Latent variable separation with variational autoencoders

Perez Rey, Luis

Award date:
2018

Link to publication

Technische Universiteit
Eindhoven
University of Technology

Department of Mathematics and Computer Science
Center for Analysis, Scientific Computing and Applications

Latent Variable Separation with Variational Autoencoders

Master's Thesis Overview

Luis Armando Pérez Rey

Supervisors:
Jim Portegies
Vlado Menkovski

Eindhoven, July 2018

Abstract

Variational autoencoders are considered an unsupervised learning method capable of estimating a suitable generative model that emulates the process that produced a dataset with respect to a set of unobserved variables denominated latent variables. The use of variational autoencoders for retrievieng the underlying structure of a dataset in terms of the latent variables is of interest for a wide range of applications. In this thesis we proposed the use of two variational autoencoders: a baseline based on the work of D.P Kingma and M. Welling [16] and I. Higgins et al. [14] and a diffusion variational autoencoder to retrieve the underlying structure of datasets with an underlying circular/toroidal structure. The proposed diffusion variational autoencoder posits a circular/toroidal latent structure for retrieving the underlying structure from a dataset. The results obtained show that the proposed baseline and diffusion variational autoencoders are capable of retrieving the underlying circular/toroidal structure for the example datasets.

Two variational autoencoders fed with the same dataset produce their own generative model to explain the same data. We have proposed a definition for the concept of Δ-reducibility between the trained generative models of variational autoencoders. The concept is based on functions called Δ-reductors that map the elements from the latent spaces of the variational autoencoders with the purpose of reducing one generative model in terms of the other up to a certain tolerance level $\Delta \in \mathbb{R}^+$. For the generative models trained with the baseline and diffusion variational autoencoders we have proposed an algorithm for constructing simple reduction maps between the recovered latent spaces to reduce the corresponding generative models. The reduction maps obtained are limited to generative models with a known underlying latent structure (circular/toroidal). The results of this thesis provides a basis for research oriented to the connection of generative models with respect to their underlying latent variables.

Acknowledgements

I would like to express my deep gratitude to Dr. Jim Portegies for his guidance and patience in our weekly meetings, for his thorough revisions to my work, for teaching me how to structure my entangled ideas into clear explanations and for giving me his support in every matter.

I would also like to thank Dr. Vlado Menkovski for his advice, support and supervision during my biggest projects during the master and for introducing me into the world of deep learning.

I would like to express my very great appreciation to Dr Rui Pires da Silva Castro for his valuable advice and assistance while making my career choices and his valuable comments to this thesis.

A very special thanks to my grandmother for all her loving care, her love will remain with me in every step that I take. I want to thank my girlfriend for supporting me in every challenge that I have faced. I also want to thank my friends for their constant presence in my life. Finally, I wish to thank my mother, my father, my sister and my uncle for their strong encouragement and support not only in my studies but in every matter of my life.

Contents

Chapter 1

Introduction

During the course of their early years, humans learn about the world that surrounds them from high dimensional sensory input received from the environment. The diverse range of stimuli in the form of images, sound, acceleration, temperature, texture, etc. is combined by the brain to produce valuable representations of the world. These learned representations are composed of features organized to enable the performance of complex tasks [21].

One interesting example of such useful representations learned by the brain is the notion of orientation and location obtained by the processing of external stimuli gathered from navigating through space. A question that arises is whether this kind of notion has been "programmed" into the brain by years of evolution or if it is something that is learned with the experience of early years.

Consider for example a dataset of images from an object that is rotating along a fixed axis, see Figure 1.1. Each observed image consists of an array of pixel values and we would like to find a suitable efficient representation to describe it. Despite the angle being not explicitly shown in each picture, for us as humans, a natural representation of these images would involve a certain notion of angle-pose associated to each frame.

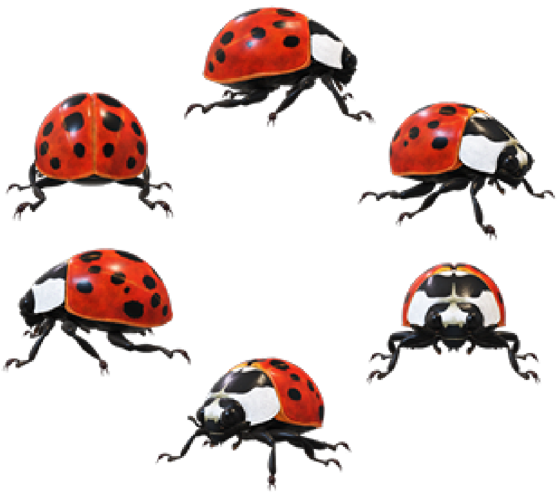

Figure 1.1: Object seen from different angles, a natural concept that can be learned from the images is the notion of orientation.

We will restrict the data that we study to cases in which there is an underlying known structure with a simple geometry (circular and toroidal). Variational autoencoders, developed by D.P. Kingma and M. Welling [18], are based on the mathematical formalism of Bayesian inference. They use neural networks to create generative models that can explain the observed data in terms of unobserved quantities which are called latent variables. We test with different variational autoencoders whether the recovered latent variables of a dataset are suitable representations that can capture the underlying structure of data.

Imagine a real setting in which two persons experience the same phenomenon. Their internal representations created by their brain of the same event might differ. It is interesting to find out whether these internal portrayals are equivalent or at least can be explained in terms of one another. In our framework different variational autoencoders produce their own latent representations of the input data. We try to answer whether different latent representations that explain a same dataset can be reduced in terms of one another. By constructing simple functions between the different latent representations we try, up to a certain tolerance level, to reduce one generative model in terms of another.

1.1 Previous work

Under the term *dimensionality reduction*, different techniques for producing lower dimensional representations of complex data are grouped. These techniques are designed with the purpose of preserving the original characteristics of the input data by assuming that high-dimensional data lies close to a lower-dimensional structure [20]. In this thesis we will focus on the task of producing simple representations of high-dimensional data that can capture its underlying geometrical structure.

The importance of having methods to obtain simple representations of data lies in the simplification of data analysis tasks. Lower dimensional representations can avoid the issues grouped into the term of *curse of dimensionality* concerning the analysis of high dimensional data. Moreover, good data representations can serve for improving the performance of tasks. The recovery of useful representations for machine learning tasks corresponds to the research field of *representation learning* [2, 9].

In this work we focus on a special subset of dimensionality reduction techniques denominated as *latent variable separation methods* [20]. These approaches are based on probabilistic models that assume the existence of unobserved quantities called latent variables that participate in the generation of the observed data [3]. Latent variable separation methods proposes that each datapoint can be expressed in terms of a lower dimensional latent variable obtained from a probability distribution determined by the observed data called the posterior distribution. Bayesian inference is a method used for approximating to the underlying posterior distribution.

Within Bayesian inference it is important to select a suitable algorithm for obtaining the approximate posterior used to identify suitable latent variable representations of data. Some of these algorithms include Markov Chain Monte Carlo (MCMC), and Expectation Maximization (EM). These methods have the downside of being computationally expensive for big datasets and difficult to assess their convergence [11].

An alternative to the computationally expensive MCMC and EM is stochastic variational inference which provides a method that is scalable to large datasets. This technique estimates the posterior over the latent variables via parameter optimization by minimizing an objective function [3, 15]. D.P Kingma and M. Welling introduced variational autoencoders (VAEs) which is a method for variational inference that performs the parameter optimization by introducing the use of neural networks.

In this thesis we have focused on the use of variational autoencoders for performing variational inference and latent variable separation for our input datasets. VAEs are considered an unsupervised learning method capable of estimating a suitable generative model that emulates the process that produced a dataset with respect to the underlying latent variables. This means that variational autoencoders can be used for both: creating new datapoints with a learned generative model and for latent variable separation.

There are different modifications to the original variational autoencoder framework that attempt to improve its performance with respect to the quality of the data produced from the learned generative model and with respect to the latent variables recovered to represent data. Some of the modification include changes to the optimization objective (INFO-VAE [27],β-VAE [14]), lossy latent representations(Lossy VAE [1]) and the addition of autoregressive flows [17]. Another extension to the VAE framework is semi-supervised method that enforces the addition of a certain feature of interest into the latent variables of the generative model (CVAE [16]).

We will focus the use of variational autoencoders to the task of recovering the underlying geometrical structure within a dataset. It is important to note that the recovered latent representations of a dataset obtained with a variational autoencoder are determined by the assumptions made for the latent variable structure. For most examples non-restrictive assumptions are made for the latent variables in a VAE. In the settings studied , it can be valuable to include assumptions that constrain the structure of the latent variables by incorporating the geometrical intuition of the analyst with respect to the analyzed data. With this in mind, recent efforts are in the direction of enforcing a certain latent variable geometry such as the hyper-spherical [6] and toroidal [22]. These assumptions on the structure of the latent variables can be used in tasks such as angle-pose estimation of images [25].

The geometrical structure of a dataset can be identified beforehand by an analyst since it can be often connected to the underlying generative process that produced the data. It is of interest to identify whether the latent variable separation models are capable of obtaining lower dimensional representations of data that capture intuitive concepts interpretable by humans [13]. With this in mind we have identified the need to propose a mathematical definition for relating the latent variables of different generative models in order to provide an interpretation to the recovered representations attained by deep learning methods.

1.2 Overview of Thesis

This thesis is focused on the study of the latent representations obtained with variational autoencoders for different datasets with underlying geometrical structures (circular/toroidal).

Our contributions:

1. Proposed a simple benchmark dataset with underlying circular and toroidal structure.
2. Proposed a baseline variational autoencoder based on [18, 14] to test whether the recovered latent structure captures the underlying structure of the input dataset.
3. Introduced the diffusion variational autoencoder. This variational autoencoder assumes a latent space with a restricted circular/toroidal structure. Moreover, it employs a different family of probability distributions for approximating the posterior different to the ones proposed in [6] and [25].
4. Analyzed the recovered latent structures recovered for the datasets together with their data reconstruction performance.
5. Provided a definition for the concept of Δ-reducibility between the trained generative models of variational autoencoders. The concept is based on functions called Δ-reductors that map

the elements from the latent spaces of the variational autoencoders with the purpose of reducing one generative model in terms of the other up to a certain tolerance level $\Delta \in \mathbb{R}^+$.

6. For the generative models trained with the baseline and diffusion variational autoencoders we have proposed an algorithm for constructing simple reduction maps between the recovered latent spaces to reduce the corresponding generative models. We analyzed the effects of the recovered reduction maps.

1.3 Thesis structure

Chapter 2 introduces the mathematical notation of the thesis together with the basic concepts for Bayesian models and the corresponding variational inference methods for approximating the posterior distribution. In Chapter 3 the underlying theory for variational autoencoders is introduced and in Chapter 4 our baseline variational autoencoder is presented. Chapter 5 presents a benchmark dataset with underlying circular geometrical structure used to train the baseline variational autoencoder. Chapter 6 shows a proof for the recovered latent structure of a simplified baseline variational autoencoder. Chapter 7 presents the proposed diffusion variational autoencoder that incorporates the assumption of a circular latent space, the results of training with the benchmark dataset are presented in this section. In Chapter 8 we present the definition for Δ-reductions between recovered generative models and in Chapter 9 we provide a method for constructing these reductions between the learned generative models of the variational autoencoders. Chapter 10 presents some results for different datasets with a circular underlying structure. Finally on Chapter 11 we extend the examples from previous chapters for datasets with an underlying todoidal geometry.

Chapter 2

Bayesian Models

In this thesis we will be focusing on mathematical models that attempt to explain the observations/data within a probabilistic formalism. This thesis will be conveyed in terms of a more general mathematical framework based on measure theory. In the next section we review some basic definitions from measure-theoretic probability theory. Once the mathematical language is introduced we proceed to describe the Bayesian methods used in this thesis to process data.

2.1 Mathematical Background

Throughout lifetime, the experiences we gather from our interactions with the environment determines our interpretation of how the world is structured. By performing actions and processing the observed outcomes, we gather information that can be used to produce future predictions. A *model* corresponds to a simple description of a process that may have produced the observed outcomes [7].

The outcome of some processes given certain actions can be unpredictable. This can be due to the the complexity of underlying mechanisms or unknown factors that produce the observed outcomes and that, in a practical sense, are impossible to identify. In these cases, we need to introduce a formal framework that is capable of producing predictions that incorporate certain degree of uncertainty associated to our lack of information. This type of settings are the subject of study for probability theory.

Probability theory can be considered as a "rational framework for thinking about uncertainty" [26]. Probability theory deals with the outcomes of *trials* or *experiments* which are "any controlled study with an outcome of an uncertain kind"[7] and involves the use of controlled actions to identify observable results. The set of all possible outcomes from an experiment is denominated *sample space* and is denoted as Ω. For example, in the experiment of throwing a fair six-sided dice, the sample space is given by the numbers written in each of the faces $\Omega = \{1, 2, 3, 4, 5, 6\}$.

Within the sample space we can define smaller subsets named events that represent the answer to questions about the outcome of an experiment. An event, denoted as $\mathcal{E} \subseteq \Omega$ is a subset of the sample space Ω. The power set 2^{Ω} is the set of all possible events of Ω. In our previous example of the throw of a fair six-sided dice we could be interested in the outcome of rolling an even number, the associated event to this question would correspond to $\mathcal{E} = \{2, 4, 6\} \subseteq \Omega$. Often, one does not consider all possible subsets of Ω, but rather a restricted collection $\mathcal{F}_{\Omega}$. This restricted collection of subsets needs to satisfy certain properties. If $\mathcal{F}_{\Omega}$ contains the empty set and it is closed under the formation of complements and countable unions we say that it is a σ-algebra. The pair $(\Omega, \mathcal{F}_{\Omega})$ with $\mathcal{F}_{\Omega}$ a σ-algebra of Ω is called a *measurable space.* [24]

Definition 2.1.1. ***σ-algebra***
A collection of subsets of Ω, $\mathcal{F} \subset 2^{\Omega}$ is called a σ- algebra if

(i) $\emptyset \in \mathcal{F}$,

(ii) $\mathcal{E} \in \mathcal{F} \implies \mathcal{E}^c = \Omega \setminus \mathcal{E} \in \mathcal{F}$,

(iii) $\mathcal{E}_1, \mathcal{E}_2, \ldots \in \mathcal{F} \implies \bigcup\limits_{i=1}^{\infty} \mathcal{E}_i \in \mathcal{F}$.

Definition 2.1.2. ***Measurable space***
The pair $(\Omega, \mathcal{F}_\Omega)$ consisting of a set Ω and a σ-algebra $\mathcal{F}_\Omega$ is called a measurable space.

In an intuitive way we can think of the concept of probability as a value that describes how likely an event is observed in an experiment. A *probability measure*, also called *probability distribution*, is a function over a σ-algebra $\mathcal{F}$, $P_\Omega : \mathcal{F}_\Omega \mapsto [0,1]$ that associates to each event a non-negative value between zero and one. The higher the probability, the more likely an event is. The special properties of the probability measure functions are described within the axioms of probability established in 1933 by the mathematician Andrey Nikolaevich Kolmogorov. The probability measure is a special case of the more general concept of *measure* studied by the field of measure theory which is out of the scope of this work. A probabilistic view of measure theory can be consulted in [26].

Definition 2.1.3. ***Probability measure***
Let $\mathcal{F}$ be a σ-algebra of Ω. A mapping $\mathbb{P} : \mathcal{F}_\Omega \mapsto [0,1]$ is called a probability measure on $(\Omega, \mathcal{F}_\Omega)$ if

(i) $\mathbb{P}_\Omega(\Omega) = 1 \ \wedge \ \mathbb{P}_\Omega(\emptyset) = 0$

(ii) If $\mathcal{E}_1, \mathcal{E}_2, \ldots$ are mutually disjoint events in $\mathcal{F}_\Omega$ then $\mathbb{P}_\Omega\left(\bigcup\limits_{i=1}^{\infty} \mathcal{E}_i\right) = \sum\limits_{i=1}^{\infty} \mathbb{P}_\Omega(\mathcal{E}_i)$

Probability theory involves the study of probability spaces which consist of the three elements previously described: the sample space Ω, the σ-algebra $\mathcal{F}_\Omega$ from Ω and the probability measure $\mathbb{P}_\Omega$ defined over $\mathcal{F}_\Omega$.

Definition 2.1.4. ***Probability space***
A probability space consists of the triple $(\Omega, \mathcal{F}_\Omega, \mathbb{P}_\Omega)$ of items such that

(i) Ω is a set,

(ii) $\mathcal{F}_\Omega \subseteq 2^{\Omega}$ is a σ-algebra of Ω,

(iii) $\mathbb{P}_\Omega$ is a probability measure on $(\Omega, \mathcal{F}_\Omega)$.

There are several consequences that can be derived from these axioms and describe the properties of such spaces. Most of these results can be consulted in probability theory texts such as [4], [12]. Here we will present some important definitions and results that will be used in the following work.

Definition 2.1.5. *Let $(\Omega_1, \mathcal{F}_{\Omega_1})$, $(\Omega_2, \mathcal{F}_{\Omega_2})$ be two measurable spaces. A function $Y : \Omega_1 \mapsto \Omega_2$ is $(\mathcal{F}_{\Omega_1}, \mathcal{F}_{\Omega_2})$-measurable if for all $B \in \mathcal{F}_{\Omega_2}$ the preimage $Y^{-1}(B) \in \mathcal{F}_{\Omega_1}$. That is,*

$$\{\omega \in \Omega_1 | Y(\omega) \in B\} \in \mathcal{F}_{\Omega_1}.$$

In probability theory measurable functions are called random variables.

Once that we have defined the concept of measure over a sample space, it is useful to mention the concept of probability density of a measure. First we need to define what is absolute continuity, then we will present the important result derived by Johann Radon and Otto Nikodym for connecting two probability measures over a sample space and finally we will present the definition of a density function with respect to a measure.

