$55.80 \pm 0.20$	$28.73 \pm 0.18$
$\mathbb{S}^{72}$	$-75.24 \pm 0.01$	$-81.39 \pm 0.14$	$53.03 \pm 0.27$	$28.36 \pm 0.16$
$(\mathbb{D}^2)^{36}$	$-75.66 \pm 0.06$	$-81.94 \pm 0.09$	$53.32 \pm 0.16$	$28.61 \pm 0.11$
$(\mathbb{H}^2)^{36}$	$-77.87 \pm 0.02$	$-83.95 \pm 0.02$	$55.71 \pm 0.35$	$28.24 \pm 0.36$
$\mathbb{H}^{72}$	$-75.03 \pm 0.11$	$-81.23 \pm 0.14$	$52.63 \pm 0.10$	$28.61 \pm 0.11$
$(\mathbb{P}^2)^{36}$	$-75.77 \pm 0.12$	$-82.07 \pm 0.02$	$53.65 \pm 0.38$	$28.43 \pm 0.39$
$\mathbb{P}^{72}$	$-75.71 \pm 0.08$	$-81.95 \pm 0.09$	$53.29 \pm 0.14$	$28.67 \pm 0.05$
$(\mathbb{D}^2)^{12} \times (\mathbb{E}^2)^{12} \times (\mathbb{P}^2)^{12}$	$-77.40 \pm 0.55$	$-83.35 \pm 0.41$	$53.90 \pm 0.40$	$29.45 \pm 0.12$
$(\mathbb{D}_1^2)^{12} \times (\mathbb{E}^2)^{12} \times (\mathbb{P}_{-1}^2)^{12}$	$-75.36 \pm 0.23$	$-81.53 \pm 0.42$	$53.02 \pm 0.39$	$28.51 \pm 0.45$
$\mathbb{D}^{24} \times \mathbb{E}^{24} \times \mathbb{P}^{24}$	<b><math>-75.11 \pm 0.05</math></b>	$-80.99 \pm 0.07$	$52.48 \pm 0.19$	$28.52 \pm 0.16$
$(\mathbb{E}^2)^{12} \times (\mathbb{H}^2)^{12} \times (\mathbb{S}^2)^{12}$	$-77.47 \pm \text{nan}$	$-83.28 \pm \text{nan}$	$54.91 \pm \text{nan}$	$28.36 \pm \text{nan}$
$(\mathbb{E}^2)^{12} \times (\mathbb{H}_{-1}^2)^{12} \times (\mathbb{S}_1^2)^{12}$	$-77.53 \pm 0.34$	$-83.95 \pm 0.40$	$55.54 \pm 0.43$	$28.42 \pm 0.08$
$\mathbb{E}^{24} \times \mathbb{H}^{24} \times \mathbb{S}^{24}$	<b><math>-75.04 \pm 0.16</math></b>	$-81.17 \pm 0.18$	$52.61 \pm 0.32$	$28.55 \pm 0.38$
$(\mathbb{U}^2)^{36}$	<b><math>-74.64 \pm 0.08</math></b>	$-80.52 \pm 0.10$	$52.04 \pm 0.10$	$28.48 \pm 0.07$
$\mathbb{U}^{72}$	$-75.46 \pm 0.09$	$-81.76 \pm 0.09$	$53.27 \pm 0.18$	$28.49 \pm 0.18$

Table 6.6: Summary of selected models (mean and standard deviation), latent space dimension 72, diagonal covariance, on the MNIST dataset.

When we scale our models up to the latent space dimension 72 (Table 6.8), we can see that the projected sphere and Poincaré ball VAEs perform better across the board than VAEs with hyperboloid or hyperspherical components. A clear trend is also that single-component VAEs (even learnable) seem to do worse than mixtures of different constant curvature components, like the  $(\mathbb{D}^2)^{12} \times (\mathbb{E}^2)^{12} \times (\mathbb{P}^2)^{12}$  VAE. However, the basic Euclidean VAE beats all the other approaches at this latent space dimension, even though our  $(\mathbb{U}^2)^{36}$  model comes remarkably close and is actually within the margin of error, along with  $(\mathbb{E}^2)^{36}$ .

**CIFAR-10 reconstruction** Due to time constraints, several of the CIFAR models have not been run 3 times, therefore, the results are rather preliminary. Do note that however, especially in higher dimensions, the variance across runs of the same model is usually not very big. If a model is only run once, its standard deviation is defined as “not a number”(nan) for convenience. The complete results can be found in Section D.3.3 of Appendix D.

## 6.4. Diagonal covariance matrix parametrization

Model	LL	ELBO	BCE	KL
$\mathbb{S}_1^6$	-136.69±0.94	-141.46±0.92	129.52±0.74	11.94±0.19
$\mathbb{D}_1^6$	-137.42±1.20	-141.95±1.94	130.70±2.18	11.25±0.26
$\mathbb{E}^6$	-136.05±0.29	-140.50±0.35	128.95±0.41	11.55±0.14
$\mathbb{H}_{-1}^6$	-137.09±0.06	-142.22±0.19	130.37±0.21	11.85±0.12
$\mathbb{P}_{-1}^6$	- <b>135.86</b> ±0.20	-140.36±0.19	128.92±0.23	11.44±0.16
$(\mathbb{S}^2)^3$	- <b>136.14</b> ±0.27	-140.68±0.32	128.98±0.27	11.70±0.13
$\mathbb{S}^6$	-136.20±0.44	-140.76±0.45	129.10±0.37	11.66±0.13
$(\mathbb{D}^2)^3$	-136.13±0.17	-140.59±0.15	129.10±0.20	11.49±0.12
$\mathbb{D}^6$	-136.30±0.08	-140.74±0.14	129.35±0.16	11.39±0.05
$(\mathbb{H}^2)^3$	-136.17±0.09	-140.65±0.17	129.26±0.07	11.39±0.16
$\mathbb{H}^6$	-136.24±0.32	-140.92±0.33	129.48±0.27	11.45±0.12
$(\mathbb{P}^2)^3$	-136.09±0.07	-140.41±0.08	129.04±0.05	11.37±0.08
$\mathbb{P}^6$	-136.05±0.44	-140.42±0.47	129.04±0.53	11.38±0.07
$\mathbb{D}^2 \times \mathbb{E}^2 \times \mathbb{P}^2$	- <b>135.89</b> ±0.40	-140.28±0.42	128.75±0.40	11.53±0.04
$\mathbb{E}^2 \times \mathbb{H}^2 \times \mathbb{S}^2$	-135.93±0.48	-140.51±0.53	128.85±0.48	11.66±0.14
$(\mathbb{U}^2)^3$	-136.21±0.07	-140.65±0.30	129.14±0.34	11.52±0.15
$\mathbb{U}^6$	- <b>136.04</b> ±0.17	-140.43±0.14	129.07±0.27	11.36±0.13

Table 6.7: Summary of selected models (mean and standard deviation), latent space dimension 6, diagonal covariance, on the Omniglot dataset.

For a latent space of dimension 6, we can observe that almost all non-Euclidean models perform better than the euclidean baseline  $\mathbb{E}^6$ . Especially well-performing is the fixed hyperboloid  $\mathbb{H}_{-1}^6$ , and the learnable hypersphere  $\mathbb{S}^6$ . For more detailed results, see Table 6.9. Unfortunately, we were not able to run more models for a better comparison.

On higher dimensions, the Euclidean baseline triumphs over the very limited models we were able to run (Table 6.10), although our universal curvature approach does not trail far behind. Conclusive statements for this latent space dimension cannot be made, as the comparison is very limited.

Curvatures for all learnable models on this dataset converge to values in the approximate range of  $(-0.15, +0.15)$ .

### 6.4.2 Summary of experimental evaluation

Concluding from the results presented above, we can safely say there does not seem to be a single approach that can “do it all”, in the spirit of the “No Free Lunch” theorem (Wolpert et al., 1997).

A very good model seems to be the Riemannian Normal Poincaré ball VAE

6.4. Diagonal covariance matrix parametrization

Model	LL	ELBO	BCE	KL
$(\mathbb{S}_1^2)^{36}$	-112.33±0.14	-118.94±0.14	91.04±0.37	27.90±0.23
$(\mathbb{D}_1^2)^{36}$	-108.66±0.24	-116.06±0.18	85.95±0.16	30.11±0.04
$\mathbb{E}^{72}$	- <b>105.89</b> ±0.16	-112.40±0.17	79.52±0.19	32.89±0.20
$\mathbb{H}_{-1}^{72}$	-111.19±0.42	-120.49±0.35	91.11±0.73	29.38±0.40
$(\mathbb{P}_{-1}^2)^{36}$	-109.05±0.09	-115.99±0.10	85.81±0.42	30.18±0.34
$\mathbb{P}_{-1}^{72}$	-111.24±0.28	-118.36±0.24	89.53±0.38	28.84±0.18
$\mathbb{S}^{72}$	-109.39±0.32	-116.42±0.32	87.22±0.58	29.20±0.28
$\mathbb{D}^{72}$	-108.81±0.08	-115.71±0.09	85.68±0.10	30.03±0.09
$\mathbb{H}^{72}$	-108.62±0.40	-115.54±0.30	85.18±0.62	30.37±0.34
$(\mathbb{P}^2)^{36}$	-108.78±0.66	-115.54±0.70	85.16±1.38	30.38±0.69
$\mathbb{P}^{72}$	-109.66±0.61	-116.50±0.68	87.09±1.43	29.42±0.75
$(\mathbb{D}^2)^{12} \times (\mathbb{E}^2)^{12} \times (\mathbb{P}^2)^{12}$	-107.02±1.56	-115.62±1.76	88.52±8.24	27.10±6.48
$(\mathbb{D}_1^2)^{12} \times (\mathbb{E}^2)^{12} \times (\mathbb{P}_{-1}^2)^{12}$	-108.06±0.47	-114.92±0.39	83.95±0.58	30.97±0.22
$(\mathbb{U}^2)^{36}$	- <b>105.98</b> ±0.05	-112.70±0.19	79.85±0.80	32.85±0.61
$\mathbb{U}^{72}$	-106.58±0.12	-113.68±0.11	81.53±0.34	32.15±0.36

Table 6.8: Summary of selected models (mean and standard deviation), latent space dimension 72, diagonal covariance, on the Omniglot dataset.

Model	LL	ELBO	BCE	KL
$\mathbb{E}^6$	-1896.19±2.54	-1905.75±3.19	1889.97±2.88	15.78±0.32
$\mathbb{H}_{-1}^6$	- <b>1888.23</b> ±2.12	-1896.56±2.93	1882.05±2.65	14.51±0.34
$\mathbb{P}_{-1}^6$	-1893.27±0.61	-1902.67±0.74	1887.44±0.83	15.23±0.16
$\mathbb{D}^6$	-1893.85±0.36	-1902.67±0.69	1887.37±0.74	15.30±0.08
$\mathbb{S}^6$	- <b>1889.76</b> ±1.62	-1897.31±1.71	1882.55±1.48	14.76±0.24
$\mathbb{P}^6$	-1891.40±2.14	-1899.68±2.74	1884.58±2.56	15.10±0.18
$\mathbb{D}^2 \times \mathbb{E}^2 \times \mathbb{P}^2$	-1899.90±4.60	-1904.63±1.46	1889.13±1.38	15.50±0.08
$\mathbb{E}^2 \times \mathbb{H}^2 \times \mathbb{S}^2$	-1895.46±0.92	-1897.57±0.94	1882.84±0.70	14.73±0.24
$(\mathbb{U}^2)^3$	-1895.09±4.27	-1904.46±5.21	1888.89±4.71	15.57±0.51

Table 6.9: Summary of selected models (mean and standard deviation), latent space dimension 6, diagonal covariance, on the CIFAR dataset.



Model	LL	ELBO	BCE	KL
$\mathbb{E}^{512}$	$-1814.12 \pm 0.16$	$-1819.06 \pm 0.20$	$1774.48 \pm 0.43$	$44.57 \pm 0.40$
$\mathbb{E}^{172} \times \mathbb{H}^{170} \times \mathbb{S}^{170}$	$-1815.42 \pm \text{nan}$	$-1820.13 \pm \text{nan}$	$1776.19 \pm \text{nan}$	$43.94 \pm \text{nan}$
$\mathbb{U}^{512}$	$-1814.37 \pm \text{nan}$	$-1819.42 \pm \text{nan}$	$1775.29 \pm \text{nan}$	$44.13 \pm \text{nan}$

Table 6.10: Summary of results (mean and standard deviation), latent space dimension 512, diagonal covariance, on the CIFAR dataset.

$\mathcal{RN} \mathbb{P}^n$ . However, it has practical limitations due to rejection sampling and an unstable implementation.

On the contrary, von Mises-Fischer spherical VAEs have almost consistently performed worse than their Wrapped Normal equivalents. Overall, Wrapped Normal VAEs in all constant curvature manifolds seem to do a good job at modeling the latent space.

A key takeaway is that our universal curvature models  $\mathbb{U}^n$  and  $(\mathbb{U}^2)^{\lfloor n/2 \rfloor}$  seem to generally outperform their corresponding Euclidean VAE baselines in lower-dimensional latent spaces and, with minor losses, manage to keep most of the competitive performance as the dimensionality goes up, contrary to VAEs with other non-Euclidean components.

## 6.5 Future work

Even though we have shown that one can approximate the true posterior very well with Normal-like distributions in Riemannian manifolds of constant curvature, there remain several promising directions of explorations.

First of all, an interesting extension of this work would be to try mixed-curvature VAEs on graph data, e.g. link prediction on social networks, as some of our models might be well suited for sparse and structured data. Another very beneficial extension would be to investigate why the obtained results have such a big variance across runs and try to reduce it. However, this is a problem that affects the Euclidean VAE as well, even if not as flagrantly.

Secondly, we have empirically noticed that it seems to be significantly harder to optimize our models in spherical spaces — they seem more prone to divergence. In discussions, other researchers have also observed similar behavior, but a more thorough investigation is not available at the moment. We have side-stepped some optimization problems by introducing products of spaces — previously, it has been reported that both spherical and hyperbolic VAEs generally do not scale well to dimensions greater than 20 or 40. For those cases, we could successfully optimize a subdivided space  $(\mathbb{S}^2)^{36}$  instead of one big manifold  $\mathbb{S}^{72}$ . However, that also does not seem to be a conclusive rule. Especially

in higher dimensions, we have noticed that our VAEs  $(\mathbb{S}^2)^{36}$  with learnable curvature and  $\mathbb{D}_1^{72}$  with fixed curvature seem to consistently diverge. In a few cases  $\mathbb{S}^{72}$  with fixed curvature and even the product  $(\mathbb{E}^2)^{12} \times (\mathbb{H}^2)^{12} \times (\mathbb{S}^2)^{12}$  with learnable curvature seemed to diverge quite often as well.

The most promising future direction seems to be the use of “Normalizing Flows” for variational inference as presented by Rezende and Mohamed (2015) and Gemici et al. (2016). More recently, it was also combined with “autoregressive flows” in Huang et al. (2018). Using normalizing flows, one should be able to achieve the desired level of complexity of the latent distribution in a VAE, which should, similarly to our work, help to approximate the true posterior of the data better. The advantage of normalizing flows is the flexibility of the modeled distributions, at the expense of being more computationally expensive.

Finally, another interesting extension would be to extend the defined geometrical models to allow for training generative adversarial networks (GANs) (Goodfellow et al., 2014) in products of constant curvature spaces and benefit from the better sharpness and quality of samples that GANs provide. Finally, one could synthesize the above to achieve adversarially trained autoencoders in Riemannian manifolds similarly to Kim et al. (2017); Makhzani et al. (2015); Pan et al. (2018) and aim to achieve good sample quality and a well-formed latent space at the same time.

# Conclusion

---

Generative modeling has gained huge popularity recently with practical applications in many industries. Variational autoencoders (VAEs) are a relatively recent addition to the family of generative models and enables both generative modeling and dimensionality reduction, with a special emphasis on learning representations that occupy the latent space in a meaningful way. This is achieved by having a prior assumption on the distribution of representation in the latent space.

By transforming the latent space and associated prior distributions onto Riemannian manifolds of constant curvature, it has previously been shown that we can learn representations on curved space, which might be beneficial in the case when our data has a strong underlying structure — for example tree-like, or circular (directional).

Generalizing on the above ideas, we have extended the theory of learning VAEs in Riemannian manifolds to products of constant curvature spaces. To be able to do that, we derived the necessary operations in several models of constant curvature spaces, extended existing probability distribution families to these manifolds, and generalized VAEs to latent spaces that are products of smaller “component” spaces. In our approach, each component of such a product space can even have a different and learnable curvature (constant across all points of the given component’s manifold).

On various datasets, we show that our approach is competitive with state of the art VAEs. Additionally, it has the appealing property that it generalizes the variational autoencoder — if the curvatures of all components go to 0, we recover the classical Euclidean VAE of Kingma and Welling (2014).

---

## Bibliography

---

- Hervé Abdi and Lynne J. Williams. Principal component analysis. *Wiley Interdisciplinary Reviews: Computational Statistics*, 2(4):433–459, 2010. doi: 10.1002/wics.101.
- Georgios Arvanitidis, Lars Kai Hansen, and Søren Hauberg. Latent space oddity: on the curvature of deep generative models. In *International Conference on Learning Representations*, 2018.
- Kayhan Batmanghelich, Ardavan Saeedi, Karthik Narasimhan, and Sam Gershman. Nonparametric Spherical Topic Modeling with Word Embeddings. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 537–542. Association for Computational Linguistics, 2016. doi: 10.18653/v1/P16-2087.
- Gary Bécigneul and Octavian-Eugen Ganea. Riemannian Adaptive Optimization Methods. In *International Conference on Learning Representations*, 2019.
- Marcel Berger. *A panoramic view of Riemannian geometry*. Springer Science & Business Media, 2012.
- David M Blei, Alp Kucukelbir, and Jon D McAuliffe. Variational Inference: A Review for Statisticians. *Journal of the American Statistical Association*, 112(518):859–877, 2017.
- Janos Bolyai. *Appendix, Scientiam Spatii absolute veram exhibens: a veritate aut falsitate Axiomatis XI. Euclidei (a priori haud unquam decidenda) independentem; adjecta ad casum falsitatis, quadratura circuli geometrica. Auctore Johanne Bolyai de eadem, Geometrarum in Exercitu Caesareo Regio Austriaco Castrensi Capiteo*. Coll. Ref., 1832.
- Silvere Bonnabel. Stochastic Gradient Descent on Riemannian Manifolds. *IEEE Transactions on Automatic Control*, 58:2217–2229, 2013.

- Samuel R. Bowman, Luke Vilnis, Oriol Vinyals, Andrew Dai, Rafal Jozefowicz, and Samy Bengio. Generating Sentences from a Continuous Space. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 10–21. Association for Computational Linguistics, 2016. doi: 10.18653/v1/K16-1002.
- M. M. Bronstein, J. Bruna, Y. LeCun, A. Szlam, and P. Vandergheynst. Geometric Deep Learning: Going beyond Euclidean data. *IEEE Signal Processing Magazine*, 34(4):18–42, July 2017. ISSN 1053-5888. doi: 10.1109/MSP.2017.2693418.
- Yuri Burda, Roger Grosse, and Ruslan Salakhutdinov. Importance Weighted Autoencoders. In *Proceedings of the 4th International Conference on Learning Representations, ICLR 2016*, 2016.
- James W Cannon, William J Floyd, Richard Kenyon, Walter R Parry, et al. Hyperbolic geometry. *Flavors of geometry*, 31:59–115, 1997.
- Xi Chen, Diederik P. Kingma, Tim Salimans, Yan Duan, Prafulla Dhariwal, John Schulman, Ilya Sutskever, and Pieter Abbeel. Variational Lossy Autoencoder. In *Proceedings of the 2nd International Conference on Learning Representations, ICLR 2014*, 2014.
- Tim R. Davidson, Luca Falorsi, Nicola De Cao, Thomas Kipf, and Jakub M. Tomczak. Hyperspherical Variational Auto-Encoders. In *UAI*, 2018.
- Jay L Devore and Kenneth N Berk. *Modern mathematical statistics with applications*. Springer, 2012.
- Bhuwan Dhingra, Christopher J Shallue, Mohammad Norouzi, Andrew M Dai, and George E Dahl. Embedding Text in Hyperbolic Spaces. *arXiv preprint arXiv:1806.04313*, 2018.
- Chuong B Do. The Multivariate Gaussian Distribution, 2008. URL <http://cs229.stanford.edu/section/gaussians.pdf>.
- Carl Doersch. Tutorial on Variational Autoencoders. *arXiv preprint arXiv:1606.05908*, 2016.
- Robert L. Foote. A Unified Pythagorean Theorem in Euclidean, Spherical, and Hyperbolic Geometries. *Mathematics Magazine*, 90(1):59–69, 2017. ISSN 0025570X, 19300980.
- Octavian Ganea, Gary Bécigneul, and Thomas Hofmann. Hyperbolic Neural Networks. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 5345–5355. Curran Associates, Inc., 2018a.

- 
- Octavian-Eugen Ganea, Gary Bécigneul, and Thomas Hofmann. Hyperbolic Entailment Cones for Learning Hierarchical Embeddings. *arXiv preprint arXiv:1804.01882*, 2018b.
- Carl Friedrich Gauss. *Theoria motus corporum coelestium in sectionibus conicis solem ambientium*, volume 7. Perthes et Besser, 1809.
- Mevlana C Gemici, Danilo Rezende, and Shakir Mohamed. Normalizing Flows on Riemannian Manifolds. *arXiv preprint arXiv:1611.02304*, 2016.
- Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative Adversarial Nets. In *Advances in Neural Information Processing Systems*, pages 2672–2680, 2014.
- Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016.
- Albert Gu, Frederic Sala, Beliz Gunel, and Christopher Ré. Learning Mixed-Curvature Representations in Product Spaces. In *International Conference on Learning Representations*, 2019.
- Søren Hauberg. Directional Statistics with the Spherical Normal Distribution. In *2018 21st International Conference on Information Fusion (FUSION)*, pages 704–711. IEEE, 2018.
- Chin-Wei Huang, David Krueger, Alexandre Lacoste, and Aaron Courville. Neural autoregressive flows. *arXiv preprint arXiv:1804.00779*, 2018.
- Edwin T Jaynes. Information theory and statistical mechanics. *Physical review*, 106(4):620, 1957.
- Michael I. Jordan, Zoubin Ghahramani, Tommi S. Jaakkola, and Lawrence K. Saul. An Introduction to Variational Methods for Graphical Models. *Machine learning*, 37(2):183–233, 1999.
- Yoon Kim, Kelly Zhang, Alexander M Rush, Yann LeCun, et al. Adversarially regularized autoencoders. *arXiv preprint arXiv:1706.04223*, 2017.
- Diederik P. Kingma and Jimmy Lei Ba. Adam: A Method for Stochastic Optimization. *International Conference on Learning Representations (ICLR)*, 2015.
- Diederik P Kingma and Max Welling. Auto-Encoding Variational Bayes. In *Proceedings of the 2nd International Conference on Learning Representations*, ICLR 2014, 2014.

- 
- R. Kleinberg. Geographic Routing Using Hyperbolic Space. In *IEEE INFOCOM 2007 - 26th IEEE International Conference on Computer Communications*, pages 1902–1909, May 2007. doi: 10.1109/INFCOM.2007.221.
- Max Kochurov, Sergey Kozlukov, Rasul Karimov, and Viktor Yanush. Geoopt: Adaptive Riemannian optimization in PyTorch. <https://github.com/geoopt/geoopt>, 2019.
- Alex Krizhevsky. Learning Multiple Layers of Features from Tiny Images. Technical report, University of Toronto, 2009.
- Solomon Kullback and Richard A. Leibler. On Information and Sufficiency. *Ann. Math. Statist.*, 22(1):79–86, 03 1951. doi: 10.1214/aoms/1177729694.
- Brenden M. Lake, Ruslan Salakhutdinov, and Joshua B. Tenenbaum. Human-level concept learning through probabilistic program induction. *Science*, 350 (6266):1332–1338, 2015. ISSN 0036-8075. doi: 10.1126/science.aab3050.
- Johann H Lambert. Observations trigonométriques. *Mem. Acad. Sci. Berlin*, 24:327–354, 1770.
- Marc T Law, Jake Snell, and Richard S Zemel. Lorentzian distance learning, 2019.
- Yann LeCun. The MNIST database of handwritten digits, 1998. URL <http://yann.lecun.com/exdb/mnist/>.
- J.M. Lee. *Riemannian Manifolds: An Introduction to Curvature*. Graduate Texts in Mathematics. Springer New York, 1997. ISBN 9780387983226.
- Hongbo Li, David Hestenes, and Alyn Rockwood. *A Universal Model for Conformal Geometries of Euclidean, Spherical and Double-Hyperbolic Spaces*, pages 77–104. Springer Berlin Heidelberg, Berlin, Heidelberg, 2001. ISBN 978-3-662-04621-0. doi: 10.1007/978-3-662-04621-0\_4.
- Francesco Locatello, Stefan Bauer, Mario Lucic, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. Challenging common assumptions in the unsupervised learning of disentangled representations. *arXiv preprint arXiv:1811.12359*, 2018.
- Alireza Makhzani, Jonathon Shlens, Navdeep Jaitly, Ian Goodfellow, and Brendan Frey. Adversarial autoencoders. *arXiv preprint arXiv:1511.05644*, 2015.
- Kanti V Mardia. Characterizations of directional distributions. In *A Modern Course on Statistical Distributions in Scientific Work*, pages 365–385. Springer, 1975.

- 
- Emile Mathieu, Charline Le Lan, Chris J Maddison, Ryota Tomioka, and Yee Whye Teh. Hierarchical Representations with Poincaré Variational Auto-Encoders. *arXiv preprint arXiv:1901.06033*, 2019.
- Loïc Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework. In *Proceedings of the 5th International Conference on Learning Representations, ICLR 2017*, 2017.
- Mervin E. Muller. A Note on a Method for Generating Points Uniformly on N-dimensional Spheres. *Commun. ACM*, 2(4):19–20, April 1959. ISSN 0001-0782. doi: 10.1145/377939.377946.
- Christian A. Naesseth, Francisco J. R. Ruiz, Scott W. Linderman, and David M. Blei. Reparameterization Gradients through Acceptance-Rejection Sampling Algorithms. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, AISTATS 2017, 20-22 April 2017, Fort Lauderdale, FL, USA*, pages 489–498, 2017.
- Yoshihiro Nagano, Shoichiro Yamaguchi, Yasuhiro Fujita, and Masanori Koyama. A differentiable gaussian-like distribution on hyperbolic space for gradient-based learning. *arXiv preprint arXiv:1902.02992*, 2019.
- Maximilian Nickel and Douwe Kiela. Learning Continuous Hierarchies in the Lorentz Model of Hyperbolic Geometry. In *Proceedings of the 35th International Conference on International Conference on Machine Learning - Volume 50, ICML’18*, 2018.
- Maximilian Nickel and Douwe Kiela. Poincaré Embeddings for Learning Hierarchical Representations. In *Advances in Neural Information Processing Systems 30*, pages 6338–6347. Curran Associates, Inc., 2017.
- Simon Pampana. The Poincaré disk is a model to “see” 2D hyperbolic space by approximating what a hyperbola looks like from below, 2016. URL <https://twitter.com/mathemaniac/status/753728363563331584>.
- Shirui Pan, Ruiqi Hu, Guodong Long, Jing Jiang, Lina Yao, and Chengqi Zhang. Adversarially regularized graph autoencoder for graph embedding. *arXiv preprint arXiv:1802.04407*, 2018.
- Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in PyTorch. In *NIPS-W*, 2017.
- Xavier Pennec. Intrinsic Statistics on Riemannian Manifolds: Basic Tools for Geometric Measurements. *Journal of Mathematical Imaging and Vision*, 25(1):127, Jul 2006. ISSN 1573-7683. doi: 10.1007/s10851-006-6228-4.



- 
- Peter Petersen, S Axler, and KA Ribet. *Riemannian Geometry*, volume 171. Springer, 2006.
- John Ratcliffe. *Foundations of Hyperbolic Manifolds*, volume 149. Springer Science & Business Media, 2006.
- Danilo Jimenez Rezende and Shakir Mohamed. Variational Inference with Normalizing Flows. *arXiv preprint arXiv:1505.05770*, 2015.
- Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic Backpropagation and Approximate Inference in Deep Generative Models. *arXiv preprint arXiv:1401.4082*, 2014.
- Frederic Sala, Chris De Sa, Albert Gu, and Christopher Re. Representation tradeoffs for hyperbolic embeddings. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 4460–4469, Stockholm, Sweden, 10–15 Jul 2018. PMLR.
- Ruslan Salakhutdinov and Iain Murray. On the Quantitative Analysis of Deep Belief Networks. In *Proceedings of the 25th International Conference on Machine Learning*, ICML’08, pages 872–879, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-205-4. doi: 10.1145/1390156.1390266.
- Rik Sarkar. Low Distortion Delaunay Embedding of Trees in Hyperbolic Plane. In Marc van Kreveld and Bettina Speckmann, editors, *Graph Drawing*, pages 355–366, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg. ISBN 978-3-642-25878-7.
- Irhum Shafkat. Intuitively Understanding Variational Autoencoders, 2018. URL <https://towardsdatascience.com/intuitively-understanding-variational-autoencoders-1bfe67eb5daf>.
- Martin Simonovsky and Nikos Komodakis. GraphVAE: Towards generation of small graphs using variational autoencoders. In *International Conference on Artificial Neural Networks*, pages 412–422. Springer, 2018.
- John Parr Snyder. *Map projections—A working manual*, volume 1395. US Government Printing Office, 1987.
- Akihiro Tanabe, Kenji Fukumizu, Shigeyuki Oba, Takashi Takenouchi, and Shin Ishii. Parameter estimation for von Mises–Fisher distributions. *Computational Statistics*, 22(1):145–157, Apr 2007. ISSN 1613-9658. doi: 10.1007/s00180-007-0030-7.
- Alexandru Tifrea, Gary Bécigneul, and Octavian-Eugen Ganea. Poincaré Glove: Hyperbolic Word Embeddings. In *International Conference on Learning Representations*, 2019.

- Gary Ulrich. Computer Generation of Distributions on the m-Sphere. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 33(2):158–163, 1984. ISSN 00359254, 14679876.
- Abraham Albert Ungar. A Gyrovector Space Approach to Hyperbolic Geometry. *Synthesis Lectures on Mathematics and Statistics*, 1(1):1–194, 2008.
- Wikimedia. Hyperbolic functions, 2009. URL [https://commons.wikimedia.org/wiki/File:Hyperbolic\\_functions-2.svg](https://commons.wikimedia.org/wiki/File:Hyperbolic_functions-2.svg).
- Wikimedia. Stereographic projection SW, 2012. URL [https://en.wikipedia.org/wiki/File:Stereographic\\_projection\\_SW.JPG](https://en.wikipedia.org/wiki/File:Stereographic_projection_SW.JPG).
- Wikimedia. Stereographic projection in 3D, 2017. URL [https://commons.wikimedia.org/wiki/File:Stereographic\\_projection\\_in\\_3D.svg](https://commons.wikimedia.org/wiki/File:Stereographic_projection_in_3D.svg).
- Benjamin Wilson and Matthias Leimeister. Gradient descent in hyperbolic space. *arXiv preprint arXiv:1805.08207*, 2018.
- Richard C. Wilson and Edwin R. Hancock. Spherical embedding and classification. In Edwin R. Hancock, Richard C. Wilson, Terry Windeatt, Ilkay Ulusoy, and Francisco Escolano, editors, *Structural, Syntactic, and Statistical Pattern Recognition*, pages 589–599, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg. ISBN 978-3-642-14980-1.
- David H Wolpert, William G Macready, et al. No Free Lunch Theorems for Optimization. *IEEE transactions on evolutionary computation*, 1(1):67–82, 1997.
- Jiacheng Xu and Greg Durrett. Spherical Latent Spaces for Stable Variational Autoencoders. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4503–4513. Association for Computational Linguistics, 2018.
- Jun-Yan Zhu, Philipp Krähenbühl, Eli Shechtman, and Alexei A. Efros. Generative Visual Manipulation on the Natural Image Manifold. In *Proceedings of European Conference on Computer Vision (ECCV)*, 2016.