Definition 2.1.6. ***Absolute Continuity***
Let $(\Omega, \mathcal{F}_\Omega)$ be a measurable space and let $\mathbb{P}_1$, $\mathbb{P}_2$ be two probability measures on $(\Omega, \mathcal{F}_\Omega)$. We say that $\mathbb{P}_2$ is absolutely continuous with respect to $\mathbb{P}_1$ if for all $\mathcal{E} \in \mathcal{F}_\Omega$ then

$$\mathbb{P}_1(\mathcal{E}) = 0 \textit{ then } \mathbb{P}_2(\mathcal{E}) = 0 \tag{2.1}$$

Theorem 1. ***Radon-Nikodym Theorem***
Let $(\Omega, \mathcal{F}_\Omega)$ be a measurable space and let $\mathbb{P}_1$, $\mathbb{P}_2$ be two probability measures on $(\Omega, \mathcal{F}_\Omega)$ such that $\mathbb{P}_2$ is absolutely continuous with respect to $\mathbb{P}_1$. Then there exists a $\mathbb{P}_1$ integrable function called the Radon-Nykodym derivative denoted by

$$\frac{d\mathbb{P}_2}{d\mathbb{P}_1} \tag{2.2}$$

Such that

$$\mathbb{P}_2 = \frac{d\mathbb{P}_2}{d\mathbb{P}_1}\mathbb{P}_1 \tag{2.3}$$

For all $\mathcal{E} \in \mathcal{F}_\Omega$, it holds that

$$\mathbb{P}_2(\mathcal{E}) = \int_{\mathcal{E}} \frac{d\mathbb{P}_2}{d\mathbb{P}_1} d\mathbb{P}_1 \tag{2.4}$$

Definition 2.1.7. ***Probability density***
Given the probability space $(\Omega, \mathcal{F}_\Omega, \mathbb{P}_\Omega)$ and a measure $\mathbb{P}$ over $(\Omega, \mathcal{F}_\Omega)$. A probability density of $\mathbb{P}_\Omega$ with respect to a measure $\mathbb{P}$ is a function $P_\Omega : \Omega \mapsto \mathbb{R}_0^+$ such that for a given event $\mathcal{E} \in \mathcal{F}_\Omega$,

$$\mathbb{P}_\Omega(\mathcal{E}) = \int_A P_\Omega(\omega) d\mathbb{P}(\omega). \tag{2.5}$$

In this thesis we will assume the existence of the various probability densities that appear.

2.2 Generative models and the latent space

Bayesian models posit that the sample space Ω of an experiment can be divided into a set of observable outcomes X called *data space* and a set of unknown unobservable outcomes Z called *latent space* such that $\Omega = X \times Z$. The probability space that describes such an experiment is given by $(X \times Z, \mathcal{F}_X \otimes \mathcal{F}_Z, \mathbb{P}_{X\times Z})$. In a trial, the outcome $(x, z) \in X \times Z$ is assumed to be sampled from the probability measure $\mathbb{P}_{X\times Z}$. The observed outcome $x \in X$ is called *datapoint*, the unobserved outcome $z \in Z$ is called *latent variable* and the measure $\mathbb{P}_{X\times Z}$ is usually referred to as the *generative model*.

We will assume that the observed outcomes obtained from an experiment can be identified with elements in the D dimensional Euclidean space, i.e. $X = \mathbb{R}^D$. This can be a sensible assumption if we consider that usually, while studying the world that surrounds us, we gather data by means of a certain device that assigns a fixed number to the properties of a system.

The case of the latent space Z is treated differently since, as it has been stated, it represents unobserved outcomes that in principle might not have a structure that can be reliably represented with Euclidean space. Additionally, the latent space is assumed to be describable in terms of a structure with a dimension smaller than that of the data space X.

For a given experiment, a datapoint $x \in X$ is assumed to be dependent on the corresponding latent variable $z \in Z$. Therefore latent variables, although unobserved, are considered to influence the observed data. Consider an observed outcome $x \in X$ and an unobserved outcome $z \in Z$, we can express the probability density of the generative model $P_{X\times Z}$ with respect to the $\mathcal{L}^D \otimes \mathbb{P}_Z$ measure in terms of the conditional density $\mathbb{P}_{X|z}$ with respect to the $\mathcal{L}^D$ measure and the marginal density P_Z with respect to the $\mathbb{P}_Z$ measure,

$$P_{X\times Z}(x,z) = P_{X|z}(x)P_Z(z). \tag{2.6}$$

The probability densities of Equation (2.6) are defined in terms of the D-dimensional Lebesgue measure $\mathcal{L}^D$ and the measure $\mathbb{P}_Z$ over Z.

Definition 2.2.1. ***Radon-Nykodim applied to generative model***
Let $\mathbb{P}_{X\times Z}$ and $\mathcal{L}^D \otimes \mathbb{P}_Z$ be probability measures over the measurable space $(X\times Z, \mathcal{F}_X \otimes \mathcal{F}_Z)$ such that $\mathbb{P}_{X\times Z}$ is absolutely continuous with respect to $\mathcal{L}^D \otimes \mathbb{P}_Z$. Then there exists the $\mathcal{L}^D \otimes \mathbb{P}_Z$-integrable function $P_{X\times Z} : X \times Z \mapsto \mathbb{R}_0^+$ such that for every event $\mathcal{E}_X \times \mathcal{E}_Z \in \mathcal{F}_X \otimes \mathcal{F}_Z$

$$\mathbb{P}_{X\times Z}(\mathcal{E}_X \times \mathcal{E}_Z) = \int_{\mathcal{E}_X\times\mathcal{E}_Z} P_{X\times Z}(x',z')d(\mathcal{L}^D \otimes \mathbb{P}_Z)(x',z') \tag{2.7}$$

We denote $P_{X\times Z}$ as the probability density function of the measure $\mathbb{P}_{X\times Z}$ with respect to the $\mathcal{L}^D \otimes \mathbb{P}_Z$ measure.

Definition 2.2.2. ***Marginal probability***
Consider the probability space $(X\times Z, \mathcal{F}_X \otimes \mathcal{F}_Z, \mathbb{P}_{X\times Z})$. Let $P_{X\times Z}$ be the probability density of $\mathbb{P}_{X\times Z}$ with respect to the $\mathcal{L}^D \otimes \mathbb{P}_Z$ measure. The marginal probability density $P_X : X \mapsto \mathbb{R}_0^+$ over X with respect to the $\mathcal{L}^D$ measure for a given outcome $x \in X$ is given by the integral

$$P_X(x) = \int_Z P_{X\times Z}(x,z')d\mathbb{P}_Z(z'). \tag{2.8}$$

Analogously, the marginal probability density $P_Z : X \mapsto \mathbb{R}_0^+$ over Z with respect to the $\mathbb{P}_Z$ measure for a given outcome $z \in Z$ is given by the integral

$$P_Z(z) = \int_X P_{X\times Z}(x',z)d\mathcal{L}^D(x'). \tag{2.9}$$

From the marginal probability densities, we define the probability spaces $(X, \mathcal{F}_X, \mathbb{P}_X)$, $(Z, \mathcal{F}_Z, \mathbb{P}_Z)$ where each measure is defined in terms of the integral of their corresponding probability densities. For a given event $A \in \mathcal{F}_X$ and $B \in \mathcal{F}_Z$ we have the marginal probability measures given by

$$\mathbb{P}_X(A) = \int_A P_X(x')d\mathcal{L}^D(x'),$$

$$\mathbb{P}_Z(B) = \int_B P_Z(z')d\mathbb{P}_Z(z').$$

Definition 2.2.3. ***Conditional probability***
Let P_X, P_Z be the marginal probability densities with respect to $\mathcal{L}^D$ and $\mathbb{P}_Z$ respectively. Let $P_{X\times Z}$ be the probability density of the generative model with respect to the $\mathcal{L}^D \otimes \mathbb{P}_Z$ measure. The probability density function over X given an outcome $z \in Z$ with $P_Z(z) \neq 0$ is given by the function $P_{X|z} : X \mapsto \mathbb{R}_0^+$ defined by

$$P_{X|z}(x) = \frac{P_{X\times Z}(x,z)}{P_Z(z)}.$$

Analogously, the probability density function over Z given an outcome $x \in X$ with $P_X(x) \neq 0$ is given by the function $P_{Z|x} : Z \mapsto \mathbb{R}_0^+$. For an outcome $z \in Z$ this function takes the value

$$P_{Z|x}(z) = \frac{P_{X\times Z}(x,z)}{P_X(x)}.$$

From the conditional probability densities, we define the probability spaces $(X, \mathcal{F}_X, \mathbb{P}_{X|z})$, $(Z, \mathcal{F}_Z, \mathbb{P}_{Z|x})$ were each measure is defined in terms of the integral of their corresponding densities. For a given *event $A \in \mathcal{F}_X$ and $B \in \mathcal{F}_Z$ we have the conditional probability measures given by*

$$\begin{aligned}\mathbb{P}_{X|z}(A) &= \int_A P_{X|z}(x')d\mathcal{L}^D(x'),\\ \mathbb{P}_{Z|x}(B) &= \int_B P_{Z|x}(z')d\mathbb{P}_Z(z').\end{aligned}$$

Equation (2.6) provides an interpretation for the process that produces the outcome $(x, z) \in X \times Z$ which can be described in two steps involving the dependence of an observed outcome x with respect to the latent variable z.

1. An unobserved outcome $z \in Z$ is sampled according to the marginal distribution $\mathbb{P}_Z$.
2. The datapoint $x \in X$ is obtained by sampling from the conditional distribution $\mathbb{P}_{X|z}$ defined by the outcome $z \in Z$.

In a practical setting, when we perform an experiment, the only information that we have available is the observed outcomes represented by datapoints $x \in X$. Given an observation $x \in X$ obtained from an experiment, the underlying conditional distribution $\mathbb{P}_{Z|x}$ over Z describes the possible latent variables involved in the generation of x through the process described by Equation (2.6). The probability measure $\mathbb{P}_{Z|x}$ is called the *posterior* distribution since it provides a model describing the latent variables after incorporating the information about observation x.

Inference corresponds to the techniques involved in estimating the properties of an underlying distribution by processing the information provided by data [7]. In particular, Bayesian inference is founded on Bayes' theorem that provides a formula for the estimation of the probability density function $P_{Z|x}$ of $\mathbb{P}_{Z|x}$ in terms of $x \in X$.

Theorem 2 (Bayes' theorem)**.** *Consider the probability space $(X \times Z, \mathcal{F}_X \otimes \mathcal{F}_Z, \mathbb{P}_{X\times Z})$ that describes the outcomes of an experiment. For a given outcome $(x, z) \in X \times Z$ such that $P_X(x) \neq 0$, let $P_{Z|x}$, P_Z, be the probability densities of the measures $\mathbb{P}_{Z|x}$, $\mathbb{P}_Z$ with respect to the $\mathbb{P}_Z$ measure. Let P_X, $P_{X|z}$ be the probability densities of the measures, $\mathbb{P}_{X|z}$, $\mathbb{P}_X$ with respect to the Lebesgue measure $\mathcal{L}^D$. Bayes' theorem states the relationship between densities is given by*

$$P_{Z|x}(z) = \frac{P_{X|z}(x)P_Z(z)}{P_X(x)}. \tag{2.10}$$

The distribution $\mathbb{P}_Z$ associated to the probability density P_Z of Equation (2.10) is called the *prior* distribution and incorporates the knowledge of the underlying latent space *before* having any information about the data. The value $P_{X|z}(x)$ represents the likelihood of datapoint x with respect to a certain latent variable z. Intuitively, it provides information about how probable it is that the latent variable z was involved in the generation of x.

For a given datapoint $x \in X$ the probability density function $P_X(x)$ is called the *evidence* of x. In order to estimate the probability density of the posterior distribution $P_{Z|x}$, the value of the evidence is needed. For some cases computing this value can be intractable, this is discussed even further in the next section where we introduce the method of *variational inference* used for approximating the posterior distribution from a dataset of N independent observations.

2.3 Variational Inference

Consider a dataset $\mathcal{X} = \{x_i\}_{i=1}^N$ of N independent and identically distributed outcomes obtained from an experiment. *Variational inference* is a method used for approximating the unknown posterior probability distribution $\mathbb{P}_{Z|x}$ for datapoint $x \in \mathcal{X}$ via parameter optimization [3]. The main idea is to identify the element of a proposed family of parametric distributions that is close to the target unknown posterior. Closeness is measured with the Kullback-Leibler divergence.

Consider the set of all probability measures over the measurable space $(Z, \mathcal{F}_Z)$ denoted as $\mathcal{P}_Z$. A given family of parametric distributions $\mathcal{Q}_Z^A \subseteq \mathcal{P}_Z$ corresponds to a set of probability measures over the measurable space $(Z, \mathcal{F}_Z)$ such that for each parameter α in a parameter set A, there is a corresponding distribution $\mathbb{Q}_Z^\alpha$

$$\mathcal{Q}_Z^A = \{\mathbb{Q}_Z^\alpha \in \mathcal{P}_Z \mid \alpha \in A\}. \tag{2.11}$$

For example, in the case where the latent space Z can be identified with the d dimensional Euclidean space $\mathbb{R}^d$ the parametric family of normal distributions over $Z = \mathbb{R}^d$ is defined as

$$\mathcal{Q}_{\mathbb{R}^d}^A = \left\{ \mathbb{Q}_{\mathbb{R}^d}^{(\mu,\Sigma)} \in \mathcal{P}_{\mathbb{R}^d} \;\middle|\; Q_{\mathbb{R}^d}^{(\mu,\Sigma)} = \frac{\det(\Sigma)^{-1/2}}{\sqrt{(2\pi)^d}} \exp\left(-\frac{1}{2}(z-\mu)^T \Sigma^{-1}(z-\mu)\right) \right\}. \tag{2.12}$$

Here $Q_{\mathbb{R}^d}^{(\mu,\Sigma)}$ is the probability density of $\mathbb{Q}_{\mathbb{R}^d}^{(\mu,\Sigma)}$ with respect to the Lebesgue measure $\mathcal{L}^d$. Each member of the parametric family has a mean vector $\mu \in \mathbb{R}^d$ and covariance matrix $\Sigma \in \mathbb{R}^{d\times d}$. The set of possible parameters is then described as $A = \{(\mu, \Sigma) \in \mathbb{R}^d \times \mathbb{R}^{d\times d}\}$.

The goal of variational inference is to find for each datapoint $x \in \mathcal{X}$ the appropriate parameters $\alpha^* \in A$ such that the optimal approximation $\mathbb{Q}_Z^{\alpha^*} \in \mathcal{Q}_Z^A$ minimizes the Kullback-Leibler divergence to the true posterior $\mathbb{P}_{Z|x}$. This condition is stated as

$$\alpha^* = \arg\min_{\alpha\in A} \mathrm{KL}(\mathbb{Q}_Z^\alpha || \mathbb{P}_{Z|x}). \tag{2.13}$$

From Bayes' theorem, in order to calculate the probability density of the posterior $P_{Z|x}$ one of the challenges is to estimate the evidence of datapoint x obtained by integrating the generative model probability density over the latent space

$$P_X(x) = \int_Z P_{X\times Z}(x,z) d\mathbb{P}_Z(z) = \int_Z P_{X|z}(x) P_Z(z) d\mathbb{P}_Z(z). \tag{2.14}$$

In general, the value of this integral is not always available either because there is no analytical expression or the amount of computations required to calculate it can make it intractable computationally [3]. The Kullback-Leibler divergence of Equation (2.13) involves the estimation of the probability density for the evidence P_X. First, we show that such term is included in the condition of equation (2.13) and then provide a workaround to its intractability.

Let Q_Z^α, $P_{Z|x}$ be the probability density functions of $\mathbb{Q}_Z^\alpha$ and $\mathbb{P}_{Z|x}$ accordingly with respect to the prior distribution $\mathbb{P}_Z$. The Kullback-Leibler divergence of the posterior $\mathbb{P}_{Z|x}$ with respect to an approximation $\mathbb{Q}_Z^\alpha$ is defined as

$$\mathrm{KL}(\mathbb{Q}_Z^\alpha || \mathbb{P}_{Z|x}) = \mathbb{E}_{\mathbb{Q}_Z^\alpha}\left[\log\left(\frac{Q_Z^\alpha}{P_{Z|x}}\right)\right] = \int_Z \log\left(\frac{Q_Z^\alpha(z)}{P_{Z|x}(z)}\right) d\mathbb{Q}_Z^\alpha(z). \tag{2.15}$$

Bayes' theorem presented in section (2) provides a way of rewriting the density of the posterior $P_{Z|x}$ in terms of the densities of the prior P_Z, the likelihood $P_{X|z}$ and the evidence P_X. By substituting equation (2.10) into the Kullback-Leibler divergence of $\mathbb{P}_{Z|x}$ with respect to $\mathbb{Q}_Z^\alpha$ we obtain the equation

$$\mathrm{KL}(\mathbb{Q}_Z^\alpha || \mathbb{P}_{Z|x}) = \int_Z \log\left(\frac{Q_Z^\alpha(z) P_X(x)}{P_{X|z_i}(x) P_Z(z)}\right) d\mathbb{Q}_Z^\alpha(z). \tag{2.16}$$

The terms can be rearranged within the integral and expressed as

$$\mathrm{KL}(\mathbb{Q}_Z^\alpha||\mathbb{P}_{Z|x}) = \int_Z \log P_X(x) + \log\left(\frac{Q_Z^\alpha(z')}{P_Z(z')}\right) - \log P_{X|z'}(x)\, d\mathbb{Q}_Z^\alpha(z'). \tag{2.17}$$

The first term log $P_X(x)$ is independent of $z \in Z$ and can be taken out of the integral. The second term corresponds to the Kullback-Leibler divergence of the prior $\mathbb{P}_Z$ with respect to $\mathbb{Q}_Z^\alpha$. The third term corresponds to the expected value of log $P_{X|\cdot}(x)$ with respect to the measure $\mathbb{Q}_Z^\alpha$. Thus, the Kullback-Leibler divergence is rewritten as

$$\mathrm{KL}(\mathbb{Q}_Z^\alpha||\mathbb{P}_{Z|x}) = \log P_X(x) + \mathrm{KL}(\mathbb{Q}_Z^\alpha||\mathbb{P}_Z) - \mathbb{E}_{z\sim\mathbb{Q}_Z^\alpha}[\log P_{X|z}(x)]. \tag{2.18}$$

The presence of the evidence $P_X(x)$ of datapoint x can already be recognized in the first term of the right hand side which we will move to the left,

$$\mathrm{KL}(\mathbb{Q}_Z^\alpha||\mathbb{P}_{Z|x}) - \log P_X(x) = \mathrm{KL}(\mathbb{Q}_Z^\alpha||\mathbb{P}_Z) - \mathbb{E}_{z\sim\mathbb{Q}_Z^\alpha}[\log P_{X|z}(x)]. \tag{2.19}$$

The terms in the right hand side of Equation (2.15) are grouped into the function $\mathscr{L} : X \times A \mapsto \mathbb{R}_0^+$ of datapoint $x \in \mathcal{X}$ and the parameters $\alpha \in A$ given by

$$\mathscr{L}(x, \alpha) = \mathbb{E}_{z\sim\mathbb{Q}_Z^\alpha}[\log P_{X|z}(x)] - \mathrm{KL}(\mathbb{Q}_Z^\alpha||\mathbb{P}_Z) \tag{2.20}$$

Since the $\log P_X(x)$ term of the left hand side is independent of the distribution $\mathbb{Q}_Z^\alpha$, we can minimize the left hand side of Equation 2.19 with respect to $\mathbb{Q}_Z^\alpha$ by maximizing the function $\mathscr{L}(x, \alpha)$ or equivalently minimize $-\mathscr{L}(x, \alpha)$ with respect to α. This new selection criteria avoids the estimation of the evidence and corresponds to

$$\alpha^* = \arg\max_{\alpha\in A} \mathscr{L}(x, \alpha) = \arg\min_{\alpha\in A} -\mathscr{L}(x, \alpha). \tag{2.21}$$

The function $\mathscr{L}$ is called the evidence lower bound (ELBO) since it provides a constraint for the logarithm of P_X such that $\log P_X(x) \geq \mathscr{L}(x, \alpha)$. Rearranging equation (2.18) shows this relation between the ELBO and log$P_X(x)$,

$$\log P_X(x) = \mathrm{KL}(\mathbb{Q}_Z^\alpha||\mathbb{P}_{Z|x}) + \mathscr{L}(x, \alpha) \geq \mathscr{L}(x, \alpha). \tag{2.22}$$

Where the second relation holds since the Kullback-Leibler divergence is non-negative for any pair of distributions, i.e. $\mathrm{KL}(\cdot||\cdot) \geq 0$. In the case where the dataset $\mathcal{X}$ corresponds to N independent and identically distributed samples from X, we can provide a lower bound to the joint probability density of the complete dataset $\mathcal{X}$,

$$\log\left(\prod_{i=1}^N P_X(x)\right) = \sum_{i=1}^N \log P_X(x) \geq \sum_{i=1}^N \mathscr{L}(x, \alpha). \tag{2.23}$$

We define the ELBO of the complete dataset $\mathcal{X}$ of independent and identically distributed datapoints as

$$\mathscr{L}(\mathcal{X}, \alpha) = \sum_{x\in\mathcal{X}} \mathscr{L}(x, \alpha) \tag{2.24}$$

In summary, variational inference tries to approximate the unknown posterior $\mathbb{P}_{Z|x}$ for a given datapoint $x \in X$ by using the available information gathered in a dataset $\mathcal{X}$ of N independent and identically distributed datapoints. The posterior is approximated by finding the optimal parameters $\alpha^* \in A$ for a proposed distribution $\mathbb{Q}_Z^{\alpha^*}$ member of a parametric family such that it maximizes the ELBO of the complete dataset $\mathcal{X}$ given by $\mathscr{L}(\mathcal{X}, \alpha^*)$

$$\alpha^* = \arg\min_{\alpha\in A} -\mathscr{L}(\mathcal{X}, \alpha). \tag{2.25}$$

The main challenge is how to carry out this parameter optimization. Up until now we have not introduced any assumptions on the underlying likelihood distribution $\mathbb{P}_{X|z}$ or the prior $\mathbb{P}_Z$ which are necessary to calculate the ELBO from equation (2.20). There are different methods that can be used for obtaining an approximate distribution based on different assumptions. In this work we will focus on the variational autoencoders developed by Kingma and Welling [18] which use neural networks to satisfy the selection criteria of the ELBO.

2.4 The evidence lower bound

The ELBO function $\mathscr{L}(x, \alpha)$ for a single datapoint $x \in \mathcal{X}$ and parameter $\alpha \in A$ is conformed of two terms. Maximizing the ELBO for datapoint x is equivalent to minimizing the negative ELBO. In the machine learning context most problems are treated as a minimization of a certain loss function. Thus, in the next sections we will consider variational inference as a technique for approximating the posterior by finding the optimal parameters α^* that minimize a loss function equal to the negative ELBO. We will provide some intuition on each term of the negative ELBO by considering the consequences of minimizing each term independently.

$$-\mathscr{L}(x, \alpha) = -\mathbb{E}_{\mathbb{Q}_Z^\alpha}[\log P_{X|\cdot}(x)] + \mathrm{KL}(\mathbb{Q}_Z^\alpha || \mathbb{P}_Z) \tag{2.26}$$

The first term of the negative ELBO corresponds to the average of $\log P_{X|\cdot}(x)$ over the latent space Z with respect to the posterior approximation distribution $\mathbb{Q}_Z^\alpha$ given by

$$-\mathbb{E}_{\mathbb{Q}_Z^\alpha}[\log P_{X|\cdot}(x)] = -\int_Z \log P_{X|z'}(x) d\mathbb{P}_Z(z'). \tag{2.27}$$

The goal of minimizing this term corresponds to finding the parameters of the approximate posterior $\mathbb{Q}_Z^\alpha$ that places its mass over the latent variables $z \in Z$ that maximize the likelihood $P_{X|z}(x)$ for observing datapoint $x \in \mathcal{X}$. This term is denoted as the reconstruction error since it penalizes the parameters that fail to maximize the probability of observing the datapoints in a dataset with respect to the latent variables sampled according to the approximate posterior.

The second term of Equation (2.20) corresponds to the Kullback-Leibler divergence between the approximate posterior $\mathbb{Q}_Z^\alpha$ and the prior $\mathbb{P}_Z$. Recall that the Kullback-Leibler divergence between two probability distribution is always nonnegative. Thus, the minimum value that this second term can take corresponds to zero. The second term of the ELBO is expressed as

$$-\mathrm{KL}(\mathbb{Q}_Z^\alpha || \mathbb{P}_Z) = -\int_Z \log \left(\frac{Q_Z^\alpha(z')}{P_Z(z')} \right) d\mathbb{P}_Z(z'). \tag{2.28}$$

Optimizing α to maximize this second term forces the approximate distribution $\mathbb{Q}_Z^\alpha$ to mimic the prior distribution. This term is referred to as the Kullback-Leibler regularization term since it restricts the shape that the approximate posterior distribution can take. Maximizing the ELBO of Equation (2.20) results in finding the optimal α^* that balances both, the reconstruction error and the Kullback-Leibler regularization term.

Chapter 3

Variational Autoencoders

Kingma and Welling [18] introduced a variational inference method that uses neural networks to minimize the negative ELBO. Within the variational autoencoder context the latent space Z is reinterpreted as the space of codes that represents the observed elements from X. For each observed outcome $x \in X$, we can obtain an encoding/latent representation by sampling from the posterior $\mathbb{P}_{Z|x}$. Thus we refer to the posterior $\mathbb{P}_{Z|x}$ as the encoding distribution over Z that describes the possible latent variables associated to observation $x \in X$. Likewise, we refer to $\mathbb{P}_{X|z}$ as the decoding distribution that describes possible observations from X that can be obtained given a specific latent variable $z \in Z$.

In a variational autoencoders both the encoder $\mathbb{P}_{Z|x}$ and decoder $\mathbb{P}_{X|z}$ distributions of the true underlying model are approximated with members of parametric families of distributions whose parameters are calculated with neural networks. The internal weights of those neural networks are optimized via the minimization of a loss function corresponding to the negative ELBO through stochastic gradient descent and backpropagation with respect to input data.

3.1 Minimization of the negative ELBO

Consider a dataset $\mathcal{X} = \{x_i\}_{i=1}^N$ of N independent and identically distributed random variables. In Section 2.3 we introduced the selection criteria for obtaining an approximation to the posterior distribution $\mathbb{P}_{Z|x}$ for an individual datapoint $x \in \mathcal{X}$ in terms of the negative ELBO function given by

$$-\mathscr{L}(x,\alpha) = -\mathbb{E}_{\mathbb{Q}_Z^\alpha}[\log P_{X|\cdot}(x_i)] + KL(\mathbb{Q}_Z^\alpha||\mathbb{P}_Z). \tag{3.1}$$

Where $\mathbb{Q}_Z^\alpha$ is an approximation to the posterior distribution $\mathbb{P}_{Z|x_i}$ and a member of the parametric family $\mathcal{Q}_Z^A$ with possible parameters in the set A. In variational autoencoders, this family is chosen such that for each member $\mathbb{Q}_Z^\alpha \in \mathcal{Q}_Z^A$, its probability density function Q_Z^α is differentiable almost everywhere with respect to α.

Furthermore, it is also assumed that the decoding distribution $\mathbb{P}_{X|z}$ for a given latent variable $z \in Z$ is also a member from a parametric family of distributions $\mathcal{P}_X^B \subseteq \mathcal{P}_X$ with

$$\mathcal{P}_X^B = \{\mathbb{P}_X^\beta \in \mathcal{P}_X \mid \beta \in B\}. \tag{3.2}$$

Chosen such that, for each member, its probability density function is differentiable almost everywhere with respect to the parameters in B.

In Chapter 2 we described the prior $\mathbb{P}_Z$ as the distribution that incorporates the assumptions of the unknown latent variables before having any information about the data. This distribution is chosen beforehand as a fixed member from a parametric family of distributions $\mathbb{P}_{Z;\gamma} \in \mathcal{P}_Z^\Gamma$ with the set of parameters Γ such that its probability density P_Z^γ is differentiable almost everywhere with respect to γ.

The parameters $\alpha \in A$ for the encoder distribution $\mathbb{Q}_Z^\alpha$ are calculated from an observation $x \in \mathcal{X}$ with the neural network function $\boldsymbol{\alpha}^{(\psi)} : X \mapsto A$. The neural network function is determined by the value of its internal weights $\psi \in \Psi$ where Ψ is the set of all possible weights and depends on the architecture of the network. Likewise, the parameters β for the decoder distribution $\mathbb{P}_X^\beta$ are calculated from a given latent variable $z \in Z$ with the neural network function $\boldsymbol{\beta}^{(\xi)} : Z \mapsto B$. Where $\xi \in \Xi$ represents corresponding internal weights of the neural network.

For a fixed prior $\mathbb{P}_Z^\gamma$ and considering the previously described assumptions, we redefine the negative ELBO function for datapoint $x \in \mathcal{X}$ as a function of the internal weights of the neural networks that calculate the parameters of the encoder and decoder distributions in $\mathcal{Q}_Z^A$ and $\mathcal{P}_X^B$. Such function $\mathscr{L} : X \times \Psi \times \Xi \mapsto \mathbb{R}_0^+$ for the neural network weights $(\psi, \xi) \in \Psi \times \Xi$ is equal to

$$-\mathscr{L}(x, \psi, \xi) = -\mathbb{E}_{\mathbb{Q}_Z^{\boldsymbol{\alpha}^{(\psi)}(x)}}\left[\log P_X^{\boldsymbol{\beta}^{(\xi)}(\cdot)}(x)\right] + \mathrm{KL}(\mathbb{Q}_Z^{\boldsymbol{\alpha}^{(\psi)}(x)}||\mathbb{P}_Z^\gamma). \tag{3.3}$$

For a dataset of N independent and identically distributed datapoints, the main goal of a variational autoencoder is to identify the optimal neural network weights $(\psi^*, \xi^*) \in \Psi \times \Xi$ that produce the parameters for the optimal encoder and decoder distributions that minimizes the negative ELBO of the complete dataset defined as

$$\mathscr{L}(\mathcal{X}, \psi, \xi) = \sum_{x \in \mathcal{X}} \mathscr{L}(x, \psi, \xi) \tag{3.4}$$

Therefore, the optimal parameters $(\psi^*, \xi^*) \in \Psi \times \Xi$ are obtained from the condition

$$(\psi^*, \xi^*) = \arg \max_{(\psi,\xi)\in\Psi\times\Xi} \mathscr{L}(\mathcal{X}, \psi, \xi) \tag{3.5}$$

The previous equation states a minimization problem which can be optimized via stochastic gradient descent coupled with backpropagation through the neural networks $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$. In general, the evidence lower bound of equation (2.20) does not have a closed analytical form. Thus, for some choices of parametric encoding and decoding families, the ELBO must be approximated via the Monte Carlo method.

3.2 Monte Carlo estimation of the evidence lower bound

Notice that for each datapoint $x \in \mathcal{X}$, the ELBO includes the computation of two expected values over the latent space Z with respect to the measure $\mathbb{Q}_Z^\alpha$. These expected values do not necessarily have an analytical expression that can be computed. For such cases, an estimation via the Monte Carlo method can be used.

For datapoint x, L samples $\{z^{(l)}\}_{l=1}^L$ are taken from latent space Z according to $\mathbb{Q}_Z^{\boldsymbol{\alpha}^{(\psi)}(x)}$ where $z^{(l)}$ is the l-th sample. The approximate expected value is obtained according to the Monte Carlo method by calculating the average with respect to these samples,

$$-\mathbb{E}_{\mathbb{Q}_Z^{\boldsymbol{\alpha}^{(\psi)}(x)}}[\log(P_X^{\boldsymbol{\beta}^{(\xi)}(\cdot)}(x))] \approx -\frac{1}{L}\sum_{l=1}^{L} \log(P_X^{\boldsymbol{\beta}^{(\xi)}(z^{(l)})}(x)). \tag{3.6}$$

The second term of the ELBO which corresponds to the Kullback-Leibler regularization term can then be estimated as

$$-\operatorname{KL}(\mathbb{Q}_Z^{\boldsymbol{\alpha}^{(\psi)}(x)}||\mathbb{P}_Z^{\gamma}) \approx \frac{1}{L}\sum_{l=1}^{L}\log\left(\frac{Q_Z^{\boldsymbol{\alpha}^{(\psi)}(x)}(z^{(l)})}{P_Z^{\gamma}(z^{(l)})}\right). \tag{3.7}$$

The number of samples L needed for this approximation depends on the amount of data possessed according to D.P. Kingma and M. Welling in [18]. For a large amount of data, for example one hundred, the value of L can be chosen as one. It is important to mention that for some distribution choices in this thesis, this term can be calculated analytically.

From both terms we define the approximate ELBO for datapoint x and the samples as $\tilde{\mathscr{L}} : X \times \Psi \times \Xi \mapsto \mathbb{R}_0^+$. For the neural network weights $(\psi, \xi) \in \Psi \times \Xi$ it is equal to

$$\tilde{\mathscr{L}}(x, \psi, \xi) = \frac{1}{L}\sum_{l=1}^{L}\log(P_X^{\boldsymbol{\beta}^{(\xi)}(z^{(l)})}(x)) - \log\left(\frac{Q_Z^{\boldsymbol{\alpha}^{\psi}(x)}(z^{(l)})}{P_Z^{\gamma}(z^{(l)})}\right). \tag{3.8}$$

The approximate ELBO for the complete dataset $\mathcal{X}$ is given by

$$\tilde{\mathscr{L}}(\mathcal{X}, \psi, \xi) = \sum_{x \in \mathcal{X}} \tilde{\mathscr{L}}(x, \psi, \xi). \tag{3.9}$$

The optimal parameters $(\psi^*, \xi^*) \in \Psi \times \Xi$ can be obtained by minimizing the approximate negative ELBO expressed as

$$(\psi^*, \xi^*) = \arg \min_{(\psi,\xi)\in\Psi\times\Xi} -\tilde{\mathscr{L}}(\mathcal{X}, \psi, \xi) \tag{3.10}$$

The optimal weights for the neural networks described in the previous section can be attained through gradient ascent by backpropagation through the neural network. An estimate of this gradient is simply obtained also with Monte Carlo sampling and expressed in terms of the approximate ELBO.

$$-\nabla_{\psi,\xi}\tilde{\mathscr{L}}(\mathcal{X}, \psi, \xi) = -\sum_{x \in \mathcal{X}} \nabla_{\psi,\xi}\tilde{\mathscr{L}}(x_i, \psi, \xi). \tag{3.11}$$

3.3 Reparametrization trick

The Monte Carlo method for estimating the ELBO for a datapoint x_i with Equation 3.8 requires samples from Z according to the distribution $\mathbb{Q}_Z^{\boldsymbol{\alpha}^{(\psi)}(x)}$. The process of sampling latent variables from Z does not allow the flow of gradients with backpropagation through the neural networks. To solve this issue, we define a function Rep_Z that can transform elements from an auxiliary probability space E into elements of Z. Such function has the property that sampling elements of Z according to $\mathbb{Q}_Z^{\boldsymbol{\alpha}^{(\psi)}(x)}$ is equivalent to sampling elements of E according to $\mathbb{P}_E$. Moreover, sampled elements from E are provided as input to the variational autoencoder and gradients can be backpropagated throughout the network.