---

# Notation

---

## Numbers and Arrays

$a$	A scalar (integer or real)
$\mathbf{a}$	A vector
$\mathbf{A}$	A matrix
$\mathbf{I}_n$	Identity matrix with $n$ rows and $n$ columns
$\mathbf{I}$	Identity matrix with dimensionality implied by context
$\mathbf{e}^{(i)}$	Standard basis vector with all 0 and a 1 at position $i$
$\text{diag}(\mathbf{a})$	A square, diagonal matrix with entries given by $\mathbf{a}$
$\text{trace}(\mathbf{A})$	Trace of matrix $\mathbf{A}$
$a_i$	Element $i$ of vector $\mathbf{a}$ , with indexing starting at 1
$A_{i,j}$	Element $i, j$ of matrix $\mathbf{A}$

## Sets

$\mathbb{A}$	A set
$\mathbb{R}, \mathbb{C}$	The set of real (complex) numbers
$\{0, 1, \dots, n\}$	The set of all integers between 0 and $n$
$[a, b]$	The real interval including $a$ and $b$
$(a, b]$	The real interval excluding $a$ but including $b$
$\mathbb{A} \setminus \mathbb{B}$	Set subtraction, i.e., the set containing the elements of $\mathbb{A}$ that are not in $\mathbb{B}$

**Calculus**

$\frac{dy}{dx}$	Derivative of $y$ with respect to $x$
$\frac{\partial y}{\partial x}$	Partial derivative of $y$ with respect to $x$
$\nabla_{\mathbf{x}}y$	Gradient of $y$ with respect to $\mathbf{x}$
$\nabla_{\mathbf{X}}y$	Matrix derivatives of $y$ with respect to $\mathbf{X}$
$\frac{\partial f}{\partial \mathbf{x}}$	Jacobian matrix $\mathbf{J} \in \mathbb{R}^{m \times n}$ of $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$
$\nabla_{\mathbf{x}}^2 f(\mathbf{x})$	The Hessian matrix of $f$ at input point $\mathbf{x}$
$\int f(\mathbf{x})d\mathbf{x}$	Definite integral over the entire domain of $\mathbf{x}$
$\int_{\mathbb{S}} f(\mathbf{x})d\mathbf{x}$	Definite integral with respect to $\mathbf{x}$ over the set $\mathbb{S}$
$x \rightarrow c^+, x \rightarrow c^-$	$x$ approaches $c$ from above (below)

**Probability and Information Theory**

$p(a)$	A probability distribution over a continuous variable, or over a variable whose type has not been specified
$a \sim p$	Random variable $a$ has distribution $p$
$\mathbf{E}_{\mathbf{x} \sim p} [f(\mathbf{x})]$	Expectation of $f(\mathbf{x})$ with respect to $p(\mathbf{x})$
$\text{Var}(f(\mathbf{x}))$	Variance of $f(\mathbf{x})$ under $p(\mathbf{x})$
$\text{Cov}(f(\mathbf{x}), g(\mathbf{x}))$	Covariance of $f(\mathbf{x})$ and $g(\mathbf{x})$ under $p(\mathbf{x})$
$H(\mathbf{x})$	Shannon entropy of the random variable $\mathbf{x}$
$D_{\text{KL}}(p \parallel q)$	Kullback-Leibler divergence of $p$ and $q$ (non-symmetric)
$\mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$	Normal (Gaussian) distribution over $\mathbf{x}$ with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$

**Functions**

$f : \mathbb{A} \rightarrow \mathbb{B}$	The function $f$ with domain $\mathbb{A}$ and range $\mathbb{B}$
$f \circ g$	Composition of the functions $f$ and $g$
$f(\mathbf{x}; \boldsymbol{\theta})$	A function of $\mathbf{x}$ parametrized by $\boldsymbol{\theta}$ . Sometimes we write $f(\mathbf{x})$ and omit the argument $\boldsymbol{\theta}$ to lighten notation.
$\log x$	Natural logarithm of $x$
$\langle \mathbf{x}, \mathbf{y} \rangle_2$	Standard scalar product in $\mathbb{R}^n$ , $\sum_{i=1}^n x_i y_i$
$\ \mathbf{x}\ _p$	$L^p$ norm of $\mathbf{x}$
$\Gamma(z)$	Gamma function, $\int_0^{\infty} x^{z-1} \exp(-x) dx$
$\mathbf{x} \odot \mathbf{y}$	Concatenation of the vectors $\mathbf{x}$ and $\mathbf{y}$

**Abbreviations**

BDP	Binary Diffusion Process (Mathieu et al., 2019)
CDF	Cummulative distribution function
CV	Computer Vision
ELBO	Evidence Lower Bound
IWAE	Importance Weighted Autoencoder (Burda et al., 2016)
KL	Kullback-Leiber (divergence)
ML	Machine Learning
MLE	Maximum Likelihood Estimation
NLU	Natural Language Understanding
PCA	Principal Component Analysis
PD	Positive definite
PDF	Probability density function
PSD	Positive semi-definite
VAE	Variational Autoencoder (Kingma and Welling, 2014)
vMF	von Mises-Fisher distribution

---

## List of Theorems

---

2.1	Remark (Euclidean constant curvature spaces are sub-divisible.)	17
3.1	Remark (Diagonal covariance in multivariate Normal distributions)	19
3.2	Remark (Maximum Likelihood characterization)	19
3.3	Remark (Maximum Entropy principle)	19
A.1	Theorem ( $\log_{\mathbf{x}}$ is the inverse of $\exp_{\mathbf{x}}$ in $\mathbb{E}^n$ )	69
A.2	Theorem (Length preservation property of $\exp_{\mathbf{x}}$ in $\mathbb{E}^n$ )	69
A.3	Remark (About the divergence of points in $\mathbb{H}_K^n$ )	70
A.4	Theorem (Exponential map in $\mathbb{H}_K^n$ )	71
A.5	Theorem (Logarithmic map in $\mathbb{H}_K^n$ )	72
A.6	Theorem ( $\log_{\mathbf{x}}^K$ is the inverse of $\exp_{\mathbf{x}}^K$ in $\mathbb{H}_K^n$ )	73
A.7	Theorem (Length preservation property of $\exp_{\mathbf{x}}^K$ in $\mathbb{H}_K^n$ )	73
A.8	Theorem ( $\text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K$ transports points to the tangent space of $\mathbf{y}$ in $\mathbb{H}_K^n$ )	74
A.9	Theorem (Parallel transport preserves angles in $\mathbb{H}_K^n$ )	75
A.10	Corollary (Parallel transport in $\mathbb{H}_K^n$ is norm-preserving)	75
A.11	Theorem (Stereographic backprojected points of $\mathbb{P}_K^n$ belong to $\mathbb{H}_K^n$ )	76
A.12	Theorem (Distance equivalence in $\mathbb{P}_K^n$ )	78
A.13	Theorem (Gyospace distance converges to Euclidean in $\mathbb{P}_K^n$ )	78
A.14	Theorem (Distance converges to Euclidean as $K \rightarrow 0^-$ in $\mathbb{P}_K^n$ )	79
A.15	Theorem (Length preservation property of $\exp_{\mathbf{x}}^K$ in $\mathbb{P}_K^n$ )	79
A.16	Theorem (Parallel transport and its inverse in $\mathbb{P}_K^n$ )	80
A.17	Theorem (Parallel transport preserves angles in $\mathbb{P}_K^n$ )	81
A.18	Corollary (Parallel transport in $\mathbb{P}_K^n$ is norm-preserving)	81
A.19	Remark (About the divergence of points in $\mathbb{S}_K^n$ )	83
A.20	Theorem (Exponential map in $\mathbb{S}_K^n$ )	83
A.21	Theorem (Logarithmic map in $\mathbb{S}_K^n$ )	84

A.22	Theorem ( $\log_{\mathbf{x}}^K$ is the inverse of $\exp_{\mathbf{x}}^K$ in $\mathbb{S}_K^n$ ) . . . . .	85
A.23	Theorem (Length preservation property of $\exp_{\mathbf{x}}^K$ in $\mathbb{S}_K^n$ ) . . . . .	85
A.24	Theorem (Parallel transport preserves angles in $\mathbb{S}_K^n$ ) . . . . .	86
A.25	Corollary (Parallel transport on $\mathbb{S}_K^n$ is norm-preserving) . . . . .	87
A.26	Theorem ( $\text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K$ transports points to the tangent space of $\mathbf{y}$ in $\mathbb{S}_K^n$ ) . . . . .	87
A.27	Remark (Homeomorphism between $\mathbb{S}_K^n$ and $\mathbb{R}^n$ ) . . . . .	88
A.28	Theorem (Stereographic backprojected points of $\mathbb{D}_K^n$ belong to $\mathbb{S}_K^n$ ) . . . . .	88
A.29	Theorem (Distance equivalence in $\mathbb{D}_K^n$ ) . . . . .	89
A.30	Theorem (Gyrospace distance converges to Euclidean in $\mathbb{D}_K^n$ ) . . . . .	90
A.31	Theorem (Distance converges to Euclidean as $K \rightarrow 0^+$ in $\mathbb{D}_K^n$ ) . . . . .	90
A.32	Theorem ( $\log_{\mathbf{x}}^K$ is the inverse of $\exp_{\mathbf{x}}^K$ in $\mathbb{D}_K^n$ ) . . . . .	91
A.33	Theorem (Length preservation property of $\exp_{\mathbf{x}}^K$ in $\mathbb{D}_K^n$ ) . . . . .	92
A.34	Theorem (Parallel transport and its inverse in $\mathbb{D}_K^n$ ) . . . . .	93
A.35	Theorem (Parallel transport preserves angles in $\mathbb{D}_K^n$ ) . . . . .	94
A.36	Corollary (Parallel transport on $\mathbb{D}_K^n$ is norm-preserving) . . . . .	94
A.37	Theorem (Möbius addition converges to Eucl. vector addition) . . . . .	94
A.38	Theorem ( $\rho_K^{-1}$ is the inverse stereographic projection) . . . . .	95
A.39	Lemma ( $\lambda_{\mathbf{x}}^K$ converges to 2 as $K \rightarrow 0$ ) . . . . .	96
A.40	Theorem ( $\exp_{\mathbf{x}}^K(\mathbf{v})$ converges to $\mathbf{x} + \mathbf{v}$ as $K \rightarrow 0$ ) . . . . .	96
A.41	Theorem ( $\log_{\mathbf{x}}^K(\mathbf{y})$ converges to $\mathbf{y} - \mathbf{x}$ as $K \rightarrow 0$ ) . . . . .	97
A.42	Lemma ( $\text{gyr}[\mathbf{x}, \mathbf{y}]\mathbf{v}$ converges to $\mathbf{v}$ as $K \rightarrow 0$ ) . . . . .	98
A.43	Theorem ( $\text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v})$ converges to $\mathbf{v}$ as $K \rightarrow 0$ ) . . . . .	98
B.1	Remark (vMF distribution on $\mathbb{S}_K^n$ ) . . . . .	101
B.2	Theorem (Probability density function of $\mathcal{WN}(\mathbf{z}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ in $\mathbb{H}^n$ ) . . . . .	101
B.3	Theorem (Probability density function of $\mathcal{WN}(\mathbf{z}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ in $\mathbb{H}_K^n$ ) . . . . .	104
B.4	Theorem (Probability density function of $\mathcal{WN}(\mathbf{z}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ in $\mathbb{S}_K^n$ ) . . . . .	107
B.5	Theorem (Probability density function of $\mathcal{WN}(\mathbf{z}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ in $\mathbb{P}_K^n$ ) . . . . .	110
B.6	Theorem (Probability density function of $\mathcal{WN}(\mathbf{z}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ in $\mathbb{D}_K^n$ ) . . . . .	111
D.1	Remark (Computability of functions with floating-point numbers) . . . . .	114
D.2	Remark (Practical limitations of constant curvature manifolds) . . . . .	114

---

## List of Figures

---

1.1	Binary tree embedded in a Poincaré ball (Mathieu et al., 2019). . .	2
2.1	Illustrative visualizations of the stereographic projection $\rho_K$ . . .	10
2.2	Visualization of the topological difference between a two-dimensional sphere, a torus, and a hyperboloid. Colors correspond to curvature (red is -1, white is 0, blue is +1). . . . .	17
3.1	Three-step transformation of a sampled point $\mathbf{v}$ in $\mathcal{T}_{\mu_0}\mathbb{H}^1$ for the hyperbolic Wrapped Normal distribution (Nagano et al., 2019). . .	24
4.1	Illustration of a VAE model. Green boxes represent sample and latent space representations, red boxes represent parameterized functions (neural networks), the purple box represents a reparameterized source of randomness, yellow denotes a probability distribution, and blue boxes are loss terms. . . . .	30
5.1	Visualization of a one-dimensional hypersphere and hyperboloid around $K = 0$ . . . . .	35
A.1	“The Poincaré disk is a model to ‘see’ a 2D hyperbolic space by approximating what a hyperbola looks like from below,” Pampena (2016). . . . .	77
A.2	Visualization of cosh of an angle in $\mathbb{H}_1^1$ (Wikimedia, 2009). . . . .	99
B.1	Surface area plots for spheres of variable radius in $n$ -dimensional spaces. . . . .	101
C.1	VAE latent space representation plots of a VAE applied to MNIST digits, using different parts of the ELBO loss for optimization (Shafkat, 2018). . . . .	112



---

D.1	Learned curvature across epochs (with standard deviation) with latent space dimension of 6, spherical covariance parametrization, on the BDP dataset. . . . .	118
D.2	Learned curvature across epochs (with standard deviation) with latent space dimension of 6, spherical covariance parametrization, on the MNIST dataset. . . . .	118
D.3	Boxplot of evaluation marginal log-likelihoods at the end of training for BDP and MNIST, with spherical covariance per component.	119
D.4	Boxplot of evaluation marginal log-likelihoods at the end of training for MNIST, with diagonal covariance per component. . . . .	123
D.5	Learned curvature across epochs (with standard deviation) with latent space dimension of 6, diagonal covariance parametrization, on the MNIST dataset. . . . .	124
D.6	Learned curvature across epochs (with standard deviation) with latent space dimension of 12, diagonal covariance parametrization, on the MNIST dataset. . . . .	124
D.7	Qualitative comparison of reconstruction quality of randomly selected runs of a selection of well-performing models on MNIST test set digits. . . . .	125
D.8	Samples from various models of a grid search around $\mathbf{0}$ of a single component's latent space on MNIST test digits. . . . .	126
D.9	Illustrative latent space visualization of a randomly selected run of the models $\mathbb{E}^2 \times \mathbb{H}^2 \times \mathbb{S}^2$ , $\mathbb{E}^6$ , $\mathbb{H}^6$ , and $\mathbb{S}^6$ on MNIST. . . . .	127
D.10	Boxplot of evaluation marginal log-likelihoods at the end of training for Omniglot, with spherical covariance per component. . . . .	128
D.11	Learned curvature across epochs (with standard deviation) with latent space dimension of 72, diagonal covariance parametrization, on the Omniglot dataset. . . . .	130
D.12	Qualitative comparison of reconstruction quality of randomly selected runs of a selection of well-performing models on Omniglot test set characters. . . . .	132
D.13	Boxplot of evaluation marginal log-likelihoods at the end of training for Omniglot, with spherical covariance per component. . . . .	133
D.14	Learned curvature across epochs (with standard deviation) with latent space dimension of 6, diagonal covariance parametrization, on the CIFAR dataset. . . . .	134
D.15	Qualitative comparison of reconstruction quality of randomly selected runs of a selection of well-performing models on CIFAR test set images. . . . .	135
D.16	Samples from various models of a grid search around $\mathbf{0}$ of a single component's latent space on cifar test digits. . . . .	136

---

## List of Tables

---

2.1	Summary of operations in $\mathbb{S}_K$ and $\mathbb{H}_K$ . . . . .	14
2.2	Summary of operations in $\mathbb{D}_K$ and $\mathbb{P}_K$ . . . . .	14
6.1	Brief overview of components and their properties. . . . .	40
6.2	Summary of results (mean and standard deviation), latent space dimension 6, spherical covariance, on the BDP dataset. . . . .	41
6.3	Summary of selected models (mean and standard deviation), latent space dimension 6, spherical covariance, on the MNIST dataset. . . . .	42
6.4	Summary of selected models (mean and standard deviation), latent space dimension 6, diagonal covariance, on the MNIST dataset. . . . .	44
6.5	Summary of selected models (mean and standard deviation), latent space dimension 12, diagonal covariance, on the MNIST dataset. . . . .	45
6.6	Summary of selected models (mean and standard deviation), latent space dimension 72, diagonal covariance, on the MNIST dataset. . . . .	46
6.7	Summary of selected models (mean and standard deviation), latent space dimension 6, diagonal covariance, on the Omniglot dataset. . . . .	47
6.8	Summary of selected models (mean and standard deviation), latent space dimension 72, diagonal covariance, on the Omniglot dataset. . . . .	48
6.9	Summary of selected models (mean and standard deviation), latent space dimension 6, diagonal covariance, on the CIFAR dataset. . . . .	48
6.10	Summary of results (mean and standard deviation), latent space dimension 512, diagonal covariance, on the CIFAR dataset. . . . .	49
A.1	Euclidean operations. . . . .	68
A.2	Hyperbolic operations. . . . .	70
A.3	Poincaré ball operations. . . . .	76
A.4	Spherical operations. . . . .	82
A.5	Spherical projected operations. . . . .	88

---

D.1	Summary of results (mean and standard-deviation) with latent space dimension of 6, spherical covariance parametrization, on the BDP dataset. . . . .	116
D.2	Summary of results (mean and standard-deviation) with latent space dimension of 6, spherical covariance parametrization, on the MNIST dataset. . . . .	117
D.3	Summary of results (mean and standard-deviation) with latent space dimension of 6, diagonal covariance parametrization, on the MNIST dataset. . . . .	120
D.4	Summary of results (mean and standard-deviation) with latent space dimension of 12, diagonal covariance parametrization, on the MNIST dataset. . . . .	121
D.5	Summary of results (mean and standard-deviation) with latent space dimension of 72, diagonal covariance parametrization, on the MNIST dataset. . . . .	122
D.6	Summary of results (mean and standard-deviation) with latent space dimension of 6, diagonal covariance parametrization, on the Omniglot dataset. . . . .	129
D.7	Summary of results (mean and standard-deviation) with latent space dimension of 72, diagonal covariance parametrization, on the Omniglot dataset. . . . .	131
D.8	Summary of results (mean and standard-deviation) with latent space dimension of 6, diagonal covariance parametrization, on the CIFAR dataset. . . . .	134
D.9	Summary of results (mean and standard-deviation) with latent space dimension of 512, diagonal covariance parametrization, on the CIFAR dataset. . . . .	135

## Appendix A

---

# Geometrical details

---

This chapter provides detailed statements and proofs for various properties of constant curvature spaces, operations thereon, and properties thereof.

## A.1 Euclidean geometry

### A.1.1 Euclidean space

An overview of all the necessary operations can be found in Table A.1.

#### Distance function

The distance function in  $\mathbb{E}^n$  is

$$d_{\mathbb{E}}(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|_2.$$

Due to the Pythagorean theorem, we can derive that

$$\begin{aligned} \|\mathbf{x} - \mathbf{y}\|_2^2 &= \langle \mathbf{x} - \mathbf{y}, \mathbf{x} - \mathbf{y} \rangle_2 = \|\mathbf{x}\|_2^2 - 2 \langle \mathbf{x}, \mathbf{y} \rangle_2 + \|\mathbf{y}\|_2^2 \\ &= \|\mathbf{x}\|_2^2 + \|\mathbf{y}\|_2^2 - 2 \|\mathbf{x}\|_2 \|\mathbf{y}\|_2 \cos^{-1} \theta_{\mathbf{x}, \mathbf{y}} \end{aligned}$$

#### Exponential map

The exponential map in  $\mathbb{E}^n$  is

$$\exp_{\mathbf{x}}(\mathbf{v}) = \mathbf{x} + \mathbf{v}.$$

---

Distance	$d_{\mathbb{E}}(\mathbf{x}, \mathbf{y}) = \ \mathbf{x} - \mathbf{y}\ _2$
Exp. map	$\exp_{\mathbf{x}}(\mathbf{v}) = \mathbf{x} + \mathbf{v}$
Log. map	$\log_{\mathbf{x}}(\mathbf{y}) = \mathbf{y} - \mathbf{x}$

---

Table A.1: Euclidean operations.

The fact that the resulting points belong to the space is trivial. Deriving the inverse function, i.e. the logarithmic map, is also trivial:

$$\log_{\mathbf{x}}(\mathbf{y}) = \mathbf{y} - \mathbf{x}.$$

**Theorem A.1** ( $\log_{\mathbf{x}}$  is the inverse of  $\exp_{\mathbf{x}}$  in  $\mathbb{E}^n$ )

$$\log_{\mathbf{x}}(\exp_{\mathbf{x}}(\mathbf{v})) = \mathbf{v}.$$

**Proof**

$$\log_{\mathbf{x}}(\exp_{\mathbf{x}}(\mathbf{v})) = \log_{\mathbf{x}}(\mathbf{x} + \mathbf{v}) = (\mathbf{x} + \mathbf{v}) - \mathbf{x} = \mathbf{v}. \quad \square$$

**Theorem A.2 (Length preservation property of  $\exp_{\mathbf{x}}$  in  $\mathbb{E}^n$ )** *For all points on the manifold  $\mathbf{x} \in \mathbb{E}^n$  and for all tangent vectors at that point  $\mathbf{v} \in \mathcal{T}_{\mathbf{x}}\mathbb{E}^n$  it holds that*

$$d_{\mathbb{E}}(\mathbf{x}, \exp_{\mathbf{x}}(\mathbf{v})) = \|\mathbf{v}\|_2.$$

**Proof**

$$d_{\mathbb{E}}(\mathbf{x}, \exp_{\mathbf{x}}(\mathbf{v})) = \|\mathbf{x} - \exp_{\mathbf{x}}(\mathbf{v})\|_2 = \|\mathbf{x} - (\mathbf{x} + \mathbf{v})\|_2 = \|-\mathbf{v}\|_2 = \|\mathbf{v}\|_2. \quad \square$$

### Parallel transport

We do not need parallel transport in the Euclidean space, as we can directly sample from a Normal distribution. In other words, we can just define parallel transport to be an identity function.

## A.2 Hyperbolic geometry

### A.2.1 Hyperboloid

An overview of all the necessary operations can be found in Table A.2.

Do note, that all the theorems for the hypersphere are essentially trivial corollaries of their equivalents in the hypersphere (and vice-versa) (Section A.3.1). Notable differences include the fact that  $R^2 = -\frac{1}{K}$ , not  $R^2 = \frac{1}{K}$ , and all the operations use the hyperbolic trigonometric functions  $\sinh$ ,  $\cosh$ , and  $\tanh$ , instead of their Euclidean counterparts. Also, we often leverage the “hyperbolic” Pythagorean theorem, in the form  $\cosh^2(\alpha) - \sinh^2(\alpha) = 1$ .

Distance	$d_{\mathbb{H}}(\mathbf{x}, \mathbf{y}) = \frac{1}{\sqrt{-K}} \cosh^{-1}(-K \langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}})$
Exp. map	$\exp_{\mathbf{x}}^K(\mathbf{v}) = \cosh(\beta) \mathbf{x} + \sinh(\beta) \frac{\mathbf{v}}{\beta}$ , where $\beta = \sqrt{-K} \ \mathbf{v}\ _{\mathcal{L}}$
Log. map	$\log_{\mathbf{x}}^K(\mathbf{y}) = \frac{\cosh^{-1}(\alpha)}{\sqrt{\alpha^2 - 1}}(\mathbf{y} - \alpha \mathbf{x})$ , where $\alpha = K \langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}}$
Par. transp.	$\text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}) = \mathbf{v} - \frac{K \langle \mathbf{y}, \mathbf{v} \rangle_{\mathcal{L}}}{1 + K \langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}}}(\mathbf{x} + \mathbf{y})$

Table A.2: Hyperbolic operations.

### Projections

Due to the definition of the space as a retraction from the ambient space, we can project a generic vector in the ambient space to the hyperboloid using the shortest Euclidean distance by normalization:

$$\text{proj}_{\mathbb{H}_K^n}(\mathbf{x}) = R \frac{\mathbf{x}}{\|\mathbf{x}\|_{\mathcal{L}}} = \frac{\mathbf{x}}{\sqrt{K} \|\mathbf{x}\|_{\mathcal{L}}}.$$

Secondly, the  $n+1$  coordinates of a point on the hyperboloid are co-dependent; they satisfy the relation  $\langle \mathbf{x}, \mathbf{x} \rangle_{\mathcal{L}} = 1/K$ . This implies, that if we are given a vector with  $n$  coordinates  $\tilde{\mathbf{x}} = (x_2, \dots, x_{n+1})$ , we can compute the missing coordinate to place it onto the hyperboloid:

$$x_1 = \sqrt{\|\tilde{\mathbf{x}}\|_2^2 - \frac{1}{K}}.$$

This is useful for example in the case of orthogonally projecting points from  $\mathcal{T}_{\mu_0} \mathbb{H}_K^n$  onto the manifold.

### Distance function

The distance function in  $\mathbb{H}_K^n$  is

$$d_{\mathbb{H}}^K(\mathbf{x}, \mathbf{y}) = R \cdot \theta_{\mathbf{x}, \mathbf{y}} = R \cosh^{-1} \left( -\frac{\langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}}}{R^2} \right) = \frac{1}{\sqrt{-K}} \cosh^{-1}(-K \langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}}).$$

**Remark (About the divergence of points in  $\mathbb{H}_K^n$ )** *Since the points on the hyperboloid  $\mathbf{x} \in \mathbb{H}_K^n$  are norm-constrained to*

$$\langle \mathbf{x}, \mathbf{x} \rangle_{\mathcal{L}} = \frac{1}{K},$$

*all the points on the hyperboloid go to infinity as  $K$  goes to  $0^-$  from below:*

$$\lim_{K \rightarrow 0^-} \langle \mathbf{x}, \mathbf{x} \rangle_{\mathcal{L}} = -\infty.$$

This confirms the intuition that the hyperboloid grows “flatter”, but to do that, it has to go away from the origin of the coordinate space  $\mathbf{0}$ . A good example of a point that diverges is the origin of the hyperboloid  $\boldsymbol{\mu}_0^K = (1/K, 0, \dots, 0)^T = (R, 0, \dots, 0)^T$ . That makes this model unsuitable for trying to learn sign-agnostic curvatures, similarly to the hypersphere.

### Exponential map

The exponential map in  $\mathbb{H}_K^n$  is

$$\exp_{\mathbf{x}}^K(\mathbf{v}) = \cosh\left(\frac{\|\mathbf{v}\|_{\mathcal{L}}}{R}\right) \mathbf{x} + \sinh\left(\frac{\|\mathbf{v}\|_{\mathcal{L}}}{R}\right) \frac{R\mathbf{v}}{\|\mathbf{v}\|_{\mathcal{L}}},$$

and in the case of  $\mathbf{x} := \boldsymbol{\mu}_0 = (R, 0, \dots, 0)^T$ :

$$\exp_{\boldsymbol{\mu}_0}^K(\mathbf{v}) = \left( \cosh\left(\frac{\|\tilde{\mathbf{v}}\|_2}{R}\right) R; \sinh\left(\frac{\|\tilde{\mathbf{v}}\|_2}{R}\right) \frac{R}{\|\tilde{\mathbf{v}}\|_2} \tilde{\mathbf{v}}^T \right)^T,$$

where  $\mathbf{v} = (0; \tilde{\mathbf{v}}^T)^T$  and  $\|\mathbf{v}\|_{\mathcal{L}} = \|\mathbf{v}\|_2 = \|\tilde{\mathbf{v}}\|_2$ .

**Theorem A.4 (Exponential map in  $\mathbb{H}_K^n$ )** *For all  $\mathbf{x} \in \mathbb{H}_K^n$  and  $\mathbf{v} \in \mathcal{T}_{\mathbf{x}}\mathbb{H}_K^n$ , the exponential map in  $\mathbb{H}_K^n$  maps tangent space vectors  $\mathbf{v}$  to the manifold:*

$$\|\exp_{\mathbf{x}}^K(\mathbf{v})\|_{\mathcal{L}}^2 = 1/K = -R^2.$$

### Proof

$$\begin{aligned} & \|\exp_{\mathbf{x}}^K(\mathbf{v})\|_{\mathcal{L}}^2 = \\ & = \left\| \cosh\left(\frac{\|\mathbf{v}\|_{\mathcal{L}}}{R}\right) \mathbf{x} + \sinh\left(\frac{\|\mathbf{v}\|_{\mathcal{L}}}{R}\right) \frac{R\mathbf{v}}{\|\mathbf{v}\|_{\mathcal{L}}} \right\|_{\mathcal{L}}^2 \\ & = \left\| \cosh\left(\frac{\|\mathbf{v}\|_{\mathcal{L}}}{R}\right) \mathbf{x} \right\|_{\mathcal{L}}^2 + \left\| \sinh\left(\frac{\|\mathbf{v}\|_{\mathcal{L}}}{R}\right) \frac{R\mathbf{v}}{\|\mathbf{v}\|_{\mathcal{L}}} \right\|_{\mathcal{L}}^2 \\ & \quad + 2 \left\langle \cosh\left(\frac{\|\mathbf{v}\|_{\mathcal{L}}}{R}\right) \mathbf{x}, \sinh\left(\frac{\|\mathbf{v}\|_{\mathcal{L}}}{R}\right) \frac{R\mathbf{v}}{\|\mathbf{v}\|_{\mathcal{L}}} \right\rangle_{\mathcal{L}} \\ & = \cosh^2\left(\frac{\|\mathbf{v}\|_{\mathcal{L}}}{R}\right) \|\mathbf{x}\|_{\mathcal{L}}^2 + \sinh^2\left(\frac{\|\mathbf{v}\|_{\mathcal{L}}}{R}\right) \frac{R^2}{\|\mathbf{v}\|_{\mathcal{L}}^2} \|\mathbf{v}\|_{\mathcal{L}}^2 \\ & \quad + 2 \underbrace{\cosh\left(\frac{\|\mathbf{v}\|_{\mathcal{L}}}{R}\right) \sinh\left(\frac{\|\mathbf{v}\|_{\mathcal{L}}}{R}\right) \frac{R}{\|\mathbf{v}\|_{\mathcal{L}}} \langle \mathbf{x}, \mathbf{v} \rangle_{\mathcal{L}}}_{\mathbf{v} \in \mathcal{T}_{\mathbf{x}}\mathbb{H}_K^n \implies \langle \mathbf{x}, \mathbf{v} \rangle_{\mathcal{L}} = 0} \\ & = -R^2 \cosh^2\left(\frac{\|\mathbf{v}\|_{\mathcal{L}}}{R}\right) + R^2 \sinh^2\left(\frac{\|\mathbf{v}\|_{\mathcal{L}}}{R}\right) \\ & = -R^2 \left( \cosh^2\left(\frac{\|\mathbf{v}\|_{\mathcal{L}}}{R}\right) - \sinh^2\left(\frac{\|\mathbf{v}\|_{\mathcal{L}}}{R}\right) \right) \\ & = -R^2. \end{aligned}$$

□

**Theorem A.5 (Logarithmic map in  $\mathbb{H}_K^n$ )** For all  $\mathbf{x}, \mathbf{y} \in \mathbb{H}_K^n$ , the logarithmic map in  $\mathbb{H}_K^n$  maps  $\mathbf{y}$  to a tangent vector at  $\mathbf{x}$ :

$$\log_{\mathbf{x}}^K(\mathbf{y}) = \frac{\cosh^{-1}(\alpha)}{\sqrt{\alpha^2 - 1}}(\mathbf{y} - \alpha\mathbf{x}),$$

where  $\alpha = K \langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}}$ .

**Proof** We show the detailed derivation of the logarithmic map as an inverse function to the exponential map  $\log_{\mathbf{x}}(\mathbf{y}) = \exp_{\mathbf{x}}^{-1}(\mathbf{y})$ , adapted from (Nagano et al., 2019).

As mentioned previously,

$$\mathbf{y} = \exp_{\mathbf{x}}^K(\mathbf{v}) = \cosh\left(\frac{\|\mathbf{v}\|_{\mathcal{L}}}{R}\right)\mathbf{x} + \sinh\left(\frac{\|\mathbf{v}\|_{\mathcal{L}}}{R}\right)\frac{R\mathbf{v}}{\|\mathbf{v}\|_{\mathcal{L}}}.$$

Solving for  $\mathbf{v}$ , we obtain

$$\mathbf{v} = \frac{\|\mathbf{v}\|_{\mathcal{L}}}{R \sinh\left(\frac{\|\mathbf{v}\|_{\mathcal{L}}}{R}\right)}\left(\mathbf{y} - \cosh\left(\frac{\|\mathbf{v}\|_{\mathcal{L}}}{R}\right)\mathbf{x}\right).$$

However, we still need to rewrite  $\|\mathbf{v}\|_{\mathcal{L}}$  in evaluable terms:

$$0 = \langle \mathbf{x}, \mathbf{v} \rangle_{\mathcal{L}} = \frac{\|\mathbf{v}\|_{\mathcal{L}}}{R \sinh\left(\frac{\|\mathbf{v}\|_{\mathcal{L}}}{R}\right)}\left(\langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}} - \cosh\left(\frac{\|\mathbf{v}\|_{\mathcal{L}}}{R}\right)\underbrace{\langle \mathbf{x}, \mathbf{x} \rangle_{\mathcal{L}}}_{-R^2}\right),$$

hence

$$\cosh\left(\frac{\|\mathbf{v}\|_{\mathcal{L}}}{R}\right) = -\frac{1}{R^2}\langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}},$$

and therefore

$$\|\mathbf{v}\|_{\mathcal{L}} = R \cosh^{-1}\left(-\frac{1}{R^2}\langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}}\right) = \frac{1}{\sqrt{-K}} \cosh^{-1}(K \langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}}) = d_{\mathbb{H}}^K(\mathbf{x}, \mathbf{y}).$$

Plugging the result back into the first equation, we obtain

$$\begin{aligned} \mathbf{v} &= \frac{\|\mathbf{v}\|_{\mathcal{L}}}{R \sinh\left(\frac{\|\mathbf{v}\|_{\mathcal{L}}}{R}\right)}\left(\mathbf{y} - \cosh\left(\frac{\|\mathbf{v}\|_{\mathcal{L}}}{R}\right)\mathbf{x}\right) \\ &= \frac{R \cosh^{-1}(\alpha)}{R \sinh\left(\frac{1}{R}R \cosh^{-1}(\alpha)\right)}\left(\mathbf{y} - \cosh\left(\frac{1}{R}R \cosh^{-1}(\alpha)\right)\mathbf{x}\right) \\ &= \frac{\cosh^{-1}(\alpha)}{\sinh(\cosh^{-1}(\alpha))}(\mathbf{y} - \cosh(\cosh^{-1}(\alpha))\mathbf{x}) \\ &= \frac{\cosh^{-1}(\alpha)}{\sqrt{\alpha^2 - 1}}(\mathbf{y} - \alpha\mathbf{x}), \end{aligned}$$



where  $\alpha = -\frac{1}{R^2} \langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}} = K \langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}}$ , and the last equality assumes  $|\alpha| > 1$ . This assumption holds, since for all points  $\mathbf{x}, \mathbf{y} \in \mathbb{H}_K^n$  it holds that  $\langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}} \leq -R^2$ , and  $\langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}} = -R^2$  if and only if  $\mathbf{x} = \mathbf{y}$ , due to Cauchy-Schwarz (Ratcliffe, 2006, Theorem 3.1.6). Hence, the only case where this would be a problem would be if  $\mathbf{x} = \mathbf{y}$ , but it is clear that the result in that case is  $\mathbf{u} = \mathbf{0}$ .  $\square$

**Theorem A.6** ( $\log_x^K$  is the inverse of  $\exp_x^K$  in  $\mathbb{H}_K^n$ )

$$\log_x^K(\exp_x^K(\mathbf{v})) = \mathbf{v}.$$

**Proof**

$$\begin{aligned} \log_x^K(\exp_x^K(\mathbf{v})) &= \\ &= \log_x^K \left( \cosh \left( \frac{\|\mathbf{v}\|_{\mathcal{L}}}{R} \right) \mathbf{x} + \sinh \left( \frac{\|\mathbf{v}\|_{\mathcal{L}}}{R} \right) \frac{R\mathbf{v}}{\|\mathbf{v}\|_{\mathcal{L}}} \right) \\ &= \frac{\cosh^{-1}(\alpha)}{\sqrt{\alpha^2 - 1}} \left( \cosh \left( \frac{\|\mathbf{v}\|_{\mathcal{L}}}{R} \right) \mathbf{x} + \sinh \left( \frac{\|\mathbf{v}\|_{\mathcal{L}}}{R} \right) \frac{R\mathbf{v}}{\|\mathbf{v}\|_{\mathcal{L}}} - \alpha \mathbf{x} \right) \\ &= \frac{\|\mathbf{v}\|_{\mathcal{L}}}{R \sqrt{\cosh^2 \left( \frac{\|\mathbf{v}\|_{\mathcal{L}}}{R} \right) - 1}} \sinh \left( \frac{\|\mathbf{v}\|_{\mathcal{L}}}{R} \right) \frac{R\mathbf{v}}{\|\mathbf{v}\|_{\mathcal{L}}} \\ &= \frac{\|\mathbf{v}\|_{\mathcal{L}}}{R \sinh \left( \frac{\|\mathbf{v}\|_{\mathcal{L}}}{R} \right)} \left( \sinh \left( \frac{\|\mathbf{v}\|_{\mathcal{L}}}{R} \right) \frac{R\mathbf{v}}{\|\mathbf{v}\|_{\mathcal{L}}} \right) \\ &= \mathbf{v}, \end{aligned}$$

where

$$\begin{aligned} \alpha &= -\frac{\langle \mathbf{x}, \exp_x^K(\mathbf{v}) \rangle_2}{R^2} \\ &= -\frac{1}{R^2} \cosh \left( \frac{\|\mathbf{v}\|_{\mathcal{L}}}{R} \right) \underbrace{\langle \mathbf{x}, \mathbf{x} \rangle_{\mathcal{L}}}_{=-R^2} - \underbrace{\sinh \left( \frac{\|\mathbf{v}\|_{\mathcal{L}}}{R} \right) \frac{1}{R\|\mathbf{v}\|_{\mathcal{L}}} \langle \mathbf{x}, \mathbf{v} \rangle_{\mathcal{L}}}_{\mathbf{v} \in \mathcal{T}_x \mathbb{H}_K^n \implies \langle \mathbf{x}, \mathbf{v} \rangle_{\mathcal{L}} = 0} \\ &= \cosh \left( \frac{\|\mathbf{v}\|_{\mathcal{L}}}{R} \right). \end{aligned} \quad \square$$

**Theorem A.7 (Length preservation property of  $\exp_x^K$  in  $\mathbb{H}_K^n$ )** For all points on the manifold  $\mathbf{x} \in \mathbb{H}_K^n$  and for all tangent vectors at that point  $\mathbf{v} \in \mathcal{T}_x \mathbb{H}_K^n$  it holds that

$$d_{\mathbb{H}}(\mathbf{x}, \exp_x^K(\mathbf{v})) = \|\mathbf{v}\|_{\mathcal{L}}.$$

**Proof**

$$\begin{aligned}
d_{\mathbb{H}}(\mathbf{x}, \exp_{\mathbf{x}}^K(\mathbf{v})) &= R \cosh^{-1} \left( -\frac{\langle \mathbf{x}, \exp_{\mathbf{x}}^K(\mathbf{v}) \rangle_{\mathcal{L}}}{R^2} \right) \\
&= R \cosh^{-1} \left( \cosh \left( \frac{\|\mathbf{v}\|_{\mathcal{L}}}{R} \right) \right) \\
&= \|\mathbf{v}\|_{\mathcal{L}},
\end{aligned}$$

where the equality

$$-\frac{\langle \mathbf{x}, \exp_{\mathbf{x}}^K(\mathbf{v}) \rangle_{\mathcal{L}}}{R^2} = \cosh \left( \frac{\|\mathbf{v}\|_{\mathcal{L}}}{R} \right)$$

corresponds to the definition of  $\alpha$  in the Proof of Theorem A.6.  $\square$

**Parallel transport**

**Derivation of parallel transport** Using the generic formula for parallel transport in manifolds for  $\mathbf{x}, \mathbf{y} \in \mathcal{M}$  and  $\mathbf{v} \in \mathcal{T}_{\mathbf{x}}\mathcal{M}$

$$\text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}) = \mathbf{v} - \frac{\langle \log_{\mathbf{x}}^K(\mathbf{y}), \mathbf{v} \rangle_{\mathcal{L}}}{d_{\mathcal{M}}(\mathbf{x}, \mathbf{y})} (\log_{\mathbf{x}}^K(\mathbf{y}) + \log_{\mathbf{y}}^K(\mathbf{x})), \quad (\text{A.1})$$

and the logarithmic map formula from Theorem A.5

$$\log_{\mathbf{x}}^K(\mathbf{y}) = \frac{\cosh^{-1}(\alpha)}{\sqrt{\alpha^2 - 1}} (\mathbf{y} - \alpha \mathbf{x}),$$

where  $\alpha = -\frac{1}{R^2} \langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}}$ , we derive parallel transport in  $\mathbb{H}_K^n$ :

$$\text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}) = \mathbf{v} + \frac{\langle \mathbf{y}, \mathbf{v} \rangle_{\mathcal{L}}}{R^2 - \langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}}} (\mathbf{x} + \mathbf{y}).$$