Definition 3.3.1. ***VAE reparametrization function***
Consider the auxiliary probability space $(E, \mathcal{F}_E, \mathbb{P}_E)$ from which we will obtain the auxiliary outcome $\epsilon \in E$ by sampling over E according to $\mathbb{P}_E$. We denominate as reparametrization function $\operatorname{Rep}_Z : A \times E \mapsto Z$ a function differentiable almost everywhere with respect to the parameters in A with the property that for a given parameter α corresponding to datapoint x_i, the law of this function $\operatorname{Rep}_Z(\alpha, \cdot)_{\#}\mathbb{P}_E$ determines a distribution over the measurable space $(Z, \mathcal{F}_Z)$. Furthermore, the reparametrization function is such that for any bounded continuous function $f : Z \mapsto \mathbb{R}$ and parameter $\alpha \in A$ then

$$\mathbb{E}_{\mathbb{P}_E}[f(\mathrm{Rep}_Z(\alpha,\cdot))] = \mathbb{E}_{\mathbb{Q}_Z^{\alpha}}[f]. \tag{3.12}$$

For datapoint x, we can transform L elements $\{\epsilon_i^{(l)}\}_{l=1}^L$ sampled from E according to $\mathbb{P}_E$ into latent variables from Z given by $z^{(l)} = \mathrm{Rep}_Z(\boldsymbol{\alpha}^{(\psi)}(x), \epsilon^{(l)})$. These latent variables have the property to be distributed according to $\mathbb{Q}_Z^{\boldsymbol{\alpha}^{(\psi)}(x)}$. Thus, these transformed latent values can be used for the estimation of the ELBO in Equation (3.8).

3.4 Variational autoencoder training

Given a dataset $\mathcal{X}$ of N independent and identically distributed outcomes obtained from an experiment we can train a variational autoencoder that can recover the suitable probability distributions that will optimize the evidence lower bound. A first step before training such autoencoder is to define the candidate latent space together with the prior and the parametric families of encoding and decoding distributions. These preliminary assumptions are summarized in the following list:

1. **Latent space:** The choice of the latent space Z is motivated by the goal of finding a suitable set of unobserved outcomes that can explain the observations in $\mathcal{X}$. Some of the important characteristics of the latent space that are taken into account are dimensionality and the intrinsic geometry which determines the overall structure of Z.

2. **Prior distribution:** The prior distribution $\mathbb{P}_Z$ as stated in Section 3 is chosen as distribution from a parametric family of distributions. It incorporates any assumptions on the overall structure of the latent space and stays fixed for the whole training of the variational autoencoder.

3. **Family of encoding distributions:** The family $\mathcal{Q}_Z^A$ is picked for the approximation to the underlying posterior $\mathbb{P}_{Z|x}$. More complex families can provide better approximations but can result in longer computations. The reparametrization function needed for training is also defined with respect to this parametric family, see Sectioon 3.

4. **Family of decoding distributions:** A suitable family $\mathcal{P}_X^B$ is chosen such that its complexity provides the sufficient expressiveness for data reconstruction with respect to the latent variables.

After stating the necessary assumptions we can train the neural networks of the variational autoencoder by performing several training steps in which the ELBO is optimized via stochastic gradient ascent. The algorithm for training a variational autoencoder from a dataset $\mathcal{X}$ is described as follows:

```
Input: $\mathcal{X} = \{x_i\}_{i=1}^N$
Result: $(\psi^*, \xi^*)$
Initialize $(\psi, \xi)$;
repeat
    for $x \in \mathcal{X}$ do
        Obtain $\boldsymbol{\alpha}^{(\psi)}(x_i)$;
        for $l = 1, 2, \ldots, L$ do
            Sample $\epsilon^{(l)}$ according to $\mathbb{P}_{\mathbb{E}}$;
            Obtain sampled latent variable as $z^{(l)} = \mathrm{Rep}_Z(\boldsymbol{\alpha}^{(\psi)}(x), \epsilon^{(l)})$;
            Obtain $\boldsymbol{\beta}^{(\xi)}(z^{(l)})$;
        end
        Calculate $-\nabla_{\psi,\xi}\tilde{\mathscr{L}}(x, \psi, \xi)$;
    end
    Calculate $-\nabla_{\psi,\xi}\tilde{\mathscr{L}}(\mathcal{X}, \psi, \xi)$;
    Update $(\psi, \xi)$ according to the estimate of $-\nabla_{\psi,\xi}\tilde{\mathscr{L}}(\mathcal{X}, \psi, \xi)$ ;
until Convergence of $(\psi, \xi)$;
```

Algorithm 1: Variational autoencoder training algorithm

Note: To avoid notational cluttering in the next sections we will suppress the explicit dependance of the encoding and decoding neural networks on the internal weights ψ, ξ. Moreover, we will distinguish a trained neural with an asterisk such that $\boldsymbol{\alpha}^* \equiv \boldsymbol{\alpha}^{(\psi^*)}$ and $\boldsymbol{\beta}^* \equiv \boldsymbol{\beta}^{(\xi^*)}$.

3.5 Neural network architecture

The chosen architecture for the encoding and decoding neural network throughout this thesis is very simple. The encoding neural network takes an input datapoint x and produced the parameters for the encoding distribution α with n dense hidden layers. On the other hand, the decoding neural network takes an input latent variable z and produces the decoding distribution parameter β with the same number of n dense layers. For the decoding and encoding neural networks the number of neurons is fixed to $D/3$ with D the dimensions of the input data. See Figure 3.1 for a representation of the neural network architectures.

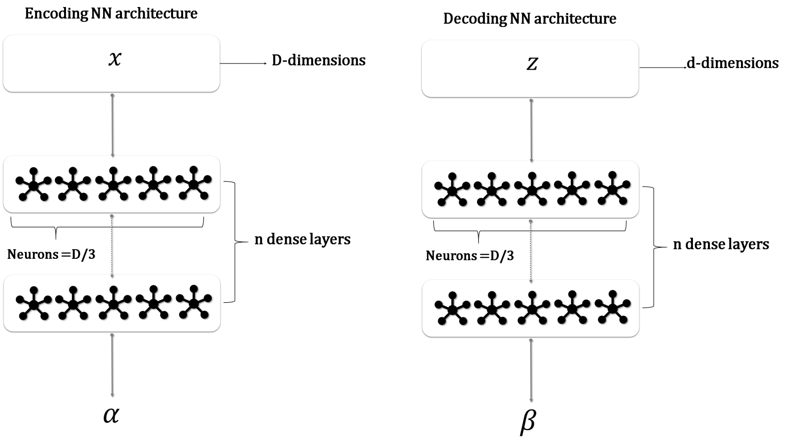

Figure 3.1: Encoding and decoding neural network architecture. The parameters are the number of hidden dense layers n and the number of neurons in each layer determined by the size of the input data dimension D.

Chapter 4

Baseline Variational Autoencoder

For a given dataset $\mathcal{X} = \{x_i\}_{i=1}^N$ of N independent and identically distributed datapoints we want to test whether a variational autoencoder is capable of recovering the underlying latent variables responsible for the generation of $\mathcal{X}$. As stated in the previous section, the variational autoencoder can be interpreted as a method for encoding data in terms of the latent variables from latent space Z.

In this chapter we present a baseline variational autoencoder that incorporates the same assumptions stated by D.P. Kingma and M. Welling [18]. The latent space Z can be identified with the d-dimensional Euclidean space $Z = \mathbb{R}^d$. The prior is chosen to be the standard normal distribution over $\mathbb{R}^d$ with probability density P_Z with respect to the Lebesgue measure $\mathcal{L}^d$ given by

$$P_Z(z) = \left(\frac{1}{\sqrt{(2\pi)^d}}\right) \exp\left(\frac{-\|z\|_2^2}{2}\right). \tag{4.1}$$

For the encoding $\mathcal{Q}_Z^A$ and decoding $\mathcal{P}_X^B$ parametric families of distributions we propose a family of normal probability distributions. This choice determines the ELBO estimation divided into the likelihood and the Kullback-Leibler regularization term. The details are presented in the next sections.

4.1 Encoding family & Kullback-Leibler regularization

Given that we assume that the latent space can be identified with the d-dimensional Euclidean space $Z = \mathbb{R}^d$, the encoding distribution used to approximate the posterior $\mathbb{P}_{Z|x}$ is chosen to be a member of the parametric family of normal distributions $\mathcal{Q}_Z^A$ defined as

$$\mathcal{Q}_Z^A = \left\{ \mathbb{Q}_Z^{(\mu,\Sigma)} \in \mathcal{P}_Z \;\middle|\; Q_Z^{(\mu,\Sigma)} = \frac{1}{\sqrt{(2\pi)^d \det(\Sigma)}} \exp\left(-\frac{1}{2}(z-\mu)^T \Sigma^{-1}(z-\mu)\right) \right\}. \tag{4.2}$$

Here $Q_Z^{(\mu,\Sigma)}$ is the probability density of the distribution $\mathbb{Q}_Z^{(\mu,\Sigma)}$ with respect to the Lebesgue measure $\mathcal{L}^d$. The parameters for each member of this family correspond to a mean vector $\mu \in \mathbb{R}^d$ and a covariance matrix $\Sigma \in \mathbb{R}^{d\times d}$. Thus, the set of possible parameters A corresponds to the pairs of mean and standard deviation

$$A = \{(\mu, \Sigma) \in \mathbb{R}^d \times \mathbb{R}^{d\times d}\}. \tag{4.3}$$

With these particular assumptions for the prior and encoding distribution, the Kullback-Leibler regularization term can be calculated analytically and does not require a Monte Carlo estimation. It corresponds to

$$\mathrm{KL}(\mathbb{Q}_Z^{(\mu,\Sigma)}||\mathbb{P}_Z) = \frac{1}{2}\left(\mathrm{tr}(\Sigma) + \|\mu\|_2^2 - \log\left(\det(\Sigma)\right) - d\right). \tag{4.4}$$

We will now analyze the case in which we restrict the parametric family $\mathcal{Q}_Z$ to distributions with diagonal covariance matrix. The elements in the diagonal are represented by the variance vector $\sigma^2 \in (\mathbb{R}_0^+)^d$. In this case, the Kullback-Leibler regularization term is simplified to

$$\mathrm{KL}(\mathbb{Q}_Z^{(\mu,\sigma^2)}||\mathbb{P}_Z) = \frac{1}{2}(\|\sigma^2\|_1 + \|\mu\|_2^2 - \|\log\,(\sigma^2)\|_1 - d). \tag{4.5}$$

In the next section we will approximate the likelihood term of the ELBO via Monte Carlo estimation. We can use the reparametrization trick introduced in Section 3.3 to obtain samples of the latent space Z according to $\mathbb{Q}_Z^{(\mu,\sigma^2)}$ with a reparametrization function $\mathrm{Rep}_Z : A \times E \mapsto Z$,

$$\mathrm{Rep}_Z((\mu,\sigma^2),\epsilon) = \mu + \epsilon \odot \sqrt{\sigma^2}. \tag{4.6}$$

The operator $\odot$ corresponds to the entry-wise Hadamard product and the auxiliary value $\epsilon \in E$ is sampled according to the standard normal distribution over $E = \mathbb{R}^d$. Finally, we will denote as $\boldsymbol{\mu_Z} : X \mapsto \mathbb{R}^d$ and $\boldsymbol{\sigma_Z^2} : X \mapsto (\mathbb{R}_0^+)^d$ the neural networks that calculate the corresponding encoding distribution parameters.

4.2 Decoding family & reconstruction error

The reconstruction error term of the negative ELBO is determined by the parametric family $\mathcal{P}_X^B$ used to obtain the decoding distribution $\mathbb{P}_{X|z}$ for a given latent variable $z \in Z$. For all the proposed variational autoencoders presented in this thesis we consider this family as the family of parametric normal distributions over data space $X = \mathbb{R}^D$,

$$\mathcal{P}_X^B = \left\{ \mathbb{P}_X^{(\mu,\Sigma)} \in \mathcal{P}_X \;\middle|\; P_X^{(\mu,\Sigma)} = \frac{1}{\sqrt{(2\pi)^D \det(\Sigma)}} \exp\left(-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right)\right\}. \tag{4.7}$$

Here $P_X^{(\mu,\Sigma)}$ is the probability density of $\mathbb{P}_X^{(\mu,\Sigma)}$ with respect to the Lebesgue measure $\mathcal{L}^D$. The parameter set is given by the pair of mean $\mu \in \mathbb{R}^D$ and covariance matrix $\Sigma \in \mathbb{R}^{D\times D}$ such that

$$B = \{(\mu,\Sigma) \in \mathbb{R}^D \times \mathbb{R}^{D\times D}\} \tag{4.8}$$

In the estimation of the negative ELBO we need to calculate the term corresponding to reconstruction error of a datapoint $x \in X$. This term involves the negative logarithm of the decoding distribution which corresponds to

$$-\log P_X^{(\mu,\Sigma)}(x) = \frac{1}{2}\left[D\log\,(2\pi) + \;\log\,(\det\Sigma) + (x-\mu)^T\Sigma^{-1}(x-\mu)\right]. \tag{4.9}$$

For the baseline variational autoencoder we will focus on the particular case in which we restrict $\mathcal{P}_X^B$ to distributions with a fixed diagonal covariance matrix $\Sigma = \sigma_X^2 I_D$ where $\sigma_X \in \mathbb{R}^+$ corresponds to a fixed standard deviation. The effects of modifying this parameter when training a variational autoencoder are presented in the next section. For datapoint $x \in X$, the logarithm of the decoding distribution becomes

$$-\log P_X^{(\mu,\sigma_X^2)}(x) = \frac{1}{2}\left[\frac{1}{\sigma_X^2}\|x-\mu\|_2^2 + D\log\,(2\pi\sigma_X^2)\right]. \tag{4.10}$$

The reconstruction error of the negative ELBO can be estimated via the Monte Carlo method introduced in Section 3.2 by sampling L elements from Z according to $\mathbb{Q}_Z^{(\mu,\sigma^2)}$. For each of the L

samples we calculate the parameters of the approximate posterior with the neural networks $\boldsymbol{\mu_Z}$ and $\boldsymbol{\sigma^2_Z}$. The location parameter of the decoding distribution $\mathbb{P}_X^{(\mu,\sigma^2)}$ is calculated with the neural network $\boldsymbol{\mu_X} : Z \mapsto \mathbb{R}^D$ while its variance is the fixed value $\sigma_X^2 \in \mathbb{R}^+$. The approximate likelihood term is expressed as

$$-\mathbb{E}_{\mathbb{Q}_Z^{(\boldsymbol{\mu_Z}(x),\boldsymbol{\sigma^2_Z}(x))}}\left[\log P_X^{(\boldsymbol{\mu_X}(\cdot),\sigma_X^2)}(x)\right] \approx \frac{1}{2L}\sum_{l=1}^{L}\frac{1}{\sigma_X^2}\|x-\boldsymbol{\mu_X}(z^{(l)})\|_2^2 + D\log\left(2\pi\sigma_X^2\right). \tag{4.11}$$

The L samples from Z space $\{z^{(l)}\}_{l=1}^{L}$ according to the posterior approximate can be obtained via the reparametrization trick discussed in the previous section by sampling L terms $\{\epsilon^{(l)}\}_{l=1}^{L}$ from $E = \mathbb{R}^d$ according to the standard normal distribution. Thus, each latent variable $z^{(l)}$ is calculated as

$$z^{(l)} = \mathrm{Rep}_Z((\boldsymbol{\mu_Z}(x), \boldsymbol{\sigma^2_Z}(x)), \epsilon^{(l)}) = \boldsymbol{\mu_Z}(x) + \epsilon^{(l)} \odot \sqrt{\boldsymbol{\sigma^2_Z}(x)} \tag{4.12}$$

4.3 The σ_X parameter as a weighting term

The parameter σ_X of the decoding normal distribution has a special importance for the negative ELBO minimization as it has been presented in [14] and [8]. Recall that in Section 2.4 we have divided the negative ELBO into two terms: the reconstruction error and the Kullback-Leibler regularization. In the baseline variational autoencoder, the reconstruction error for a latent variable z and datapoint x includes the logarithm of the decoding distribution

$$-\log P_X^{(\boldsymbol{\mu_X}(z),\sigma_X^2)}(x) = \frac{1}{2\sigma_X^2}\|x-\boldsymbol{\mu_X}(z)\|_2^2 + D\log(2\pi\sigma_X^2). \tag{4.13}$$

The first term in the logarithm corresponds to a squared distance between an input datapoint x and the mean value for the decoding distribution $\boldsymbol{\mu_X}(z)$ for a latent variable z. If we consider the calculated mean $\boldsymbol{\mu_X}(z)$ as the average reconstructed datapoint associated to the latent variable z, then minimizing the reconstruction error for the baseline variational autoencoder is minimizing the squared error between the reconstruction and the original datapoint. The second term of the logarithm is constant for a fixed value of σ_X^2, minimizing the negative ELBO with respect to the neural network weights is equivalent to minimizing the quantity

$$\frac{1}{\sigma_X^2}\mathbb{E}_{\mathbb{Q}_Z^{(\boldsymbol{\mu_Z}(x),\boldsymbol{\sigma}^2_{\boldsymbol{Z}}(x))}}\left[\frac{1}{2}\|x-\boldsymbol{\mu_X}(\cdot)\|_2^2\right] + \mathrm{KL}\left(\mathbb{Q}_Z^{(\boldsymbol{\mu_Z}(x),\boldsymbol{\sigma}^2_{\boldsymbol{Z}}(x))}||\mathbb{P}_Z\right) \tag{4.14}$$

In this case we can notice that the parameter $1/\sigma_X^2$ acts as a weight that modulates the contributions of the reconstruction error with respect to the Kullback-Leibler regularization term as discussed by C. Doersch in [8], an alternative interpretation in terms of a constrained optimization problem is presented by I. Higgins et al. in the β-VAE of [14]. Increasing the value of σ_X results in a higher contribution of the Kullback-Leibler regularization, while decreasing σ_X makes the reconstruction error term more important.

The intuition behind the effects of parameter σ_X is connected to its relationship with the decoding distribution. Recall that the decoding distribution $P_X^{(\boldsymbol{\mu_X}(z),\sigma_X^2)}$ provides for a latent variable z a distribution of all possible datapoints that can be reconstructed from it. Decreasing the value of σ_X conditions the normal decoding distribution to be narrower and to produce more certain reconstructions, therefore the reconstruction error has a higher contribution. On the other hand, increasing σ_X conditions the decoding distribution to produce less certain reconstructions and the Kullback-Leibler regularization becomes more important.