A special form of parallel transport exists for when the source vector is  $\boldsymbol{\mu}_0 = (R, 0, \dots, 0)^T$ :

$$\text{PT}_{\boldsymbol{\mu}_0 \rightarrow \mathbf{y}}^K(\mathbf{v}) = \mathbf{v} + \frac{\langle \mathbf{y}, \mathbf{v} \rangle_2}{R^2 + Ry_1} \begin{pmatrix} y_1 + R \\ y_2 \\ \vdots \\ y_{n+1} \end{pmatrix}.$$

**Theorem A.8** ( $\text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K$  transports points to the tangent space of  $\mathbf{y}$  in  $\mathbb{H}_K^n$ )  
For all points on the manifold  $\mathbf{x}, \mathbf{y} \in \mathbb{H}_K^n$  and a tangent vector  $\mathbf{v} \in \mathcal{T}_{\mathbf{x}}\mathbb{H}_K^n$  it holds that

$$\text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}) \in \mathcal{T}_{\mathbf{y}}\mathbb{H}_K^n.$$

**Proof**

$$\begin{aligned}
\langle \mathbf{y}, \text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}) \rangle_{\mathcal{L}} &= \left\langle \mathbf{y}, \mathbf{v} + \frac{\langle \mathbf{y}, \mathbf{v} \rangle_{\mathcal{L}}}{R^2 - \langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}}} (\mathbf{x} + \mathbf{y}) \right\rangle_{\mathcal{L}} \\
&= \langle \mathbf{y}, \mathbf{v} \rangle_{\mathcal{L}} + \frac{\langle \mathbf{y}, \mathbf{v} \rangle_{\mathcal{L}}}{R^2 - \langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}}} \langle \mathbf{y}, \mathbf{x} + \mathbf{y} \rangle_{\mathcal{L}} \\
&= \langle \mathbf{y}, \mathbf{v} \rangle_{\mathcal{L}} + \frac{\langle \mathbf{y}, \mathbf{v} \rangle_{\mathcal{L}}}{R^2 - \langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}}} (\langle \mathbf{y}, \mathbf{x} \rangle_{\mathcal{L}} + \langle \mathbf{y}, \mathbf{y} \rangle_{\mathcal{L}}) \\
&= \langle \mathbf{y}, \mathbf{v} \rangle_{\mathcal{L}} + \frac{\langle \mathbf{y}, \mathbf{v} \rangle_{\mathcal{L}}}{R^2 - \langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}}} (\langle \mathbf{y}, \mathbf{x} \rangle_{\mathcal{L}} - R^2) \\
&= \langle \mathbf{y}, \mathbf{v} \rangle_{\mathcal{L}} - \frac{\langle \mathbf{y}, \mathbf{v} \rangle_{\mathcal{L}}}{R^2 - \langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}}} (R^2 - \langle \mathbf{y}, \mathbf{x} \rangle_{\mathcal{L}}) \\
&= \langle \mathbf{y}, \mathbf{v} \rangle_{\mathcal{L}} - \langle \mathbf{y}, \mathbf{v} \rangle_{\mathcal{L}} = 0,
\end{aligned}$$

which implies  $\text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}) \in \mathcal{T}_{\mathbf{y}} \mathbb{H}_K^n$ .  $\square$

**Theorem A.9 (Parallel transport preserves angles in  $\mathbb{H}_K^n$ )** For all points on the manifold  $\mathbf{x}, \mathbf{y} \in \mathbb{H}_K^n$  and tangent vectors  $\mathbf{v}, \mathbf{v}' \in \mathcal{T}_{\mathbf{x}} \mathbb{H}_K^n$  it holds that

$$\langle \text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}), \text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}') \rangle_{\mathcal{L}} = \langle \mathbf{v}, \mathbf{v}' \rangle_{\mathcal{L}}.$$

**Proof**

$$\begin{aligned}
&\langle \text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}), \text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}') \rangle_{\mathcal{L}} = \\
&= \left\langle \mathbf{v} + \frac{\langle \mathbf{y}, \mathbf{v} \rangle_{\mathcal{L}}}{R^2 - \langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}}} (\mathbf{x} + \mathbf{y}), \mathbf{v}' + \frac{\langle \mathbf{y}, \mathbf{v}' \rangle_{\mathcal{L}}}{R^2 - \langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}}} (\mathbf{x} + \mathbf{y}) \right\rangle_{\mathcal{L}} \\
&= \langle \mathbf{v}, \mathbf{v}' \rangle_{\mathcal{L}} \\
&\quad + \frac{\langle \mathbf{y}, \mathbf{v}' \rangle_{\mathcal{L}}}{R^2 - \langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}}} \underbrace{\langle \mathbf{v}, \mathbf{x} + \mathbf{y} \rangle_{\mathcal{L}}}_{\langle \mathbf{v}, \mathbf{y} \rangle_{\mathcal{L}}} + \frac{\langle \mathbf{y}, \mathbf{v} \rangle_{\mathcal{L}}}{R^2 - \langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}}} \underbrace{\langle \mathbf{v}', \mathbf{x} + \mathbf{y} \rangle_{\mathcal{L}}}_{\langle \mathbf{v}', \mathbf{y} \rangle_{\mathcal{L}}} \\
&\quad + \frac{\langle \mathbf{y}, \mathbf{v} \rangle_{\mathcal{L}} \langle \mathbf{y}, \mathbf{v}' \rangle_{\mathcal{L}}}{(R^2 - \langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}})^2} \underbrace{\langle \mathbf{x} + \mathbf{y}, \mathbf{x} + \mathbf{y} \rangle_{\mathcal{L}}}_{-R^2 + 2\langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}} - R^2} \\
&= \langle \mathbf{v}, \mathbf{v}' \rangle_{\mathcal{L}} + 2 \frac{\langle \mathbf{y}, \mathbf{v}' \rangle_{\mathcal{L}} \langle \mathbf{y}, \mathbf{v} \rangle_{\mathcal{L}}}{R^2 - \langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}}} - 2 \frac{\langle \mathbf{y}, \mathbf{v} \rangle_{\mathcal{L}} \langle \mathbf{y}, \mathbf{v}' \rangle_{\mathcal{L}}}{(R^2 - \langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}})^2} (R^2 - \langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}}) \\
&= \langle \mathbf{v}, \mathbf{v}' \rangle_{\mathcal{L}}. \quad \square
\end{aligned}$$

**Corollary (Parallel transport in  $\mathbb{H}_K^n$  is norm-preserving)**

$$\|\text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v})\|_{\mathcal{L}} = \|\mathbf{v}\|_{\mathcal{L}}.$$

**Proof**

$$\|\text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v})\|_{\mathcal{L}}^2 = \langle \text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}), \text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}) \rangle_{\mathcal{L}} = \langle \mathbf{v}, \mathbf{v} \rangle_{\mathcal{L}} = \|\mathbf{v}\|_{\mathcal{L}}^2,$$

where the second equality corresponds to Theorem A.9.  $\square$

Möbius add.	$\mathbf{x} \oplus_K \mathbf{y} = \frac{(1 - 2K \langle \mathbf{x}, \mathbf{y} \rangle_2 - K \ \mathbf{y}\ _2^2) \mathbf{x} + (1 + K \ \mathbf{x}\ _2^2) \mathbf{y}}{1 - 2K \langle \mathbf{x}, \mathbf{y} \rangle_2 + K^2 \ \mathbf{x}\ _2^2 \ \mathbf{y}\ _2^2}$
Distance	$d_{\mathbb{P}}(\mathbf{x}, \mathbf{y}) = \frac{1}{\sqrt{-K}} \cosh^{-1} \left( 1 - \frac{2K \ \mathbf{x} - \mathbf{y}\ _2^2}{(1 + K \ \mathbf{x}\ _2^2)(1 + K \ \mathbf{y}\ _2^2)} \right)$
Gyr. dist.	$d_{\mathbb{P}_{\text{gyr}}}(\mathbf{x}, \mathbf{y}) = \frac{2}{\sqrt{-K}} \tanh^{-1}(\sqrt{-K} \ \mathbf{x} \oplus_K \mathbf{y}\ _2)$
Lambda	$\lambda_{\mathbf{x}}^K = \frac{2}{1 + K \ \mathbf{x}\ _2^2}$
Exp. map	$\exp_{\mathbf{x}}^K(\mathbf{v}) = \mathbf{x} \oplus_K \left( \tanh \left( \sqrt{-K} \frac{\lambda_{\mathbf{x}}^K \ \mathbf{v}\ _2}{2} \right) \frac{\mathbf{v}}{\sqrt{-K} \ \mathbf{v}\ _2} \right)$
Log. map	$\log_{\mathbf{x}}^K(\mathbf{y}) = \frac{2}{\sqrt{-K} \lambda_{\mathbf{x}}^K} \tanh^{-1} \left( \sqrt{-K} \ \mathbf{z}\ _2 \right) \frac{\mathbf{z}}{\ \mathbf{z}\ _2},$ where $\mathbf{z} = \mathbf{x} \oplus_K \mathbf{y}$
Gyration	$\text{gyr}[\mathbf{x}, \mathbf{y}]\mathbf{v} = \ominus_K(\mathbf{x} \oplus_K \mathbf{y}) \oplus_K (\mathbf{x} \oplus_K (\mathbf{y} \oplus_K \mathbf{v}))$
Par. transp.	$\text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}) = \frac{\lambda_{\mathbf{x}}^K}{\lambda_{\mathbf{y}}^K} \text{gyr}[\mathbf{y}, \mathbf{x}]\mathbf{v}$ $\text{PT}_{\mu_0 \rightarrow \mathbf{y}}^K(\mathbf{v}) = \frac{2}{\lambda_{\mathbf{y}}^K} \mathbf{v}, \quad \text{PT}_{\mathbf{x} \rightarrow \mu_0}^K(\mathbf{v}) = \frac{\lambda_{\mathbf{x}}^K}{2} \mathbf{v}$

Table A.3: Poincaré ball operations.

## A.2.2 Poincaré ball

An overview of all the necessary operations can be found in Table A.3.

Do note, that all the theorems for the projected hypersphere are essentially trivial corollaries of their equivalents in the Poincaré ball (and vice-versa) (Section A.3.2). Notable differences include the fact that  $R^2 = -\frac{1}{K}$ , not  $R^2 = \frac{1}{K}$ , and all the operations use the hyperbolic trigonometric functions  $\sinh$ ,  $\cosh$ , and  $\tanh$ , instead of their Euclidean counterparts. Also, we often leverage the “hyperbolic” Pythagorean theorem, in the form  $\cosh^2(\alpha) - \sinh^2(\alpha) = 1$ .

### Stereographic projection

**Theorem A.11 (Stereographic backprojected points of  $\mathbb{P}_K^n$  belong to  $\mathbb{H}_K^n$ )**

For all  $\mathbf{y} \in \mathbb{P}_K^n$ ,

$$\|\rho_K^{-1}(\mathbf{y})\|_{\mathcal{L}}^2 = \frac{1}{K}.$$

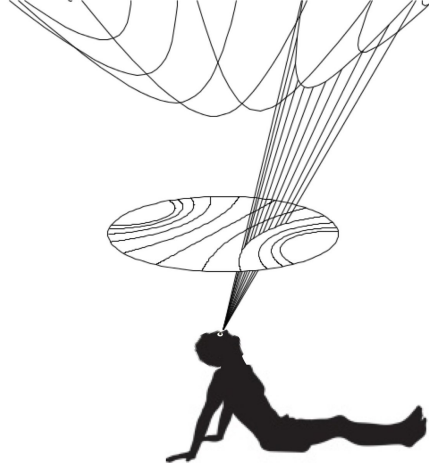


Figure A.1: “The Poincaré disk is a model to ‘see’ a 2D hyperbolic space by approximating what a hyperbola looks like from below,” Pampena (2016).

**Proof**

$$\begin{aligned}
\|\rho_K^{-1}(\mathbf{y})\|_{\mathcal{L}}^2 &= \left\| \left( \frac{1}{\sqrt{|K|}} \frac{K \|\mathbf{y}\|_2^2 - 1}{K \|\mathbf{y}\|_2^2 + 1}; \frac{2\mathbf{y}^T}{K \|\mathbf{y}\|_2^2 + 1} \right)^T \right\|_{\mathcal{L}}^2 \\
&= - \left( \frac{1}{\sqrt{|K|}} \frac{K \|\mathbf{y}\|_2^2 - 1}{K \|\mathbf{y}\|_2^2 + 1} \right)^2 + \frac{4 \|\mathbf{y}\|_2^2}{(K \|\mathbf{y}\|_2^2 + 1)^2} \\
&= \frac{1}{|K|} \frac{-(K \|\mathbf{y}\|_2^2 - 1)^2 + 4|K| \|\mathbf{y}\|_2^2}{(K \|\mathbf{y}\|_2^2 + 1)^2} \\
&= \frac{1}{-K} \frac{-(K \|\mathbf{y}\|_2^2 - 1)^2 - 4K \|\mathbf{y}\|_2^2}{(K \|\mathbf{y}\|_2^2 + 1)^2} \\
&= \frac{1}{K} \frac{(K \|\mathbf{y}\|_2^2 - 1)^2 + 4K \|\mathbf{y}\|_2^2}{(K \|\mathbf{y}\|_2^2 + 1)^2} \\
&= \frac{1}{K} \frac{K^2 \|\mathbf{y}\|_2^4 + 2K \|\mathbf{y}\|_2^2 + 1}{(K \|\mathbf{y}\|_2^2 + 1)^2} \\
&= \frac{1}{K} \frac{(K \|\mathbf{y}\|_2^2 + 1)^2}{(K \|\mathbf{y}\|_2^2 + 1)^2} = \frac{1}{K}.
\end{aligned}$$

□

### Distance function

The distance function in  $\mathbb{P}_K^n$  is (derived from the hyperboloid distance function using the stereographic projection  $\rho_K$ ):

$$\begin{aligned} d_{\mathbb{P}}(\mathbf{x}, \mathbf{y}) &= d_{\mathbb{H}}(\rho_K^{-1}(\mathbf{x}), \rho_K^{-1}(\mathbf{y})) \\ &= \frac{1}{\sqrt{-K}} \cosh^{-1} \left( 1 - \frac{2K \|\mathbf{x} - \mathbf{y}\|_2^2}{(1 + K \|\mathbf{x}\|_2^2)(1 + K \|\mathbf{y}\|_2^2)} \right) \\ &= R \cosh^{-1} \left( 1 + \frac{2R^2 \|\mathbf{x} - \mathbf{y}\|_2^2}{(R^2 - \|\mathbf{x}\|_2^2)(R^2 - \|\mathbf{y}\|_2^2)} \right) \end{aligned}$$

**Theorem A.12 (Distance equivalence in  $\mathbb{P}_K^n$ )** For all  $K < 0$  and for all pairs of points  $\mathbf{x}, \mathbf{y} \in \mathbb{P}_K^n$ , the Poincaré distance between them equals the gyrospace distance

$$d_{\mathbb{P}}(\mathbf{x}, \mathbf{y}) = d_{\mathbb{P}_{gyr}}(\mathbf{x}, \mathbf{y}).$$

**Proof** Proven using Mathematica (File: `distance_limits.ws`), proof involves heavy algebra.  $\square$

**Theorem A.13 (Gyrospace distance converges to Euclidean in  $\mathbb{P}_K^n$ )** For any fixed pair of points  $\mathbf{x}, \mathbf{y} \in \mathbb{P}_K^n$ , the Poincaré gyrospace distance between them converges to the Euclidean distance in the limit (up to a constant) as  $K \rightarrow 0^-$ :

$$\lim_{K \rightarrow 0^-} d_{\mathbb{P}_{gyr}}(\mathbf{x}, \mathbf{y}) = 2 \|\mathbf{x} - \mathbf{y}\|_2.$$

**Proof**

$$\begin{aligned} \lim_{K \rightarrow 0^-} d_{\mathbb{P}_{gyr}}(\mathbf{x}, \mathbf{y}) &= 2 \lim_{K \rightarrow 0^-} \left[ \frac{\tanh^{-1}(\sqrt{-K} \|\mathbf{x} \oplus_K \mathbf{y}\|_2)}{\sqrt{-K}} \right] \\ &= 2 \lim_{K \rightarrow 0^-} \left[ \frac{\tanh^{-1}(\sqrt{-K} \|\mathbf{y} - \mathbf{x}\|_2)}{\sqrt{-K}} \right] \\ &= 2 \|\mathbf{y} - \mathbf{x}\|_2, \end{aligned}$$

where the second equality holds because of the theorem of limits of composed functions, where

$$\begin{aligned} f(a) &= \frac{\tanh^{-1}(a\sqrt{-K})}{\sqrt{-K}} \\ g(K) &= \|\mathbf{x} \oplus_K \mathbf{y}\|_2. \end{aligned}$$

We see that

$$\lim_{K \rightarrow 0^-} g(K) = \|\mathbf{y} - \mathbf{x}\|_2$$

due to Theorem A.37, and

$$\lim_{a \rightarrow \|\mathbf{x} - \mathbf{y}\|_2} f(a) = \frac{\tanh^{-1}(a\sqrt{-K})}{\sqrt{-K}}$$

Additionally for the last equality, we need the fact that

$$\lim_{x \rightarrow 0} \frac{\tanh^{-1}(a\sqrt{|x|})}{\sqrt{|x|}} = a. \quad \square$$

**Theorem A.14 (Distance converges to Euclidean as  $K \rightarrow 0^-$  in  $\mathbb{P}_K^n$ )** For any fixed pair of points  $\mathbf{x}, \mathbf{y} \in \mathbb{P}_K^n$ , the Poincaré distance between them converges to the Euclidean distance in the limit (up to a constant) as  $K \rightarrow 0^-$ :

$$\lim_{K \rightarrow 0^-} d_{\mathbb{P}}(\mathbf{x}, \mathbf{y}) = 2 \|\mathbf{x} - \mathbf{y}\|_2.$$

**Proof** Theorem A.12 and A.13. □

### Exponential map

As derived and proven in Ganea et al. (2018a), the exponential map in  $\mathbb{P}_K^n$  and its inverse is

$$\begin{aligned} \exp_{\mathbf{x}}^K(\mathbf{v}) &= \mathbf{x} \oplus_K \left( \tanh \left( \sqrt{-K} \frac{\lambda_{\mathbf{x}}^K \|\mathbf{v}\|_2}{2} \right) \frac{\mathbf{v}}{\sqrt{-K} \|\mathbf{v}\|_2} \right) \\ \log_{\mathbf{x}}^K(\mathbf{y}) &= \frac{2}{\sqrt{-K} \lambda_{\mathbf{x}}^K} \tanh^{-1} \left( \sqrt{-K} \|\mathbf{x} \oplus_K \mathbf{y}\|_2 \right) \frac{-\mathbf{x} \oplus_K \mathbf{y}}{\|\mathbf{x} \oplus_K \mathbf{y}\|_2} \end{aligned}$$

In the case of  $\mathbf{x} := \boldsymbol{\mu}_0 = (0, \dots, 0)^T$  they simplify to:

$$\begin{aligned} \exp_{\boldsymbol{\mu}_0}^K(\mathbf{v}) &= \tanh \left( \sqrt{-K} \|\mathbf{v}\|_2 \right) \frac{\mathbf{v}}{\sqrt{-K} \|\mathbf{v}\|_2} \\ \log_{\boldsymbol{\mu}_0}^K(\mathbf{y}) &= \tanh^{-1} \left( \sqrt{-K} \|\mathbf{y}\|_2 \right) \frac{\mathbf{y}}{\|\mathbf{y}\|_2}. \end{aligned}$$

**Theorem A.15 (Length preservation property of  $\exp_{\mathbf{x}}^K$  in  $\mathbb{P}_K^n$ )** For all points on the manifold  $\mathbf{x} \in \mathbb{P}_K^n$  and for all tangent vectors at that point  $\mathbf{v} \in \mathcal{T}_{\mathbf{x}}\mathbb{P}_K^n$  it holds that

$$d_{\mathbb{P}_{gr}}(\mathbf{x}, \exp_{\mathbf{x}}^K(\mathbf{v})) = \lambda_{\mathbf{x}}^K \|\mathbf{v}\|_2.$$

**Proof**

$$\begin{aligned}
d_{\mathbb{P}_{\text{gyr}}}(\mathbf{x}, \exp_{\mathbf{x}}^K(\mathbf{v})) &= \\
&= \frac{2}{\sqrt{-K}} \tanh^{-1} \left( \sqrt{-K} \left\| -\mathbf{x} \oplus_K \exp_{\mathbf{x}}^K(\mathbf{v}) \right\|_2 \right) \\
&= \frac{2}{\sqrt{-K}} \tanh^{-1} \left( \sqrt{-K} \left\| -\mathbf{x} \oplus_K \left( \mathbf{x} \oplus_K \left( \tanh \left( \sqrt{-K} \frac{\lambda_{\mathbf{x}}^K \|\mathbf{v}\|_2}{2} \right) \frac{\mathbf{v}}{\sqrt{-K} \|\mathbf{v}\|_2} \right) \right) \right\|_2 \right) \\
&= \frac{2}{\sqrt{-K}} \tanh^{-1} \left( \sqrt{-K} \left\| \tanh \left( \sqrt{-K} \frac{\lambda_{\mathbf{x}}^K \|\mathbf{v}\|_2}{2} \right) \frac{\mathbf{v}}{\sqrt{-K} \|\mathbf{v}\|_2} \right\|_2 \right) \\
&= \frac{2}{\sqrt{-K}} \tanh^{-1} \left( \sqrt{-K} \tanh \left( \sqrt{-K} \frac{\lambda_{\mathbf{x}}^K \|\mathbf{v}\|_2}{2} \right) \frac{\|\mathbf{v}\|_2}{\sqrt{-K} \|\mathbf{v}\|_2} \right) \\
&= \frac{2}{\sqrt{-K}} \tanh^{-1} \left( \tanh \left( \sqrt{-K} \frac{\lambda_{\mathbf{x}}^K \|\mathbf{v}\|_2}{2} \right) \right) \\
&= \frac{2}{\sqrt{-K}} \sqrt{-K} \frac{\lambda_{\mathbf{x}}^K \|\mathbf{v}\|_2}{2} \\
&= \lambda_{\mathbf{x}}^K \|\mathbf{v}\|_2,
\end{aligned}$$

where the third equality holds because of the left-cancellation law (Ganea et al., 2018a, Section 2.3).  $\square$

**Parallel transport**

Ganea et al. (2018a); Kochurov et al. (2019) have also derived and implemented the parallel transport operation for the Poincaré ball:

$$\begin{aligned}
\text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}) &= \frac{\lambda_{\mathbf{x}}^K}{\lambda_{\mathbf{y}}^K} \text{gyr}[\mathbf{y}, -\mathbf{x}]\mathbf{v}, \\
\text{PT}_{\mu_0 \rightarrow \mathbf{y}}^K(\mathbf{v}) &= \frac{2}{\lambda_{\mathbf{y}}^K} \mathbf{v}, \\
\text{PT}_{\mathbf{x} \rightarrow \mu_0}^K(\mathbf{v}) &= \frac{\lambda_{\mathbf{x}}^K}{2} \mathbf{v},
\end{aligned}$$

where

$$\text{gyr}[\mathbf{x}, \mathbf{y}]\mathbf{v} = -(\mathbf{x} \oplus_K \mathbf{y}) \oplus_K (\mathbf{x} \oplus_K (\mathbf{y} \oplus_K \mathbf{v}))$$

is the gyration operation (Ungar, 2008, Definition 1.11).

**Theorem A.16 (Parallel transport and its inverse in  $\mathbb{P}_{\mathbf{K}}^n$ )**

$$\text{PT}_{\mathbf{y} \rightarrow \mathbf{x}}^K(\text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v})) = \mathbf{v}.$$



**Proof** We only use this fact for  $\mathbf{x}$  or  $\mathbf{y}$  equal to  $\boldsymbol{\mu}_0$ , and for that, it is trivial. Otherwise, one can prove it using the properties from Ungar (2008).  $\square$

Unfortunately, on the Poincaré ball,  $\langle \cdot, \cdot \rangle_{\mathbf{x}}$  has a form that changes with respect to  $\mathbf{x}$ , unlike in the hyperboloid. This means that the following theorems do not hold with respect to  $\langle \cdot, \cdot \rangle_2$ .

**Theorem A.17 (Parallel transport preserves angles in  $\mathbb{P}_K^n$ )** *For all points on the manifold  $\mathbf{x}, \mathbf{y} \in \mathbb{P}_K^n$  and tangent vectors  $\mathbf{v}, \mathbf{v}' \in \mathcal{T}_{\mathbf{x}}\mathbb{P}_K^n$  it holds that*

$$\langle \text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}), \text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}') \rangle_{\mathbf{y}} = \langle \mathbf{v}, \mathbf{v}' \rangle_{\mathbf{x}}.$$

**Proof**

$$\begin{aligned} \langle \text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}), \text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}') \rangle_{\mathbf{y}} &= (\lambda_{\mathbf{y}}^K)^2 \langle \text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}), \text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}') \rangle_2 \\ &= (\lambda_{\mathbf{y}}^K)^2 \left( \frac{\lambda_{\mathbf{x}}^K}{\lambda_{\mathbf{y}}^K} \right)^2 \langle \text{gyr}[\mathbf{y}, -\mathbf{x}]\mathbf{v}, \text{gyr}[\mathbf{y}, -\mathbf{x}]\mathbf{v}' \rangle_2 \\ &= (\lambda_{\mathbf{x}}^K)^2 \langle \mathbf{v}, \mathbf{v}' \rangle_2 \\ &= \langle \mathbf{v}, \mathbf{v}' \rangle_{\mathbf{x}}, \end{aligned}$$

where  $\langle \text{gyr}[\mathbf{x}, \mathbf{y}]\mathbf{v}, \text{gyr}[\mathbf{x}, \mathbf{y}]\mathbf{v}' \rangle_2 = \langle \mathbf{v}, \mathbf{v}' \rangle_2$  is proven in Ungar (2008, Equation 1.32).  $\square$

**Corollary (Parallel transport in  $\mathbb{P}_K^n$  is norm-preserving)**

$$\| \text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}) \|_{\mathbf{y}} = \| \mathbf{v} \|_{\mathbf{x}},$$

and hence

$$\| \text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}) \|_2 = \frac{\lambda_{\mathbf{x}}^K}{\lambda_{\mathbf{y}}^K} \| \mathbf{v} \|_2.$$

**Proof**

$$\begin{aligned} (\lambda_{\mathbf{y}}^K)^2 \| \text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}) \|_2^2 &= \| \text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}) \|_{\mathbf{y}}^2 \\ &= \langle \text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}), \text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}) \rangle_{\mathbf{y}} \\ &= \langle \mathbf{v}, \mathbf{v} \rangle_{\mathbf{x}} \\ &= \| \mathbf{v} \|_{\mathbf{x}}^2 = (\lambda_{\mathbf{x}}^K)^2 \| \mathbf{v} \|_2^2, \end{aligned}$$

where the third equality corresponds to Theorem A.17.  $\square$

Distance	$d_{\mathbb{S}}(\mathbf{x}, \mathbf{y}) = \frac{1}{\sqrt{K}} \cos^{-1}(K \langle \mathbf{x}, \mathbf{y} \rangle_2)$
Exp. map	$\exp_{\mathbf{x}}^K(\mathbf{v}) = \cos\left(\sqrt{K} \ \mathbf{v}\ _2\right) \mathbf{x} + \sin\left(\sqrt{K} \ \mathbf{v}\ _2\right) \frac{\mathbf{v}}{\sqrt{K} \ \mathbf{v}\ _2}$
Log. map	$\log_{\mathbf{x}}^K(\mathbf{y}) = \frac{\cos^{-1}(\alpha)}{\sqrt{1 - \alpha^2}} (\mathbf{y} - \alpha \mathbf{x}), \alpha = K \langle \mathbf{x}, \mathbf{y} \rangle_2$
Par. transp.	$\text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}) = \mathbf{v} - \frac{K \langle \mathbf{y}, \mathbf{v} \rangle_2}{1 + K \langle \mathbf{x}, \mathbf{y} \rangle_2} (\mathbf{x} + \mathbf{y})$

Table A.4: Spherical operations.

## A.3 Spherical geometry

### A.3.1 Hypersphere

An overview of all the necessary operations can be found in Table A.4.

Do note, that all the theorems for the hypersphere are essentially trivial corollaries of their equivalents in the hyperboloid (Section A.2.1). Notable differences include the fact that  $R^2 = \frac{1}{K}$ , not  $R^2 = -\frac{1}{K}$ , and all the operations use the Euclidean trigonometric functions  $\sin$ ,  $\cos$ , and  $\tan$ , instead of their hyperbolic counterparts. Also, we often leverage the Pythagorean theorem, in the form  $\sin^2(\alpha) + \cos^2(\alpha) = 1$ .

### Projections

Due to the definition of the space as a retraction from the ambient space, we can project a generic vector in the ambient space to the hypersphere using the shortest Euclidean distance by normalization:

$$\text{proj}_{\mathbb{S}_K^{n-1}}(\mathbf{x}) = R \frac{\mathbf{x}}{\|\mathbf{x}\|_2} = \frac{\mathbf{x}}{\sqrt{K} \|\mathbf{x}\|_2}.$$

Secondly, the  $n + 1$  coordinates of a point on the sphere are co-dependent; they satisfy the relation  $\langle \mathbf{x}, \mathbf{x} \rangle_2 = 1/K$ . This implies, that if we are given a vector with  $n$  coordinates  $\tilde{\mathbf{x}} = (x_2, \dots, x_{n+1})$ , we can compute the missing coordinate to place it onto the sphere:

$$x_1 = \sqrt{\frac{1}{K} - \|\tilde{\mathbf{x}}\|_2^2}.$$

This is useful for example in the case of orthogonally projecting points from  $\mathcal{T}_{\mu_0} \mathbb{S}_K^n$  onto the manifold.

### Distance function

The distance function in  $\mathbb{S}_K^n$  is

$$d_{\mathbb{S}}^K(\mathbf{x}, \mathbf{y}) = R \cdot \theta_{\mathbf{x}, \mathbf{y}} = R \cos^{-1} \left( \frac{\langle \mathbf{x}, \mathbf{y} \rangle_2}{R^2} \right) = \frac{1}{\sqrt{K}} \cos^{-1} (K \langle \mathbf{x}, \mathbf{y} \rangle_2).$$

**Remark (About the divergence of points in  $\mathbb{S}_K^n$ )** *Since the points on the hypersphere  $\mathbf{x} \in \mathbb{S}_K^n$  are norm-constrained to*

$$\langle \mathbf{x}, \mathbf{x} \rangle_2 = \frac{1}{K},$$

*all the points on the sphere go to infinity as  $K$  goes to  $0^+$  from above:*

$$\lim_{K \rightarrow 0^+} \langle \mathbf{x}, \mathbf{x} \rangle_2 = \infty.$$

This confirms the intuition that the sphere grows “flatter”, but to do that, it has to go away from the origin of the coordinate space  $\mathbf{0}$ . A good example of a point that diverges is the north pole of the sphere  $\boldsymbol{\mu}_0^K = (1/K, 0, \dots, 0)^T = (R, 0, \dots, 0)^T$ . That makes this model unsuitable for trying to learn sign-agnostic curvatures, similarly to the hyperboloid.

### Exponential map

**Theorem A.20 (Exponential map in  $\mathbb{S}_K^n$ )** *For all  $\mathbf{x} \in \mathbb{S}_K^n$  and  $\mathbf{v} \in \mathcal{T}_{\mathbf{x}}\mathbb{S}_K^n$ , the exponential map in  $\mathbb{S}_K^n$  maps tangent space vectors  $\mathbf{v}$  to the manifold:*

$$\|\exp_{\mathbf{x}}^K(\mathbf{v})\|_2^2 = 1/K = R^2.$$

**Proof**

$$\begin{aligned}
& \|\exp_{\mathbf{x}}^K(\mathbf{v})\|_2^2 = \\
& = \left\| \cos\left(\frac{\|\mathbf{v}\|_2}{R}\right) \mathbf{x} + \sin\left(\frac{\|\mathbf{v}\|_2}{R}\right) \frac{R\mathbf{v}}{\|\mathbf{v}\|_2} \right\|_2^2 \\
& = \left\| \cos\left(\frac{\|\mathbf{v}\|_2}{R}\right) \mathbf{x} \right\|_2^2 + \left\| \sin\left(\frac{\|\mathbf{v}\|_2}{R}\right) \frac{R\mathbf{v}}{\|\mathbf{v}\|_2} \right\|_2^2 \\
& \quad + 2 \left\langle \cos\left(\frac{\|\mathbf{v}\|_2}{R}\right) \mathbf{x}, \sin\left(\frac{\|\mathbf{v}\|_2}{R}\right) \frac{R\mathbf{v}}{\|\mathbf{v}\|_2} \right\rangle_2 \\
& = \cos^2\left(\frac{\|\mathbf{v}\|_2}{R}\right) \|\mathbf{x}\|_2^2 + \sin^2\left(\frac{\|\mathbf{v}\|_2}{R}\right) \frac{R^2}{\|\mathbf{v}\|_2^2} \|\mathbf{v}\|_2^2 \\
& \quad + 2 \underbrace{\cos\left(\frac{\|\mathbf{v}\|_2}{R}\right) \sin\left(\frac{\|\mathbf{v}\|_2}{R}\right) \frac{R}{\|\mathbf{v}\|_2} \langle \mathbf{x}, \mathbf{v} \rangle_2}_{\mathbf{v} \in \mathcal{T}_{\mathbf{x}} \mathbb{S}_K^n \implies \langle \mathbf{x}, \mathbf{v} \rangle_2 = 0} \\
& = \cos^2\left(\frac{\|\mathbf{v}\|_2}{R}\right) R^2 + \sin^2\left(\frac{\|\mathbf{v}\|_2}{R}\right) R^2 \\
& = R^2 \left( \cos^2\left(\frac{\|\mathbf{v}\|_2}{R}\right) + \sin^2\left(\frac{\|\mathbf{v}\|_2}{R}\right) \right) \\
& = R^2. \quad \square
\end{aligned}$$

**Theorem A.21 (Logarithmic map in  $\mathbb{S}_K^n$ )** For all  $\mathbf{x}, \mathbf{y} \in \mathbb{S}_K^n$ , the logarithmic map in  $\mathbb{S}_K^n$  maps  $\mathbf{y}$  to a tangent vector at  $\mathbf{x}$ :

$$\log_{\mathbf{x}}^K(\mathbf{y}) = \frac{\cos^{-1}(\alpha)}{\sqrt{1-\alpha^2}} (\mathbf{y} - \alpha \mathbf{x}),$$

where  $\alpha = K \langle \mathbf{x}, \mathbf{y} \rangle_2$ .

**Proof** Analogous to the proof of Theorem A.22.

As mentioned previously,

$$\mathbf{y} = \exp_{\mathbf{x}}^K(\mathbf{v}) = \cos\left(\frac{\|\mathbf{v}\|_2}{R}\right) \mathbf{x} + \sin\left(\frac{\|\mathbf{v}\|_2}{R}\right) \frac{R\mathbf{v}}{\|\mathbf{v}\|_2}.$$

Solving for  $\mathbf{v}$ , we obtain

$$\mathbf{v} = \frac{\|\mathbf{v}\|_2}{R \sin\left(\frac{\|\mathbf{v}\|_2}{R}\right)} \left( \mathbf{y} - \cos\left(\frac{\|\mathbf{v}\|_2}{R}\right) \mathbf{x} \right).$$

However, we still need to rewrite  $\|\mathbf{v}\|_2$  in evaluable terms:

$$0 = \langle \mathbf{x}, \mathbf{v} \rangle_2 = \frac{\|\mathbf{v}\|_2}{R \sin\left(\frac{\|\mathbf{v}\|_2}{R}\right)} \left( \langle \mathbf{x}, \mathbf{y} \rangle_2 - \cos\left(\frac{\|\mathbf{v}\|_2}{R}\right) \underbrace{\langle \mathbf{x}, \mathbf{x} \rangle_2}_{R^2} \right),$$

hence

$$\cos\left(\frac{\|\mathbf{v}\|_2}{R}\right) = \frac{1}{R^2} \langle \mathbf{x}, \mathbf{y} \rangle_2,$$

and therefore

$$\|\mathbf{v}\|_2 = R \cos^{-1}\left(\frac{1}{R^2} \langle \mathbf{x}, \mathbf{y} \rangle_2\right) = \frac{1}{\sqrt{K}} \cos^{-1}(K \langle \mathbf{x}, \mathbf{y} \rangle_2) = d_{\mathbb{S}}^K(\mathbf{x}, \mathbf{y}).$$

Plugging the result back into the first equation, we obtain

$$\begin{aligned} \mathbf{v} &= \frac{\|\mathbf{v}\|_2}{R \sin\left(\frac{\|\mathbf{v}\|_2}{R}\right)} \left( \mathbf{y} - \cos\left(\frac{\|\mathbf{v}\|_2}{R}\right) \mathbf{x} \right) \\ &= \frac{R \cos^{-1}(\alpha)}{R \sin\left(\frac{1}{R} R \cos^{-1}(\alpha)\right)} \left( \mathbf{y} - \cos\left(\frac{1}{R} R \cos^{-1}(\alpha)\right) \mathbf{x} \right) \\ &= \frac{\cos^{-1}(\alpha)}{\sin(\cos^{-1}(\alpha))} (\mathbf{y} - \cos(\cos^{-1}(\alpha)) \mathbf{x}) \\ &= \frac{\cos^{-1}(\alpha)}{\sqrt{1 - \alpha^2}} (\mathbf{y} - \alpha \mathbf{x}), \end{aligned}$$

where  $\alpha = \frac{1}{R^2} \langle \mathbf{x}, \mathbf{y} \rangle_2 = K \langle \mathbf{x}, \mathbf{y} \rangle_2$ , and the last equality assumes  $|\alpha| > 1$ . This assumption holds, since for all points  $\mathbf{x}, \mathbf{y} \in \mathbb{S}_K^n$  it holds that  $\langle \mathbf{x}, \mathbf{y} \rangle_2 \leq R^2$ , and  $\langle \mathbf{x}, \mathbf{y} \rangle_2 = R^2$  if and only if  $\mathbf{x} = \mathbf{y}$ , due to Cauchy-Schwarz (Ratcliffe, 2006, Theorem 3.1.6). Hence, the only case where this would be a problem would be if  $\mathbf{x} = \mathbf{y}$ , but it is clear that the result in that case is  $\mathbf{u} = \mathbf{0}$ .  $\square$

**Theorem A.22** ( $\log_x^K$  is the inverse of  $\exp_x^K$  in  $\mathbb{S}_K^n$ )

$$\log_x^K(\exp_x^K(\mathbf{v})) = \mathbf{v}.$$

**Proof**

$$\begin{aligned} \log_x^K(\exp_x^K(\mathbf{v})) &= \\ &= \log_x^K\left(\cos\left(\frac{\|\mathbf{v}\|_2}{R}\right) \mathbf{x} + \sin\left(\frac{\|\mathbf{v}\|_2}{R}\right) \frac{R\mathbf{v}}{\|\mathbf{v}\|_2}\right) \\ &= \frac{\cos^{-1}(\alpha)}{\sqrt{1 - \alpha^2}} \left( \cos\left(\frac{\|\mathbf{v}\|_2}{R}\right) \mathbf{x} + \sin\left(\frac{\|\mathbf{v}\|_2}{R}\right) \frac{R\mathbf{v}}{\|\mathbf{v}\|_2} - \alpha \mathbf{x} \right) \\ &= \frac{\|\mathbf{v}\|_2}{R \sqrt{1 - \cos^2\left(\frac{\|\mathbf{v}\|_2}{R}\right)}} \sin\left(\frac{\|\mathbf{v}\|_2}{R}\right) \frac{R\mathbf{v}}{\|\mathbf{v}\|_2} \\ &= \frac{\|\mathbf{v}\|_2}{R \sin\left(\frac{\|\mathbf{v}\|_2}{R}\right)} \left( \sin\left(\frac{\|\mathbf{v}\|_2}{R}\right) \frac{R\mathbf{v}}{\|\mathbf{v}\|_2} \right) \\ &= \mathbf{v}, \end{aligned}$$

where

$$\begin{aligned}
\alpha &= \frac{\langle \mathbf{x}, \exp_{\mathbf{x}}^K(\mathbf{v}) \rangle_2}{R^2} \\
&= \frac{1}{R^2} \cos\left(\frac{\|\mathbf{v}\|_2}{R}\right) \underbrace{\langle \mathbf{x}, \mathbf{x} \rangle_2}_{=R^2} + \underbrace{\sin\left(\frac{\|\mathbf{v}\|_2}{R}\right) \frac{1}{R\|\mathbf{v}\|_2} \langle \mathbf{x}, \mathbf{v} \rangle_2}_{\mathbf{v} \in \mathcal{T}_{\mathbf{x}}\mathbb{S}_K^n \implies \langle \mathbf{x}, \mathbf{v} \rangle_2 = 0} \\
&= \cos\left(\frac{\|\mathbf{v}\|_2}{R}\right). \quad \square
\end{aligned}$$

**Theorem A.23 (Length preservation property of  $\exp_{\mathbf{x}}^K$  in  $\mathbb{S}_K^n$ )** For all points on the manifold  $\mathbf{x} \in \mathbb{S}_K^n$  and for all tangent vectors at that point  $\mathbf{v} \in \mathcal{T}_{\mathbf{x}}\mathbb{S}_K^n$  it holds that

$$d_{\mathbb{S}}(\mathbf{x}, \exp_{\mathbf{x}}^K(\mathbf{v})) = \|\mathbf{v}\|_2.$$

**Proof**

$$\begin{aligned}
d_2(\mathbf{x}, \exp_{\mathbf{x}}^K(\mathbf{v})) &= R \cos^{-1}\left(\frac{\langle \mathbf{x}, \exp_{\mathbf{x}}^K(\mathbf{v}) \rangle_2}{R^2}\right) \\
&= R \cos^{-1}\left(\cos\left(\frac{\|\mathbf{v}\|_2}{R}\right)\right) \\
&= \|\mathbf{v}\|_2,
\end{aligned}$$

where the equality

$$-\frac{\langle \mathbf{x}, \exp_{\mathbf{x}}^K(\mathbf{v}) \rangle_2}{R^2} = \cos\left(\frac{\|\mathbf{v}\|_2}{R}\right)$$

corresponds to the definition of  $\alpha$  in the Proof of Theorem A.22. □

**Parallel transport**

Using the generic formula for parallel transport in manifolds (Equation A.2.1) for  $\mathbf{x}, \mathbf{y} \in \mathbb{S}_K^n$  and  $\mathbf{v} \in \mathcal{T}_{\mathbf{x}}\mathbb{S}_K^n$  and the spherical logarithmic map formula

$$\log_{\mathbf{x}}^K(\mathbf{y}) = \frac{\cos^{-1}(\alpha)}{\sqrt{1-\alpha^2}}(\mathbf{y} - \alpha\mathbf{x}),$$

where  $\alpha = K \langle \mathbf{x}, \mathbf{y} \rangle_2$ , we derive parallel transport in  $\mathbb{S}_K^n$ :

$$\begin{aligned}
\text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}) &= \mathbf{v} - \frac{\langle \mathbf{y}, \mathbf{v} \rangle_2}{R^2 + \langle \mathbf{x}, \mathbf{y} \rangle_2}(\mathbf{x} + \mathbf{y}) \\
&= \mathbf{v} - \frac{K \langle \mathbf{y}, \mathbf{v} \rangle_2}{1 + K \langle \mathbf{x}, \mathbf{y} \rangle_2}(\mathbf{x} + \mathbf{y}).
\end{aligned}$$

A special form of parallel transport exists for when the source vector is  $\boldsymbol{\mu}_0 = (R, 0, \dots, 0)^T$ :

$$\text{PT}_{\boldsymbol{\mu}_0 \rightarrow \mathbf{y}}^K(\mathbf{v}) = \mathbf{v} - \frac{\langle \mathbf{y}, \mathbf{v} \rangle_2}{R^2 + Ry_1} \begin{pmatrix} y_1 + R \\ y_2 \\ \vdots \\ y_{n+1} \end{pmatrix}.$$

**Theorem A.24 (Parallel transport preserves angles in  $\mathbb{S}_K^n$ )** For all points on the manifold  $\mathbf{x}, \mathbf{y} \in \mathbb{S}_K^n$  and tangent vectors  $\mathbf{v}, \mathbf{v}' \in \mathcal{T}_{\mathbf{x}}\mathbb{S}_K^n$  it holds that

$$\langle \text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}), \text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}') \rangle_2 = \langle \mathbf{v}, \mathbf{v}' \rangle_2.$$

**Proof**

$$\begin{aligned} & \langle \text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}), \text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}') \rangle_2 = \\ & = \left\langle \mathbf{v} - \frac{\langle \mathbf{y}, \mathbf{v} \rangle_2}{R^2 + \langle \mathbf{x}, \mathbf{y} \rangle_2} (\mathbf{x} + \mathbf{y}), \mathbf{v}' - \frac{\langle \mathbf{y}, \mathbf{v}' \rangle_2}{R^2 + \langle \mathbf{x}, \mathbf{y} \rangle_2} (\mathbf{x} + \mathbf{y}) \right\rangle_2 \\ & = \langle \mathbf{v}, \mathbf{v}' \rangle_2 \\ & \quad - \frac{\langle \mathbf{y}, \mathbf{v}' \rangle_2}{R^2 + \langle \mathbf{x}, \mathbf{y} \rangle_2} \underbrace{\langle \mathbf{v}, \mathbf{x} + \mathbf{y} \rangle_2}_{\langle \mathbf{v}, \mathbf{y} \rangle_2} - \frac{\langle \mathbf{y}, \mathbf{v} \rangle_2}{R^2 + \langle \mathbf{x}, \mathbf{y} \rangle_2} \underbrace{\langle \mathbf{v}', \mathbf{x} + \mathbf{y} \rangle_2}_{\langle \mathbf{v}', \mathbf{y} \rangle_2} \\ & \quad + \frac{\langle \mathbf{y}, \mathbf{v} \rangle_2 \langle \mathbf{y}, \mathbf{v}' \rangle_2}{(R^2 + \langle \mathbf{x}, \mathbf{y} \rangle_2)^2} \underbrace{\langle \mathbf{x} + \mathbf{y}, \mathbf{x} + \mathbf{y} \rangle_2}_{R^2 + 2\langle \mathbf{x}, \mathbf{y} \rangle_2 + R^2} \\ & = \langle \mathbf{v}, \mathbf{v}' \rangle_2 - 2 \frac{\langle \mathbf{y}, \mathbf{v}' \rangle_2 \langle \mathbf{y}, \mathbf{v} \rangle_2}{R^2 + \langle \mathbf{x}, \mathbf{y} \rangle_2} + 2 \frac{\langle \mathbf{y}, \mathbf{v} \rangle_2 \langle \mathbf{y}, \mathbf{v}' \rangle_2}{(R^2 + \langle \mathbf{x}, \mathbf{y} \rangle_2)^2} (R^2 + \langle \mathbf{x}, \mathbf{y} \rangle_2) \\ & = \langle \mathbf{v}, \mathbf{v}' \rangle_2. \quad \square \end{aligned}$$

**Corollary (Parallel transport on  $\mathbb{S}_K^n$  is norm-preserving)**

$$\|\text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v})\|_2 = \|\mathbf{v}\|_2.$$

**Proof**

$$\|\text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v})\|_2^2 = \langle \text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}), \text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}) \rangle_2 = \langle \mathbf{v}, \mathbf{v} \rangle_2 = \|\mathbf{v}\|_2^2,$$

where the second equality corresponds to Theorem A.24.  $\square$

**Theorem A.26 ( $\text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K$  transports points to the tangent space of  $\mathbf{y}$  in  $\mathbb{S}_K^n$ )** For all points on the manifold  $\mathbf{x}, \mathbf{y} \in \mathbb{S}_K^n$  and a tangent vector  $\mathbf{v} \in \mathcal{T}_{\mathbf{x}}\mathbb{S}_K^n$  it holds that

$$\text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}) \in \mathcal{T}_{\mathbf{y}}\mathbb{S}_K^n.$$

**Proof**

$$\begin{aligned}
\langle \mathbf{y}, \text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}) \rangle_2 &= \left\langle \mathbf{y}, \mathbf{v} - \frac{\langle \mathbf{y}, \mathbf{v} \rangle_2}{R^2 + \langle \mathbf{x}, \mathbf{y} \rangle_2} (\mathbf{x} + \mathbf{y}) \right\rangle_2 \\
&= \langle \mathbf{y}, \mathbf{v} \rangle_2 - \frac{\langle \mathbf{y}, \mathbf{v} \rangle_2}{R^2 + \langle \mathbf{x}, \mathbf{y} \rangle_2} \langle \mathbf{y}, \mathbf{x} + \mathbf{y} \rangle_2 \\
&= \langle \mathbf{y}, \mathbf{v} \rangle_2 - \frac{\langle \mathbf{y}, \mathbf{v} \rangle_2}{R^2 + \langle \mathbf{x}, \mathbf{y} \rangle_2} (\langle \mathbf{y}, \mathbf{x} \rangle_2 + \langle \mathbf{y}, \mathbf{y} \rangle_2) \\
&= \langle \mathbf{y}, \mathbf{v} \rangle_2 - \frac{\langle \mathbf{y}, \mathbf{v} \rangle_2}{R^2 + \langle \mathbf{x}, \mathbf{y} \rangle_2} (\langle \mathbf{y}, \mathbf{x} \rangle_2 + R^2) \\
&= \langle \mathbf{y}, \mathbf{v} \rangle_2 - \langle \mathbf{y}, \mathbf{v} \rangle_2 = 0,
\end{aligned}$$

which implies  $\text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}) \in \mathcal{T}_{\mathbf{y}} \mathbb{S}_K^n$ .  $\square$

### A.3.2 Projected hypersphere

An overview of all the necessary operations can be found in Table A.5.

Do note, that all the theorems for the projected hypersphere are essentially trivial corollaries of their equivalents in the Poincaré ball (and vice-versa) (Section A.2.2). Notable differences include the fact that  $R^2 = \frac{1}{K}$ , not  $R^2 = -\frac{1}{K}$ , and all the operations use the Euclidean trigonometric functions  $\sin$ ,  $\cos$ , and  $\tan$ , instead of their hyperbolic counterparts. Also, we often leverage the Pythagorean theorem, in the form  $\sin^2(\alpha) + \cos^2(\alpha) = 1$ .

#### Stereographic projection

**Remark (Homeomorphism between  $\mathbb{S}_K^n$  and  $\mathbb{R}^n$ )** We notice that  $\rho_K$  is not a homeomorphism between the  $n$ -dimensional sphere and  $\mathbb{R}^n$ , as it is not defined at  $-\boldsymbol{\mu}_0 = (-R; \mathbf{0}^T)^T$ . If we additionally changed compactified the plane by adding a point “at infinity” and set it equal to  $\rho_K(\boldsymbol{\mu}_0)$ ,  $\rho_K$  would become a homeomorphism. For an illustration, see Figure 2.1b and imagine where Earth’s south pole would be represented if the projection was not cut off at a given latitude.

**Theorem A.28 (Stereographic backprojected points of  $\mathbb{D}_K^n$  belong to  $\mathbb{S}_K^n$ )**  
For all  $\mathbf{y} \in \mathbb{D}_K^n$ ,

$$\|\rho_K^{-1}(\mathbf{y})\|_2^2 = \frac{1}{K}.$$



Möbius add.	$\mathbf{x} \oplus_K \mathbf{y} = \frac{(1 - 2K \langle \mathbf{x}, \mathbf{y} \rangle_2 - K \ \mathbf{y}\ _2^2) \mathbf{x} + (1 + K \ \mathbf{x}\ _2^2) \mathbf{y}}{1 - 2K \langle \mathbf{x}, \mathbf{y} \rangle_2 + K^2 \ \mathbf{x}\ _2^2 \ \mathbf{y}\ _2^2}$
Distance	$d_{\mathbb{D}}(\mathbf{x}, \mathbf{y}) = \frac{1}{\sqrt{K}} \cos^{-1} \left( 1 - \frac{2K \ \mathbf{x} - \mathbf{y}\ _2^2}{(1 + K \ \mathbf{x}\ _2^2)(1 + K \ \mathbf{y}\ _2^2)} \right)$
Gyr. dist.	$d_{\mathbb{D}_{\text{gyr}}}(\mathbf{x}, \mathbf{y}) = \frac{2}{\sqrt{K}} \tan^{-1}(\sqrt{K} \ \mathbf{x} \oplus_K \mathbf{y}\ _2)$
Lambda	$\lambda_{\mathbf{x}}^K = \frac{2}{1 + K \ \mathbf{x}\ _2^2}$
Exp. map	$\exp_{\mathbf{x}}^K(\mathbf{v}) = \mathbf{x} \oplus_K \left( \tan \left( \sqrt{K} \frac{\lambda_{\mathbf{x}}^K \ \mathbf{v}\ _2}{2} \right) \frac{\mathbf{v}}{\sqrt{K} \ \mathbf{v}\ _2} \right)$
Log. map	$\log_{\mathbf{x}}^K(\mathbf{y}) = \frac{2}{\sqrt{K} \lambda_{\mathbf{x}}^K} \tan^{-1} \left( \sqrt{K} \ \mathbf{x} \oplus_K \mathbf{y}\ _2 \right) \frac{-\mathbf{x} \oplus_K \mathbf{y}}{\ \mathbf{x} \oplus_K \mathbf{y}\ _2}$
Gyration	$\text{gyr}[\mathbf{x}, \mathbf{y}]\mathbf{v} = \ominus_K(\mathbf{x} \oplus_K \mathbf{y}) \oplus_K(\mathbf{x} \oplus_K(\mathbf{y} \oplus_K \mathbf{v}))$
Par. transp.	$\text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}) = \frac{\lambda_{\mathbf{x}}^K}{\lambda_{\mathbf{y}}^K} \text{gyr}[\mathbf{y}, -\mathbf{x}]\mathbf{v}$
	$\text{PT}_{\mu_0 \rightarrow \mathbf{y}}^K(\mathbf{v}) = \frac{2}{\lambda_{\mathbf{y}}^K} \mathbf{v}, \quad \text{PT}_{\mathbf{x} \rightarrow \mu_0}^K(\mathbf{v}) = \frac{\lambda_{\mathbf{x}}^K}{2} \mathbf{v}$

Table A.5: Spherical projected operations.

**Proof**

$$\begin{aligned}
\|\rho_K^{-1}(\mathbf{y})\|_2^2 &= \left\| \left( \frac{1}{\sqrt{|K|}} \frac{K \|\mathbf{y}\|_2^2 - 1}{K \|\mathbf{y}\|_2^2 + 1}, \frac{2\mathbf{y}^T}{K \|\mathbf{y}\|_2^2 + 1} \right)^T \right\|_2^2 \\
&= \left( \frac{1}{\sqrt{|K|}} \frac{K \|\mathbf{y}\|_2^2 - 1}{K \|\mathbf{y}\|_2^2 + 1} \right)^2 + \frac{4 \|\mathbf{y}\|_2^2}{(K \|\mathbf{y}\|_2^2 + 1)^2} \\
&= \frac{1}{|K|} \frac{(K \|\mathbf{y}\|_2^2 - 1)^2 + 4|K| \|\mathbf{y}\|_2^2}{(K \|\mathbf{y}\|_2^2 + 1)^2} \\
&= \frac{1}{K} \frac{(K \|\mathbf{y}\|_2^2 - 1)^2 + 4K \|\mathbf{y}\|_2^2}{(K \|\mathbf{y}\|_2^2 + 1)^2} \\
&= \frac{1}{K} \frac{K^2 \|\mathbf{y}\|_2^4 + 2K \|\mathbf{y}\|_2^2 + 1}{(K \|\mathbf{y}\|_2^2 + 1)^2} \\
&= \frac{1}{K} \frac{(K \|\mathbf{y}\|_2^2 + 1)^2}{(K \|\mathbf{y}\|_2^2 + 1)^2} = \frac{1}{K}. \quad \square
\end{aligned}$$

### Distance function

The distance function in  $\mathbb{D}_K^n$  is (derived from the spherical distance function using the stereographic projection  $\rho_K$ ):

$$\begin{aligned} d_{\mathbb{D}}(\mathbf{x}, \mathbf{y}) &= d_{\mathbb{S}}(\rho_K^{-1}(\mathbf{x}), \rho_K^{-1}(\mathbf{y})) \\ &= \frac{1}{\sqrt{K}} \cos^{-1} \left( 1 - \frac{2K \|\mathbf{x} - \mathbf{y}\|_2^2}{(1 + K \|\mathbf{x}\|_2^2)(1 + K \|\mathbf{y}\|_2^2)} \right) \\ &= R \cos^{-1} \left( 1 - \frac{2R^2 \|\mathbf{x} - \mathbf{y}\|_2^2}{(R^2 + \|\mathbf{x}\|_2^2)(R^2 + \|\mathbf{y}\|_2^2)} \right) \end{aligned}$$

**Theorem A.29 (Distance equivalence in  $\mathbb{D}_K^n$ )** *For all  $K > 0$  and for all pairs of points  $\mathbf{x}, \mathbf{y} \in \mathbb{D}_K^n$ , the spherical projected distance between them equals the gyrospace distance*

$$d_{\mathbb{D}}(\mathbf{x}, \mathbf{y}) = d_{\mathbb{D}_{gyr}}(\mathbf{x}, \mathbf{y}).$$

**Proof** Proven using Mathematica (File: `distance_limits.ws`), proof involves heavy algebra.  $\square$

**Theorem A.30 (Gyrospace distance converges to Euclidean in  $\mathbb{D}_K^n$ )** *For any fixed pair of points  $\mathbf{x}, \mathbf{y} \in \mathbb{D}_K^n$ , the spherical projected gyrospace distance between them converges to the Euclidean distance in the limit (up to a constant) as  $K \rightarrow 0^+$ :*

$$\lim_{K \rightarrow 0^+} d_{\mathbb{D}_{gyr}}(\mathbf{x}, \mathbf{y}) = 2 \|\mathbf{x} - \mathbf{y}\|_2.$$

**Proof**

$$\begin{aligned} \lim_{K \rightarrow 0^+} d_{\mathbb{D}_{gyr}}(\mathbf{x}, \mathbf{y}) &= 2 \lim_{K \rightarrow 0^+} \left[ \frac{\tan^{-1}(\sqrt{K} \|\mathbf{x} \oplus_K \mathbf{y}\|_2)}{\sqrt{K}} \right] \\ &= 2 \lim_{K \rightarrow 0^+} \left[ \frac{\tan^{-1}(\sqrt{K} \|\mathbf{y} - \mathbf{x}\|_2)}{\sqrt{K}} \right] \\ &= 2 \|\mathbf{y} - \mathbf{x}\|_2, \end{aligned}$$

where the second equality holds because of the theorem of limits of composed functions, where

$$\begin{aligned} f(a) &= \frac{\tan^{-1}(a\sqrt{K})}{\sqrt{K}} \\ g(K) &= \|\mathbf{x} \oplus_K \mathbf{y}\|_2. \end{aligned}$$

We see that

$$\lim_{K \rightarrow 0^-} g(K) = \|\mathbf{y} - \mathbf{x}\|_2$$

due to Theorem A.37, and

$$\lim_{a \rightarrow \|\mathbf{x} - \mathbf{y}\|_2} f(a) = \frac{\tan^{-1}(a\sqrt{K})}{\sqrt{K}}$$

Additionally for the last equality, we need the fact that

$$\lim_{x \rightarrow 0} \frac{\tanh^{-1}(a\sqrt{|x|})}{\sqrt{|x|}} = a. \quad \square$$

**Theorem A.31 (Distance converges to Euclidean as  $K \rightarrow 0^+$  in  $\mathbb{D}_K^n$ )**  
For any fixed pair of points  $\mathbf{x}, \mathbf{y} \in \mathbb{D}_K^n$ , the spherical projected distance between them converges to the Euclidean distance in the limit (up to a constant) as  $K \rightarrow 0^+$ :

$$\lim_{K \rightarrow 0^+} d_{\mathbb{D}}(\mathbf{x}, \mathbf{y}) = 2 \|\mathbf{x} - \mathbf{y}\|_2.$$

**Proof** Theorem A.29 and A.30. □

### Exponential map

Analogously to the derivation of the exponential map in  $\mathbb{P}_K^n$  in Ganea et al. (2018a, Section 2.3–2.4), we can derive Möbius scalar multiplication in  $\mathbb{D}_K^n$ :

$$\begin{aligned} r \otimes_K \mathbf{x} &= \frac{1}{i\sqrt{K}} \tanh(r \tanh^{-1}(i\sqrt{K} \|\mathbf{x}\|_2)) \frac{\mathbf{x}}{\|\mathbf{x}\|_2} \\ &= \frac{1}{i\sqrt{K}} \tanh(ri \tan^{-1}(\sqrt{K} \|\mathbf{x}\|_2)) \frac{\mathbf{x}}{\|\mathbf{x}\|_2} \\ &= \frac{1}{\sqrt{K}} \tan(r \tan^{-1}(\sqrt{K} \|\mathbf{x}\|_2)) \frac{\mathbf{x}}{\|\mathbf{x}\|_2}, \end{aligned}$$

where we use the fact that  $\tanh^{-1}(ix) = i \tan^{-1}(x)$  and  $\tanh(ix) = i \tan(x)$ . We can easily see that  $1 \otimes_K \mathbf{x} = \mathbf{x}$ .

Hence, the geodesic has the form of

$$\gamma_{\mathbf{x} \rightarrow \mathbf{y}}(t) = \mathbf{x} \oplus_K t \otimes_K (-\mathbf{x} \oplus_K \mathbf{y}),$$

and therefore the exponential map in  $\mathbb{D}_K^n$  is:

$$\exp_{\mathbf{x}}^K(\mathbf{v}) = \mathbf{x} \oplus_K \left( \tan \left( \sqrt{K} \frac{\lambda_{\mathbf{x}}^K \|\mathbf{v}\|_2}{2} \right) \frac{\mathbf{v}}{\sqrt{K} \|\mathbf{v}\|_2} \right).$$

The inverse formula can also be computed:

$$\log_{\mathbf{x}}^K(\mathbf{y}) = \frac{2}{\sqrt{K}\lambda_{\mathbf{x}}^K} \tan^{-1} \left( \sqrt{K} \|\mathbf{x} \oplus_K \mathbf{y}\|_2 \right) \frac{-\mathbf{x} \oplus_K \mathbf{y}}{\|\mathbf{x} \oplus_K \mathbf{y}\|_2}$$

In the case of  $\mathbf{x} := \boldsymbol{\mu}_0 = (0, \dots, 0)^T$  they simplify to:

$$\begin{aligned} \exp_{\boldsymbol{\mu}_0}^K(\mathbf{v}) &= \tan \left( \sqrt{K} \|\mathbf{v}\|_2 \right) \frac{\mathbf{v}}{\sqrt{K} \|\mathbf{v}\|_2} \\ \log_{\boldsymbol{\mu}_0}^K(\mathbf{y}) &= \tan^{-1} \left( \sqrt{K} \|\mathbf{y}\|_2 \right) \frac{\mathbf{y}}{\sqrt{K} \|\mathbf{y}\|_2}. \end{aligned}$$

**Theorem A.32** ( $\log_{\mathbf{x}}^K$  is the inverse of  $\exp_{\mathbf{x}}^K$  in  $\mathbb{D}_K^n$ )

$$\log_{\mathbf{x}}^K(\exp_{\mathbf{x}}^K(\mathbf{v})) = \mathbf{v}.$$

**Proof**

$$\begin{aligned} \log_{\mathbf{x}}^K(\exp_{\mathbf{x}}^K(\mathbf{v})) &= \frac{2}{\sqrt{K}\lambda_{\mathbf{x}}^K} \tan^{-1} \left( \sqrt{K} \|\mathbf{x} \oplus_K \exp_{\mathbf{x}}^K(\mathbf{v})\|_2 \right) \frac{-\mathbf{x} \oplus_K \exp_{\mathbf{x}}^K(\mathbf{v})}{\|\mathbf{x} \oplus_K \exp_{\mathbf{x}}^K(\mathbf{v})\|_2} \\ &= \frac{2}{\sqrt{K}\lambda_{\mathbf{x}}^K} \tan^{-1} \left( \sqrt{K} \|\mathbf{x} \oplus_K \mathbf{y}\|_2 \right) \frac{-\mathbf{x} \oplus_K \mathbf{y}}{\|\mathbf{x} \oplus_K \mathbf{y}\|_2} \\ &= \frac{2}{\sqrt{K}\lambda_{\mathbf{x}}^K} \tan^{-1} \left( \sqrt{K} \frac{1}{\sqrt{K}} \tan \left( \sqrt{K} \frac{\lambda_{\mathbf{x}}^K \|\mathbf{v}\|_2}{2} \right) \right) \\ &\quad \cdot \frac{\tan \left( \sqrt{K} \frac{\lambda_{\mathbf{x}}^K \|\mathbf{v}\|_2}{2} \right) \frac{\mathbf{v}}{\sqrt{K} \|\mathbf{v}\|_2}}{\frac{1}{\sqrt{K}} \tan \left( \sqrt{K} \frac{\lambda_{\mathbf{x}}^K \|\mathbf{v}\|_2}{2} \right)} \\ &= \frac{2}{\sqrt{K}\lambda_{\mathbf{x}}^K} \tan^{-1} \left( \tan \left( \sqrt{K} \frac{\lambda_{\mathbf{x}}^K \|\mathbf{v}\|_2}{2} \right) \right) \frac{\mathbf{v}}{\|\mathbf{v}\|_2} \\ &= \frac{2}{\sqrt{K}\lambda_{\mathbf{x}}^K} \sqrt{K} \frac{\lambda_{\mathbf{x}}^K \|\mathbf{v}\|_2}{2} \frac{\mathbf{v}}{\|\mathbf{v}\|_2} = \mathbf{v}, \end{aligned}$$

where the third equality is based on the fact that

$$\begin{aligned} -\mathbf{x} \oplus_K \exp_{\mathbf{x}}^K(\mathbf{v}) &= -\mathbf{x} \oplus_K \left( \mathbf{x} \oplus_K \left( \tan \left( \sqrt{K} \frac{\lambda_{\mathbf{x}}^K \|\mathbf{v}\|_2}{2} \right) \frac{\mathbf{v}}{\sqrt{K} \|\mathbf{v}\|_2} \right) \right) \\ &= \tan \left( \sqrt{K} \frac{\lambda_{\mathbf{x}}^K \|\mathbf{v}\|_2}{2} \right) \frac{\mathbf{v}}{\sqrt{K} \|\mathbf{v}\|_2}, \end{aligned}$$

and

$$\|\mathbf{x} \oplus_K \exp_{\mathbf{x}}^K(\mathbf{v})\|_2 = \frac{1}{\sqrt{K}} \tan \left( \sqrt{K} \frac{\lambda_{\mathbf{x}}^K \|\mathbf{v}\|_2}{2} \right). \quad \square$$

**Theorem A.33 (Length preservation property of  $\exp_x^K$  in  $\mathbb{D}_K^n$ )** For all points on the manifold  $\mathbf{x} \in \mathbb{D}_K^n$  and for all tangent vectors at that point  $\mathbf{v} \in \mathcal{T}_{\mathbf{x}}\mathbb{D}_K^n$  it holds that

$$d_{\mathbb{D}_{\text{gyr}}}(\mathbf{x}, \exp_{\mathbf{x}}^K(\mathbf{v})) = \lambda_{\mathbf{x}}^K \|\mathbf{v}\|_2.$$

**Proof**

$$\begin{aligned} d_{\mathbb{D}_{\text{gyr}}}(\mathbf{x}, \exp_{\mathbf{x}}^K(\mathbf{v})) &= \\ &= \frac{2}{\sqrt{K}} \tan^{-1} \left( \sqrt{K} \left\| -\mathbf{x} \oplus_K \exp_{\mathbf{x}}^K(\mathbf{v}) \right\|_2 \right) \\ &= \frac{2}{\sqrt{K}} \tan^{-1} \left( \sqrt{K} \left\| -\mathbf{x} \oplus_K \left( \mathbf{x} \oplus_K \left( \tan \left( \sqrt{K} \frac{\lambda_{\mathbf{x}}^K \|\mathbf{v}\|_2}{2} \right) \frac{\mathbf{v}}{\sqrt{K} \|\mathbf{v}\|_2} \right) \right) \right\|_2 \right) \\ &= \frac{2}{\sqrt{K}} \tan^{-1} \left( \sqrt{K} \left\| \tan \left( \sqrt{K} \frac{\lambda_{\mathbf{x}}^K \|\mathbf{v}\|_2}{2} \right) \frac{\mathbf{v}}{\sqrt{K} \|\mathbf{v}\|_2} \right\|_2 \right) \\ &= \frac{2}{\sqrt{K}} \tan^{-1} \left( \sqrt{K} \tan \left( \sqrt{K} \frac{\lambda_{\mathbf{x}}^K \|\mathbf{v}\|_2}{2} \right) \frac{\|\mathbf{v}\|_2}{\sqrt{K} \|\mathbf{v}\|_2} \right) \\ &= \frac{2}{\sqrt{K}} \tan^{-1} \left( \tan \left( \sqrt{K} \frac{\lambda_{\mathbf{x}}^K \|\mathbf{v}\|_2}{2} \right) \right) \\ &= \frac{2}{\sqrt{K}} \sqrt{K} \frac{\lambda_{\mathbf{x}}^K \|\mathbf{v}\|_2}{2} \\ &= \lambda_{\mathbf{x}}^K \|\mathbf{v}\|_2, \end{aligned}$$

where the third equality holds because of the left-cancellation law (Ganea et al., 2018a, Section 2.3).  $\square$

### Parallel transport

Similarly to the Poincaré ball, we can derive the parallel transport operation for the projected sphere:

$$\begin{aligned} \text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}) &= \frac{\lambda_{\mathbf{x}}^K}{\lambda_{\mathbf{y}}^K} \text{gyr}[\mathbf{y}, -\mathbf{x}]\mathbf{v}, \\ \text{PT}_{\boldsymbol{\mu}_0 \rightarrow \mathbf{y}}^K(\mathbf{v}) &= \frac{2}{\lambda_{\mathbf{y}}^K} \mathbf{v}, \\ \text{PT}_{\mathbf{x} \rightarrow \boldsymbol{\mu}_0}^K(\mathbf{v}) &= \frac{\lambda_{\mathbf{x}}^K}{2} \mathbf{v}, \end{aligned}$$

where

$$\text{gyr}[\mathbf{x}, \mathbf{y}]\mathbf{v} = -(\mathbf{x} \oplus_K \mathbf{y}) \oplus_K (\mathbf{x} \oplus_K (\mathbf{y} \oplus_K \mathbf{v}))$$

is the gyration operation (Ungar, 2008, Definition 1.11).

**Theorem A.34 (Parallel transport and its inverse in  $\mathbb{D}_K^n$ )**

$$\text{PT}_{\mathbf{y} \rightarrow \mathbf{x}}^K(\text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v})) = \mathbf{v}.$$

**Proof** We only use this fact for  $\mathbf{x}$  or  $\mathbf{y}$  equal to  $\boldsymbol{\mu}_0$ , and for that, it is trivial. Otherwise, one can prove it using the properties from Ungar (2008).  $\square$

Unfortunately, on the projected sphere,  $\langle \cdot, \cdot \rangle_{\mathbf{x}}$  has a form that changes with respect to  $\mathbf{x}$ , similarly to the Poincaré ball and unlike in the hypersphere. This means that the following theorems do not hold with respect to  $\langle \cdot, \cdot \rangle_2$ .