The effects of the parameter σ_X will be explored throughout this thesis for the different datasets used.

Chapter 5

Benchmark Dataset

In order to test the baseline variational autoencoder we have devised a simple dataset $\mathcal{X}$ of D-dimensional vectors generated artificially by sampling elements from a known generative model $\mathbb{P}_{X\times Z}$. In this simple case, a datapoint is generated from a latent variable sampled according to a distribution $\mathbb{P}_Z$. The datapoint $x \in \mathbb{R}^D$ is obtained from the sampled latent variable by calculating a D-dimensional vector according to $F : Z \mapsto \mathbb{R}^D$ and adding some Gaussian noise.

In this chapter we introduce the process used to generate dataset $\mathcal{X}$ in terms of F. The function F is chosen in such a way that it induces a circular structure for the generated data. After describing the dataset we present the results obtained by training the baseline variational autoencoder by assuming a 2-dimensional Euclidean latent space $Z = \mathbb{R}^2$.

5.1 Dataset description

As a first study case, we have considered an artificial experiment with an observable data space X given by D-dimensional vectors $X = \mathbb{R}^D$. Each observation x in the dataset $\mathcal{X}$ is generated from a sampled element of the interval $\Phi = [-\pi, \pi)$. We will later identify the set Φ as the latent space from the underlying Bayesian model.

To generate the observations we have used the function $F : \Phi \mapsto X$ which calculates the average datapoint for a given phase $\varphi \in \Phi$. The function F is defined as $F(\varphi) = (f_j(\varphi))_{j=1}^D$ where each individual function $f_j : \Phi \mapsto \mathbb{R}$ corresponds to

$$f_j(\varphi) = \sin\left(\frac{2\pi j}{D} + \varphi\right). \tag{5.1}$$

Notice that for each of these individual functions, the value of $\varphi \in \Phi$ can be considered as the phase of a discrete sine function over the interval $[0, 1]$ with angular frequency 2π . Hence, for a particular datapoint $x \in X$ we will refer to $\varphi \in \Phi$ as its corresponding underlying phase. A representation of the function F for the phase $\varphi = 0$ is shown in Figure 5.1.

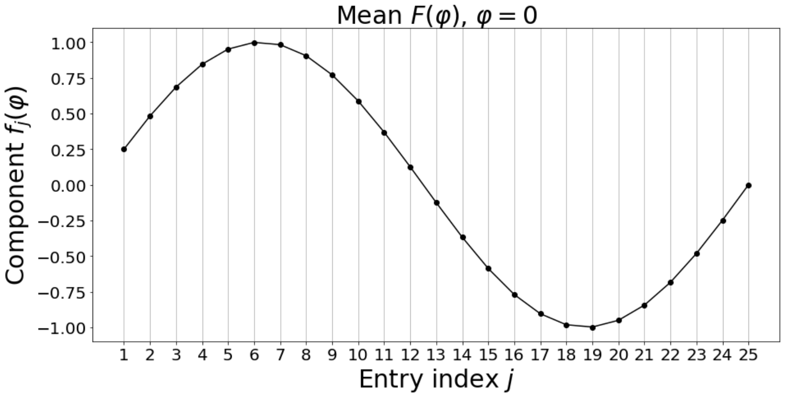

Figure 5.1: Example of function $F(\varphi)$ for phase $\varphi = 0$ with $D = 25$. The horizontal axis corresponds to the entry index i of function F, the vertical axis shows the value of $f_i(\varphi)$. The line indicates the shape of the continuous sinusoid function.

Consider the probability space $(\Phi, \mathcal{F}_\Phi, \mathbb{P}_\Phi)$ with $\mathbb{P}_\Phi$ the uniform probability measure over Φ. Each element x_i in the dataset $\mathcal{X}_\sigma = \{x_i\}_{i=1}^N$, where $\sigma \in \mathbb{R}^+$ represents the amount of Gaussian noise added, is obtained by following the process consisting of the steps:

1. Sample a phase φ_i from Φ according to the uniform probability distribution $\mathbb{P}_\Phi$ over Φ.

2. Sample datapoint x_i from a normal distribution with mean $F(\varphi_i)$ and diagonal covariance matrix $\Sigma = \sigma^2 I_d$ with $\sigma \in \mathbb{R}^+$.

Translating this setting to the Bayesian model context presented in Chapter 2, we consider the set Φ as the latent space $Z = \Phi$ of a Bayesian model with a uniform probability measure $\mathbb{P}_\Phi$ defined in terms of the probability density with respect to the $\mathcal{L}^1$ measure,

$$P_\Phi(\varphi) = \frac{1}{2\pi}. \tag{5.2}$$

For a given phase $\varphi \in \Phi$, the conditional distribution $\mathbb{P}_{X|\varphi}$ from which data is sampled corresponds to the normal distribution with location parameter $F(\varphi)$ and variance determined by the standard deviation $\sigma \in \mathbb{R}^+$ which can be interpreted as the amount of noise added to the function F. The probability density of this conditional distribution with respect to the $\mathcal{L}^D$ measure is given by

$$P_{X|\varphi}(x) = \frac{1}{(\sigma\sqrt{2\pi})^D}\exp\left(-\frac{\|x - F(\varphi)\|_2^2}{2\sigma^2}\right).$$

According to Equation (2.6) presented in Chapter 2, the probability density of the generative model $\mathbb{P}_{X\times\Phi}$ for a given datapoint $x \in X$ and a phase $\varphi \in \Phi$ is given by

$$P_{X\times\Phi}(x, \varphi) = P_{X|\varphi}(x) \cdot P_\Phi(\varphi) = \frac{1}{2\pi(\sigma\sqrt{2\pi})^D}\exp\left(-\frac{\|x - F(\varphi)\|_2^2}{2\sigma^2}\right).$$

From the generative model that we have described we will produce different datasets that can be characterized in terms of the value σ used for the conditional distribution $\mathbb{P}_{X\times\Phi}$. The dataset with N datapoints generated according to a generative model with parameter σ will be denoted as $\mathcal{X}_\sigma$. In Figure 5.2 we present an example datapoint generated from the latent phase $\varphi = 0$ for the corresponding datasets $\mathcal{X}_1$, $\mathcal{X}_{0.1}$ and $\mathcal{X}_{0.01}$.

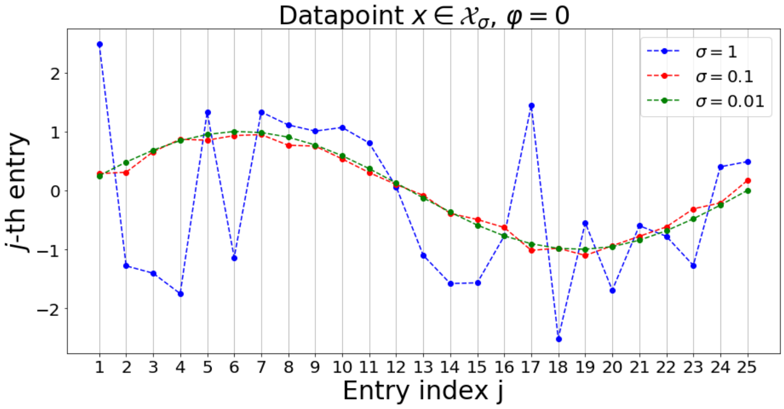

Figure 5.2: Example datapoint x from dataset $\mathcal{X}_\sigma$ with phase $\varphi = 0$, $D = 25$ and $\sigma \in \{1, 0.1, 0.01\}$. The horizontal axis corresponds to the entry index j and the vertical axis shows the j-th datapoint entry. The dotted lines shown the interpolated curve form by the datapoint entries.

5.2 Structure of the benchmark dataset

The generative model that produced the dataset $\mathcal{X}_\sigma$ is determined entirely by the function F and the underlying probability distribution. Notice the periodicity of F with respect to the phases such that $F(0) = \lim_{\theta \to 2\pi} F(\theta)$. In an intuitive way, we can expect that the periodicity of the function F induces a circular structure upon the dataset $\mathcal{X}$.

We consider that the encoding distribution learned in a variational autoencoder aims at identifying the appropriate latent variables that explain each datapoint via the use of neural networks. According to M. Carreira-Perpinam [5] for the cases in which the dataset has a circular structure, in order to recover a continuous encoding of the datapoints into the latent space it is needed a periodic set of latent variable representations.

The phases in the latent space $Z = \Phi$ can be embedded into $\mathbb{R}^2$ to enforce the periodicity of the latent variables. In particular, we can embed the phases from Φ into the unit circle within $\mathbb{R}^2$ via the function $\mathrm{Emb}_{\mathbb{R}^2} : \mathbb{R} \mapsto \mathbb{R}^2$, defined as

$$\mathrm{Emb}_{\mathbb{R}^2}(\varphi) = (\cos(\varphi), \sin(\varphi)) \tag{5.3}$$

For the purpose of visualizing such embedding we will consider the set Φ_{Vis} of 100 phases corresponding to the regular partition of $\Phi = [-\pi, \pi)$ given by

$$\Phi_{\mathrm{Vis}} = \left\{ -\pi + \frac{2\pi i}{100} \right\}_{i=0}^{99} \tag{5.4}$$

In Figure 5.3 we present the embedded phases from Φ_{Vis} in $\mathbb{R}^2$ together with the corresponding value of $F(\varphi)$. It is important to realize that such continuous latent variable structure can only be obtained from at least a 2-dimensional space. Therefore, in the next section we will describe the results of training a variational autoencoder with the benchmark dataset by assuming a latent space $Z = \mathbb{R}^2$ with the purpose of identifying a suitable periodic encoding of the datapoints into the latent space.

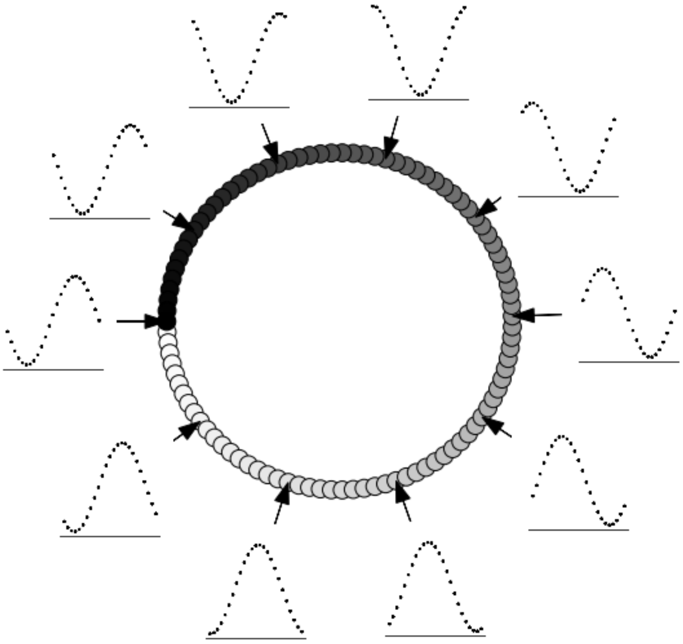

Figure 5.3: Embedding of phases $\varphi \in \Phi_{\mathrm{Vis}}$ into $\mathbb{R}^2$ with the embedding function $\mathrm{Emb}_{\mathbb{R}^2}(\varphi)$. Each point in the unitary circle represents an embedded phase which is responsible of generating a datapoint according to $x = F(\varphi)$.

5.3 Baseline variational autoencoder training

For training the baseline variational autoencoder, we generated three datasets $\mathcal{X}_\sigma$ with different values of σ denoted by $\mathcal{X}_{0.01}$, $\mathcal{X}_{0.1}$ and $\mathcal{X}_1$. Each dataset consists of $N = 10000$ datapoints obtained via the generative model described in the first section of this chapter. The number of latent samples is chosen as $L = 1$.

For each dataset $\mathcal{X}_\sigma$ we have trained three different variational autoencoders with the decoding distribution parameter $\sigma_X \in \{0.01, 0.1, 1\}$. Moreover, due to the random initialization of the neural network weights the outcome of each training is variable. For each pair of dataset $\mathcal{X}_\sigma$ and σ_X we have repeated five times the neural network training to evaluate the reproducibility of the results.

The variational autoencoders have been trained for 10^5 epochs via stochastic gradient descent using the Adam optimizer with initial learning rate of 10^{-4}, $\beta_1 = 0.9$ and $\beta_2 = 0.999$. The architecture of the neural networks is shown in Section 3.5. Two dense layers where used for both the encoder and decoder neural networks with $D/3 = \lfloor 25/3 \rfloor$ neurons each.

5.4 Qualitative results

To analyze and evaluate the trained variational autoencoders we generated an extra dataset $\mathcal{X}_{\text{Vis}}$ from the regular partition Φ_{Vis} described in Equation (5.4),

$$\mathcal{X}_{\text{Vis}} = \left\{ x \in \mathbb{R}^D \middle| \; x = F(\varphi) \; ; \; \varphi \in \Phi_{\text{Vis}} \right\}. \tag{5.5}$$

This auxiliary dataset corresponds to noiseless datapoints generated from the function F evaluated at equally spaced phases within $\Phi = [-\pi, \pi)$. It is used to effectively visualize the learned approximate posterior $\mathbb{Q}_Z^{(\boldsymbol{\mu_Z}(x), \boldsymbol{\sigma^2}(x))}$ of the trained variational autoencoder characterized by the location and scale parameters calculated with the neural networks $\boldsymbol{\mu_Z}$ and $\boldsymbol{\sigma^2_Z}$ respectively.

We will refer to the value $\boldsymbol{\mu_Z}(x)$ as the latent variable representation of datapoint x. Since each datapoint $x \in \mathcal{X}_{\text{Vis}}$ can be associated to the underlying phase that generated it, our purpose is to study the learned relationships between the latent variable representations for datapoints with consecutive phases. As we stated in the previous section, the function F has induced circular structure upon each dataset, thus we are expecting to obtain a periodic latent variable representation.

The approximate posterior distribution $\mathbb{Q}_Z^{(\boldsymbol{\mu_Z}(x), \boldsymbol{\sigma^2_Z}(x))}$ for a datapoint x can be visualized with respect to the high probability areas in latent space associated to each datapoint $x \in \mathcal{X}_{\text{Vis}}$. In the case of normal encoding distributions these areas are visualized with an ellipse centered at $\boldsymbol{\mu_Z}(x) \in \mathbb{R}^2$. The major and minor axis correspond to the entries of the standard deviation vector $\sqrt{\boldsymbol{\sigma^2_Z}(x)} \in (\mathbb{R}^+)^2$. For each datapoint x, the corresponding ellipses are given by the set of points in $Z = \mathbb{R}^2$ defined as

$$\left\{ z \in \mathbb{R}^2 \;\middle|\; \|(z - \boldsymbol{\mu_Z}(x)) \oslash \sqrt{\boldsymbol{\sigma^2_Z}(x)}\|_2 \leq 1 \right\}. \tag{5.6}$$

Here the operation $\oslash$ corresponds to the Hadamard element-wise division. The representation of the approximate posterior for datapoints in $\mathcal{X}_{\text{Vis}}$ is presented in Figure 5.4. Each plot corresponds to the obtained approximate posterior for dataset $\mathcal{X}_\sigma$ (Rows) and parameter σ_X (Columns).

Each of the plots in Figure 5.4 shows that the learned latent representations of the datapoints in dataset $\mathcal{X}_{\text{Vis}}$ forms a loop with respect to the underlying consecutive phases that generated the data. Thus, the latent representation has a periodic behavior with respect to the underlying phases as expected but the markers do not necessarily lie on a circle in latent space.. It is important to notice the different shapes obtained by varying the amount of noise in each dataset $\mathcal{X}_\sigma$ and the proposed decoding distribution parameter σ_X.

First of all consider a row of plots in Figure 5.4 with a fixed dataset $\mathcal{X}_\sigma$ and a parameter $\sigma_X \in \{1, 0.1, 0.01\}$. As it was discussed in Section 4.3, the value of σ_X acts a weighing factor within the negative ELBO by modulating the contribution of the reconstruction error and the Kullback-Leibler regularization.

The combination of forcing more precise reconstructions and the decreased effect of the Kullback-Leibler regularization results in posterior approximates that for a given datapoint x produce more precise latent embeddings. The approximate posterior $\mathbb{Q}_Z^{(\boldsymbol{\mu_Z}(x), \boldsymbol{\sigma^2}(x))}$ resemble less the prior having a smaller standard deviation $\boldsymbol{\sigma^2_Z}(x)$ and therefore smaller high-probability ellipses.

On the other hand when we increase the value of σ_X, we allow less precise reconstructions of datapoints with respect to the latent variables. The Kullback-Leibler regularization has a higher contribution and the encoding distribution has a higher standard deviation represented with bigger ellipses which overlap. This overlapping represents the uncertainty of the encoder distribution to determine for a given datapoint the corresponding latent variable that generated it.

We can observe the effects of using the datasets $\mathcal{X}_{0.01}, \mathcal{X}_{0.1}$ and $\mathcal{X}_1$ with different noise levels. The plots shown in each column of Figure 5.4 present the recovered structure of the latent variables for a fixed parameter σ_X across each dataset. We can identify that for datasets $\mathcal{X}_{0.01}$ and $\mathcal{X}_{0.1}$ the learnt latent representations have a similar structure.

For the dataset with the highest noise $\mathcal{X}_1$ and the parameters $\sigma_X \in 0.1, 0.01$, there is still a noticeable cycle in the latent representation for datapoints with consecutive phases, but the structure appears to be discontinuous with an intricate shape and small standard deviation (ellipses have a very small scale). The variational autoencoders with small values of $\sigma_X \in \{0.1, 0.01\}$ have decoding distributions that enforce less variability of the reconstructed datapoints with respect to the latent variables. It seems that due to the variability of the input dataset $\mathcal{X}_\sigma$ the encoding distribution is less capable of identifying the shape of the underlying latent structure.