**Theorem A.35 (Parallel transport preserves angles in  $\mathbb{D}_K^n$ )** *For all points on the manifold  $\mathbf{x}, \mathbf{y} \in \mathbb{D}_K^n$  and tangent vectors  $\mathbf{v}, \mathbf{v}' \in \mathcal{T}_{\mathbf{x}}\mathbb{D}_K^n$  it holds that*

$$\langle \text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}), \text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}') \rangle_{\mathbf{y}} = \langle \mathbf{v}, \mathbf{v}' \rangle_{\mathbf{x}}.$$

**Proof**

$$\begin{aligned} \langle \text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}), \text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}') \rangle_{\mathbf{y}} &= (\lambda_{\mathbf{y}}^K)^2 \langle \text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}), \text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}') \rangle_2 \\ &= (\lambda_{\mathbf{y}}^K)^2 \left( \frac{\lambda_{\mathbf{x}}^K}{\lambda_{\mathbf{y}}^K} \right)^2 \langle \text{gyr}[\mathbf{y}, -\mathbf{x}]\mathbf{v}, \text{gyr}[\mathbf{y}, -\mathbf{x}]\mathbf{v}' \rangle_2 \\ &= (\lambda_{\mathbf{x}}^K)^2 \langle \mathbf{v}, \mathbf{v}' \rangle_2 \\ &= \langle \mathbf{v}, \mathbf{v}' \rangle_{\mathbf{x}}, \end{aligned}$$

where  $\langle \text{gyr}[\mathbf{x}, \mathbf{y}]\mathbf{v}, \text{gyr}[\mathbf{x}, \mathbf{y}]\mathbf{v}' \rangle_2 = \langle \mathbf{v}, \mathbf{v}' \rangle_2$  is proven in Ungar (2008, Equation 1.32).  $\square$

**Corollary (Parallel transport on  $\mathbb{D}_K^n$  is norm-preserving)**

$$\|\text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v})\|_{\mathbf{y}} = \|\mathbf{v}\|_{\mathbf{x}},$$

and hence

$$\|\text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v})\|_2 = \frac{\lambda_{\mathbf{x}}^K}{\lambda_{\mathbf{y}}^K} \|\mathbf{v}\|_2.$$

**Proof**

$$\begin{aligned} (\lambda_{\mathbf{y}}^K)^2 \|\text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v})\|_2^2 &= \|\text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v})\|_{\mathbf{y}}^2 \\ &= \langle \text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}), \text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}) \rangle_{\mathbf{y}} \\ &= \langle \mathbf{v}, \mathbf{v} \rangle_{\mathbf{x}} \\ &= \|\mathbf{v}\|_{\mathbf{x}}^2 = (\lambda_{\mathbf{x}}^K)^2 \|\mathbf{v}\|_2^2, \end{aligned}$$

where the third equality corresponds to Theorem A.35.  $\square$

## A.4 Miscellaneous properties

**Theorem A.37** (Möbius addition converges to Eucl. vector addition)

$$\lim_{K \rightarrow 0} (\mathbf{x} \oplus_K \mathbf{y}) = \mathbf{x} + \mathbf{y}.$$

*Note: This theorem works from both sides, hence applies to the Poincaré ball as well as the projected spherical space. Observe that the Möbius addition has the same form for both spaces.*

**Proof**

$$\begin{aligned} \lim_{K \rightarrow 0} (\mathbf{x} \oplus_K \mathbf{y}) &= \lim_{K \rightarrow 0} \left[ \frac{(1 - 2K \langle \mathbf{x}, \mathbf{y} \rangle_2 - K \|\mathbf{y}\|_2^2) \mathbf{x} + (1 + K \|\mathbf{x}\|_2^2) \mathbf{y}}{1 - 2K \langle \mathbf{x}, \mathbf{y} \rangle_2 + K^2 \|\mathbf{x}\|_2^2 \|\mathbf{y}\|_2^2} \right] \\ &= \mathbf{x} + \mathbf{y}. \end{aligned} \quad \square$$

**Theorem A.38** ( $\rho_K^{-1}$  is the inverse stereographic projection)

For all  $(\xi; \mathbf{x}^T)^T \in \mathcal{M}_K^n$ ,  $\xi \in \mathbb{R}$

$$\rho_K^{-1}(\rho((\xi; \mathbf{x}^T)^T)) = \mathbf{x},$$

where  $\mathcal{M} \in \{\mathbb{S}, \mathbb{H}\}$ .

**Proof**

$$\begin{aligned} \rho_K^{-1}(\rho_K((\xi; \mathbf{x}^T)^T)) &= \rho_K^{-1} \left( \frac{\mathbf{x}}{1 - \sqrt{|K|}\xi} \right) \\ &= \left( \frac{1}{\sqrt{|K|}} \frac{K \left\| \frac{\mathbf{x}}{1 - \sqrt{|K|}\xi} \right\|_2^2 - 1}{K \left\| \frac{\mathbf{x}}{1 - \sqrt{|K|}\xi} \right\|_2^2 + 1}; \frac{\frac{2\mathbf{x}^T}{1 - \sqrt{|K|}\xi}}{K \left\| \frac{\mathbf{x}}{1 - \sqrt{|K|}\xi} \right\|_2^2 + 1} \right)^T \\ &= \frac{1/\sqrt{|K|}}{K \left\| \frac{\mathbf{x}}{1 - \sqrt{|K|}\xi} \right\|_2^2 + 1} \left( K \left\| \frac{\mathbf{x}}{1 - \sqrt{|K|}\xi} \right\|_2^2 - 1; \frac{2\sqrt{|K|}\mathbf{x}^T}{1 - \sqrt{|K|}\xi} \right)^T \\ &= \frac{1/\sqrt{|K|}}{\frac{K\|\mathbf{x}\|_2^2}{(1 - \sqrt{|K|}\xi)^2} + 1} \left( \frac{K\|\mathbf{x}\|_2^2}{(1 - \sqrt{|K|}\xi)^2} - 1; \frac{2\sqrt{|K|}\mathbf{x}^T}{1 - \sqrt{|K|}\xi} \right)^T \end{aligned}$$

We observe that  $\|\mathbf{x}\|_2^2 = \frac{1}{K} - \xi^2$ , because  $\mathbf{x} \in \mathcal{M}_K^n$ . Therefore

$$\begin{aligned}
& \rho_K^{-1}(\rho_K((\xi; \mathbf{x}^T)^T)) = \\
& = \dots \quad (\text{above}) \\
& = \frac{1/\sqrt{|K|}}{K \frac{\frac{1}{K} - \xi^2}{(1 - \sqrt{|K|}\xi)^2} + 1} \left( K \frac{\frac{1}{K} - \xi^2}{(1 - \sqrt{|K|}\xi)^2} - 1; \frac{2\sqrt{|K|}\mathbf{x}^T}{1 - \sqrt{|K|}\xi} \right)^T \\
& = \frac{1/\sqrt{|K|}}{\frac{(1 - \sqrt{|K|}\xi)(1 + \sqrt{|K|}\xi)}{(1 - \sqrt{|K|}\xi)^2} + 1} \left( \frac{(1 - \sqrt{|K|}\xi)(1 + \sqrt{|K|}\xi)}{(1 - \sqrt{|K|}\xi)^2} - 1; \frac{2\sqrt{|K|}\mathbf{x}^T}{1 - \sqrt{|K|}\xi} \right)^T \\
& = \frac{1/\sqrt{|K|}}{\frac{1 + \sqrt{|K|}\xi}{1 - \sqrt{|K|}\xi} + 1} \left( \frac{1 + \sqrt{|K|}\xi}{1 - \sqrt{|K|}\xi} - 1; \frac{2\sqrt{|K|}\mathbf{x}^T}{1 - \sqrt{|K|}\xi} \right)^T \\
& = \frac{1/\sqrt{|K|}}{\frac{1 + \sqrt{|K|}\xi + 1 - \sqrt{|K|}\xi}{1 - \sqrt{|K|}\xi}} \left( \frac{1 + \sqrt{|K|}\xi - 1 + \sqrt{|K|}\xi}{1 - \sqrt{|K|}\xi}; \frac{2\sqrt{|K|}\mathbf{x}^T}{1 - \sqrt{|K|}\xi} \right)^T \\
& = \frac{1}{2\sqrt{|K|}} \left( 2\sqrt{|K|}\xi; 2\sqrt{|K|}\mathbf{x}^T \right)^T = (\xi; \mathbf{x}^T)^T.
\end{aligned}$$

□

**Lemma ( $\lambda_{\mathbf{x}}^K$  converges to 2 as  $K \rightarrow 0$ )** For all  $\mathbf{x}$  in  $\mathbb{P}_K^n$  or  $\mathbb{D}_K^n$ , it holds that

$$\lim_{K \rightarrow 0} \lambda_{\mathbf{x}}^K = 2.$$

**Proof**

$$\lim_{K \rightarrow 0} \lambda_{\mathbf{x}}^K = \lim_{K \rightarrow 0} \frac{2}{1 + K \|\mathbf{x}\|_2^2} = 2. \quad \square$$

**Theorem A.40 ( $\exp_{\mathbf{x}}^K(\mathbf{v})$  converges to  $\mathbf{x} + \mathbf{v}$  as  $K \rightarrow 0$ )** For all  $\mathbf{x}$  in the Poincaré ball  $\mathbb{P}_K^n$  or the projected sphere  $\mathbb{D}_K^n$  and  $\mathbf{v} \in \mathcal{T}_{\mathbf{x}}\mathcal{M}$ , it holds that

$$\lim_{K \rightarrow 0} \exp_{\mathbf{x}}^K(\mathbf{v}) = \exp_{\mathbf{x}}(\mathbf{v}) = \mathbf{x} + \mathbf{v},$$

hence the exponential map converges to its Euclidean variant.



**Proof** For the positive case  $K > 0$

$$\begin{aligned}
\lim_{K \rightarrow 0^+} \exp_x^K(\mathbf{v}) &= \lim_{K \rightarrow 0^+} \left( \mathbf{x} \oplus_K \left( \tan_K \left( \sqrt{|K|} \frac{\lambda_x^K \|\mathbf{v}\|_2}{2} \right) \frac{\mathbf{v}}{\sqrt{|K|} \|\mathbf{v}\|_2} \right) \right) \\
&= \mathbf{x} + \lim_{K \rightarrow 0^+} \left( \tan_K \left( \sqrt{|K|} \frac{\lambda_x^K \|\mathbf{v}\|_2}{2} \right) \frac{\mathbf{v}}{\sqrt{|K|} \|\mathbf{v}\|_2} \right) \\
&= \mathbf{x} + \frac{\mathbf{v}}{\|\mathbf{v}\|_2} \lim_{K \rightarrow 0^+} \frac{\tan \left( \sqrt{K} \frac{\lambda_x^K \|\mathbf{v}\|_2}{2} \right)}{\sqrt{K} \|\mathbf{v}\|_2} \\
&= \mathbf{x} + \mathbf{v},
\end{aligned}$$

due to several applications of the theorem of limits of composed functions, Lemma A.39, and the fact that

$$\lim_{\alpha \rightarrow 0} \frac{\tan(\sqrt{\alpha}a)}{\sqrt{\alpha}} = a.$$

The negative case  $K < 0$  is analogous. □

**Theorem A.41** ( $\log_x^K(\mathbf{y})$  converges to  $\mathbf{y} - \mathbf{x}$  as  $K \rightarrow 0$ ) For all  $\mathbf{x}, \mathbf{y}$  in the Poincaré ball  $\mathbb{F}_K^n$  or the projected sphere  $\mathbb{D}_K^n$ , it holds that

$$\lim_{K \rightarrow 0} \log_x^K(\mathbf{y}) = \log_x(\mathbf{y}) = \mathbf{y} - \mathbf{x},$$

hence the logarithmic map converges to its Euclidean variant.

**Proof** Firstly,

$$\mathbf{z} = -\mathbf{x} \oplus_K \mathbf{y} \xrightarrow{K \rightarrow 0} \mathbf{y} - \mathbf{x},$$

due to Theorem A.37. For the positive case  $K > 0$

$$\begin{aligned}
\lim_{K \rightarrow 0^+} \log_x^K(\mathbf{y}) &= \lim_{K \rightarrow 0^+} \left( \frac{2}{\sqrt{|K|} \lambda_x^K} \tan_K^{-1} \left( \sqrt{|K|} \|\mathbf{z}\|_2 \right) \frac{\mathbf{z}}{\|\mathbf{z}\|_2} \right) \\
&= \lim_{K \rightarrow 0^+} \left( \frac{2 \tan_K^{-1} \left( \sqrt{|K|} \|\mathbf{z}\|_2 \right)}{\lambda_x^K \sqrt{|K|} \|\mathbf{z}\|_2} \mathbf{z} \right) \\
&= \lim_{K \rightarrow 0^+} \frac{2}{\lambda_x^K} \cdot \lim_{K \rightarrow 0^+} \frac{\tan^{-1} \left( \sqrt{K} \|\mathbf{z}\|_2 \right)}{\sqrt{K} \|\mathbf{z}\|_2} \cdot \lim_{K \rightarrow 0^+} \mathbf{z} \\
&= 1 \cdot 1 \cdot (\mathbf{x} - \mathbf{y}) = \mathbf{x} - \mathbf{y},
\end{aligned}$$

due to several applications of the theorem of limits of composed functions, product rule for limits, Lemma A.39, and the fact that

$$\lim_{\alpha \rightarrow 0} \frac{\tan^{-1}(\sqrt{\alpha}a)}{\sqrt{\alpha}} = a.$$

The negative case  $K < 0$  is analogous. □

**Lemma (gyr[ $\mathbf{x}, \mathbf{y}$ ]v converges to  $\mathbf{v}$  as  $K \rightarrow 0$ )** For all  $\mathbf{x}, \mathbf{y}$  in the Poincaré ball  $\mathbb{P}_K^n$  or the projected sphere  $\mathbb{D}_K^n$  and  $\mathbf{v} \in \mathcal{T}_{\mathbf{x}}\mathcal{M}$ , it holds that

$$\lim_{K \rightarrow 0} \text{gyr}[\mathbf{x}, \mathbf{y}]\mathbf{x} = \mathbf{v},$$

hence gyration converges to an identity function.

**Proof**

$$\begin{aligned} \lim_{K \rightarrow 0} \text{gyr}[\mathbf{x}, \mathbf{y}]\mathbf{v} &= \lim_{K \rightarrow 0} (\ominus_K(\mathbf{x} \oplus_K \mathbf{y}) \oplus_K (\mathbf{x} \oplus_K (\mathbf{y} \oplus_K \mathbf{v}))) \\ &= -(\mathbf{x} + \mathbf{y}) + (\mathbf{x} + (\mathbf{y} + \mathbf{v})) \\ &= -\mathbf{x} - \mathbf{y} + \mathbf{x} + \mathbf{y} + \mathbf{v} = \mathbf{v}, \end{aligned}$$

due to Theorem A.37 and the theorem of limits of composed functions.  $\square$

**Theorem A.43 (PT $_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v})$  converges to  $\mathbf{v}$  as  $K \rightarrow 0$ )** For all  $\mathbf{x}, \mathbf{y}$  in the Poincaré ball  $\mathbb{P}_K^n$  or the projected sphere  $\mathbb{D}_K^n$  and  $\mathbf{v} \in \mathcal{T}_{\mathbf{x}}\mathcal{M}$ , it holds that

$$\lim_{K \rightarrow 0} PT_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}) = \mathbf{v}.$$

**Proof**

$$\begin{aligned} \lim_{K \rightarrow 0} PT_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}) &= \lim_{K \rightarrow 0} \left( \frac{\lambda_{\mathbf{x}}^K}{\lambda_{\mathbf{y}}^K} \text{gyr}[\mathbf{y}, -\mathbf{x}]\mathbf{v} \right) \\ &= \lim_{K \rightarrow 0} \underbrace{\frac{\lambda_{\mathbf{x}}^K}{\lambda_{\mathbf{y}}^K}}_{\xrightarrow{K \rightarrow 0} 1} \cdot \lim_{K \rightarrow 0} \underbrace{\text{gyr}[\mathbf{y}, -\mathbf{x}]\mathbf{v}}_{\xrightarrow{K \rightarrow 0} \mathbf{v}} \\ &= \mathbf{v}, \end{aligned}$$

due to the product rule for limits, Lemma A.39, and Lemma A.42.  $\square$

## A.5 Angles in constant curvature spaces

In the Euclidean space, we can define the notion of an “angle” between the two vectors (or equivalently hyperplanes) thanks to the Cauchy-Schwarz theorem:

$$\langle \mathbf{x}, \mathbf{y} \rangle_2 = \|\mathbf{x}\|_2 \|\mathbf{y}\|_2 \cos \theta_{\mathbf{x}, \mathbf{y}},$$

i.e. the scalar product decomposes into a product of the norms of the two normal (orthogonal) vectors and the cosine of the angle  $\theta_{\mathbf{x}, \mathbf{y}}$  between them. Consequently, we have

$$\theta_{\mathbf{x}, \mathbf{y}} = \cos^{-1} \left( \frac{\langle \mathbf{x}, \mathbf{y} \rangle_2}{\|\mathbf{x}\|_2 \|\mathbf{y}\|_2} \right),$$

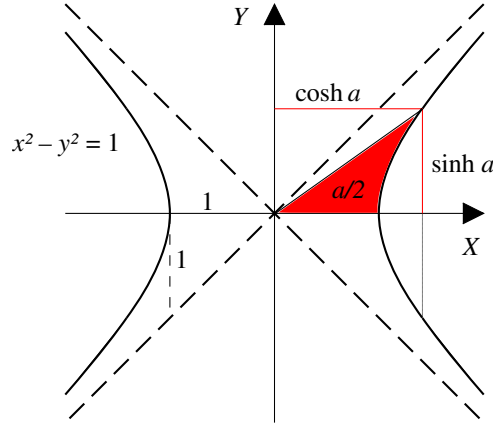


Figure A.2: Visualization of  $\cosh$  of an angle in  $\mathbb{H}_1^1$  (Wikimedia, 2009).

which gives us a specific formula for the “angle” between  $\mathbf{x}$  and  $\mathbf{y}$ .

Since the inner product at every point in the hypersphere  $\mathbb{S}_K^n$  and the hyperboloid  $\mathbb{H}_K^n$  is the same, we can define a notion of angles between points on these manifolds. For the hypersphere, the inner product coincides with the inner product in the ambient Euclidean space, therefore angles and norms correspond to the Euclidean variants as well.

For the hyperboloid model  $\mathbb{H}_K^n$ , we have

$$\langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}} = \|\mathbf{x}\|_{\mathcal{L}} \|\mathbf{y}\|_{\mathcal{L}} \cosh \theta_{\mathbf{x}, \mathbf{y}} = -R^2 \cosh \theta_{\mathbf{x}, \mathbf{y}},$$

due to the hyperboloid variant of Cauchy-Schwarz (Ratcliffe, 2006, Theorem 3.1.6) (see Figure A.2). Subsequently

$$\theta_{\mathbf{x}, \mathbf{y}} = \cosh^{-1} \frac{\langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}}}{\|\mathbf{x}\|_{\mathcal{L}} \|\mathbf{y}\|_{\mathcal{L}}} = \cosh^{-1} \left( -\frac{\langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}}}{R^2} \right) = \cosh^{-1} (K \langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}}),$$

because

$$\|\mathbf{x}\|_{\mathcal{L}} = \sqrt{\langle \mathbf{x}, \mathbf{x} \rangle_{\mathcal{L}}} = \sqrt{-R^2} = iR \quad \forall \mathbf{x} \in \mathbb{H}_K^n,$$

and

$$\|\mathbf{x}\|_{\mathcal{L}} \|\mathbf{y}\|_{\mathcal{L}} = \left( \sqrt{-R^2} \right)^2 = (iR)^2 = -R^2 \quad \forall \mathbf{x}, \mathbf{y} \in \mathbb{H}_K^n,$$

which simplifies the angle formula in that space.

## Appendix B

---

# Probability details

---

### B.1 Hyperspherical uniform distribution

The notion of a “uniform” distribution with uniform probability mass at every point defined on a surface in  $\mathbb{R}^n$  can be naturally applied to hyperspheres  $\mathbb{S}_K^n$ , where for any point  $\mathbf{x} \in \mathbb{R}^{n+1}$  it holds that

$$U(\mathbf{x}; \mathbb{S}_K^n) = \begin{cases} \frac{1}{S_n(R)} & \text{if } \mathbf{x} \in \mathbb{S}_K^n \\ 0 & \text{otherwise,} \end{cases}$$

where  $S_n(R)$  denotes the surface area of an  $\mathbb{S}_K^n$  with radius  $R = 1/\sqrt{K}$

$$S_n(R) = \frac{2 \left( \pi^{\frac{n+1}{2}} \right)}{\Gamma \left( \frac{n+1}{2} \right)} R^n.$$

This distribution is useful, as it provides a good prior for representations on the sphere.

We see that the probability  $U(\mathbf{x}; \mathbb{S}_K^n)$  only depends on the curvature and dimensionality (i.e. does not depend on  $\mathbf{x}$ ), and is constant non-zero on all points of  $\mathbb{S}_K^n$ . Figure B.1 shows a plot of how the surface area of a hypersphere changes with respect to its parameters.

To efficiently sample from the distribution  $U(\mathbb{S}_K^n)$ , we sample  $\mathbf{x} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ ,  $\mathbf{x} \in \mathbb{R}^{n+1}$ . We then normalize to obtain a sample

$$\frac{R}{\|\mathbf{x}\|_2} \mathbf{x} \sim U(\mathbb{S}_K^n).$$

For a proof of this method and a more thorough discussion, see Muller (1959).

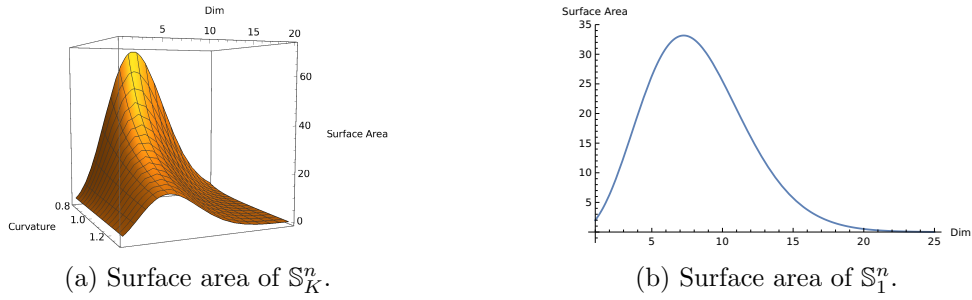


Figure B.1: Surface area plots for spheres of variable radius in  $n$ -dimensional spaces.

## B.2 Von Mises-Fisher distribution

**Remark (vMF distribution on  $\mathbb{S}_K^n$ )**

$$\mathbf{x} \sim \text{vMF}(\boldsymbol{\mu}, \kappa, K) \approx R\mathbf{x} \sim \text{vMF}(\boldsymbol{\mu}, \kappa'),$$

where  $\kappa' \propto \kappa \cdot K^{n/2} = \kappa \cdot \frac{1}{R^n}$ .

Even though this is just a (crude) approximation, the intuition behind it is that the vMF distribution on the unitary hypersphere with a given  $\kappa$  should scale with respect to the radius approximately like the uniform distribution does.

## B.3 Wrapped Normal distributions

**Theorem B.2 (Probability density function of  $\mathcal{WN}(z; \boldsymbol{\mu}, \boldsymbol{\Sigma})$  in  $\mathbb{H}^n$ )**

$$\log \mathcal{WN}(z; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \log \mathcal{N}(\mathbf{v}; \mathbf{0}, \boldsymbol{\Sigma}) - (n-1) \log \left( \frac{\sinh(\|\mathbf{u}\|_{\mathcal{L}})}{\|\mathbf{u}\|_{\mathcal{L}}} \right),$$

where  $\mathbf{u} = \log_{\boldsymbol{\mu}}(z)$  and  $\mathbf{v} = \text{PT}_{\boldsymbol{\mu} \rightarrow \boldsymbol{\mu}_0}(\mathbf{u}) \in \mathcal{T}_{\boldsymbol{\mu}_0} \mathbb{H}^n$ .

**Proof** This was shown by Nagano et al. (2019). We reproduce it here because the following theorems about Wrapped Normal distributions in other manifolds build on this.

Given two random variables  $\mathbf{x}, \mathbf{y}$  such that  $\mathbf{y} = f(\mathbf{x})$  for an invertible and continuous map  $f$  (Devore and Berk, 2012, Section 5.4), it holds that

$$\log p(\mathbf{y}) = \log p(\mathbf{x}) - \log \det \left( \frac{\partial f}{\partial \mathbf{x}} \right).$$

In our case,  $f = \exp_{\boldsymbol{\mu}} \circ \text{PT}_{\boldsymbol{\mu}_0 \rightarrow \boldsymbol{\mu}}$ , and  $f^{-1} = \text{PT}_{\boldsymbol{\mu} \rightarrow \boldsymbol{\mu}_0} \circ \log_{\boldsymbol{\mu}}$ .

The determinant decomposes using the chain rule and the fact that  $\det(\mathbf{AB}) = \det(\mathbf{A}) \det(\mathbf{B})$ :

$$\det \left( \frac{\partial f(\mathbf{v})}{\partial \mathbf{v}} \right) = \det \left( \frac{\partial \exp_{\mu}(\mathbf{u})}{\partial \mathbf{u}} \right) \cdot \det \left( \frac{\partial \text{PT}_{\mu_0 \rightarrow \mu}(\mathbf{v})}{\partial \mathbf{v}} \right).$$

The derivative of parallel transport  $\text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}(\mathbf{v})$  for any  $\mathbf{x}, \mathbf{y} \in \mathbb{H}^n$  and  $\mathbf{v} \in \mathcal{T}_{\mathbf{x}}\mathbb{H}^n$  is a map  $\text{dPT}_{\mathbf{x} \rightarrow \mathbf{y}}(\mathbf{v}) : \mathcal{T}_{\mathbf{v}}(\mathcal{T}_{\mathbf{x}}\mathbb{H}^n)$ . Using the orthonormal basis (with respect to the Lorentz product)  $\{\boldsymbol{\xi}_1, \dots, \boldsymbol{\xi}_n\}$ , we can compute the determinant by computing the change with respect to each basis vector.

$$\begin{aligned} \text{dPT}_{\mathbf{x} \rightarrow \mathbf{y}}(\boldsymbol{\xi}) &= \left. \frac{\partial}{\partial \varepsilon} \right|_{\varepsilon=0} \text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}(\mathbf{v} + \varepsilon \boldsymbol{\xi}) \\ &= \left. \frac{\partial}{\partial \varepsilon} \right|_{\varepsilon=0} \left[ (\mathbf{v} + \varepsilon \boldsymbol{\xi}) + \frac{\langle \mathbf{y}, \mathbf{v} + \varepsilon \boldsymbol{\xi} \rangle_{\mathcal{L}}}{1 - \langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}}} (\mathbf{x} + \mathbf{y}) \right] \\ &= \left[ \boldsymbol{\xi} + \frac{\langle \mathbf{y}, \boldsymbol{\xi} \rangle_{\mathcal{L}}}{1 - \langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}}} (\mathbf{x} + \mathbf{y}) \right]_{\varepsilon=0} \\ &= \text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}(\boldsymbol{\xi}). \end{aligned}$$

Since parallel transport preserves norms and vectors in the orthonormal basis have norm 1, the change is  $\|\text{dPT}_{\mathbf{x} \rightarrow \mathbf{y}}(\boldsymbol{\xi})\|_{\mathcal{L}} = \|\text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}(\boldsymbol{\xi})\|_{\mathcal{L}} = 1$ .

For computing the determinant of the exponential map Jacobian, we choose the orthonormal basis  $\{\boldsymbol{\xi}_1 = \mathbf{u}/\|\mathbf{u}\|_{\mathcal{L}}, \boldsymbol{\xi}_2, \dots, \boldsymbol{\xi}_n\}$ , where we just completed the basis based on the first vector. We again look at the change with respect to each basis vector. For the basis vector  $\boldsymbol{\xi}_1$ :

$$\begin{aligned} \text{dexp}_{\mathbf{x}}(\boldsymbol{\xi}_1) &= \left. \frac{\partial}{\partial \varepsilon} \right|_{\varepsilon=0} \exp_{\mathbf{x}} \left( \mathbf{u} + \varepsilon \frac{\mathbf{u}}{\|\mathbf{u}\|_{\mathcal{L}}} \right) \\ &= \left. \frac{\partial}{\partial \varepsilon} \right|_{\varepsilon=0} \left[ \cosh \left( \left\| \mathbf{u} + \varepsilon \frac{\mathbf{u}}{\|\mathbf{u}\|_{\mathcal{L}}} \right\|_{\mathcal{L}} \right) \mathbf{x} + \frac{\sinh \left( \left\| \mathbf{u} + \varepsilon \frac{\mathbf{u}}{\|\mathbf{u}\|_{\mathcal{L}}} \right\|_{\mathcal{L}} \right)}{\left\| \mathbf{u} + \varepsilon \frac{\mathbf{u}}{\|\mathbf{u}\|_{\mathcal{L}}} \right\|_{\mathcal{L}}} \left( \mathbf{u} + \varepsilon \frac{\mathbf{u}}{\|\mathbf{u}\|_{\mathcal{L}}} \right) \right] \\ &= \left. \frac{\partial}{\partial \varepsilon} \right|_{\varepsilon=0} \left[ \cosh (|\|\mathbf{u}\|_{\mathcal{L}} + \varepsilon|) \mathbf{x} + \frac{\sinh (|\|\mathbf{u}\|_{\mathcal{L}} + \varepsilon|)}{\|\mathbf{u}\|_{\mathcal{L}} |\|\mathbf{u}\|_{\mathcal{L}} + \varepsilon|} (\|\mathbf{u}\|_{\mathcal{L}} + \varepsilon) \mathbf{u} \right] \\ &= \left[ \frac{(\|\mathbf{u}\|_{\mathcal{L}} + \varepsilon) \sinh (|\|\mathbf{u}\|_{\mathcal{L}} + \varepsilon|)}{|\|\mathbf{u}\|_{\mathcal{L}} + \varepsilon|} \mathbf{x} + \frac{\cosh (|\|\mathbf{u}\|_{\mathcal{L}} + \varepsilon|)}{\|\mathbf{u}\|_{\mathcal{L}}} \mathbf{u} \right]_{\varepsilon=0} \\ &= \frac{\|\mathbf{u}\|_{\mathcal{L}} \sinh (\|\mathbf{u}\|_{\mathcal{L}})}{\|\mathbf{u}\|_{\mathcal{L}}} \mathbf{x} + \frac{\cosh (\|\mathbf{u}\|_{\mathcal{L}})}{\|\mathbf{u}\|_{\mathcal{L}}} \mathbf{u} \\ &= \sinh (\|\mathbf{u}\|_{\mathcal{L}}) \mathbf{x} + \cosh (\|\mathbf{u}\|_{\mathcal{L}}) \frac{\mathbf{u}}{\|\mathbf{u}\|_{\mathcal{L}}}, \end{aligned}$$

where the third equality is due to

$$\left\| \mathbf{u} + \varepsilon \frac{\mathbf{u}}{\|\mathbf{u}\|_{\mathcal{L}}} \right\|_{\mathcal{L}} = \left\| \left( 1 + \frac{\varepsilon}{\|\mathbf{u}\|_{\mathcal{L}}} \right) \mathbf{u} \right\|_{\mathcal{L}} = \left| 1 + \frac{\varepsilon}{\|\mathbf{u}\|_{\mathcal{L}}} \right| \|\mathbf{u}\|_{\mathcal{L}} = |\|\mathbf{u}\|_{\mathcal{L}} + \varepsilon|.$$

For every other basis vector  $\boldsymbol{\xi}_k$  where  $k > 1$ :

$$\begin{aligned}
 d \exp_{\mathbf{x}}(\boldsymbol{\xi}) &= \\
 &= \left. \frac{\partial}{\partial \varepsilon} \right|_{\varepsilon=0} \exp_{\mathbf{x}}(\mathbf{u} + \varepsilon \boldsymbol{\xi}) \\
 &= \left. \frac{\partial}{\partial \varepsilon} \right|_{\varepsilon=0} \left[ \cosh(\|\mathbf{u} + \varepsilon \boldsymbol{\xi}\|_{\mathcal{L}}) \mathbf{x} + \frac{\sinh(\|\mathbf{u} + \varepsilon \boldsymbol{\xi}\|_{\mathcal{L}})}{\|\mathbf{u} + \varepsilon \boldsymbol{\xi}\|_{\mathcal{L}}} (\mathbf{u} + \varepsilon \boldsymbol{\xi}) \right] \\
 &= \left. \frac{\partial}{\partial \varepsilon} \right|_{\varepsilon=0} \left[ \cosh\left(\sqrt{\|\mathbf{u}\|_{\mathcal{L}}^2 + \varepsilon^2}\right) \mathbf{x} + \frac{\sinh\left(\sqrt{\|\mathbf{u}\|_{\mathcal{L}}^2 + \varepsilon^2}\right)}{\sqrt{\|\mathbf{u}\|_{\mathcal{L}}^2 + \varepsilon^2}} (\mathbf{u} + \varepsilon \boldsymbol{\xi}) \right] \\
 &= \left[ \frac{\varepsilon \cosh\left(\sqrt{\|\mathbf{u}\|_{\mathcal{L}}^2 + \varepsilon^2}\right)}{\|\mathbf{u}\|_{\mathcal{L}}^2 + \varepsilon^2} (\mathbf{u} + \varepsilon \boldsymbol{\xi}) \right. \\
 &\quad \left. + \frac{(\|\mathbf{u}\|_{\mathcal{L}}^2 \boldsymbol{\xi} - \varepsilon \mathbf{u} + \varepsilon(\|\mathbf{u}\|_{\mathcal{L}}^2 + \varepsilon^2) \mathbf{x}) \sinh\left(\sqrt{\|\mathbf{u}\|_{\mathcal{L}}^2 + \varepsilon^2}\right)}{(\|\mathbf{u}\|_{\mathcal{L}}^2 + \varepsilon^2)^{3/2}} \right]_{\varepsilon=0} \\
 &= \frac{\|\mathbf{u}\|_{\mathcal{L}}^2 \sinh(\|\mathbf{u}\|_{\mathcal{L}})}{(\|\mathbf{u}\|_{\mathcal{L}}^2)^{3/2}} \boldsymbol{\xi} = \frac{\sinh(\|\mathbf{u}\|_{\mathcal{L}})}{\|\mathbf{u}\|_{\mathcal{L}}} \boldsymbol{\xi}.
 \end{aligned}$$

The third equality holds because

$$\begin{aligned}
 \|\mathbf{u} + \varepsilon \boldsymbol{\xi}\|_{\mathcal{L}}^2 &= \|\mathbf{u}\|_{\mathcal{L}}^2 + \varepsilon^2 \|\boldsymbol{\xi}\|_{\mathcal{L}}^2 - 2 \langle \mathbf{u}, \varepsilon \boldsymbol{\xi} \rangle_{\mathcal{L}} \\
 &= \|\mathbf{u}\|_{\mathcal{L}}^2 + \varepsilon^2 - 2\varepsilon \langle \mathbf{u}, \boldsymbol{\xi} \rangle_{\mathcal{L}} \\
 &= \|\mathbf{u}\|_{\mathcal{L}}^2 + \varepsilon^2,
 \end{aligned}$$

where the last equality relies on the fact that the basis is orthogonal, and  $\mathbf{u}$  is parallel to  $\boldsymbol{\xi}_1 = \mathbf{u}/\|\mathbf{u}\|_{\mathcal{L}}$ , hence it is orthogonal to all the other basis vectors.

Because the basis is orthonormal the determinant is a product of the norms of the computed change for each basis vector. Therefore,

$$\det \left( \frac{\partial \text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}(\mathbf{v})}{\partial \mathbf{v}} \right) = 1^n = 1.$$

Additionally, the following two properties hold:

$$\begin{aligned}
 \left\| \sinh(\|\mathbf{u}\|_{\mathcal{L}}) \mathbf{x} + \cosh(\|\mathbf{u}\|_{\mathcal{L}}) \frac{\mathbf{u}}{\|\mathbf{u}\|_{\mathcal{L}}} \right\|_{\mathcal{L}}^2 &= \sinh^2(\|\mathbf{u}\|_{\mathcal{L}}) \|\mathbf{x}\|_{\mathcal{L}}^2 + \cosh^2(\|\mathbf{u}\|_{\mathcal{L}}) \frac{\|\mathbf{u}\|_{\mathcal{L}}^2}{\|\mathbf{u}\|_{\mathcal{L}}^2} \\
 &= -\sinh^2(\|\mathbf{u}\|_{\mathcal{L}}) + \cosh^2(\|\mathbf{u}\|_{\mathcal{L}}) = 1,
 \end{aligned}$$

and

$$\left\| \frac{\sinh(\|\mathbf{u}\|_{\mathcal{L}})}{\|\mathbf{u}\|_{\mathcal{L}}} \boldsymbol{\xi} \right\|_{\mathcal{L}}^2 = \frac{\sinh^2(\|\mathbf{u}\|_{\mathcal{L}})}{\|\mathbf{u}\|_{\mathcal{L}}^2} \|\boldsymbol{\xi}\|_{\mathcal{L}}^2 = \frac{\sinh^2(\|\mathbf{u}\|_{\mathcal{L}})}{\|\mathbf{u}\|_{\mathcal{L}}^2}.$$

□

Therefore, we obtain

$$\det \left( \frac{\partial \exp_{\mathbf{x}}(\mathbf{u})}{\partial \mathbf{u}} \right) = 1 \cdot \left( \frac{\sinh(\|\mathbf{u}\|_{\mathcal{L}})}{\|\mathbf{u}\|_{\mathcal{L}}} \right)^{n-1}.$$

Finally,

$$\det \left( \frac{\partial f(\mathbf{v})}{\partial \mathbf{v}} \right) = \det \left( \frac{\partial \exp_{\boldsymbol{\mu}}(\mathbf{u})}{\partial \mathbf{u}} \right) \cdot \det \left( \frac{\partial \text{PT}_{\boldsymbol{\mu}_0 \rightarrow \boldsymbol{\mu}}(\mathbf{v})}{\partial \mathbf{v}} \right) = \left( \frac{\sinh(\|\mathbf{u}\|_{\mathcal{L}})}{\|\mathbf{u}\|_{\mathcal{L}}} \right)^{n-1}.$$

**Theorem B.3 (Probability density function of  $\mathcal{WN}(z; \boldsymbol{\mu}, \boldsymbol{\Sigma})$  in  $\mathbb{H}_K^n$ )**

$$\log \mathcal{WN}(z; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \log \mathcal{N}(\mathbf{v}; \mathbf{0}, \boldsymbol{\Sigma}) - (n-1) \log \left( \frac{R \sinh \left( \frac{\|\mathbf{u}\|_{\mathcal{L}}}{R} \right)}{\|\mathbf{u}\|_{\mathcal{L}}} \right),$$

where  $\mathbf{u} = \log_{\boldsymbol{\mu}}^K(z)$ ,  $\mathbf{v} = \text{PT}_{\boldsymbol{\mu} \rightarrow \boldsymbol{\mu}_0}^K(\mathbf{u})$ , and  $R = 1/\sqrt{-K}$ .

**Proof** The theorem is very similar to Theorem B.2. The difference is that in this one, we do not assume unitary radius  $R = 1 = 1/\sqrt{-K}$ . Hence, our transformation function has the form  $f = \exp_{\boldsymbol{\mu}}^K \circ \text{PT}_{\boldsymbol{\mu}_0 \rightarrow \boldsymbol{\mu}}^K$ , and  $f^{-1} = \text{PT}_{\boldsymbol{\mu} \rightarrow \boldsymbol{\mu}_0}^K \circ \log_{\boldsymbol{\mu}}^K$ .

The derivative of parallel transport  $\text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v})$  for any  $\mathbf{x}, \mathbf{y} \in \mathbb{H}_K^n$  and  $\mathbf{v} \in \mathcal{T}_{\mathbf{x}} \mathbb{H}_K^n$  is a map  $\text{dPT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}) : \mathcal{T}_{\mathbf{v}}(\mathcal{T}_{\mathbf{x}} \mathbb{H}_K^n)$ . Using the orthonormal basis (with respect to the Lorentz product)  $\{\boldsymbol{\xi}_1, \dots, \boldsymbol{\xi}_n\}$ , we can compute the determinant by computing the change with respect to each basis vector.

$$\begin{aligned} \text{dPT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\boldsymbol{\xi}) &= \frac{\partial}{\partial \varepsilon} \Big|_{\varepsilon=0} \text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v} + \varepsilon \boldsymbol{\xi}) \\ &= \frac{\partial}{\partial \varepsilon} \Big|_{\varepsilon=0} \left[ (\mathbf{v} + \varepsilon \boldsymbol{\xi}) + \frac{\langle \mathbf{y}, \mathbf{v} + \varepsilon \boldsymbol{\xi} \rangle_{\mathcal{L}}}{R^2 - \langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}}} (\mathbf{x} + \mathbf{y}) \right] \\ &= \left[ \boldsymbol{\xi} + \frac{\langle \mathbf{y}, \boldsymbol{\xi} \rangle_{\mathcal{L}}}{R^2 - \langle \mathbf{x}, \mathbf{y} \rangle_{\mathcal{L}}} (\mathbf{x} + \mathbf{y}) \right]_{\varepsilon=0} \\ &= \text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\boldsymbol{\xi}). \end{aligned}$$

Since parallel transport preserves norms and vectors in the orthonormal basis have norm 1, the change is  $\|\text{dPT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\boldsymbol{\xi})\|_{\mathcal{L}} = \|\text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}^K(\boldsymbol{\xi})\|_{\mathcal{L}} = 1$ .



For computing the determinant of the exponential map Jacobian, we choose the orthonormal basis  $\{\boldsymbol{\xi}_1 = \mathbf{u}/\|\mathbf{u}\|_{\mathcal{L}}, \boldsymbol{\xi}_2, \dots, \boldsymbol{\xi}_n\}$ , where we just completed the basis based on the first vector. We again look at the change with respect to each basis vector. For the basis vector  $\boldsymbol{\xi}_1$ :

$$\begin{aligned}
 d \exp_x^K(\boldsymbol{\xi}_1) &= \\
 &= \frac{\partial}{\partial \varepsilon} \Big|_{\varepsilon=0} \exp_x^K \left( \mathbf{u} + \varepsilon \frac{\mathbf{u}}{\|\mathbf{u}\|_{\mathcal{L}}} \right) \\
 &= \frac{\partial}{\partial \varepsilon} \Big|_{\varepsilon=0} \left[ \cosh \left( \frac{\|\mathbf{u}\|_{\mathcal{L}} + \varepsilon}{R} \right) \mathbf{x} + \frac{R \sinh \left( \frac{\|\mathbf{u}\|_{\mathcal{L}} + \varepsilon}{R} \right)}{\|\mathbf{u}\|_{\mathcal{L}} \|\mathbf{u}\|_{\mathcal{L}} + \varepsilon} (\|\mathbf{u}\|_{\mathcal{L}} + \varepsilon) \mathbf{u} \right] \\
 &= \left[ \frac{(\|\mathbf{u}\|_{\mathcal{L}} + \varepsilon) \sinh \left( \frac{\|\mathbf{u}\|_{\mathcal{L}} + \varepsilon}{R} \right)}{R \|\mathbf{u}\|_{\mathcal{L}} + \varepsilon} \mathbf{x} + \frac{\cosh \left( \frac{\|\mathbf{u}\|_{\mathcal{L}} + \varepsilon}{R} \right)}{\|\mathbf{u}\|_{\mathcal{L}}} \mathbf{u} \right]_{\varepsilon=0} \\
 &= \sinh \left( \frac{\|\mathbf{u}\|_{\mathcal{L}}}{R} \right) \frac{\mathbf{x}}{R} + \cosh \left( \frac{\|\mathbf{u}\|_{\mathcal{L}}}{R} \right) \frac{\mathbf{u}}{\|\mathbf{u}\|_{\mathcal{L}}},
 \end{aligned}$$

where the second equality is due to

$$\left\| \mathbf{u} + \varepsilon \frac{\mathbf{u}}{\|\mathbf{u}\|_{\mathcal{L}}} \right\|_{\mathcal{L}} = \left\| \left( 1 + \frac{\varepsilon}{\|\mathbf{u}\|_{\mathcal{L}}} \right) \mathbf{u} \right\|_{\mathcal{L}} = \left| 1 + \frac{\varepsilon}{\|\mathbf{u}\|_{\mathcal{L}}} \right| \|\mathbf{u}\|_{\mathcal{L}} = \|\mathbf{u}\|_{\mathcal{L}} + \varepsilon.$$

For every other basis vector  $\boldsymbol{\xi}_k$  where  $k > 1$ :

$$\begin{aligned}
 d \exp_{\mathbf{x}}^K(\boldsymbol{\xi}) &= \\
 &= \frac{\partial}{\partial \varepsilon} \Big|_{\varepsilon=0} \exp_{\mathbf{x}}^K(\mathbf{u} + \varepsilon \boldsymbol{\xi}) \\
 &= \frac{\partial}{\partial \varepsilon} \Big|_{\varepsilon=0} \left[ \cosh \left( \frac{\|\mathbf{u} + \varepsilon \boldsymbol{\xi}\|_{\mathcal{L}}}{R} \right) \mathbf{x} + \frac{R \sinh \left( \frac{\|\mathbf{u} + \varepsilon \boldsymbol{\xi}\|_{\mathcal{L}}}{R} \right)}{\|\mathbf{u} + \varepsilon \boldsymbol{\xi}\|_{\mathcal{L}}} (\mathbf{u} + \varepsilon \boldsymbol{\xi}) \right] \\
 &= \frac{\partial}{\partial \varepsilon} \Big|_{\varepsilon=0} \left[ \cosh \left( \frac{\sqrt{\|\mathbf{u}\|_{\mathcal{L}}^2 + \varepsilon^2}}{R} \right) \mathbf{x} + \frac{R \sinh \left( \frac{\sqrt{\|\mathbf{u}\|_{\mathcal{L}}^2 + \varepsilon^2}}{R} \right)}{\sqrt{\|\mathbf{u}\|_{\mathcal{L}}^2 + \varepsilon^2}} (\mathbf{u} + \varepsilon \boldsymbol{\xi}) \right] \\
 &= \left[ \frac{\varepsilon \cosh \left( \frac{\sqrt{\|\mathbf{u}\|_{\mathcal{L}}^2 + \varepsilon^2}}{R} \right)}{\|\mathbf{u}\|_{\mathcal{L}}^2 + \varepsilon^2} (\mathbf{u} + \varepsilon \boldsymbol{\xi}) \right. \\
 &\quad \left. + \frac{(R^2 \|\mathbf{u}\|_{\mathcal{L}}^2 \boldsymbol{\xi} - R^2 \varepsilon \mathbf{u} + \varepsilon (\|\mathbf{u}\|_{\mathcal{L}}^2 + \varepsilon^2) \mathbf{x}) \sinh \left( \frac{\sqrt{\|\mathbf{u}\|_{\mathcal{L}}^2 + \varepsilon^2}}{R} \right)}{R (\|\mathbf{u}\|_{\mathcal{L}}^2 + \varepsilon^2)^{3/2}} \right]_{\varepsilon=0} \\
 &= \frac{R^2 \|\mathbf{u}\|_{\mathcal{L}}^2 \sinh \left( \frac{\|\mathbf{u}\|_{\mathcal{L}}}{R} \right)}{R (\|\mathbf{u}\|_{\mathcal{L}}^2)^{3/2}} \boldsymbol{\xi} = \frac{R \sinh \left( \frac{\|\mathbf{u}\|_{\mathcal{L}}}{R} \right)}{\|\mathbf{u}\|_{\mathcal{L}}} \boldsymbol{\xi},
 \end{aligned}$$

where the third equality holds because

$$\begin{aligned}
 \|\mathbf{u} + \varepsilon \boldsymbol{\xi}\|_{\mathcal{L}}^2 &= \|\mathbf{u}\|_{\mathcal{L}}^2 + \varepsilon^2 \|\boldsymbol{\xi}\|_{\mathcal{L}}^2 - 2 \langle \mathbf{u}, \varepsilon \boldsymbol{\xi} \rangle_{\mathcal{L}} \\
 &= \|\mathbf{u}\|_{\mathcal{L}}^2 + \varepsilon^2 - 2\varepsilon \langle \mathbf{u}, \boldsymbol{\xi} \rangle_{\mathcal{L}} \\
 &= \|\mathbf{u}\|_{\mathcal{L}}^2 + \varepsilon^2,
 \end{aligned}$$

where the last equality relies on the fact that the basis is orthogonal, and  $\mathbf{u}$  is parallel to  $\boldsymbol{\xi}_1 = \mathbf{u} / \|\mathbf{u}\|_{\mathcal{L}}$ , hence it is orthogonal to all the other basis vectors.

Because the basis is orthonormal the determinant is a product of the norms of the computed change for each basis vector. Therefore,

$$\det \left( \frac{\partial \text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}(\mathbf{v})}{\partial \mathbf{v}} \right) = 1^n = 1.$$

Additionally, the following two properties hold:

$$\begin{aligned}
 \left\| \mathrm{d} \exp_{\mathbf{x}}^K \left( \frac{\mathbf{u}}{\|\mathbf{u}\|_{\mathcal{L}}} \right) \right\|_{\mathcal{L}}^2 &= \left\| \sinh \left( \frac{\|\mathbf{u}\|_{\mathcal{L}}}{R} \right) \frac{\mathbf{x}}{R} + \cosh \left( \frac{\|\mathbf{u}\|_{\mathcal{L}}}{R} \right) \frac{\mathbf{u}}{\|\mathbf{u}\|_{\mathcal{L}}} \right\|_{\mathcal{L}}^2 \\
 &= \sinh^2 \left( \frac{\|\mathbf{u}\|_{\mathcal{L}}}{R} \right) \frac{\|\mathbf{x}\|_{\mathcal{L}}^2}{R^2} + \cosh^2 \left( \frac{\|\mathbf{u}\|_{\mathcal{L}}}{R} \right) \frac{\|\mathbf{u}\|_{\mathcal{L}}^2}{\|\mathbf{u}\|_{\mathcal{L}}^2} \\
 &= -\sinh^2 \left( \frac{\|\mathbf{u}\|_{\mathcal{L}}}{R} \right) + \cosh^2 \left( \frac{\|\mathbf{u}\|_{\mathcal{L}}}{R} \right) = 1.
 \end{aligned}$$

and

$$\begin{aligned}
 \left\| \mathrm{d} \exp_{\mathbf{x}}^K (\boldsymbol{\xi}) \right\|_{\mathcal{L}}^2 &= \left\| \frac{R \sinh \left( \frac{\|\mathbf{u}\|_{\mathcal{L}}}{R} \right)}{\|\mathbf{u}\|_{\mathcal{L}}} \boldsymbol{\xi} \right\|_{\mathcal{L}}^2 \\
 &= \frac{R^2 \sinh^2 \left( \frac{\|\mathbf{u}\|_{\mathcal{L}}}{R} \right)}{\|\mathbf{u}\|_{\mathcal{L}}^2} \|\boldsymbol{\xi}\|_{\mathcal{L}}^2 \\
 &= \frac{R^2 \sinh^2 \left( \frac{\|\mathbf{u}\|_{\mathcal{L}}}{R} \right)}{\|\mathbf{u}\|_{\mathcal{L}}^2}. \quad \square
 \end{aligned}$$

Therefore, we obtain

$$\det \left( \frac{\partial \exp_{\mathbf{x}}^K(\mathbf{u})}{\partial \mathbf{u}} \right) = 1 \cdot \left( \frac{R \sinh \left( \frac{\|\mathbf{u}\|_{\mathcal{L}}}{R} \right)}{\|\mathbf{u}\|_{\mathcal{L}}} \right)^{n-1}.$$

Finally,

$$\det \left( \frac{\partial f(\mathbf{v})}{\partial \mathbf{v}} \right) = \det \left( \frac{\partial \exp_{\mu}^K(\mathbf{u})}{\partial \mathbf{u}} \right) \cdot \det \left( \frac{\partial \mathrm{PT}_{\mu_0 \rightarrow \mu}^K(\mathbf{v})}{\partial \mathbf{v}} \right) = \left( \frac{R \sinh \left( \frac{\|\mathbf{u}\|_{\mathcal{L}}}{R} \right)}{\|\mathbf{u}\|_{\mathcal{L}}} \right)^{n-1}.$$

**Theorem B.4 (Probability density function of  $\mathcal{WN}(z; \boldsymbol{\mu}, \boldsymbol{\Sigma})$  in  $\mathbb{S}_K^n$ )**

$$\log \mathcal{WN}(z; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \log \mathcal{N}(\mathbf{v}; \mathbf{0}, \boldsymbol{\Sigma}) - (n-1) \log \left( \frac{R \left| \sin \left( \frac{\|\mathbf{u}\|_2}{R} \right) \right|}{\|\mathbf{u}\|_2} \right),$$

where  $\mathbf{u} = \log_{\boldsymbol{\mu}}^K(z)$ ,  $\mathbf{v} = \mathrm{PT}_{\boldsymbol{\mu} \rightarrow \boldsymbol{\mu}_0}^K(\mathbf{u})$ , and  $R = 1/\sqrt{K}$ .

**Proof** The theorem is very similar to Theorem B.3. The difference is that in this one, our manifold changes from  $\mathbb{H}_K^n$  to  $\mathbb{S}_K^n$ , hence  $K > 0$ . Our transformation function has the form  $f = \exp_{\boldsymbol{\mu}}^K \circ \mathrm{PT}_{\boldsymbol{\mu}_0 \rightarrow \boldsymbol{\mu}}^K$ , and  $f^{-1} = \mathrm{PT}_{\boldsymbol{\mu} \rightarrow \boldsymbol{\mu}_0}^K \circ \log_{\boldsymbol{\mu}}^K$ .

The derivative of parallel transport  $PT_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v})$  for any  $\mathbf{x}, \mathbf{y} \in \mathbb{S}_K^n$  and  $\mathbf{v} \in \mathcal{T}_{\mathbf{x}}\mathbb{S}_K^n$  is a map  $dPT_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v}) : \mathcal{T}_{\mathbf{v}}(\mathcal{T}_{\mathbf{x}}\mathbb{S}_K^n)$ . Using the orthonormal basis (with respect to the Lorentz product)  $\{\boldsymbol{\xi}_1, \dots, \boldsymbol{\xi}_n\}$ , we can compute the determinant by computing the change with respect to each basis vector.

$$\begin{aligned} dPT_{\mathbf{x} \rightarrow \mathbf{y}}^K(\boldsymbol{\xi}) &= \left. \frac{\partial}{\partial \varepsilon} \right|_{\varepsilon=0} PT_{\mathbf{x} \rightarrow \mathbf{y}}^K(\mathbf{v} + \varepsilon \boldsymbol{\xi}) \\ &= \left. \frac{\partial}{\partial \varepsilon} \right|_{\varepsilon=0} \left[ (\mathbf{v} + \varepsilon \boldsymbol{\xi}) - \frac{\langle \mathbf{y}, \mathbf{v} + \varepsilon \boldsymbol{\xi} \rangle_2}{R^2 + \langle \mathbf{x}, \mathbf{y} \rangle_2} (\mathbf{x} + \mathbf{y}) \right] \\ &= \left[ \boldsymbol{\xi} - \frac{\langle \mathbf{y}, \boldsymbol{\xi} \rangle_2}{R^2 + \langle \mathbf{x}, \mathbf{y} \rangle_2} (\mathbf{x} + \mathbf{y}) \right]_{\varepsilon=0} \\ &= PT_{\mathbf{x} \rightarrow \mathbf{y}}^K(\boldsymbol{\xi}). \end{aligned}$$

Since parallel transport preserves norms and vectors in the orthonormal basis have norm 1, the change is  $\|dPT_{\mathbf{x} \rightarrow \mathbf{y}}^K(\boldsymbol{\xi})\|_2 = \|PT_{\mathbf{x} \rightarrow \mathbf{y}}^K(\boldsymbol{\xi})\|_2 = 1$ .

For computing the determinant of the exponential map Jacobian, we choose the orthonormal basis  $\{\boldsymbol{\xi}_1 = \mathbf{u}/\|\mathbf{u}\|_2, \boldsymbol{\xi}_2, \dots, \boldsymbol{\xi}_n\}$ , where we just completed the basis based on the first vector. We again look at the change with respect to each basis vector. For the basis vector  $\boldsymbol{\xi}_1$ :

$$\begin{aligned} d\exp_{\mathbf{x}}^K(\boldsymbol{\xi}_1) &= \\ &= \left. \frac{\partial}{\partial \varepsilon} \right|_{\varepsilon=0} \exp_{\mathbf{x}}^K \left( \mathbf{u} + \varepsilon \frac{\mathbf{u}}{\|\mathbf{u}\|_2} \right) \\ &= \left. \frac{\partial}{\partial \varepsilon} \right|_{\varepsilon=0} \left[ \cos \left( \frac{\|\mathbf{u}\|_2 + \varepsilon}{R} \right) \mathbf{x} + \frac{R \sin \left( \frac{\|\mathbf{u}\|_2 + \varepsilon}{R} \right)}{\|\mathbf{u}\|_2 \|\mathbf{u}\|_2 + \varepsilon} (\|\mathbf{u}\|_2 + \varepsilon) \mathbf{u} \right] \\ &= \left[ -\frac{(\|\mathbf{u}\|_2 + \varepsilon) \sin \left( \frac{\|\mathbf{u}\|_2 + \varepsilon}{R} \right)}{R \|\mathbf{u}\|_2 + \varepsilon} \mathbf{x} + \frac{\cos \left( \frac{\|\mathbf{u}\|_2 + \varepsilon}{R} \right)}{\|\mathbf{u}\|_2} \mathbf{u} \right]_{\varepsilon=0} \\ &= \cos \left( \frac{\|\mathbf{u}\|_2}{R} \right) \frac{\mathbf{u}}{\|\mathbf{u}\|_2} - \sin \left( \frac{\|\mathbf{u}\|_2}{R} \right) \frac{\mathbf{x}}{R}, \end{aligned}$$

where the second equality is due to

$$\left\| \mathbf{u} + \varepsilon \frac{\mathbf{u}}{\|\mathbf{u}\|_2} \right\|_2 = \left\| \left( 1 + \frac{\varepsilon}{\|\mathbf{u}\|_2} \right) \mathbf{u} \right\|_2 = \left| 1 + \frac{\varepsilon}{\|\mathbf{u}\|_2} \right| \|\mathbf{u}\|_2 = \|\mathbf{u}\|_2 + \varepsilon.$$

For every other basis vector  $\boldsymbol{\xi}_k$  where  $k > 1$ :

$$\begin{aligned}
 d \exp_{\mathbf{x}}^K(\boldsymbol{\xi}) &= \\
 &= \left. \frac{\partial}{\partial \varepsilon} \right|_{\varepsilon=0} \exp_{\mathbf{x}}^K(\mathbf{u} + \varepsilon \boldsymbol{\xi}) \\
 &= \left. \frac{\partial}{\partial \varepsilon} \right|_{\varepsilon=0} \left[ \cos \left( \frac{\|\mathbf{u} + \varepsilon \boldsymbol{\xi}\|_2}{R} \right) \mathbf{x} + \frac{R \sin \left( \frac{\|\mathbf{u} + \varepsilon \boldsymbol{\xi}\|_2}{R} \right)}{\|\mathbf{u} + \varepsilon \boldsymbol{\xi}\|_2} (\mathbf{u} + \varepsilon \boldsymbol{\xi}) \right] \\
 &= \left. \frac{\partial}{\partial \varepsilon} \right|_{\varepsilon=0} \left[ \cos \left( \frac{\sqrt{\|\mathbf{u}\|_2^2 + \varepsilon^2}}{R} \right) \mathbf{x} + \frac{R \sin \left( \frac{\sqrt{\|\mathbf{u}\|_2^2 + \varepsilon^2}}{R} \right)}{\sqrt{\|\mathbf{u}\|_2^2 + \varepsilon^2}} (\mathbf{u} + \varepsilon \boldsymbol{\xi}) \right] \\
 &= \left[ \frac{\varepsilon \cos \left( \frac{\sqrt{\|\mathbf{u}\|_2^2 + \varepsilon^2}}{R} \right)}{\|\mathbf{u}\|_2^2 + \varepsilon^2} (\mathbf{u} + \varepsilon \boldsymbol{\xi}) \right. \\
 &\quad \left. + \frac{(R^2 \|\mathbf{u}\|_2^2 \boldsymbol{\xi} - R^2 \varepsilon \mathbf{u} - \varepsilon (\|\mathbf{u}\|_2^2 + \varepsilon^2) \mathbf{x}) \sin \left( \frac{\sqrt{\|\mathbf{u}\|_2^2 + \varepsilon^2}}{R} \right)}{R (\|\mathbf{u}\|_2^2 + \varepsilon^2)^{3/2}} \right]_{\varepsilon=0} \\
 &= \frac{R^2 \|\mathbf{u}\|_2^2 \sin \left( \frac{\|\mathbf{u}\|_2}{R} \right)}{R (\|\mathbf{u}\|_2^2)^{3/2}} \boldsymbol{\xi} = \frac{R \sin \left( \frac{\|\mathbf{u}\|_2}{R} \right)}{\|\mathbf{u}\|_2} \boldsymbol{\xi},
 \end{aligned}$$

where the third equality holds because

$$\begin{aligned}
 \|\mathbf{u} + \varepsilon \boldsymbol{\xi}\|_2^2 &= \|\mathbf{u}\|_2^2 + \varepsilon^2 \|\boldsymbol{\xi}\|_2^2 - 2 \langle \mathbf{u}, \varepsilon \boldsymbol{\xi} \rangle_2 \\
 &= \|\mathbf{u}\|_2^2 + \varepsilon^2 - 2\varepsilon \langle \mathbf{u}, \boldsymbol{\xi} \rangle_2 \\
 &= \|\mathbf{u}\|_2^2 + \varepsilon^2,
 \end{aligned}$$

where the last equality relies on the fact that the basis is orthogonal, and  $\mathbf{u}$  is parallel to  $\boldsymbol{\xi}_1 = \mathbf{u} / \|\mathbf{u}\|_2$ , hence it is orthogonal to all the other basis vectors.

Because the basis is orthonormal the determinant is a product of the norms of the computed change for each basis vector. Therefore,

$$\det \left( \frac{\partial \text{PT}_{\mathbf{x} \rightarrow \mathbf{y}}(\mathbf{v})}{\partial \mathbf{v}} \right) = 1^n = 1.$$

Additionally, the following two properties hold:

$$\begin{aligned} \left\| d \exp_{\mathbf{x}}^K \left( \frac{\mathbf{u}}{\|\mathbf{u}\|_2} \right) \right\|_2^2 &= \left\| \cos \left( \frac{\|\mathbf{u}\|_2}{R} \right) \frac{\mathbf{u}}{\|\mathbf{u}\|_2} - \sin \left( \frac{\|\mathbf{u}\|_2}{R} \right) \frac{\mathbf{x}}{R} \right\|_2^2 \\ &= \sin^2 \left( \frac{\|\mathbf{u}\|_2}{R} \right) \frac{\|\mathbf{x}\|_2^2}{R^2} + \cos^2 \left( \frac{\|\mathbf{u}\|_2}{R} \right) \frac{\|\mathbf{u}\|_2^2}{\|\mathbf{u}\|_2^2} \\ &= \sin^2 \left( \frac{\|\mathbf{u}\|_2}{R} \right) + \cos^2 \left( \frac{\|\mathbf{u}\|_2}{R} \right) = 1. \end{aligned}$$

and

$$\begin{aligned} \left\| d \exp_{\mathbf{x}}^K (\boldsymbol{\xi}) \right\|_2^2 &= \left\| \frac{R \sin \left( \frac{\|\mathbf{u}\|_2}{R} \right)}{\|\mathbf{u}\|_2} \boldsymbol{\xi} \right\|_2^2 \\ &= \frac{R^2 \sin^2 \left( \frac{\|\mathbf{u}\|_2}{R} \right)}{\|\mathbf{u}\|_2^2} \|\boldsymbol{\xi}\|_2^2 \\ &= \frac{R^2 \sin^2 \left( \frac{\|\mathbf{u}\|_2}{R} \right)}{\|\mathbf{u}\|_2^2}. \quad \square \end{aligned}$$

Therefore, we obtain

$$\det \left( \frac{\partial \exp_{\mathbf{x}}^K(\mathbf{u})}{\partial \mathbf{u}} \right) = 1 \cdot \left( \frac{R \left| \sin \left( \frac{\|\mathbf{u}\|_2}{R} \right) \right|}{\|\mathbf{u}\|_2} \right)^{n-1}.$$

Finally,

$$\det \left( \frac{\partial f(\mathbf{v})}{\partial \mathbf{v}} \right) = \det \left( \frac{\partial \exp_{\mu}^K(\mathbf{u})}{\partial \mathbf{u}} \right) \cdot \det \left( \frac{\partial \text{PT}_{\mu_0 \rightarrow \mu}^K(\mathbf{v})}{\partial \mathbf{v}} \right) = \left( \frac{R \sin \left| \left( \frac{\|\mathbf{u}\|_2}{R} \right) \right|}{\|\mathbf{u}\|_2} \right)^{n-1}.$$

**Theorem B.5 (Probability density function of  $\mathcal{WN}(\mathbf{z}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$  in  $\mathbb{P}_K^n$ )**

$$\log \mathcal{WN}_{\mathbb{P}_K^n}(\mathbf{z}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \log \mathcal{WN}_{\mathbb{H}_K^n}(\rho_K^{-1}(\mathbf{z}); \rho_K^{-1}(\boldsymbol{\mu}), \boldsymbol{\Sigma}).$$

**Proof** Follows from Theorem B.3 and A.11.

Also proven by (Mathieu et al., 2019) in a slightly different form for a scalar scale parameter  $\mathcal{WN}(\mathbf{z}; \boldsymbol{\mu}, \sigma^2 \mathbf{I})$ . Given

$$\begin{aligned} \log \mathcal{N}(\mathbf{z}; \boldsymbol{\mu}, \sigma^2 \mathbf{I}) &= -\frac{d_{\mathbb{E}}(\boldsymbol{\mu}, \mathbf{z})^2}{2\sigma^2} - \frac{n}{2} \log(2\pi\sigma^2) \\ \log \mathcal{WN}(\mathbf{z}; \boldsymbol{\mu}, \sigma^2 \mathbf{I}) &= -\frac{d_{\mathbb{P}}^K(\boldsymbol{\mu}, \mathbf{z})^2}{2\sigma^2} - \frac{n}{2} \log(2\pi\sigma^2) \\ &\quad + (n-1) \log \left( \frac{\sqrt{-K} d_{\mathbb{P}}^K(\boldsymbol{\mu}, \mathbf{z})}{\sinh(\sqrt{-K} d_{\mathbb{P}}^K(\boldsymbol{\mu}, \mathbf{z}))} \right). \quad \square \end{aligned}$$

**Theorem B.6 (Probability density function of  $\mathcal{WN}(z; \boldsymbol{\mu}, \boldsymbol{\Sigma})$  in  $\mathbb{D}_K^n$ )**

$$\log \mathcal{WN}_{\mathbb{D}_K^n}(z; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \log \mathcal{WN}_{\mathbb{S}_K^n}(\rho_K^{-1}(z); \rho_K^{-1}(\boldsymbol{\mu}), \boldsymbol{\Sigma}).$$

**Proof** Follows from Theorem B.4 and A.28. □

---

## Variational Autoencoders

---

### C.1 Why use Variational Autoencoders?

The motivation behind variational autoencoders is to be able to bring approximate variational inference (Section 4.2) to autoencoders (Section 4.1). The main benefit of this is learning “smooth” latent space representations, compared to standard autoencoders, as is illustrated in Figure C.1.

Essentially, if we only optimize the reconstruction term of the ELBO (similar to an autoencoder), the model tends to position learned representations in the space arbitrarily so that it can reconstruct as well as possible and ends up with a latent space that has “empty” parts where no observed data samples get encoded. Hence, if we take two input data points, encode them, and try to decode some of the latent representations on the shortest path between them, we will essentially end up with very abrupt changes in the reconstructed samples. This case loosely corresponds to a non-variational autoencoder. In

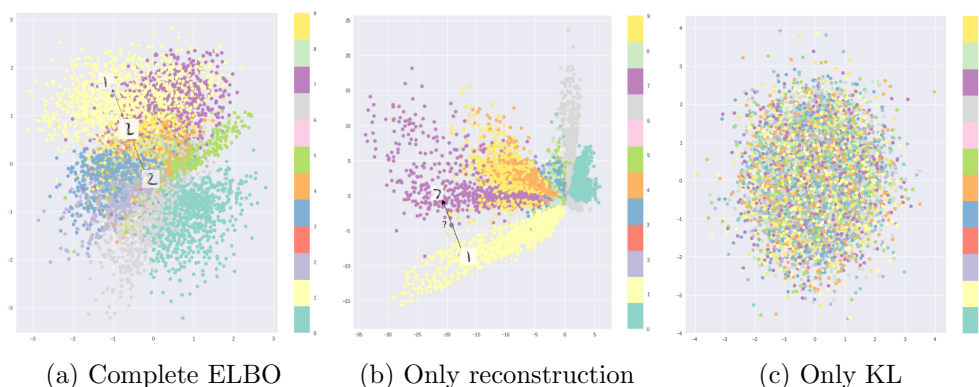


Figure C.1: VAE latent space representation plots of a VAE applied to MNIST digits, using different parts of the ELBO loss for optimization (Shafkat, 2018).



the case of a classical deterministic autoencoder, there is not even a formal way to “sample” from the model, as it is not stochastic.

If we optimize the complete ELBO, we get a regularization term in the form of the KL divergence between our decoder posterior and the chosen prior on the latent space. This means that the model will try to distribute all the latent representations of samples according to the chosen prior. A generalization of this intuition is the  $\beta$ -VAE (Matthey et al., 2017).

On the other end of the spectrum, if we only optimize the regularization part of the ELBO, we will end up with collapsed posteriors (Bowman et al., 2016; Chen et al., 2014). If the prior is a Normal distribution  $\mathcal{N}(\boldsymbol{\mu}, \sigma^2 \mathbf{I})$ , our encoder will force  $\sigma$  towards 0. Even if we constrain  $\sigma \geq 1$ , we will collapse to the prior  $\mathcal{N}(\mathbf{0}, \mathbf{I})$ .

---

## Extended results

---

### D.1 Implementation remarks

**Remark (Computability of functions with floating-point numbers)**

*Do note, that several functions we employ in either the geometrical operations, or probability distributions in Riemannian manifolds of constant curvature require some care to implement with numerical stability.*

*Notably, the arguments of  $\sinh$  and  $\cosh$  need to be clamped to around  $[-85, 85]$  to not run into infinite floating-point numbers.*

*Likewise,  $\cosh^{-1}$  and  $\tanh^{-1}$  need to be clamped to an  $\varepsilon$ -neighborhood on the edges of their domains. We use  $\varepsilon = 10^{-8}$ . Similarly,  $\sqrt{x}$  needs to be implemented as  $\sqrt{\max(\varepsilon, x)}$  to have stable gradients.*

*Do note, that both  $\log \circ \sinh$  and  $\log \circ \cosh$  have explicit forms that reduce to a “log-sum-exp” expression, which can be computed more efficiently and stably using the log-sum-exp function provided by a given numerical computation library.*

*Finally, some operations are more easily computable in practice when we reformulate them. For example, the original distance in a Poincaré ball (using  $\cosh^{-1}$ ) is very numerically unstable, while the gyrospace distance (using  $\tanh^{-1}$ ) is dramatically more stable.*

**Remark (Practical limitations of constant curvature manifolds)**

*As noted in Remark D.1, functions used in spaces of constant curvature often have to be artificially limited to stabilize training.*

*Additionally, for training in spaces like the Poincaré ball, we have to project to an “open ball”, which means again choosing a fixed  $\varepsilon$  constant, to move points away from the boundary. The distances on the hypersphere are by nature limited to  $R \cdot \pi$  (antipodal points), therefore (spherical) distances on the projected hypersphere are also limited.*

*For these reasons, the representable distances in these manifolds are often limited to only a few units. As we can see from our experiments, this does not impact learning representations much, but it might be a problem in other domains where the absolute values of distances is important as well.*

## D.2 Spherical covariance matrix

Model	LL	ELBO	BCE	KL
$(\mathbb{S}_1^2)^3$	-55.89±0.36	-56.72±0.40	51.01±0.31	5.72±0.09
$\mathbb{S}_1^6$	-55.81±0.35	-56.57±0.44	51.16±0.78	5.41±0.42
$(\text{vMF } \mathbb{S}_1^2)^3$	-57.87±1.52	-58.64±1.63	53.96±2.16	4.68±0.53
$\text{vMF } \mathbb{S}_1^6$	-58.78±0.83	-60.74±2.29	56.03±2.64	4.71±0.48
$(\mathbb{D}_1^2)^3$	-56.01±0.24	-56.67±0.31	51.02±0.40	5.65±0.10
$\mathbb{D}_1^6$	-55.78±0.07	-56.38±0.06	50.85±0.20	5.53±0.24
$(\mathbb{E}^2)^3$	-56.34±0.45	-56.94±0.50	51.32±0.55	5.62±0.19
$\mathbb{E}^6$	-56.28±0.56	-56.99±0.59	51.58±0.69	5.41±0.29
$(\mathbb{H}_{-1}^2)^3$	-56.08±0.52	-56.80±0.54	50.94±0.38	5.86±0.25
$\mathbb{H}_{-1}^6$	-56.18±0.32	-57.10±0.21	51.48±0.47	5.62±0.31
$(\mathbb{P}_{-1}^2)^3$	-55.98±0.62	-56.49±0.62	50.96±0.61	5.52±0.31
$\mathbb{P}_{-1}^6$	-56.74±0.55	-57.61±0.74	52.01±0.71	5.60±0.24
$(\mathcal{RN } \mathbb{P}_{-1}^2)^3$	-54.99±0.12	-55.90±0.13	52.42±0.71	3.48±0.60
$(\mathbb{S}^2)^3$	-56.05±0.21	-56.69±0.36	51.07±0.21	5.61±0.22
$(\text{vMF } \mathbb{S}^2)^3$	-57.56±0.88	-57.80±0.89	52.68±1.62	5.12±0.84
$\mathbb{S}^6$	-56.06±0.51	-56.65±0.64	50.93±0.38	5.72±0.40
$\text{vMF } \mathbb{S}^6$	-58.21±0.92	-59.87±1.51	54.99±1.79	4.88±0.39
$(\mathbb{D}^2)^3$	-56.06±0.36	-56.69±0.54	50.95±0.40	5.74±0.17
$\mathbb{D}^6$	-56.10±0.25	-56.69±0.17	50.90±0.19	5.79±0.03
$(\mathbb{H}^2)^3$	-55.80±0.32	-56.72±0.16	51.14±0.39	5.58±0.28
$\mathbb{H}^6$	-56.03±0.21	-56.82±0.20	50.99±0.16	5.83±0.27
$(\mathbb{P}^2)^3$	-56.29±0.05	-57.11±0.22	51.41±0.19	5.69±0.30
$\mathbb{P}^6$	-56.40±0.31	-57.13±0.25	51.17±0.33	5.96±0.27
$(\mathcal{RN } \mathbb{P}^2)^3$	-56.25±0.56	-57.26±0.45	53.16±1.07	4.11±0.64
$\mathbb{D}^2 \times \mathbb{E}^2 \times \mathbb{P}^2$	-55.87±0.22	-56.35±0.22	50.67±0.57	5.69±0.43
$\mathbb{D}_1^2 \times \mathbb{E}^2 \times \mathbb{P}_{-1}^2$	-56.06±0.41	-56.86±0.65	51.23±0.67	5.64±0.11
$\mathbb{D}^2 \times \mathbb{E}^2 \times (\mathcal{RN } \mathbb{P}^2)$	-56.35±0.82	-57.06±0.78	51.89±0.71	5.17±0.12
$\mathbb{D}_1^2 \times \mathbb{E}^2 \times (\mathcal{RN } \mathbb{P}_{-1}^2)$	-56.17±0.43	-56.75±0.56	51.80±0.80	4.95±0.32
$\mathbb{E}^2 \times \mathbb{H}^2 \times \mathbb{S}^2$	-55.92±0.42	-56.54±0.45	51.13±0.74	5.41±0.40
$\mathbb{E}^2 \times \mathbb{H}_{-1}^2 \times \mathbb{S}_1^2$	-56.04±0.57	-56.71±0.77	51.09±0.86	5.62±0.12
$\mathbb{E}^2 \times \mathbb{H}^2 \times (\text{vMF } \mathbb{S}^2)$	-55.82±0.43	-56.32±0.47	51.10±0.67	5.21±0.20
$\mathbb{E}^2 \times \mathbb{H}_{-1}^2 \times (\text{vMF } \mathbb{S}_1^2)$	-55.77±0.51	-56.34±0.65	51.33±0.57	5.01±0.17
$(\mathbb{U}^2)^3$	-55.56±0.15	-56.05±0.32	50.68±0.23	5.37±0.10
$\mathbb{U}^6$	-55.84±0.38	-56.46±0.41	50.66±0.38	5.81±0.18

Table D.1: Summary of results (mean and standard-deviation) with latent space dimension of 6, spherical covariance parametrization, on the BDP dataset.