It is interesting to notice that for this noisy dataset $\mathcal{X}_1$, if we choose the value $\sigma_X = 1$, the structure of the latent variables becomes smooth and circular. Therefore, by allowing more uncertainty in the data reconstructions we can recover a latent structure that is closer to the expected circular shape.

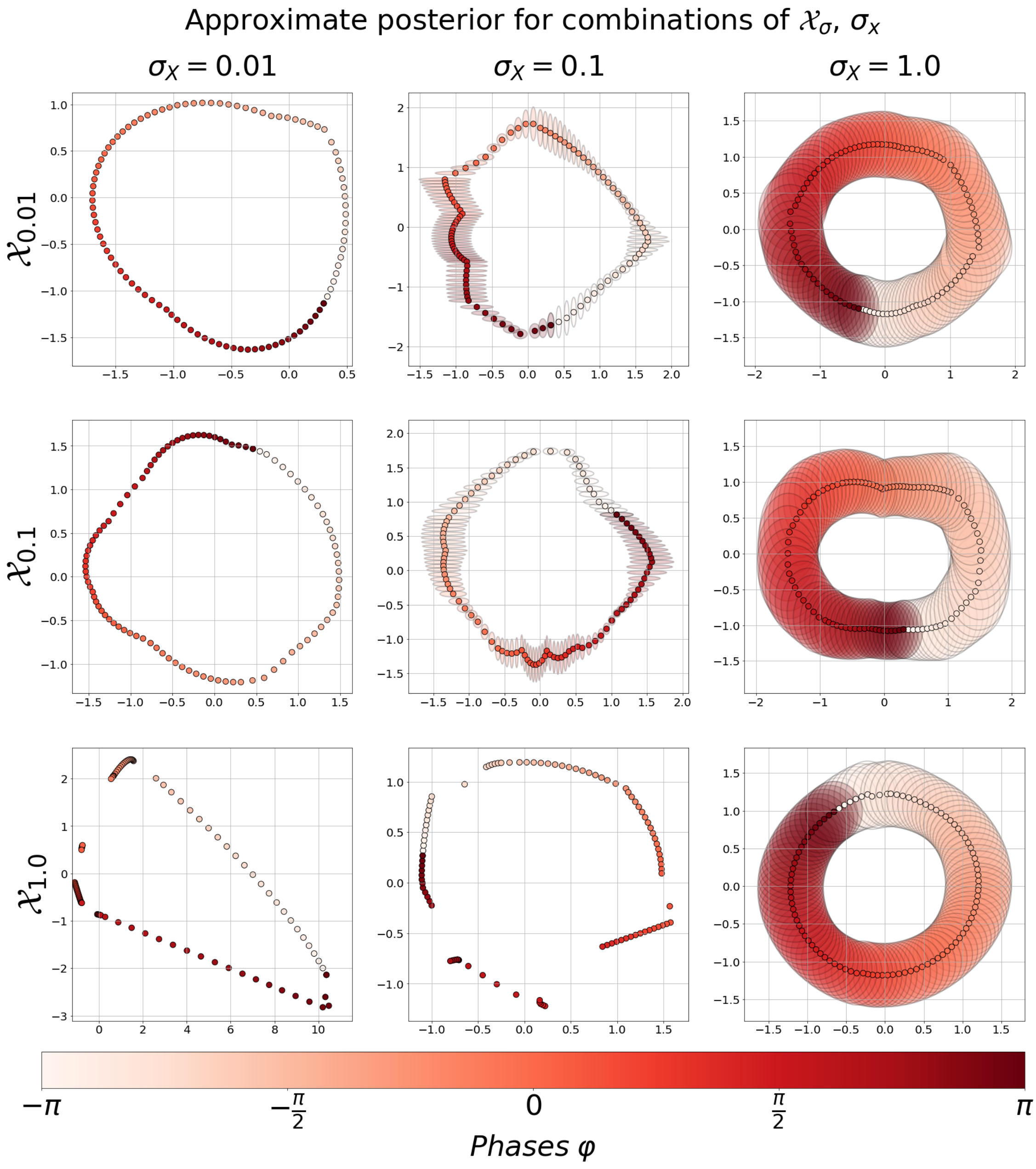

Figure 5.4: Representation of the approximate posterior $\mathbb{Q}_Z^{(\boldsymbol{\mu_Z}(x),\boldsymbol{\sigma^2}(x))}$ obtained by training the baseline variational autoencoder for the different combinations of input dataset $\mathcal{X}_\sigma$ (Rows) and decoding distribution parameter σ_X (Columns). The markers in each plot represents the calculated values for the encoding distribution's mean $\boldsymbol{\mu_Z}(x)$ for each datapoint $x \in \mathcal{X}_{\text{Vis}}$. The hue of each marker represent the underlying phases $\varphi \in \Phi_{\text{Vis}}$ corresponding to each datapoint $x \in \mathcal{X}_{\text{Vis}}$ and helps identify the relationships between datapoints with consecutive phases. The ellipses represent the high probability regions of the encoding distribution for each datapoint determined by $\boldsymbol{\sigma^2_Z}(x)$. In some cases these ellipses have a smaller size compared to the size of the mean markers and can not be observed.

5.5 Quantitative results

For each of the trained variational autoencoders we calculated the quantities corresponding to the negative evidence lower bound and the squared error. In order to test the reproducibility of the results for each pair of dataset $\mathcal{X}_\sigma$ and parameter σ_X five variational autoencoders are trained. Each of the quantities presented in Table 5.1 correspond to the average quantities over these repetitions, the standard deviation for each quantity is also presented. Now we describe each of the quantities presented in Table 5.1 for a single repetition.

In order to analyze the reconstruction capabilities of the variational autoencoder we measure the squared error SE : $x \mapsto \mathbb{R}_0^+$ between an input datapoint $x \in \mathcal{X}_{\text{Vis}}$ with respect to the reconstructed datapoint generated by the variational autoencoder. The reconstructed datapoint is obtained by first encoding the datapoint in latent space with $\boldsymbol{\mu_Z}(x)$ and then decoding this latent variable with the neural network $\boldsymbol{\mu_X}(\boldsymbol{\mu_Z}(x))$. The squared error corresponds to

$$\text{SE}(x) = \|x - \boldsymbol{\mu_X}(\boldsymbol{\mu_Z}(x))\|_2^2 \tag{5.7}$$

To evaluate the reconstruction capabilities for the entire dataset $\mathcal{X}_{\text{Vis}}$ we aggregate the squared error results by averaging them with respect to the cardinality of the dataset $|\mathcal{X}_{\text{Vis}}| = 100$. The mean squared error MSE of $\mathcal{X}_{\text{Vis}}$ corresponds to

$$\text{MSE}(\mathcal{X}_{\text{Vis}}) = \frac{1}{|\mathcal{X}_{\text{Vis}}|} \sum_{x \in \mathcal{X}_{\text{Vis}}} \text{SE}(x). \tag{5.8}$$

In this section we also present the values of the negative ELBO averaged over the complete dataset. The negative ELBO is formed of the reconstruction error and the Kullback-Leibler regularization. In the baseline variational autoencoder, the averaged reconstruction error is estimated using the Monte Carlo method described in Section 3.2 by sampling $L = 100$ latent variables according to the reparametrization trick presented in Section 3.3. The reconstruction error averaged over the entire dataset is calculated as

$$\frac{1}{L \cdot |\mathcal{X}_{\text{Vis}}|} \sum_{x \in \mathcal{X}_{\text{Vis}}} \sum_{l=1}^{L} \frac{1}{\sigma_X^2} \|x - \boldsymbol{\mu_X}(z^{(l)})\|_2^2 + \frac{D}{2} \log\left(2\pi\sigma_X^2\right). \tag{5.9}$$

The averaged Kullback-Leibler regularization is calculated according to

$$\frac{1}{2 \cdot |\mathcal{X}_{\text{Vis}}|} \sum_{x \in \mathcal{X}_{\text{Vis}}} \|\boldsymbol{\sigma_Z^2}(x)\|_1 + \|\boldsymbol{\mu}(x)\|_2^2 - \|\log\left(\boldsymbol{\sigma_Z^2}(x)\right)\|_1 - 2 \tag{5.10}$$

Therefore the negative ELBO averaged over dataset $\mathcal{X}_{\text{Vis}}$ corresponds to the sum of the two aforementioned terms. The reproducibility of all the described quantities is tested by averaging the values over the five trained neural networks. Uncertainty intervals are calculated from the standard deviation of the measurements across the five trained neural networks.

We can notice that calculated quantities for datasets $\mathcal{X}_{0.01}$ and $\mathcal{X}_{0.1}$ are the same for each value of σ_X taking into account the uncertainty intervals. This goes in accordance to the observed qualitative behavior described in the previous section where the recovered encoder distributions for both datasets present a similar behavior.

The only observable difference is that the mean squared error for the dataset with the lowest noise $\mathcal{X}_{0.01}$ and $\sigma_X = 0.01$ is lower by one order of magnitude compared to the result obtained by training with $\mathcal{X}_{0.1}$ and $\sigma_X = 0.01$.

A possible explanation can be that decreasing the value of σ_X forces the decoding distribution to produce more precise reconstructions of a datapoint for a given latent variable. Moreover,

Table 5.1: Values for the average negative ELBO, Kullback-Leibler regularization, reconstruction error and mean squared error. Each value is obtained by averaging the results of five repetitions for each corresponding variational autoencoder trained with dataset $\mathcal{X}_\sigma$ and parameter σ_X. The calculated quantities are obtained with respect to dataset $\mathcal{X}_{\text{Vis}}$. For the reconstruction error $L = 100$ samples from the latent space are taken according to the trained posterior distribution.

$\mathcal{X}_\sigma$	σ_X	**-ELBO**	**KL Regularization**	**Reconstruction**	**MSE**
$\mathcal{X}_{0.01}$	0.01	-79.90 ± 0.19	11.18 ± 0.15	-91.09 ± 0.04	$(1.59 \pm 0.64) \times 10^{-5}$
	0.1	-28.87 ± 0.03	4.95 ± 0.02	-33.81 ± 0.02	$(1.68 \pm 0.23) \times 10^{-3}$
	1	25.41 ± 0.01	1.74 ± 0.01	23.67 ± 0.01	$(4.22 \pm 0.34) \times 10^{-2}$
$\mathcal{X}_{0.1}$	0.01	-79.01 ± 0.11	11.10 ± 0.07	-90.11 ± 0.08	$(2.08 \pm 0.17) \times 10^{-4}$
	0.1	-28.74 ± 0.16	5.04 ± 0.10	-33.77 ± 0.06	$(1.52 \pm 0.17) \times 10^{-3}$
	1	25.37 ± 0.01	1.75 ± 0.01	23.62 ± 0.07	$(4.51 \pm 1.08) \times 10^{-2}$
$\mathcal{X}_1$	0.01	919.28 ± 213.64	17.53 ± 3.35	901.75 ± 215.37	$(1.99 \pm 0.43) \times 10^{-2}$
	0.1	-12.99 ± 3.09	7.79 ± 0.3	-20.79 ± 2.90	$(2.59 \pm 0.59) \times 10^{-2}$
	1	25.53 ± 0.01	1.79 ± 0.01	23.74 ± 0.01	$(4.61 \pm 0.15) \times 10^{-3}$

decreasing the value of σ_X increases the contribution of the reconstruction error to the negative ELBO and masks the Kullback-Leibler regularization term which forces the posterior to resemble the prior standard normal distribution.

On the other hand, the quantities calculated for dataset $\mathcal{X}_1$ have higher standard deviations which means that the recovered variational autoencoders have more variability. The highest variability is obtained for the variational autoencoders with $\sigma_X = 0.01$ and the least variability is observed for models with $\sigma_X = 1$. It is important to notice that for the noisy dataset $\mathcal{X}_1$, the variational autoencoder with $\sigma_X = 1$ produces reconstructions with a low mean squared error and less variability in the other calculated quantities.

As we have seen in the results obtained with the baseline variational autoencoder, we can recover from the benchmark dataset a latent representation that captures the underlying periodic latent structure. Moreover, we have analyzed the results obtained for different values of input noise within the dataset and for the parameter σ_X. It is important to note that our qualitative results for the latent representations for different values of σ_X follow the results presented in [13, 14]. Compared to these literature results we have analyzed as well the behavior of the approximate posterior's standard deviation $\boldsymbol{\sigma_Z}$. In conclusion our baseline variational autoencoder is capable of recovering periodic latent variables by producing latent representations of data with cyclic behavior. The parameter σ_X participates as a tradeoff between the shape of the recovered latent structures and the reconstruction error.

After having discussed the obtained results for the baseline variational autoencoder applied to the benchmark dataset, we will introduce in the next chapter a proof for the recovered latent structure for a special restricted and simplified case of the baseline variational autoencoder. This proof provides a very simple setting in which the recovered latent structure for the benchmark dataset is circular.

Chapter 6

Benchmark Dataset and the Simplified Baseline Variational Autoencoder

As it was presented in the previous chapter, the baseline variational autoencoder assumes an Euclidean latent space and recovers a cyclic latent variable representation for the benchmark dataset. In particular, for the benchmark dataset we have identified that in order to retrieve a circular structure for the latent variables the smallest dimension required is $d = 2$ such that $Z = \mathbb{R}^2$.

Consider the benchmark dataset $\mathcal{X}$ described in Chapter 5 where each datapoint is given in terms of the function $F : \Phi \mapsto X$ which maps a phase/latent variable $\varphi \in \Phi$ with $\Phi = [-\pi, \pi)$ into the D-dimensional Euclidean space $X = \mathbb{R}^D$ via

$$F(\varphi) = \left(\sin\left(\frac{2\pi i}{D} + \varphi\right)\right)_{i=1}^{D} \tag{6.1}$$

As it was stated, this dataset has a circular structure induced by the function F due to its periodicity. In this chapter we prove that minimizing the negative ELBO with the baseline variational autoencoder subject to certain restrictions and simplifications, enforces a circular structure of the latent variables.

6.1 Frequency domain benchmark dataset

Each of the datapoints in the benchmark dataset described in Chapter 4 corresponds to a discrete sinusoid signal with a fixed frequency and a given phase. In order to simplify our problem, we will be working with an equivalent representation of our data in the frequency domain with respect to the discrete orthonormal basis of sines and cosines through the discrete Fourier series. The i-th data entry of $x = F(\varphi)$ is obtained from the linear combination of the Fourier basis functions as

$$\sin\left(\frac{2\pi i}{D} + \varphi\right) = a_0 + \sum_{k=1}^{D-1} a_k \cos\left(\frac{2\pi k i}{D}\right) + b_k \sin\left(\frac{2\pi k i}{D}\right). \tag{6.2}$$

The discrete Fourier series takes the D-dimensional datapoint $x = F(\varphi)$ and produces an equivalent representation in terms of the $(D \times 2)$-dimensional vector $(a_k, b_k)_{k=0}^{D-1} \in \mathbb{R}^{D\times 2}$ where the coefficient pair $(a_k, b_k) \in \mathbb{R}^2$ represents the contribution of the k-th frequency [19]. Notice that the i-th datapoint entry can be decomposed by using the trigonometric formula

$$\sin\left(\frac{2\pi i}{D}+\varphi\right)=\sin(\varphi)\cos\left(\frac{2\pi i}{D}\right)+\cos(\varphi)\sin\left(\frac{2\pi i}{D}\right) \tag{6.3}$$

The previous formula shows that the coefficients of the discrete Fourier series are non zero only for the frequency component with $k=1$. Therefore, the k-th entry of the Fourier vector is given by

$$(a_k,b_k)=\begin{cases}(\sin(\varphi),\cos(\varphi)) & \text{if } k=1\\ (0,0), & \text{otherwise}\end{cases} \tag{6.4}$$

Therefore a datapoint is characterized in the frequency domain by only the components of one frequency. The non-zero components are represented with 2-dimensional vectors. We denote the obtained transformed data space as $X'=\mathbb{R}^2$. Each datapoint $x\in X'$ is obtained with respect to the phase φ via the function $\mathscr{F}:\Phi\mapsto\mathbb{R}^2$ given by

$$\mathscr{F}(\varphi)=(\sin(\varphi),\cos(\varphi)) \tag{6.5}$$

The generative model for the transformed data space corresponds to $\mathbb{P}_{X'\times\Phi}$ which can be described in terms of the normal probability density function with respect to the measure $\mathcal{L}^2\otimes\mathcal{L}^1$ with diagonal covariance matrix $\Sigma=\sigma^2 I_2$ is given by

$$P_{X'\times\Phi}(x,\varphi)=P_{X'|\varphi}(x)P_\Phi(\varphi)=\frac{1}{4\pi^2\sigma^2}\exp\left(-\frac{\|x-\mathscr{F}(\varphi)\|_2^2}{2\sigma^2}\right). \tag{6.6}$$

Which can be subdivided into a normal conditional distribution $P_{X'|\varphi}(x)$ with probability density

$$P_{X'|\varphi}(x)=\frac{1}{2\pi\sigma^2}\exp\left(-\frac{\|x-\mathscr{F}(\varphi)\|_2^2}{2\sigma^2}\right) \tag{6.7}$$

And the corresponding uniform distribution over Φ,

$$P_\Phi(\varphi)=\frac{1}{2\pi}. \tag{6.8}$$

6.2 Simplified baseline variational autoencoder

Moreover, within the baseline variational autoencoder context we propose the encoding and decoding distributions as members of parametric families of normal distributions. In this chapter we will restrict ourselves even further to the case in which the covariance matrices of the encoding and decoding distributions correspond to diagonal matrices $\Sigma_Z=\sigma_Z^2\cdot I_2$, $\Sigma_{X'}=\sigma_{X'}^2\cdot I_D$. Here the diagonal is characterized entirely by the constant values $\sigma_Z^2\in\mathbb{R}^+$ and $\sigma_{X'}^2\in\mathbb{R}^+$.