## D.2. Spherical covariance matrix

Model	LL	ELBO	BCE	KL
$(\mathbb{S}_1^2)^3$	-96.77±0.26	-101.66±0.32	87.04±0.49	14.62±0.18
(vMF $\mathbb{S}_1^2$ ) <sup>3</sup>	-97.72±0.22	-102.98±0.15	87.77±0.18	15.21±0.07
$\mathbb{S}_1^6$	-96.71±0.17	-101.55±0.30	86.90±0.30	14.65±0.10
vMF $\mathbb{S}_1^6$	-97.03±0.14	-102.12±0.26	87.42±0.28	14.69±0.03
$(\mathbb{D}_1^2)^3$	-97.84±0.10	-102.75±0.22	88.43±0.12	14.33±0.13
$\mathbb{D}_1^6$	-98.21±0.23	-103.02±0.14	88.44±0.05	14.58±0.11
$(\mathbb{E}^2)^3$	-97.04±0.14	-101.44±0.18	86.77±0.22	14.67±0.22
$\mathbb{E}^6$	-97.16±0.15	-101.67±0.14	87.17±0.26	14.50±0.20
$(\mathbb{H}_{-1}^2)^3$	-97.31±0.09	-102.20±0.29	87.81±0.23	14.39±0.13
$\mathbb{H}_{-1}^6$	-97.10±0.44	-101.89±0.33	87.32±0.22	14.56±0.20
$(\mathbb{P}_{-1}^2)^3$	-97.56±0.04	-102.33±0.22	87.93±0.32	14.40±0.10
$(\mathcal{RN} \mathbb{P}_{-1}^2)^3$	-92.54±0.19	-97.19±0.21	88.42±0.20	8.76±0.04
$\mathbb{P}_{-1}^6$	-97.80±0.05	-102.60±0.04	88.14±0.08	14.46±0.07
$(\mathbb{S}^2)^3$	-96.46±0.12	-101.30±0.17	86.79±0.25	14.51±0.09
(vMF $\mathbb{S}^2$ ) <sup>3</sup>	-97.62±0.30	-102.72±0.37	87.48±0.37	15.24±0.03
$\mathbb{S}^6$	-96.72±0.15	-101.39±0.16	86.69±0.15	14.70±0.13
vMF $\mathbb{S}^6$	-96.72±0.18	-101.55±0.21	86.82±0.23	14.73±0.02
$(\mathbb{D}^2)^3$	-97.68±0.24	-102.51±0.44	88.11±0.34	14.41±0.11
$\mathbb{D}^6$	-97.72±0.15	-102.31±0.16	87.70±0.22	14.61±0.06
$(\mathbb{H}^2)^3$	-97.37±0.13	-102.07±0.24	87.56±0.30	14.51±0.11
$\mathbb{H}^6$	-97.47±0.16	-102.18±0.20	87.64±0.23	14.53±0.07
$(\mathbb{P}^2)^3$	-97.62±0.05	-102.34±0.16	87.92±0.16	14.43±0.06
$(\mathcal{RN} \mathbb{P}^2)^3$	-94.16±0.68	-98.65±0.66	89.27±0.79	9.38±0.15
$\mathbb{P}^6$	-97.71±0.24	-102.55±0.21	88.24±0.23	14.32±0.04
$\mathbb{D}^2 \times \mathbb{E}^2 \times \mathbb{P}^2$	-97.48±0.18	-102.22±0.29	87.85±0.17	14.37±0.13
$\mathbb{D}_1^2 \times \mathbb{E}^2 \times \mathbb{P}_{-1}^2$	-97.58±0.13	-102.23±0.15	87.75±0.15	14.49±0.15
$\mathbb{D}^2 \times \mathbb{E}^2 \times (\mathcal{RN} \mathbb{P}^2)$	-96.43±0.47	-101.31±0.51	88.82±0.50	12.50±0.03
$\mathbb{D}_1^2 \times \mathbb{E}^2 \times (\mathcal{RN} \mathbb{P}_{-1}^2)$	-96.18±0.21	-100.91±0.31	88.58±0.47	12.33±0.19
$\mathbb{E}^2 \times \mathbb{H}^2 \times \mathbb{S}^2$	-96.80±0.20	-101.60±0.33	87.13±0.19	14.47±0.17
$\mathbb{E}^2 \times \mathbb{H}_{-1}^2 \times \mathbb{S}_1^2$	-96.76±0.09	-101.48±0.13	86.99±0.17	14.49±0.05
$\mathbb{E}^2 \times \mathbb{H}^2 \times (\text{vMF } \mathbb{S}^2)$	-96.56±0.27	-101.49±0.28	86.58±0.36	14.91±0.14
$\mathbb{E}^2 \times \mathbb{H}_{-1}^2 \times (\text{vMF } \mathbb{S}_1^2)$	-96.76±0.39	-101.82±0.13	87.08±0.06	14.74±0.13
$(\mathbb{U}^2)^3$	-97.12±0.04	-101.68±0.06	87.13±0.14	14.55±0.16
$\mathbb{U}^6$	-97.26±0.16	-102.05±0.18	87.54±0.21	14.51±0.11

Table D.2: Summary of results (mean and standard-deviation) with latent space dimension of 6, spherical covariance parametrization, on the MNIST dataset.

## D.2. Spherical covariance matrix

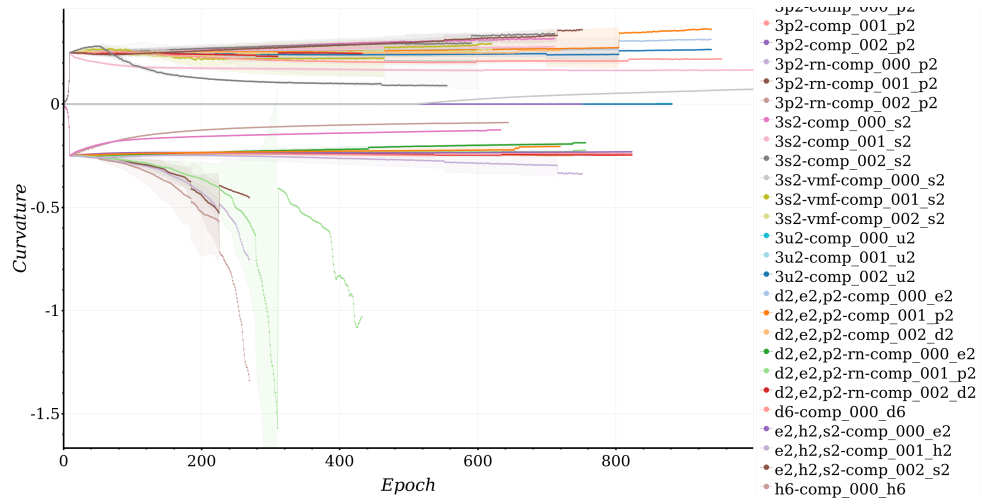


Figure D.1: Learned curvature across epochs (with standard deviation) with latent space dimension of 6, spherical covariance parametrization, on the BDP dataset.

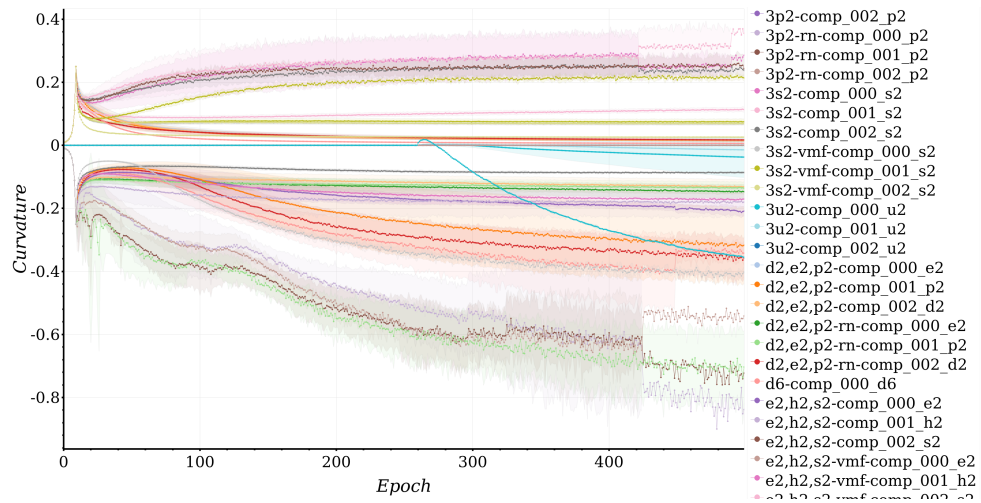
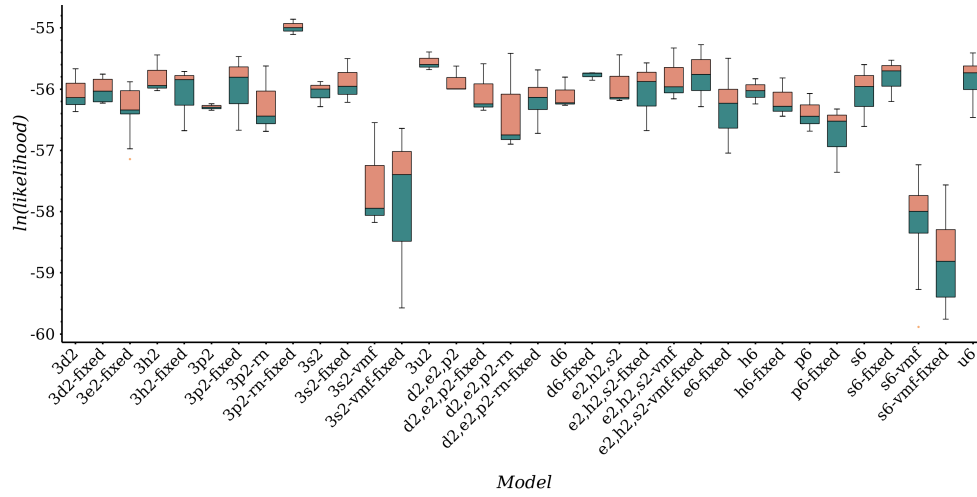
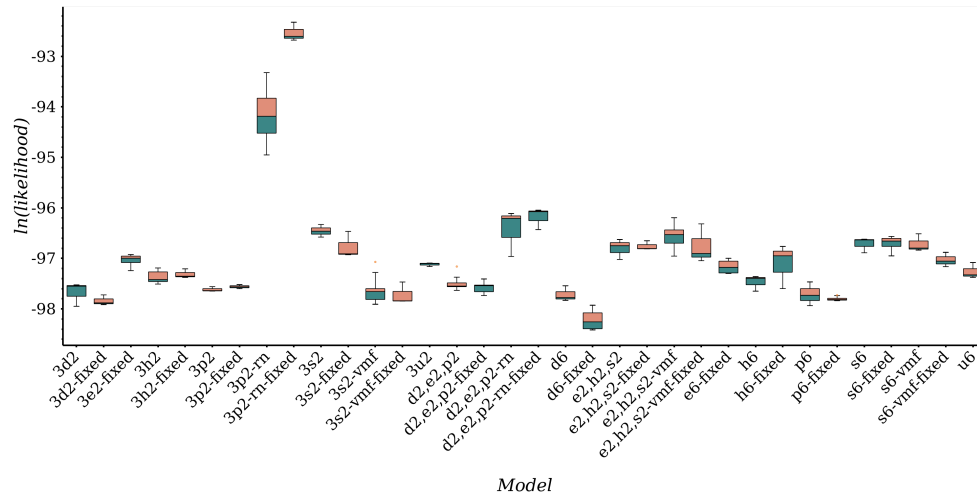


Figure D.2: Learned curvature across epochs (with standard deviation) with latent space dimension of 6, spherical covariance parametrization, on the MNIST dataset.



(a) BDP  $\mathcal{M}$ -VAEs with  $\dim(\mathcal{M}) = 6$ .



(b) MNIST  $\mathcal{M}$ -VAEs with  $\dim(\mathcal{M}) = 6$ .

Figure D.3: Boxplot of evaluation marginal log-likelihoods at the end of training for BDP and MNIST, with spherical covariance per component.

## D.3 Diagonal covariance matrix

### D.3.1 Dynamically binarized MNIST reconstruction

Model	LL	ELBO	BCE	KL
$(\mathbb{S}_1^2)^3$	$-96.57 \pm 0.04$	$-101.34 \pm 0.12$	$86.88 \pm 0.17$	$14.45 \pm 0.10$
$\mathbb{S}_1^6$	$-96.51 \pm 0.09$	$-101.29 \pm 0.18$	$86.71 \pm 0.20$	$14.58 \pm 0.13$
$(\mathbb{D}_1^2)^3$	$-97.81 \pm 0.14$	$-102.58 \pm 0.23$	$88.31 \pm 0.25$	$14.27 \pm 0.02$
$\mathbb{D}_1^6$	$-97.89 \pm 0.10$	$-102.65 \pm 0.10$	$88.39 \pm 0.16$	$14.26 \pm 0.08$
$(\mathbb{E}^2)^3$	$-96.94 \pm 0.34$	$-101.34 \pm 0.41$	$86.89 \pm 0.36$	$14.44 \pm 0.11$
$\mathbb{E}^6$	$-96.88 \pm 0.16$	$-101.36 \pm 0.08$	$86.90 \pm 0.14$	$14.46 \pm 0.07$
$(\mathbb{H}_{-1}^2)^3$	$-97.19 \pm 0.32$	$-102.06 \pm 0.28$	$87.63 \pm 0.37$	$14.42 \pm 0.10$
$\mathbb{H}_{-1}^6$	$-97.38 \pm 0.73$	$-102.22 \pm 0.95$	$87.75 \pm 0.59$	$14.47 \pm 0.37$
$(\mathbb{P}_{-1}^2)^3$	$-97.57 \pm 0.12$	$-102.22 \pm 0.18$	$87.83 \pm 0.30$	$14.39 \pm 0.13$
$\mathbb{P}_{-1}^6$	$-97.33 \pm 0.15$	$-102.02 \pm 0.35$	$87.71 \pm 0.36$	$14.31 \pm 0.04$
$(\mathbb{S}^2)^3$	$-96.78 \pm 0.35$	$-101.43 \pm 0.24$	$86.93 \pm 0.28$	$14.50 \pm 0.05$
$\mathbb{S}^6$	$-96.44 \pm 0.20$	$-101.18 \pm 0.36$	$86.74 \pm 0.38$	$14.44 \pm 0.05$
$(\mathbb{D}^2)^3$	$-97.61 \pm 0.19$	$-102.37 \pm 0.26$	$87.96 \pm 0.21$	$14.41 \pm 0.06$
$\mathbb{D}^6$	$-97.53 \pm 0.22$	$-102.31 \pm 0.38$	$87.97 \pm 0.37$	$14.34 \pm 0.08$
$(\mathbb{H}^2)^3$	$-96.86 \pm 0.31$	$-101.61 \pm 0.30$	$87.13 \pm 0.30$	$14.48 \pm 0.08$
$\mathbb{H}^6$	$-96.90 \pm 0.26$	$-101.48 \pm 0.35$	$87.18 \pm 0.48$	$14.30 \pm 0.15$
$(\mathbb{P}^2)^3$	$-97.52 \pm 0.02$	$-102.30 \pm 0.07$	$88.11 \pm 0.07$	$14.19 \pm 0.12$
$\mathbb{P}^6$	$-97.26 \pm 0.16$	$-102.00 \pm 0.17$	$87.58 \pm 0.16$	$14.42 \pm 0.08$
$\mathbb{D}^2 \times \mathbb{E}^2 \times \mathbb{P}^2$	$-97.37 \pm 0.14$	$-102.12 \pm 0.19$	$87.78 \pm 0.23$	$14.34 \pm 0.12$
$\mathbb{D}_1^2 \times \mathbb{E}^2 \times \mathbb{P}_{-1}^2$	$-97.29 \pm 0.16$	$-101.86 \pm 0.16$	$87.54 \pm 0.17$	$14.32 \pm 0.04$
$\mathbb{E}^2 \times \mathbb{H}^2 \times \mathbb{S}^2$	$-96.71 \pm 0.19$	$-101.34 \pm 0.16$	$86.91 \pm 0.17$	$14.43 \pm 0.06$
$\mathbb{E}^2 \times \mathbb{H}_{-1}^2 \times \mathbb{S}_1^2$	$-96.66 \pm 0.27$	$-101.46 \pm 0.44$	$87.02 \pm 0.38$	$14.44 \pm 0.08$
$(\mathbb{U}^2)^3$	$-97.06 \pm 0.13$	$-101.66 \pm 0.19$	$87.22 \pm 0.12$	$14.44 \pm 0.07$
$\mathbb{U}^6$	$-96.90 \pm 0.10$	$-101.68 \pm 0.07$	$87.27 \pm 0.11$	$14.42 \pm 0.12$

Table D.3: Summary of results (mean and standard-deviation) with latent space dimension of 6, diagonal covariance parametrization, on the MNIST dataset.

We also present an illustrative latent space visualization of a randomly selected run of the models  $\mathbb{E}^2 \times \mathbb{H}^2 \times \mathbb{S}^2$ ,  $\mathbb{E}^6$ ,  $\mathbb{H}^6$ , and  $\mathbb{S}^6$  with spherical covariance (Figure D.9).  $\mathbb{E}^2$  is visualized directly,  $\mathbb{S}^2$  is visualized using a Lambert azimuthal equal-area projection (Snyder, 1987, Chapter 24),  $\mathbb{H}_2$  is transformed to the Poincaré ball model using Equation 2.3. All other latent space sizes were first projected using the respective transformation (to Poincaré ball, Lambert projection) if applicable, and then projected to  $\mathbb{R}^2$  using Principal Component



## D.3. Diagonal covariance matrix

Model	LL	ELBO	BCE	KL
$(\mathbb{S}_1^2)^6$	-79.92±0.21	-84.88±0.14	62.83±0.21	22.06±0.07
$\mathbb{S}_1^{12}$	-80.72±0.34	-85.73±0.36	63.86±0.32	21.87±0.04
$(\mathbb{D}_1^2)^6$	-80.53±0.10	-85.59±0.08	63.62±0.12	21.97±0.16
$\mathbb{D}_1^{12}$	-80.81±0.12	-86.40±0.17	64.42±0.19	21.98±0.06
$(\mathbb{E}^2)^6$	-79.51±0.10	-83.91±0.12	61.84±0.06	22.07±0.13
$\mathbb{E}^{12}$	-79.51±0.09	-83.95±0.06	61.66±0.10	22.29±0.04
$(\mathbb{H}_{-1}^2)^6$	-80.54±0.23	-86.05±0.52	63.78±0.26	22.27±0.26
$\mathbb{H}_{-1}^{12}$	-79.37±0.14	-84.76±0.08	62.32±0.05	22.44±0.10
$(\mathbb{P}_{-1}^2)^6$	-80.39±0.07	-85.46±0.15	63.48±0.22	21.98±0.17
$\mathbb{P}_{-1}^{12}$	-80.88±0.20	-85.87±0.45	63.66±0.59	22.21±0.17
$(\mathbb{S}^2)^6$	-79.95±0.14	-84.90±0.25	62.83±0.34	22.07±0.17
$\mathbb{S}^{12}$	-79.99±0.27	-84.78±0.26	62.89±0.29	21.89±0.18
$(\mathbb{D}^2)^6$	-80.40±0.09	-85.38±0.08	63.49±0.12	21.89±0.18
$\mathbb{D}^{12}$	-80.37±0.16	-85.26±0.19	63.24±0.15	22.02±0.13
$(\mathbb{H}^2)^6$	-80.13±0.08	-85.22±0.24	63.32±0.34	21.90±0.10
$\mathbb{H}^{12}$	-79.77±0.10	-84.58±0.15	62.49±0.10	22.09±0.20
$(\mathbb{P}^2)^6$	-80.31±0.08	-85.35±0.10	63.57±0.17	21.79±0.07
$\mathbb{P}^{12}$	-80.66±0.09	-85.55±0.03	63.55±0.17	22.00±0.14
$(\mathbb{D}^2)^2 \times (\mathbb{E}^2)^2 \times (\mathbb{P}^2)^2$	-80.30±0.31	-85.22±0.40	63.52±0.48	21.70±0.11
$(\mathbb{D}_1^2)^2 \times (\mathbb{E}^2)^2 \times (\mathbb{P}_{-1}^2)^2$	-80.14±0.11	-85.00±0.08	62.99±0.16	22.01±0.24
$\mathbb{D}^4 \times \mathbb{E}^4 \times \mathbb{P}^4$	-80.17±0.11	-84.95±0.27	62.87±0.39	22.08±0.18
$\mathbb{D}_1^4 \times \mathbb{E}^4 \times \mathbb{P}_{-1}^4$	-80.14±0.20	-84.99±0.26	63.06±0.26	21.92±0.08
$(\mathbb{E}^2)^2 \times (\mathbb{H}^2)^2 \times (\mathbb{S}^2)^2$	-79.59±0.25	-84.43±0.20	62.68±0.20	21.75±0.20
$(\mathbb{E}^2)^2 \times (\mathbb{H}_{-1}^2)^2 \times (\mathbb{S}_1^2)^2$	-79.87±0.45	-84.82±0.61	62.66±0.42	22.17±0.20
$\mathbb{E}^4 \times \mathbb{H}^4 \times \mathbb{S}^4$	-79.69±0.14	-84.45±0.12	62.64±0.28	21.81±0.21
$\mathbb{E}^4 \times \mathbb{H}_{-1}^4 \times \mathbb{S}_1^4$	-79.77±0.09	-84.75±0.03	62.68±0.25	22.07±0.24
$(\mathbb{U}^2)^6$	-79.61±0.06	-84.13±0.04	61.92±0.22	22.21±0.23
$\mathbb{U}^{12}$	-80.01±0.30	-84.86±0.51	62.90±0.63	21.96±0.16

Table D.4: Summary of results (mean and standard-deviation) with latent space dimension of 12, diagonal covariance parametrization, on the MNIST dataset.

Analysis (Abdi and Williams, 2010, PCA) and visualized directly.

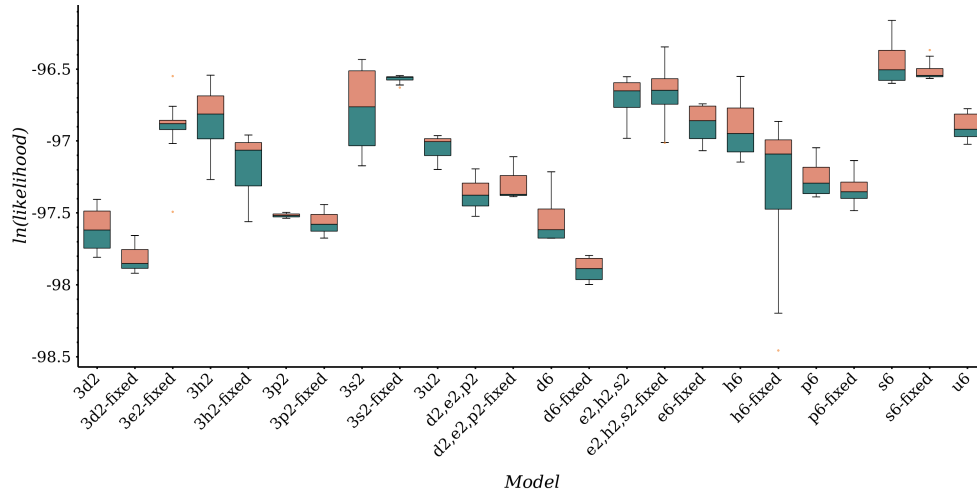
---

D.3. Diagonal covariance matrix

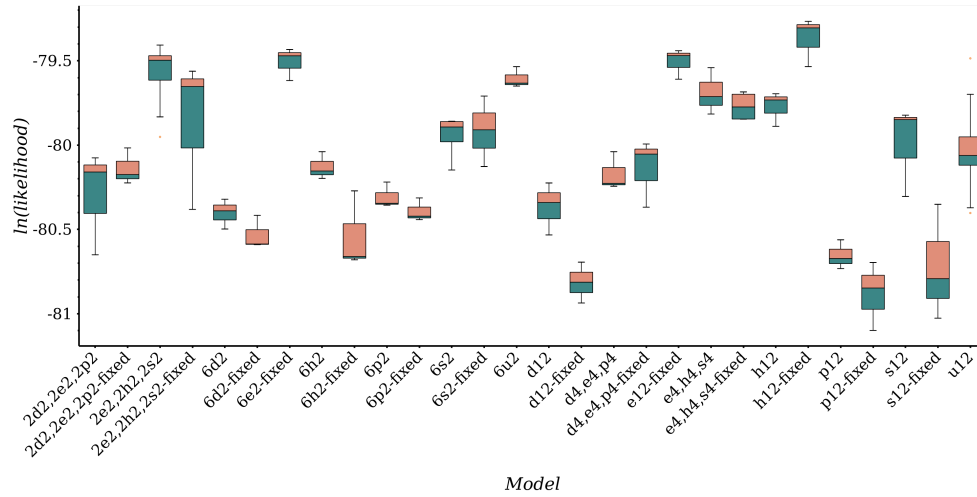
Model	LL	ELBO	BCE	KL
$(\mathbb{S}_1^2)^{36}$	-78.43±0.44	-84.99±0.49	56.88±0.28	28.11±0.56
$(\mathbb{D}_1^2)^{36}$	-76.03±0.17	-83.04±0.25	54.35±0.15	28.69±0.17
$(\mathbb{E}^2)^{36}$	-74.53±0.06	-80.05±0.10	50.91±0.17	29.15±0.07
$\mathbb{E}^{72}$	-74.42±0.06	-80.09±0.12	51.45±0.30	28.63±0.20
$(\mathbb{H}_{-1}^2)^{36}$	-77.92±0.32	-84.76±0.55	56.85±0.60	27.91±0.42
$\mathbb{H}_{-1}^{72}$	-77.30±0.12	-86.98±0.09	58.04±0.29	28.94±0.25
$(\mathbb{P}_{-1}^2)^{36}$	-76.11±0.08	-82.63±0.19	53.89±0.36	28.74±0.30
$\mathbb{P}_{-1}^{72}$	-77.50±0.05	-84.53±0.13	55.80±0.20	28.73±0.18
$\mathbb{S}^{72}$	-75.24±0.01	-81.39±0.14	53.03±0.27	28.36±0.16
$(\mathbb{D}^2)^{36}$	-75.66±0.06	-81.94±0.09	53.32±0.16	28.61±0.11
$\mathbb{D}^{72}$	-77.11±2.21	-83.94±2.81	54.94±2.55	29.00±1.31
$(\mathbb{H}^2)^{36}$	-77.87±0.02	-83.95±0.02	55.71±0.35	28.24±0.36
$\mathbb{H}^{72}$	-75.03±0.11	-81.23±0.14	52.63±0.10	28.61±0.11
$(\mathbb{P}^2)^{36}$	-75.77±0.12	-82.07±0.02	53.65±0.38	28.43±0.39
$\mathbb{P}^{72}$	-75.71±0.08	-81.95±0.09	53.29±0.14	28.67±0.05
$(\mathbb{D}^2)^{12} \times (\mathbb{E}^2)^{12} \times (\mathbb{P}^2)^{12}$	-77.40±0.55	-83.35±0.41	53.90±0.40	29.45±0.12
$(\mathbb{D}_1^2)^{12} \times (\mathbb{E}^2)^{12} \times (\mathbb{P}_{-1}^2)^{12}$	-75.36±0.23	-81.53±0.42	53.02±0.39	28.51±0.45
$\mathbb{D}^{24} \times \mathbb{E}^{24} \times \mathbb{P}^{24}$	-75.11±0.05	-80.99±0.07	52.48±0.19	28.52±0.16
$(\mathbb{E}^2)^{12} \times (\mathbb{H}_{-1}^2)^{12} \times (\mathbb{S}_1^2)^{12}$	-77.53±0.34	-83.95±0.40	55.54±0.43	28.42±0.08
$\mathbb{E}^{24} \times \mathbb{H}^{24} \times \mathbb{S}^{24}$	-75.04±0.16	-81.17±0.18	52.61±0.32	28.55±0.38
$(\mathbb{U}^2)^{36}$	-74.64±0.08	-80.52±0.10	52.04±0.10	28.48±0.07
$\mathbb{U}^{72}$	-75.46±0.09	-81.76±0.09	53.27±0.18	28.49±0.18

Table D.5: Summary of results (mean and standard-deviation) with latent space dimension of 72, diagonal covariance parametrization, on the MNIST dataset.

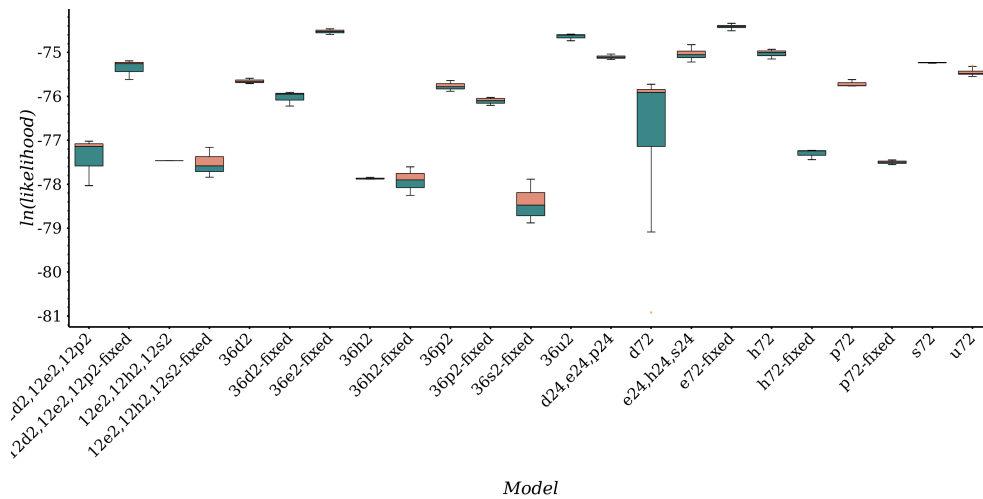
### D.3. Diagonal covariance matrix



(a) MNIST  $\mathcal{M}$ -VAEs with  $\dim(\mathcal{M}) = 6$ .



(b) MNIST  $\mathcal{M}$ -VAEs with  $\dim(\mathcal{M}) = 12$ .



(c) MNIST  $\mathcal{M}$ -VAEs with  $\dim(\mathcal{M}) = 72$ .

Figure D.4: Boxplot of evaluation marginal log-likelihoods at the end of training for MNIST, with diagonal covariance per component.

### D.3. Diagonal covariance matrix

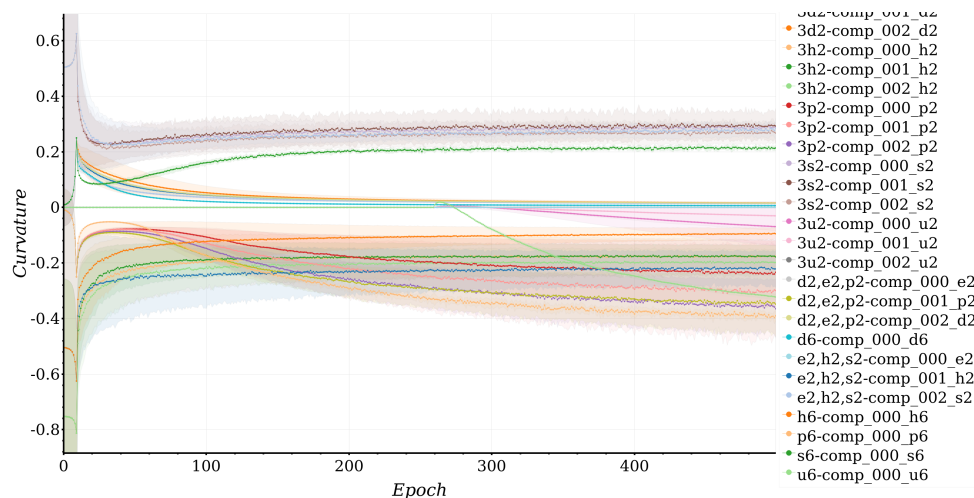


Figure D.5: Learned curvature across epochs (with standard deviation) with latent space dimension of 6, diagonal covariance parametrization, on the MNIST dataset.

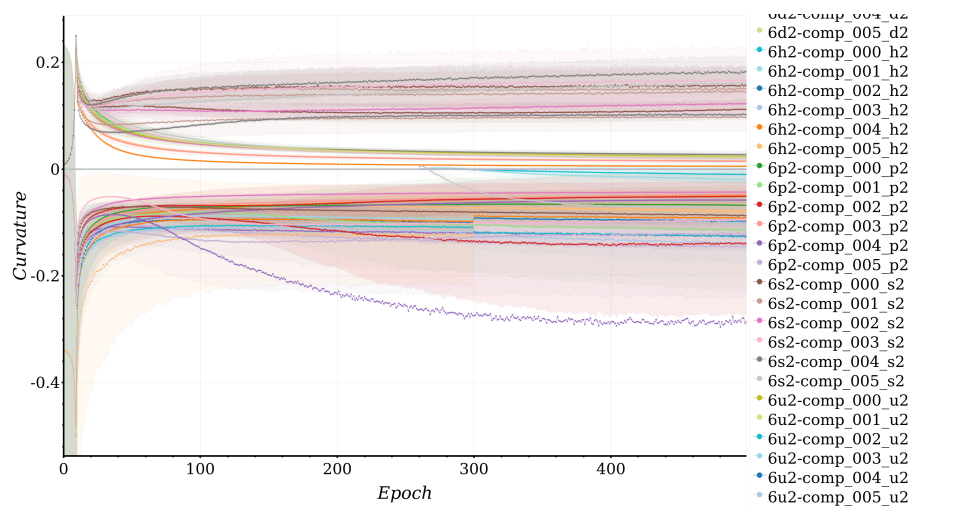


Figure D.6: Learned curvature across epochs (with standard deviation) with latent space dimension of 12, diagonal covariance parametrization, on the MNIST dataset.

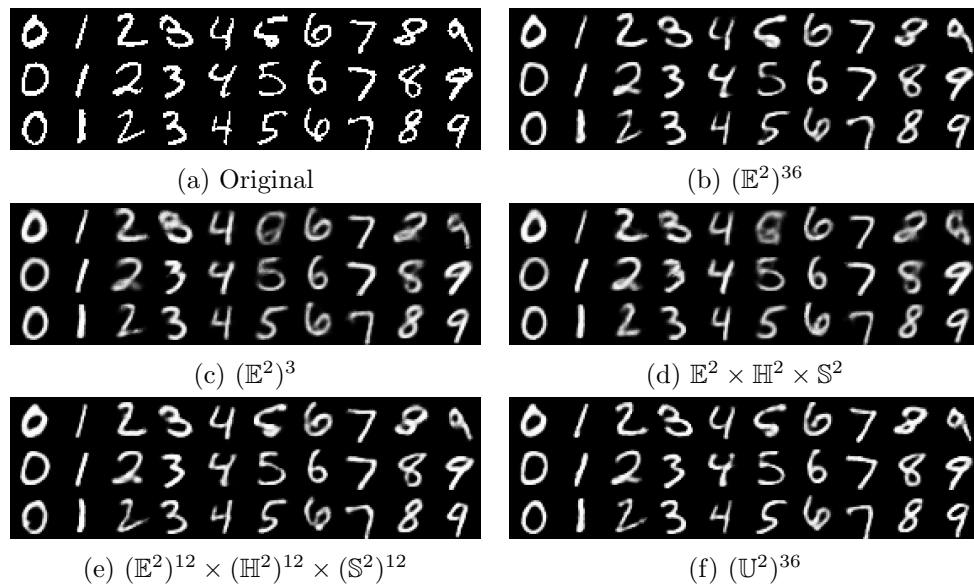


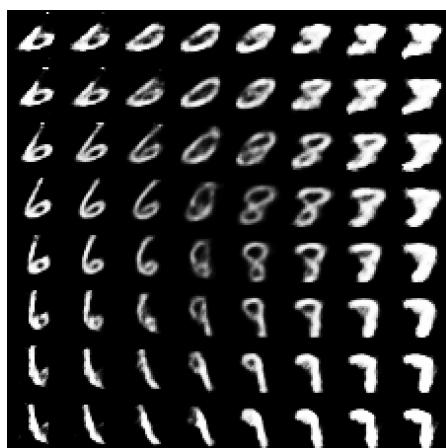
Figure D.7: Qualitative comparison of reconstruction quality of randomly selected runs of a selection of well-performing models on MNIST test set digits.



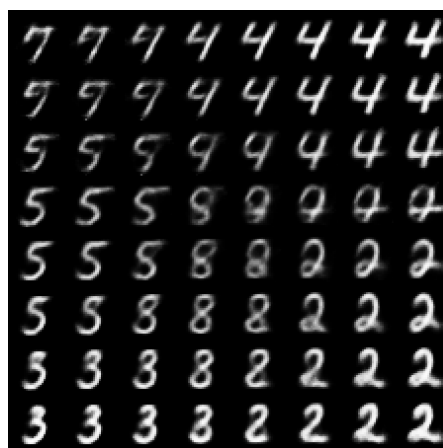
(a) First component of  $(\mathbb{E}^2)^3$ .



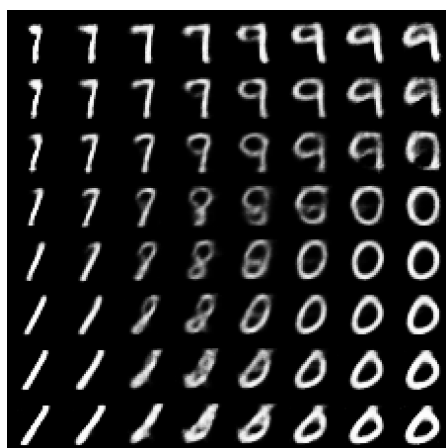
(b) Second component of  $(\mathbb{E}^2)^3$ .



(c) First (negative) component of  $(\mathbb{U}^2)^3$ .



(d) Second (negative) component of  $(\mathbb{U}^2)^3$ .



(e)  $\mathbb{P}^2$  of  $\mathbb{E}^2 \times \mathbb{P}^2 \times \mathbb{D}^2$ .



(f)  $\mathbb{D}^2$  of  $\mathbb{E}^2 \times \mathbb{P}^2 \times \mathbb{D}^2$ .

Figure D.8: Samples from various models of a grid search around  $\mathbf{0}$  of a single component's latent space on MNIST test digits.

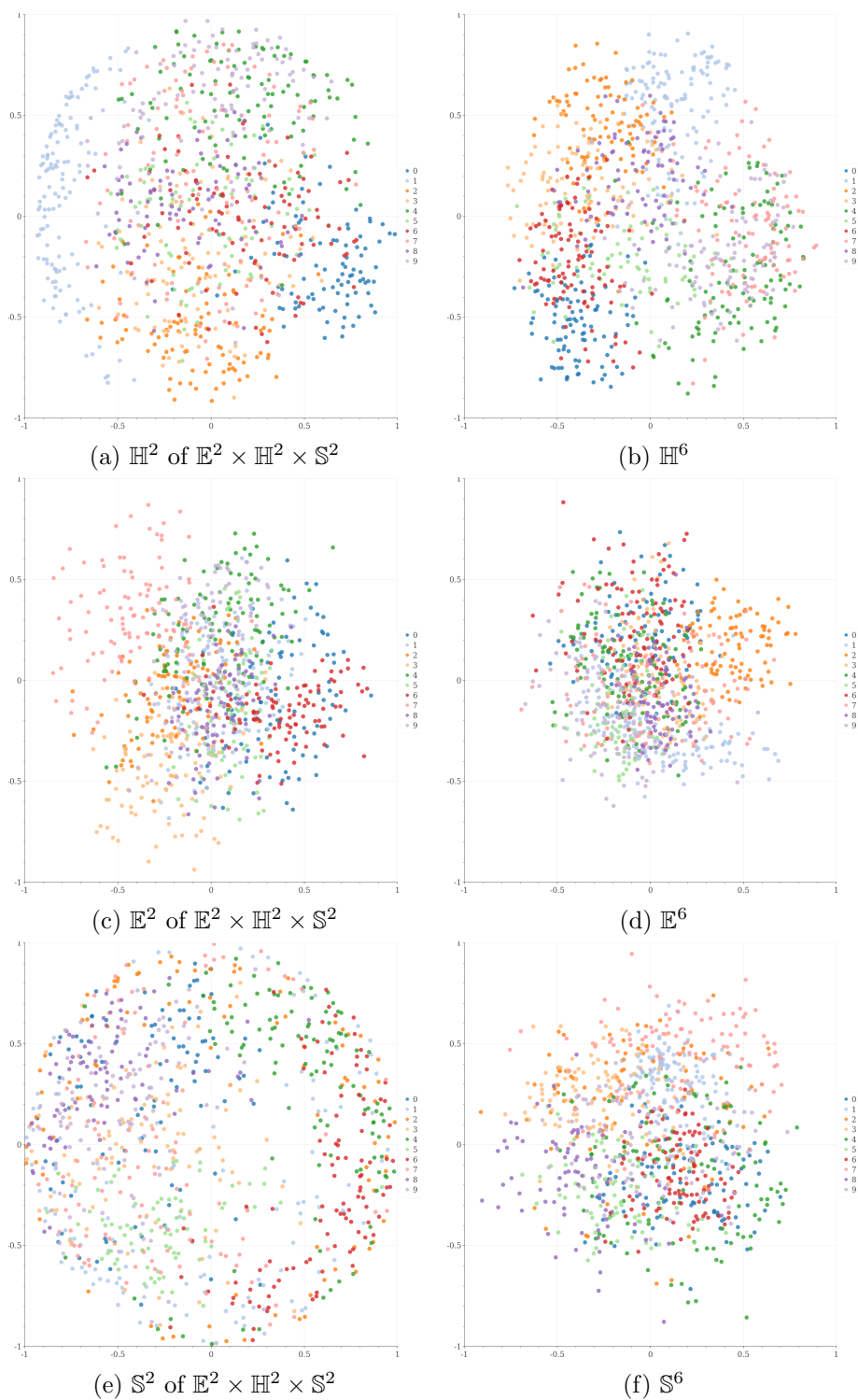
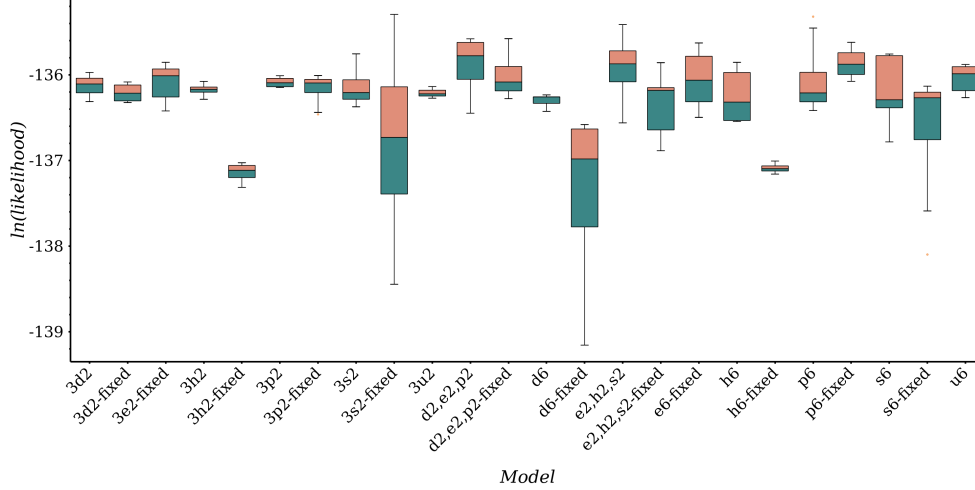
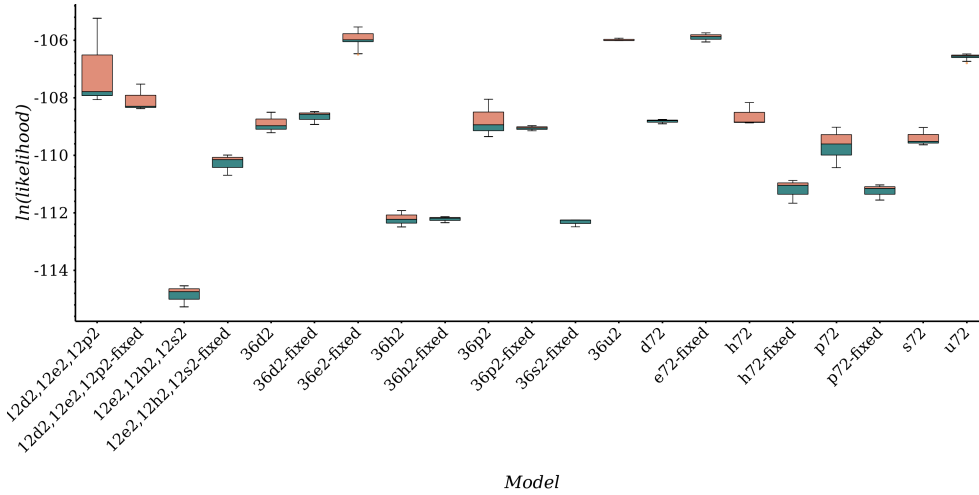


Figure D.9: Illustrative latent space visualization of a randomly selected run of the models  $\mathbb{E}^2 \times \mathbb{H}^2 \times \mathbb{S}^2$ ,  $\mathbb{E}^6$ ,  $\mathbb{H}^6$ , and  $\mathbb{S}^6$  on MNIST.

D.3.2 Dynamically binarized Omniglot reconstruction



(a) Omniglot  $\mathcal{M}$ -VAEs with  $\dim(\mathcal{M}) = 6$ .



(b) Omniglot  $\mathcal{M}$ -VAEs with  $\dim(\mathcal{M}) = 72$ .

Figure D.10: Boxplot of evaluation marginal log-likelihoods at the end of training for Omniglot, with spherical covariance per component.



Model	LL	ELBO	BCE	KL
$(\mathbb{S}_1^2)^3$	$-136.80 \pm 1.31$	$-141.68 \pm 1.52$	$131.73 \pm 5.65$	$9.95 \pm 4.33$
$\mathbb{S}_1^6$	$-136.69 \pm 0.94$	$-141.46 \pm 0.92$	$129.52 \pm 0.74$	$11.94 \pm 0.19$
$(\mathbb{D}_1^2)^3$	$-136.21 \pm 0.12$	$-140.44 \pm 0.17$	$128.93 \pm 0.14$	$11.51 \pm 0.04$
$\mathbb{D}_1^6$	$-137.42 \pm 1.20$	$-141.95 \pm 1.94$	$130.70 \pm 2.18$	$11.25 \pm 0.26$
$(\mathbb{E}^2)^3$	$-136.08 \pm 0.21$	$-140.46 \pm 0.24$	$128.85 \pm 0.34$	$11.62 \pm 0.14$
$\mathbb{E}^6$	$-136.05 \pm 0.29$	$-140.50 \pm 0.35$	$128.95 \pm 0.41$	$11.55 \pm 0.14$
$(\mathbb{H}_{-1}^2)^3$	$-137.14 \pm 0.13$	$-141.87 \pm 0.16$	$130.18 \pm 0.21$	$11.69 \pm 0.10$
$\mathbb{H}_{-1}^6$	$-137.09 \pm 0.06$	$-142.22 \pm 0.19$	$130.37 \pm 0.21$	$11.85 \pm 0.12$
$(\mathbb{P}_{-1}^2)^3$	$-136.16 \pm 0.20$	$-140.63 \pm 0.32$	$129.29 \pm 0.34$	$11.34 \pm 0.03$
$\mathbb{P}_{-1}^6$	$-135.86 \pm 0.20$	$-140.36 \pm 0.19$	$128.92 \pm 0.23$	$11.44 \pm 0.16$
$(\mathbb{S}^2)^3$	$-136.14 \pm 0.27$	$-140.68 \pm 0.32$	$128.98 \pm 0.27$	$11.70 \pm 0.13$
$\mathbb{S}^6$	$-136.20 \pm 0.44$	$-140.76 \pm 0.45$	$129.10 \pm 0.37$	$11.66 \pm 0.13$
$(\mathbb{D}^2)^3$	$-136.13 \pm 0.17$	$-140.59 \pm 0.15$	$129.10 \pm 0.20$	$11.49 \pm 0.12$
$\mathbb{D}^6$	$-136.30 \pm 0.08$	$-140.74 \pm 0.14$	$129.35 \pm 0.16$	$11.39 \pm 0.05$
$(\mathbb{H}^2)^3$	$-136.17 \pm 0.09$	$-140.65 \pm 0.17$	$129.26 \pm 0.07$	$11.39 \pm 0.16$
$\mathbb{H}^6$	$-136.24 \pm 0.32$	$-140.92 \pm 0.33$	$129.48 \pm 0.27$	$11.45 \pm 0.12$
$(\mathbb{P}^2)^3$	$-136.09 \pm 0.07$	$-140.41 \pm 0.08$	$129.04 \pm 0.05$	$11.37 \pm 0.08$
$\mathbb{P}^6$	$-136.05 \pm 0.44$	$-140.42 \pm 0.47$	$129.04 \pm 0.53$	$11.38 \pm 0.07$
$\mathbb{D}^2 \times \mathbb{E}^2 \times \mathbb{P}^2$	$-135.89 \pm 0.40$	$-140.28 \pm 0.42$	$128.75 \pm 0.40$	$11.53 \pm 0.04$
$\mathbb{D}_1^2 \times \mathbb{E}^2 \times \mathbb{P}_{-1}^2$	$-136.01 \pm 0.31$	$-140.52 \pm 0.35$	$129.02 \pm 0.27$	$11.50 \pm 0.11$
$\mathbb{E}^2 \times \mathbb{H}^2 \times \mathbb{S}^2$	$-135.93 \pm 0.48$	$-140.51 \pm 0.53$	$128.85 \pm 0.48$	$11.66 \pm 0.14$
$\mathbb{E}^2 \times \mathbb{H}_{-1}^2 \times \mathbb{S}_1^2$	$-136.34 \pm 0.41$	$-141.02 \pm 0.46$	$129.24 \pm 0.47$	$11.78 \pm 0.10$
$(\mathbb{U}^2)^3$	$-136.21 \pm 0.07$	$-140.65 \pm 0.30$	$129.14 \pm 0.34$	$11.52 \pm 0.15$
$\mathbb{U}^6$	$-136.04 \pm 0.17$	$-140.43 \pm 0.14$	$129.07 \pm 0.27$	$11.36 \pm 0.13$

Table D.6: Summary of results (mean and standard-deviation) with latent space dimension of 6, diagonal covariance parametrization, on the Omniglot dataset.

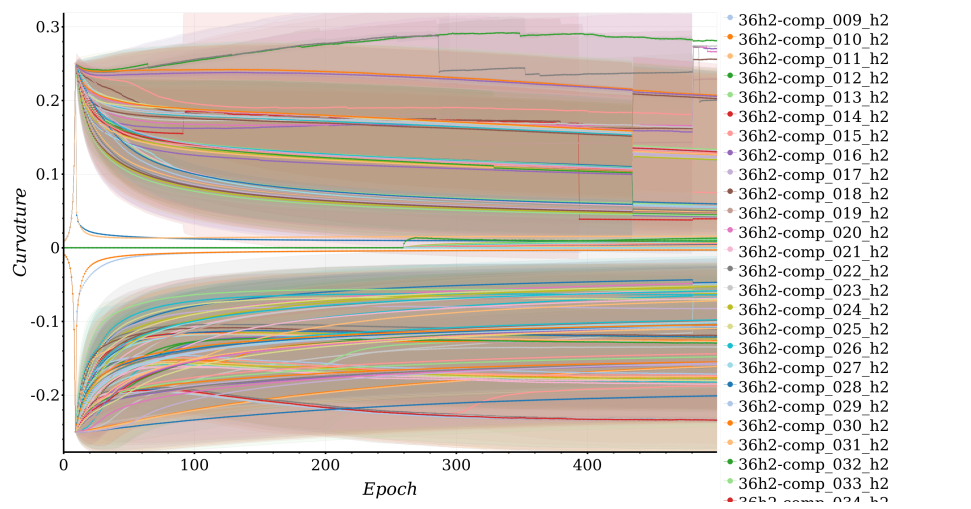


Figure D.11: Learned curvature across epochs (with standard deviation) with latent space dimension of 72, diagonal covariance parametrization, on the Omniglot dataset.

D.3. Diagonal covariance matrix

Model	LL	ELBO	BCE	KL
$(\mathbb{S}_1^2)^{36}$	-112.33±0.14	-118.94±0.14	91.04±0.37	27.90±0.23
$(\mathbb{D}_1^2)^{36}$	-108.66±0.24	-116.06±0.18	85.95±0.16	30.11±0.04
$(\mathbb{E}^2)^{36}$	-105.96±0.33	-112.41±0.35	79.80±0.72	32.61±0.41
$\mathbb{E}^{72}$	-105.89±0.16	-112.40±0.17	79.52±0.19	32.89±0.20
$(\mathbb{H}_{-1}^2)^{36}$	-112.22±0.11	-119.06±0.15	91.30±0.47	27.76±0.35
$\mathbb{H}_{-1}^{72}$	-111.19±0.42	-120.49±0.35	91.11±0.73	29.38±0.40
$(\mathbb{P}_{-1}^2)^{36}$	-109.05±0.09	-115.99±0.10	85.81±0.42	30.18±0.34
$\mathbb{P}_{-1}^{72}$	-111.24±0.28	-118.36±0.24	89.53±0.38	28.84±0.18
$\mathbb{S}^{72}$	-109.39±0.32	-116.42±0.32	87.22±0.58	29.20±0.28
$(\mathbb{D}^2)^{36}$	-108.89±0.36	-115.65±0.45	85.29±0.74	30.37±0.30
$\mathbb{D}^{72}$	-108.81±0.08	-115.71±0.09	85.68±0.10	30.03±0.09
$(\mathbb{H}^2)^{36}$	-112.21±0.28	-118.74±0.30	91.03±0.76	27.71±0.47
$\mathbb{H}^{72}$	-108.62±0.40	-115.54±0.30	85.18±0.62	30.37±0.34
$(\mathbb{P}^2)^{36}$	-108.78±0.66	-115.54±0.70	85.16±1.38	30.38±0.69
$\mathbb{P}^{72}$	-109.66±0.61	-116.50±0.68	87.09±1.43	29.42±0.75
$(\mathbb{D}^2)^{12} \times (\mathbb{E}^2)^{12} \times (\mathbb{P}^2)^{12}$	-107.02±1.56	-115.62±1.76	88.52±8.24	27.10±6.48
$(\mathbb{D}_1^2)^{12} \times (\mathbb{E}^2)^{12} \times (\mathbb{P}_{-1}^2)^{12}$	-108.06±0.47	-114.92±0.39	83.95±0.58	30.97±0.22
$(\mathbb{E}^2)^{12} \times (\mathbb{H}^2)^{12} \times (\mathbb{S}^2)^{12}$	-114.85±0.38	-120.98±0.15	95.12±0.49	25.86±0.40
$(\mathbb{E}^2)^{12} \times (\mathbb{H}_{-1}^2)^{12} \times (\mathbb{S}_1^2)^{12}$	-110.28±0.37	-116.90±0.42	87.71±0.82	29.19±0.41
$(\mathbb{U}^2)^{36}$	-105.98±0.05	-112.70±0.19	79.85±0.80	32.85±0.61
$\mathbb{U}^{72}$	-106.58±0.12	-113.68±0.11	81.53±0.34	32.15±0.36

Table D.7: Summary of results (mean and standard-deviation) with latent space dimension of 72, diagonal covariance parametrization, on the Omniglot dataset.



(a) Original

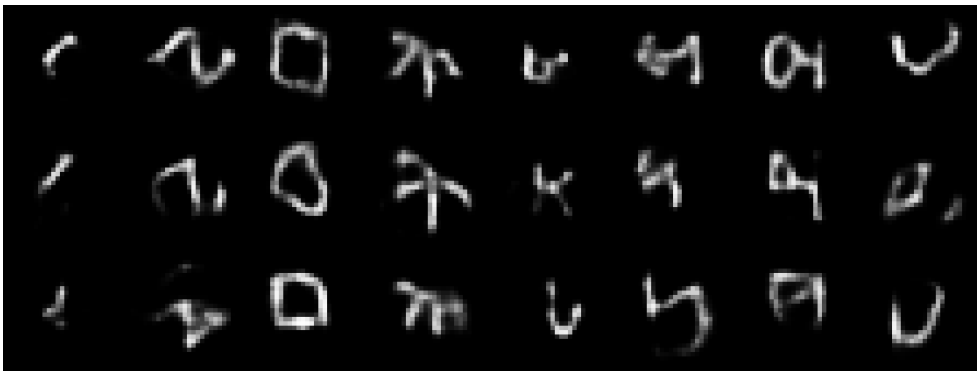
(b)  $(\mathbb{E}^2)^{36}$ (c)  $(\mathbb{U}^2)^{36}$ 

Figure D.12: Qualitative comparison of reconstruction quality of randomly selected runs of a selection of well-performing models on Omniglot test set characters.

## D.3.3 CIFAR reconstruction

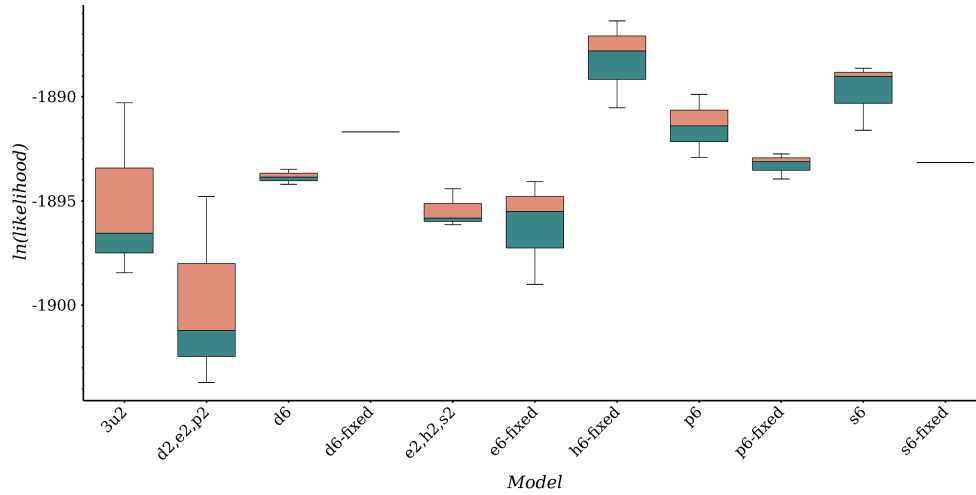
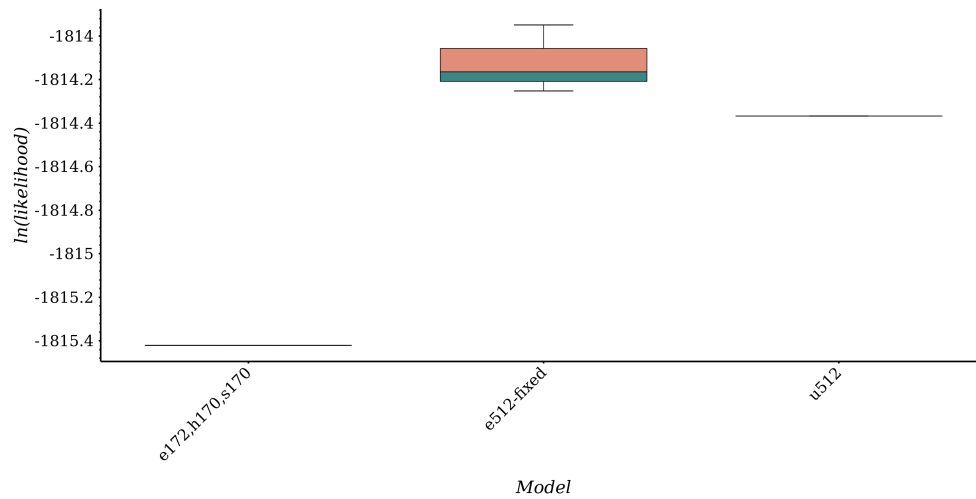
(a) CIFAR  $\mathcal{M}$ -VAEs with  $\dim(\mathcal{M}) = 6$ .(b) CIFAR  $\mathcal{M}$ -VAEs with  $\dim(\mathcal{M}) = 512$ .

Figure D.13: Boxplot of evaluation marginal log-likelihoods at the end of training for Omniglot, with spherical covariance per component.

Model	LL	ELBO	BCE	KL
$S_1^6$	$-1893.16 \pm \text{nan}$	$-1901.32 \pm \text{nan}$	$1885.97 \pm \text{nan}$	$15.35 \pm \text{nan}$
$D_1^6$	$-1891.69 \pm \text{nan}$	$-1897.77 \pm \text{nan}$	$1884.12 \pm \text{nan}$	$13.64 \pm \text{nan}$
$E^6$	$-1896.19 \pm 2.54$	$-1905.75 \pm 3.19$	$1889.97 \pm 2.88$	$15.78 \pm 0.32$
$H_{-1}^6$	$-1888.23 \pm 2.12$	$-1896.56 \pm 2.93$	$1882.05 \pm 2.65$	$14.51 \pm 0.34$
$P_{-1}^6$	$-1893.27 \pm 0.61$	$-1902.67 \pm 0.74$	$1887.44 \pm 0.83$	$15.23 \pm 0.16$
$D^6$	$-1893.85 \pm 0.36$	$-1902.67 \pm 0.69$	$1887.37 \pm 0.74$	$15.30 \pm 0.08$
$S^6$	$-1889.76 \pm 1.62$	$-1897.31 \pm 1.71$	$1882.55 \pm 1.48$	$14.76 \pm 0.24$
$P^6$	$-1891.40 \pm 2.14$	$-1899.68 \pm 2.74$	$1884.58 \pm 2.56$	$15.10 \pm 0.18$
$D^2 \times E^2 \times P^2$	$-1899.90 \pm 4.60$	$-1904.63 \pm 1.46$	$1889.13 \pm 1.38$	$15.50 \pm 0.08$
$E^2 \times H^2 \times S^2$	$-1895.46 \pm 0.92$	$-1897.57 \pm 0.94$	$1882.84 \pm 0.70$	$14.73 \pm 0.24$
$(U^2)^3$	$-1895.09 \pm 4.27$	$-1904.46 \pm 5.21$	$1888.89 \pm 4.71$	$15.57 \pm 0.51$

Table D.8: Summary of results (mean and standard-deviation) with latent space dimension of 6, diagonal covariance parametrization, on the CIFAR dataset.

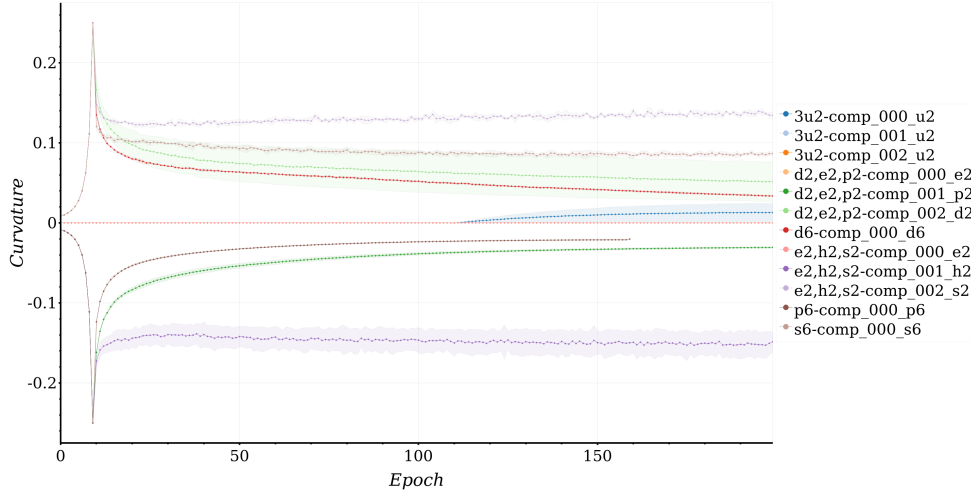


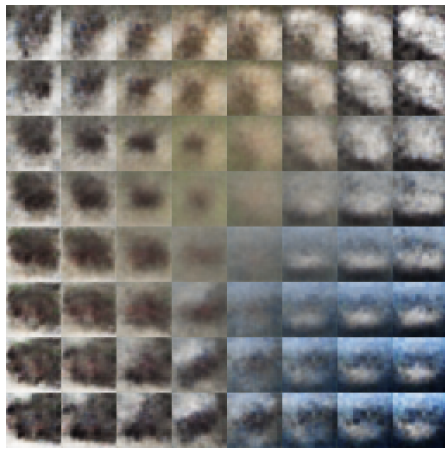
Figure D.14: Learned curvature across epochs (with standard deviation) with latent space dimension of 6, diagonal covariance parametrization, on the CIFAR dataset.

Model	LL	ELBO	BCE	KL
$\mathbb{E}^{512}$	$-1814.12 \pm 0.16$	$-1819.06 \pm 0.20$	$1774.48 \pm 0.43$	$44.57 \pm 0.40$
$\mathbb{E}^{172} \times \mathbb{H}^{170} \times \mathbb{S}^{170}$	$-1815.42 \pm nan$	$-1820.13 \pm nan$	$1776.19 \pm nan$	$43.94 \pm nan$
$\mathbb{U}^{512}$	$-1814.37 \pm nan$	$-1819.42 \pm nan$	$1775.29 \pm nan$	$44.13 \pm nan$

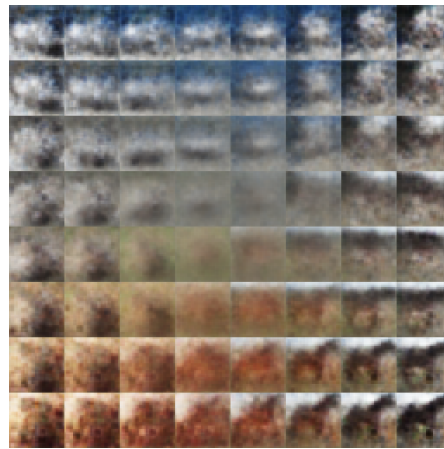
Table D.9: Summary of results (mean and standard-deviation) with latent space dimension of 512, diagonal covariance parametrization, on the CIFAR dataset.



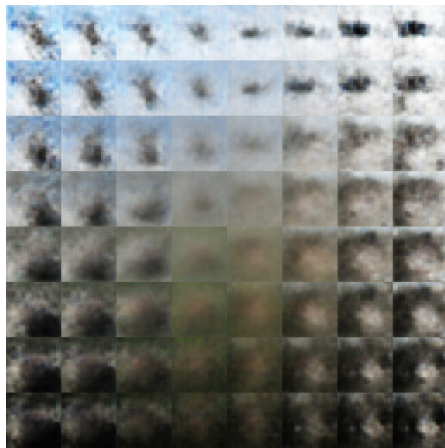
Figure D.15: Qualitative comparison of reconstruction quality of randomly selected runs of a selection of well-performing models on CIFAR test set images.



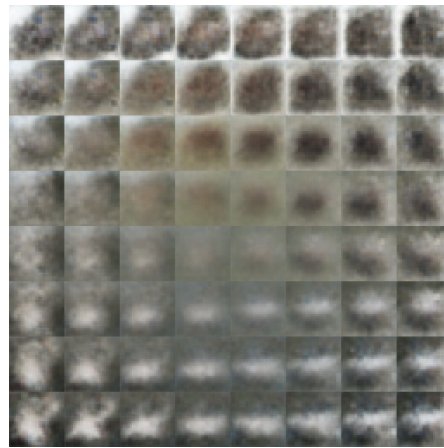
(a) 2<sup>nd</sup> pair of latent dim. of  $\mathbb{E}^6$ .



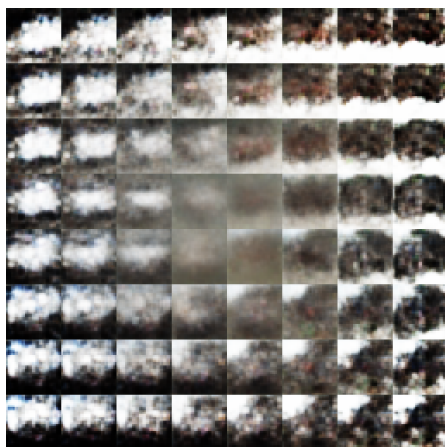
(b) 3<sup>rd</sup> pair of latent dim. of  $\mathbb{E}^6$ .



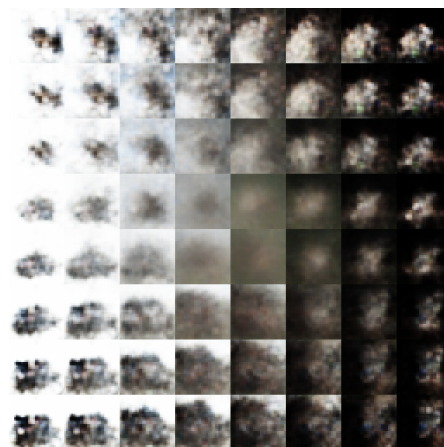
(c) 1<sup>st</sup> (positive) component of  $(\mathbb{U}^2)^3$ .



(d) 2<sup>nd</sup> (positive) component of  $(\mathbb{U}^2)^3$ .



(e)  $\mathbb{P}^2$  of  $\mathbb{E}^2 \times \mathbb{P}^2 \times \mathbb{D}^2$ .



(f)  $\mathbb{D}^2$  of  $\mathbb{E}^2 \times \mathbb{P}^2 \times \mathbb{D}^2$ .

Figure D.16: Samples from various models of a grid search around  $\mathbf{0}$  of a single component's latent space on cifar test digits.





Eidgenössische Technische Hochschule Zürich  
Swiss Federal Institute of Technology Zurich

## Declaration of originality

The signed declaration of originality is a component of every semester paper, Bachelor's thesis, Master's thesis and any other degree paper undertaken during the course of studies, including the respective electronic versions.

Lecturers may also require a declaration of originality for other written papers compiled for their courses.

I hereby confirm that I am the sole author of the written work here enclosed and that I have compiled it in my own words. Parts excepted are corrections of form and content by the supervisor.

**Title of work** (in block letters):

MIXED-CURVATURE VARIATIONAL AUTOENCODERS

**Authored by** (in block letters):

*For papers written by groups the names of all authors are required.*

**Name(s):**

SKOPEK

**First name(s):**

ONDREJ

With my signature I confirm that

- I have committed none of the forms of plagiarism described in the 'Citation etiquette' information sheet.
- I have documented all methods, data and processes truthfully.
- I have not manipulated any data.
- I have mentioned all persons who were significant facilitators of the work.

I am aware that the work may be screened electronically for plagiarism.

**Place, date**

Zürich, 4.9.2019

**Signature(s)**

*For papers written by groups the names of all authors are required. Their signatures collectively guarantee the entire content of the written paper.*