The conventional neural networks make use of nonlinear transformations for approximating functions. In particular variational autoencoders use them to calculate the encoder and decoder distribution parameters. Due to the non-linearities, it is difficult to identify analytically the optimal neural network weights obtained by minimizing the loss function corresponding to the negative ELBO. Therefore we will study the baseline variational autoencoder when we restrict ourselves to simple linear matrix multiplications for calculating the location parameters for the encoder and decoder distributions. Thus, our neural network functions $\boldsymbol{\mu}_{\boldsymbol{Z}}^{(G)}:X'\mapsto Z$ and $\boldsymbol{\mu}_{\boldsymbol{X'}}^{(H)}:Z\mapsto X'$ are given by

$$\boldsymbol{\mu}_{\boldsymbol{Z}}^{(G)}(x)=Gx \tag{6.9}$$

$$\boldsymbol{\mu}_{\boldsymbol{X'}}^{(H)}(z)=Hz \tag{6.10}$$

Where the neural weights correspond to the matrices $G\in\mathbb{R}^{2\times D}$ and $H\in\mathbb{R}^{D\times 2}$. We will identify that, in this very simple case, one particular minimizer matrix for the encoding distribution G

sends the image of $\mathscr{F}$ into a circle within latent space.

6.3 Optimal solution

Theorem 3. *Let $\mathbb{P}_{X'\times\Phi}$ be the generative model with probability density function with respect to the $\mathcal{L}^2 \otimes \mathcal{L}^1$ measure given by*

$$P_{X'\times\Phi}(x,\varphi) = \frac{1}{4\pi^2\sigma^2}\exp\left(\frac{-\|x-\mathscr{F}(\varphi)\|_2^2}{2\sigma^2}\right) \tag{6.11}$$

Consider the baseline variational autoencoder with the restrictions stated in the previous section. An optimal encoding neural network $\boldsymbol{\mu}_{\boldsymbol{Z}} : \mathbb{R}^2 \mapsto \mathbb{R}^2$ with weights G that minimizes the average negative ELBO according with respect to the marginal distribution $\mathbb{P}_{X'}$ has the property that for all $\varphi \in \Phi$ and $x = \mathscr{F}(\varphi)$

$$\|\boldsymbol{\mu}_{\boldsymbol{Z}}^{(G)}(x)\|_2 = \|Gx\|_2 = \max\left\{\frac{\sigma_Z}{\Gamma}\left(\frac{\sqrt{\Gamma}-\sigma_{X'}\sigma_Z}{\sigma_{X'}\sigma_Z}\right), 0\right\} \tag{6.12}$$

With $\Gamma = \left(\frac{1}{2}+\sigma^2\right)$

Proof. Our goal is to prove that an optimal encoding neural network $\boldsymbol{\mu}_{\boldsymbol{Z}}^{(G)}$ that minimizes the negative ELBO averaged over the data space X' according to $\mathbb{P}_{X'}$. First we will define the ELBO for a datapoint $x \in X'$ as the function $\mathscr{L} : X' \mapsto \mathbb{R}$ given by

$$-\mathscr{L}(x) = \mathrm{KL}\left(\mathbb{Q}_Z^{(\boldsymbol{\mu}_{\boldsymbol{Z}}(x),\sigma_Z^2)}||\mathbb{P}_Z\right) - \mathbb{E}_{\mathbb{Q}_Z^{(\boldsymbol{\mu}_{\boldsymbol{Z}}(x),\sigma_Z^2)}}\left[\log P_{X'}^{(\boldsymbol{\mu}_{\boldsymbol{X'}}(\cdot),\sigma_{X'}^2)}(x)\right]$$

By substituting the Kullback-Leibler regularization and the reconstruction error in terms of the encoding and decoding neural networks we obtain:

$$-\mathscr{L}(x) = \frac{1}{2}\left(2\sigma_Z^2 + \|\boldsymbol{\mu}_{\boldsymbol{Z}}(x)\|_2^2 - 2\log(\sigma_Z^2) - 2\right) + \mathbb{E}_{\mathbb{Q}_Z^{(\boldsymbol{\mu}_{\boldsymbol{Z}}(x),\sigma_Z^2)}}\left[\frac{1}{2\sigma_X^2}\|x-\boldsymbol{\mu}_{\boldsymbol{X'}}(\cdot)\|_2^2 + \log(2\pi\sigma_{X'}^2)\right]$$

Notice that we can group the terms that do not involve the encoding and decoding neural networks whose weights are optimized.

$$C = \sigma_Z^2 - \log(\sigma_Z^2) + \log(2\pi\sigma_{X'}^2) - 1.$$

Substituting the constant into the negative ELBO we obtain,

$$-\mathscr{L}(x) = \frac{1}{2}\left(\|\boldsymbol{\mu}_{\boldsymbol{Z}}(x)\|_2^2 + \frac{1}{\sigma_X^2}\mathbb{E}_{\mathbb{Q}_Z^{(\boldsymbol{\mu}_{\boldsymbol{Z}}(x),\sigma_Z^2)}}\left[\|x-\boldsymbol{\mu}_{\boldsymbol{X'}}(\cdot)\|_2^2\right]\right) + C.$$

First we will calculate the expected value with respect to the posterior approximate within the ELBO function. We can change this expected value over Z with respect to the approximate posterior $\mathbb{Q}_Z^{(\boldsymbol{\mu}_{\boldsymbol{Z}}(x),\sigma_Z^2)}$ by using the reparametrization trick defined in Section 3.3. The reparametrization function is defined as

$$\mathrm{Rep}((\boldsymbol{\mu}_{\boldsymbol{Z}}(\boldsymbol{x}),\sigma_Z),\epsilon) = \boldsymbol{\mu}_{\boldsymbol{Z}}(x) + \sigma_Z\epsilon.$$

If we choose the auxiliary the auxiliary value $\epsilon \in E = \mathbb{R}^2$ to be distributed according to the standard normal distribution $\mathbb{P}_E$ then we can change the expected value over Z into an expected value over E,

$$\mathbb{E}_{\mathbb{Q}_Z^{(\boldsymbol{\mu}_{\boldsymbol{Z}}(x),\sigma_Z^2)}}\left[\|x-\boldsymbol{\mu}_{\boldsymbol{X'}}(\cdot)\|_2^2\right] = \mathbb{E}_{\mathbb{P}_E}\left[\|x-\boldsymbol{\mu}_{\boldsymbol{X'}}(\boldsymbol{\mu}_{\boldsymbol{Z}}(x)+\sigma_Z(\cdot))\|_2^2\right].$$

By substituting the encoding and decoding neural networks as matrix multiplications we can calculate the expected value with respect to the auxiliary space E according to,

$$\begin{aligned}\mathbb{E}_{\mathbb{P}_E}\left[\|x-\boldsymbol{\mu_{X'}}(\boldsymbol{\mu_Z}(x)+\sigma_Z(\cdot))\|_2^2\right] &= \mathbb{E}_{\mathbb{P}_E}\left[\|x-H(Gx+\sigma_Z(\cdot))\|_2^2\right] \\ &= \|(HG-I_2)x\|_2^2 + 2\sigma_Z((HG-I_2)x)^T(H\mathbb{E}_{\mathbb{P}_E}\left[(\cdot)\right]) + \sigma_Z^2\mathbb{E}_{\mathbb{P}_E}\left[\|H(\cdot)\|_2^2\right].\end{aligned}$$

Since the auxiliary variables are distributed according to a standard normal distribution, the second term with $\mathbb{E}_{\mathbb{P}_E}[(\cdot)] = (0,0)$ cancels out. The third term involves the expected value of the norm of the auxiliary variable multiplied by matrix H which is calculated as

$$\mathbb{E}_{\mathbb{P}_E}\left[\|H(\cdot)\|_2^2\right] = \|H\mathbb{E}_{\mathbb{P}_E}\left[(\cdot)\right]\|_2^2 + \mathrm{tr}(H^TH\Sigma_E).$$

Since $\mathbb{P}_E$ corresponds to the standard normal distribution in $\mathbb{R}^2$ then $\mathbb{E}_{\mathbb{P}_E}[(\cdot)] = (0,0)$ and the covariance matrix $\Sigma_E = I_2$,

$$\mathbb{E}_{\mathbb{P}_E}\left[\|H(\cdot)\|_2^2\right] = \mathrm{tr}(H^TH) = \|H\|_F^2.$$

Here the operator $\|\cdot\|_F$ corresponds to the Frobenius norm of a matrix. Therefore we have the final form for the expected value over the auxiliary space given by

$$\mathbb{E}_{\mathbb{P}_E}\left[\|x-\boldsymbol{\mu_{X'}}(\boldsymbol{\mu_Z}(x)+\sigma_Z(\cdot))\|_2^2\right] = \|(HG-I_2)x\|_2^2 + \sigma_Z^2\|H\|_F^2,$$

which we can subsitute into the negative ELBO function.

$$-\mathscr{L}(x) = \frac{1}{2}\left(\|Gx\|_2^2 + \frac{1}{\sigma_{X'}^2}\|(HG-I_2)x\|_2^2 + \frac{\sigma_Z^2}{\sigma_{X'}^2}\|H\|_F^2\right) + C.$$

Recall that our goal is to find the matrices G, H that minimize the expected value with respect to $\mathbb{P}_{X'}$ of the negative ELBO expressed by

$$\mathbb{E}_{\mathbb{P}_{X'}}\left[-\mathscr{L}(x)\right] = \mathbb{E}_{\mathbb{P}_{X'}}\left[\frac{1}{2}\left(\|Gx\|_2^2 + \frac{1}{\sigma_{X'}^2}\|(HG-I_2)x\|_2^2 + \frac{\sigma_Z^2}{\sigma_{X'}^2}\|H\|_F^2\right)\right] + C. \tag{6.13}$$

The expected value of

$$\mathbb{E}_{\mathbb{P}'_X}\left[\|G(\cdot)\|_2^2\right] = \|G\mathbb{E}_{\mathbb{P}_{X'}}\left[(\cdot)\right]\|_2^2 + \mathrm{tr}(G^TG\Sigma_{X'})$$

In order to calculate the expected value we neet to identify the mean of a datapoint according to $\mathbb{P}_{X'}$ which is calculated as

$$\mathbb{E}_{\mathbb{P}_{X'}}[\cdot] = \int_{X'} x \int_\Phi P_{X'|\varphi}(x)P_\Phi(\varphi)d\mathcal{L}^1(\varphi)d\mathcal{L}^2(x). \tag{6.14}$$

For a fixed phase φ, the expected value of datapoint x distributed according to the conditional distribution is calculated with the function $\mathscr{F}(\varphi)$

$$\mathbb{E}_{\mathbb{P}_{X'|\varphi}}[(\cdot)] = \int_{X'} x\, P_{X'|\varphi}(x)d\mathcal{L}^2(x) = \mathscr{F}(\varphi).$$

Subsituting into Equation (6.14) we obtain the mean for a datapoint distributed according to the marginal $\mathbb{P}_{X'}$

$$\mathbb{E}_{\mathbb{P}_{X'}}[\cdot] = \frac{1}{2\pi}\int_\Phi \mathscr{F}(\varphi)d\mathcal{L}^1(\varphi) = \frac{1}{2\pi}\int_\Phi (\cos(\varphi),\sin(\varphi))d\mathcal{L}^1(\varphi) = (0,0)$$

Since the mean of a datapoint distributed according to $\mathbb{P}_{X'}$ is $(0,0)$, we can calculate the i-th diagonal entry from the covariance matrix according to the second moment of the conditional distribution for the i-th datapoint entry $\mathbb{E}_{x_i \sim \mathbb{P}_{X'|\varphi}}\left[x_i^2\right]$,

$$(\Sigma_{X'})_{i,i} = \int_\Phi \mathbb{E}_{x_i \sim \mathbb{P}_{X'|\varphi}}\left[x_i^2\right] d\mathcal{L}^1(\varphi).$$

We have that the second moment of the i-th entry of a datapoint x_i can be calculated with the formula

$$\mathbb{E}_{x_i \sim \mathbb{P}_{X'|\varphi}}\left[x_i^2\right] = \mathbb{E}_{x_i \sim \mathbb{P}_{X'|\varphi}}\left[x_i\right]^2 + \mathbb{V}_{x_i \sim \mathbb{P}_{X'|\varphi}}[x_i]$$

Here the operator $\mathbb{V}_{x_i \sim \mathbb{P}_{X'|\varphi}}$ corresponds to the variance of the i-th datapoint entry which is equal to σ^2. Substituting the mean value for the conditional distribution $\mathbb{E}_{\mathbb{P}_{X'|\varphi}}\left[(\cdot)\right] = \mathscr{F}(\varphi)$ we have that the i-th element of the diagonal covariance matrix is calculated as

$$(\Sigma_{X'})_{i,i} = \frac{1}{2\pi}\int_\Phi \left(\mathbb{E}_{\mathbb{P}_{X'|\varphi}}\left[(\cdot)\right]_i^2 + \sigma^2\right) d\mathcal{L}^1(\varphi) = \frac{1}{2\pi}\int_\Phi \left(\mathscr{F}(\varphi)_i^2 + \sigma^2\right) d\mathcal{L}^1(\varphi)$$

We have that the i-th entry of the conditional distribution mean corresponds to the i-th value of the function $\mathscr{F}(\varphi) = (\sin(\varphi), \cos(\varphi))$, therefore

$$(\Sigma_{X'})_{1,1} = \frac{1}{2\pi}\int_\Phi \left(\sin(\varphi)^2 + \sigma^2\right) d\mathcal{L}^1(\varphi)$$
$$(\Sigma_{X'})_{2,2} = \frac{1}{2\pi}\int_\Phi \left(\cos(\varphi)^2 + \sigma^2\right) d\mathcal{L}^1(\varphi)$$

The covariance matrix then corresponds to

$$\Sigma_{X'} = \left(\frac{1}{2} + \sigma^2\right) I_2 = \Gamma I_2.$$

For notation simplicity, we have rewritten the constant diagonal terms as Γ, i.e.

$$\Gamma = \left(\frac{1}{2} + \sigma^2\right).$$

Therefore the first term in Equation (6.13) is calculated as

$$\mathbb{E}_{\mathbb{P}_{X'}}\left[\|Gx\|_2^2\right] = \|G\mathbb{E}_{\mathbb{P}_{X'}}[(\cdot)]\|_2^2 + \operatorname{tr}(GG^T\Sigma_{X'}) = \Gamma\|G\|_F^2.$$

The second term of Equation (6.13) is obtained in a similar fashion,

$$\mathbb{E}_{\mathbb{P}_{X'}}\left[\|(HG - I_2)x\|_2^2\right] = \Gamma\|(HG - I)\|_F^2.$$

Substituting both expected values into the negative ELBO we have the formula

$$\mathbb{E}_{\mathbb{P}_{X'}}\left[-\mathscr{L}(x)\right] = \frac{1}{2}\left(\Gamma\|G\|_F^2 + \frac{\Gamma}{2\sigma_X^2}\|HG - I_2\|_F^2 + \frac{\sigma_Z^2}{\sigma_X^2}\|H\|_F^2\right) + C.$$

We can simplify our minimization objective by taking out the constants which are independent of the parameters that we are aiming to optimize. The simplified objective function used to obtain the optimal parameters is renamed as a function Loss : $\mathbb{R}^{2\times 2} \times \mathbb{R}^{2\times 2} \mapsto \mathbb{R}$. The condition for minimization is therefore

$$\arg\min_{G\in\mathbb{R}^{2\times 2}, H\in\mathbb{R}^{2\times 2}} \operatorname{Loss}(G, H) = \arg\min_{G\in\mathbb{R}^{2\times 2}, H\in\mathbb{R}^{2\times 2}} \Gamma\|G\|_F^2 + \frac{\Gamma}{\sigma_X^2}\|HG - I_2\|_F^2 + \frac{\sigma_Z^2}{\sigma_X^2}\|H\|_F^2.$$

The necessary condition that the parameters G, H must fulfill to be a local minimum is that the gradient must be equal to zero, that is

$$\begin{aligned} \nabla_G(\text{Loss}(G,H)) &= 2\Gamma G + \frac{2\Gamma}{\sigma_X^2}\left(\frac{1}{2}+\sigma^2\right)H^T(HG-I_2) = 0, \\ \nabla_H(\text{Loss}(G,H)) &= 2\frac{\sigma_Z^2}{\sigma_X^2}H + \frac{2\Gamma}{\sigma_X^2}(HG-I_2)G^T = 0. \end{aligned} \tag{6.15}$$

Multiplying the first condition in Equation (6.15) to the right by G^T and the second to the left by H^T we can substract them to obtain

$$H^T H = \Gamma\frac{\sigma_{X'}^2}{\sigma_Z^2}GG^T$$

Let G and H be optimizers of the loss function. If we express matrix G in terms of its singular value decomposition $G = U_G\Lambda_G V_G^T$ where $U_G, V_G^T \in \mathbb{R}^{2,2}$ are orthogonal matrices and $\Lambda_G \in \mathbb{R}^{2,2}$ is a diagonal matrix then we can rewrite the loss function as

$$\Gamma\|U_G\Lambda_G V_G^T\|_F^2 + \frac{\Gamma}{\sigma_X^2}\|HU_G\Lambda_G V_G^T - I_2\|_F^2 + \frac{\sigma_Z^2}{\sigma_X^2}\|H\|_F^2$$

Since U_G and V_G^T are orthogonal matrices, we have that $U_G^T U_G = I_2$ and $V_G^T V_G = I_2$. Because of this property, we have that for any matrix $A \in \mathbb{R}^{2\times 2}$, the Frobenius norm of A multiplied by an orthogonal matrix U_G is

$$\|U_G A\|_F^2 = \text{tr}(A^T U_G^T U_G A) = \text{tr}(A^T A) = \|A\|_F^2$$

We can then express the identity matrix of Equation (6.3) as $I_2 = V_G V_G^T$ and introduce identity matrices in the second and third term such that

$$\Gamma\|\Lambda_G\|_F^2 + \frac{\Gamma}{\sigma_X^2}\|V_G V_G^T H U_G\Lambda_G V_G^T - V_G V_G^T\|_F^2 + \frac{\sigma_Z^2}{\sigma_X^2}\|H\|_F^2$$

We can then factorize from the left and from the right the matrix V_G^T and V_G in the second term and remove them from the norm since they are orthogonal,

$$\Gamma\|\Lambda_G\|_F^2 + \frac{\Gamma}{\sigma_X^2}\|V_G(V_G^T H U_G\Lambda_G - I_2)V_G^T\|_F^2 + \frac{\sigma_Z^2}{\sigma_X^2}\|H\|_F^2$$

Due to the orthogonality of V_G then

$$\Gamma\|\Lambda_G\|_F^2 + \frac{\Gamma}{\sigma_X^2}\|V_G^T H U_G\Lambda_G - I_2\|_F^2 + \frac{\sigma_Z^2}{\sigma_X^2}\|H\|_F^2$$

Finally since multiplying by the orthogonal matrices V_G^T and U_G does not affect the Frobenius norm, we introduce them to the third term

$$\Gamma\|\Lambda_G\|_F^2 + \frac{\Gamma}{\sigma_X^2}\|V_G^T H U_G\Lambda_G - I_2\|_F^2 + \frac{\sigma_Z^2}{\sigma_X^2}\|V_G^T H U_G\|_F^2$$

If we define the optimizer matrices $G' = \Lambda_G$ and $H' = V_G^T H U_G$ then our loss function becomes

$$\text{Loss}(G',H') = \Gamma\|\Lambda_G\|_F^2 + \frac{\Gamma}{\sigma_X^2}\|H'\Lambda_G - I_2\|_F^2 + \frac{\sigma_Z^2}{\sigma_X^2}\|H'\|_F^2$$

By performing a polar decomposition of matrix $H' = Q_{H'}S_{H'}$ with $Q_{H'} \in \mathbb{R}^2$ an orthogonal matrix and $S_{H'}$ a symmetric matrix with $S_{H'} = \sqrt{H'^T H'}$ therefore by the condition in Equation

(6.3)

$$S_{H'}^2 = \Gamma \frac{\sigma_{X'}^2}{\sigma_Z^2} G' G'^T = \Gamma \frac{\sigma_{X'}^2}{\sigma_Z^2} \Lambda_G^2$$

Therefore $S_{H'} = \sqrt{\Gamma} \frac{\sigma_{X'}}{\sigma_Z} \Lambda_G$, substituting this into the loss function

$$\text{Loss}(G', H') = \Gamma \|\Lambda_G\|_F^2 + \frac{\Gamma}{\sigma_X^2} \left\| \left(\sqrt{\Gamma} \frac{\sigma_{X'}}{\sigma_Z} \right) Q_{H'} \Lambda_G^2 - I_2 \right\|_F^2 + \Gamma \|\Lambda_G\|_F^2$$

We can express the second term of the minimization objective in terms of the norm of the projections into the canonical vectors $e_i \in \mathbb{R}^2$.

$$\begin{aligned} \left\| \left(\sqrt{\Gamma} \frac{\sigma_{X'}}{\sigma_Z} \right) Q_{H'} \Lambda_G^2 - I_2 \right\|_F^2 &= \sum_{i=1}^{2} \left\| \left(\left(\sqrt{\Gamma} \frac{\sigma_{X'}}{\sigma_Z} \right) Q_{H'} \Lambda_G^2 - I_2 \right) e_i \right\|_2^2 \\ &\geq \sum_{i=1}^{2} \left\| \left(\left(\sqrt{\Gamma} \frac{\sigma_{X'}}{\sigma_Z} \right) \Lambda_G^2 - I_2 \right) e_i \right\|_2^2 = \left\| \left(\sqrt{\Gamma} \frac{\sigma_{X'}}{\sigma_Z} \right) \Lambda_G^2 - I_2 \right\|_F^2 . \end{aligned}$$

In this way we have reduced the dependence of our minimization objective to only the matrix Λ_G, therefore we are searching to minimize

$$\arg \min_{\Lambda_G \in \mathbb{R}^{2\times 2}} \Gamma \|\Lambda_G\|_F^2 + \frac{\Gamma}{\sigma_{X'}^2} \left\| \left(\sqrt{\Gamma} \frac{\sigma_{X'}}{\sigma_Z} \right) \Lambda_G^2 - I_2 \right\|_F^2 + \Gamma \|\Lambda_G\|_F^2 .$$

In general the diagonal matrix Λ_G can be described in terms of its diagonal entries (λ_1, λ_2). It can be seen from the fact that the minimization problem splits into two independent minimization problems that the diagonal entries must be equal, i.e. $\lambda_1 = \lambda_2 = \lambda$. Therefore the minimization criteria can be simplified to only depend on the scalar $\lambda \in \mathbb{R}$

$$\arg \min_{\lambda \in \mathbb{R}} 4\Gamma \lambda^2 + \frac{\Gamma}{\sigma_{X'}^2} \left[2 \frac{\sigma_{X'}^2 \Gamma}{\sigma_Z^2} \lambda^4 - 4 \left(\frac{\sqrt{\Gamma} \sigma_{X'}}{\sigma_Z} \right) \lambda^2 + 2 \right]$$

We can then identify the sufficient conditions for a global minimum by taking the gradients with respect to λ^2 of the loss function. By identifying that the second derivative of the loss function is always positive we conclude that it is convex.

$$\begin{aligned} \nabla_{\lambda^2} \text{Loss}(\lambda) &= 4\Gamma \left(1 - \frac{\sqrt{\Gamma}}{\sigma_{X'} \sigma_Z} + \frac{\lambda^2 \Gamma}{\sigma_Z^2} \right) = 0 \\ \nabla_{\lambda^2}^2 \text{Loss}(\lambda) &= \frac{2\Gamma^2}{\sigma_Z^2} > 0 \end{aligned}$$

The critical value for λ^2 corresponds to

$$\lambda^2 = \frac{\sigma_Z}{\Gamma} \left(\frac{\sqrt{\Gamma} - \sigma_{X'} \sigma_Z}{\sigma_{X'} \sigma_Z} \right) . \qquad (6.16)$$

Notice that due to the convexity of the loss function, there is always a critical solution, even for negative values of λ^2. Since we are interested in the value of λ we restrict the domain of the possible critical values for λ^2 from Equation (6.16) to only non-negative values. In the case where $\sqrt{\Gamma} - \sigma_{X'} \sigma_Z < 0$ the optimal lies in the boundary of our restricted domain i.e. the optimal value corresponds to $\lambda^2 = 0$. We have that the value for an optimal λ subject to the restriction to only non-negative values is

$$\lambda = \max \left\{ \frac{\sigma_Z}{\Gamma} \left(\frac{\sqrt{\Gamma} - \sigma_{X'}\sigma_Z}{\sigma_{X'}\sigma_Z} \right), 0 \right\}$$

Therefore we have identified that a minimal solution to the optimization of the negative ELBO for the simplified baseline variational autoencoder satisfies

$$\|G'x\|_2^2 = \|\Lambda_G x\|_2^2 = \max \left\{ \frac{\sigma_Z}{\Gamma} \left(\frac{\sqrt{\Gamma} - \sigma_{X'}\sigma_Z}{\sigma_{X'}\sigma_Z} \right), 0 \right\} \tag{6.17}$$

□

Even though this is a very restrictive result that limits the expressiveness of the baseline variational autoencoder it provides a proof for a simple setting in which we can naturally recover the circular underlying latent structure of the dataset. In the next section we will now further explore the consequences of restricting the latent space of a variational autoencoder. We will propose a variational autoencoder that assumes a circular latent space and propose a special family of parametric distributions to approximate the posterior.

Chapter 7

Diffusion Variational Autoencoder

In the previous chapter we introduced a benchmark dataset in which we have the complete information about the generative model $\mathbb{P}_{X\times Z}$ and the latent variables $Z = \Phi$ used to generate it. Moreover, we have noticed that the function $F : \Phi \mapsto X$ induces a spherical structure over the benchmark dataset with respect to the latent variables. The baseline variational autoencoder used in the previous chapter proposes a latent space corresponding to the Euclidean space without any special structure or restrictive assumptions.

In this chapter we will study the case in which we incorporate to a variational autoencoder the assumption that the latent space has a circular geometry. The purpose of this is to identify the effects of imposing the assumptions of having circular latent variables that generated dataset $\mathcal{X}$ and test whether a periodic latent structure is recovered. In this setting, we propose that the latent space can be identified with the unit circle $Z = \mathcal{S}^1$ embedded in $\mathbb{R}^2$,

$$\mathcal{S}^1 = \{z \in \mathbb{R}^2 \mid \|z\|_2^2 = 1\}. \tag{7.1}$$

We describe the latent space $Z = \mathcal{S}^1$ in terms of an auxiliary set $\Theta = [-\pi, \pi)$ and an embedding function $\mathrm{Emb}_{\mathbb{R}^2} : \Theta \mapsto \mathbb{R}^2$. Moreover, we will introduce a family of parametric distributions based on the solutions to the diffusion equation with periodic boundary conditions to approximate the posterior. For this variational autoencoder we propose the decoding distribution as a member of the parametric family of normal distributions just like in the baseline variational autoencoder.

7.1 Encoding family & Kullback-Leibler regularization

Let $(\Theta, \mathcal{F}_\Theta, \mathbb{P}_\Theta)$ be a probability space over the interval $\Theta = [-\pi, \pi)$ were the measure $\mathbb{P}_\Theta$ corresponds to the uniform probability distribution over Θ. The probability density P_Θ of $\mathbb{P}_\Theta$ with respect to $\mathcal{L}^1$ for a certain $\theta \in \Theta$ corresponds to

$$P_\Theta(\theta) = \frac{1}{2\pi}. \tag{7.2}$$

We define the latent space Z as the unit circle $Z = \mathcal{S}^1$ embedded in $\mathbb{R}^2$ described in terms of the elements of the interval Θ through the measurable function $\mathrm{Emb}_{\mathbb{R}^2} : \Theta \mapsto \mathbb{R}^2$,

$$\mathrm{Emb}_{\mathbb{R}^2}(\theta) = (\cos(\theta), \sin(\theta)). \tag{7.3}$$

Thus, the latent space corresponding to the unit circle is described with respect to the set Θ as the image of the embedding function,

$$\mathcal{S}^1 = \{z \in \mathbb{R}^2 \mid z = \mathrm{Emb}_{\mathbb{R}^2}(\theta)\ ;\ \theta \in \Theta\}. \tag{7.4}$$

We propose the prior distribution $\mathbb{P}_{\mathcal{S}^1}$ over the latent space $\mathcal{S}^1$ as the pushforward measure of the measurable embedding function $\mathrm{Emb}_{\mathbb{R}^2 \#}\mathbb{P}_\Theta$ which corresponds to the uniform measure over $\mathcal{S}^1$.

For estimating the posterior distribution, first we choose a family of parametric distributions $\mathcal{P}_\Theta^A$ defined in terms of the probability densities $P_\Theta^{(\mu_\Theta, t_\Theta)}$ obtained from the solutions to the diffusion equation in the domain Θ with periodic boundary conditions $P_\Theta^{(\mu_\Theta,t_\Theta)}(-\pi) = \lim_{\theta\to\pi} P_\Theta^{(\mu_\Theta,t_\Theta)}(\theta)$ (See derivation in Appendix A). This family is characterized as

$$\mathcal{P}_\Theta^A = \left\{ \mathbb{P}_\Theta^{(\mu,t)} \in \mathcal{P}_\Theta \;\middle|\; P_\Theta^{(\mu,t)}(\theta) = \frac{1}{2\pi}\sum_{m=0}^{\infty} \cos(m(\theta-\mu))\exp(-m^2 t) \right\} \tag{7.5}$$

Here the parameter $\mu \in \Theta$ corresponds to the location parameter and $t \in \mathbb{R}_0^+$ corresponds to the scaling parameter

$$A = \{(\mu, t) \in \Theta \times \mathbb{R}_0^+\}. \tag{7.6}$$

From the parametric family over the set Θ we define its counterpart in the latent space determined by the pushforward measure of the embedding. The parametric family of encoding distributions that approximate to the posterior is thus defined in terms of the same parameter set A as

$$\mathcal{Q}_{\mathcal{S}^1}^A = \left\{ \mathbb{Q}_{\mathcal{S}^1}^{(\mu,t)} \in \mathcal{P}_Z \mid \mathbb{Q}_{\mathcal{S}^1}^{(\mu,t)} = \mathrm{Emb}_{\mathbb{R}^2 \#}\mathbb{P}_\Theta^{(\mu,t)} \right\}. \tag{7.7}$$

The Kullback-Leibler divergence regularization term of the ELBO can be calculated in terms of the probability distributions over the set Θ since $\mathrm{KL}(\mathbb{Q}_{\mathcal{S}^1}^{(\mu,t)}||\mathbb{P}_{\mathcal{S}^1}) = \mathrm{KL}(\mathbb{P}_\Theta^{(\mu,t)}||\mathbb{P}_\Theta)$ with

$$\mathrm{KL}\left(\mathbb{P}_\Theta^{(\mu,t)}||\mathbb{P}_\Theta\right) = \int_\Theta \log\left(\sum_{m=0}^{\infty}\cos(m(\theta-\mu))\exp(-m^2 t)\right) d\mathbb{P}_\Theta^{(\mu,t)}(\theta). \tag{7.8}$$

This integral can be approximated with the Monte Carlo method introduced in Section 3.2 by sampling L elements $\{\theta^{(l)}\}_{l=1}^L$ from the interval Θ according to $\mathcal{Q}_\Theta^{(\mu,t)}$. The parameters μ, t from the distribution $\mathcal{Q}_\Theta^{(\mu,t)}$ are calculated with the neural networks $\boldsymbol{\mu}_{\boldsymbol{\Theta}} : X \mapsto \mathbb{R}$ and $\boldsymbol{t}_{\boldsymbol{\Theta}} : X \mapsto \mathbb{R}_0^+$ with a datapoint $x \in X$. The Monte Carlo estimate of the Kullback-Leibler divergence is given by

$$\mathrm{KL}(\mathbb{P}_\Theta^{(\boldsymbol{\mu}_{\boldsymbol{\Theta}}(x),\boldsymbol{t}_{\boldsymbol{\Theta}}(x))}||\mathbb{P}_\Theta) \approx \frac{1}{L}\sum_{i=1}^{L} P_\Theta^{(\boldsymbol{\mu}_{\boldsymbol{\Theta}}(x),\boldsymbol{t}_{\boldsymbol{\Theta}}(x))}(\theta^{(l)})\log\left(\frac{P_\Theta^{(\boldsymbol{\mu}_{\boldsymbol{\Theta}}(x),\boldsymbol{t}_{\boldsymbol{\Theta}}(x))}(\theta^{(l)})}{P_\Theta(\theta^{(l)})}\right). \tag{7.9}$$

The sampling of elements $\{\theta^{(l)}\}_{l=1}^L$ from Θ in the variational autoencoder is performed by applying the reparametrization trick introduced in Section 3.3. The reparametrization function $\mathrm{Rep}_\Theta : A \times \mathbb{R} \mapsto \Theta$ produces each sampled value $\theta^{(l)}$ from the calculated parameters μ, t of $\mathcal{Q}_\Theta^{(\mu,t)}$ and the auxiliary element $\epsilon^{(l)}$ sampled from space $E = \mathbb{R}$ according to the standard normal distribution. Therefore, the sampled $\theta^{(l)}$ is calculated as

$$\theta^{(l)} = \mathrm{Rep}_\Theta((\boldsymbol{\mu}_{\boldsymbol{\Theta}}(x), \boldsymbol{t}_{\boldsymbol{\Theta}}(x)), \epsilon) = [(\pi + \boldsymbol{\mu}_{\boldsymbol{\Theta}}(x) + \epsilon \cdot \boldsymbol{t}_{\boldsymbol{\Theta}}(x)) \bmod (2\pi)] - \pi. \tag{7.10}$$

7.2 Benchmark dataset training

The conditions used for training with the benchmark dataset correspond to the same ones used for the baseline variational autoencoder. The three datasets $\mathcal{X}_{0.01}$, $\mathcal{X}_{0.1}$ and $\mathcal{X}_1$ with different levels of noise were employed for training. The number of latent samples is chosen as $L = 1$.

For each dataset three variational autoencoders were trained with decoding distribution parameter $\sigma_X \in \{0.01, 0.1, 1\}$. The number of epochs and the optimizer used for training were the same as the ones presented in Section 5.3. The architecture of the neural networks is shown in

Section 3.5. Two dense layers where used for both the encoder and decoder neural networks with $D/3 = \lfloor 25/3 \rfloor$ neurons each.

7.3 Qualitative results

To analyze and evaluate the results obtained by the trained variational autoencoder the same dataset $\mathcal{X}_{\text{Vis}}$ defined in Section 5.4 was used.

The approximate posterior distribution $\mathbb{Q}_{\mathcal{S}^1}^{(\boldsymbol{\mu}^*_{\boldsymbol{\Theta}}(x),\boldsymbol{t}^*_{\boldsymbol{\Theta}}(x))}$ for a datapoint x is visualized with respect to its high probability region within the latent space $Z = \mathcal{S}^1$. In the diffusion variational autoencoder, these regions are characterized as arcs of the unit circle. Each arc spans an angle range within Θ determined by the scale parameter $\boldsymbol{t}_{\boldsymbol{\Theta}}(x)$. The center of each arc associated to a datapoint x is located at the embedded mean value $\text{Emb}_{\mathbb{R}^2}\boldsymbol{\mu}_{\boldsymbol{\Theta}}(x)$. Each arc is therefore described as

$$\left\{\text{Emb}_{\mathbb{R}^2}(\theta) \in \mathbb{R}^2 \;\middle|\; \|\theta - \boldsymbol{\mu}_{\boldsymbol{\Theta}}(x)\|_2 \leq \boldsymbol{t}_{\boldsymbol{\Theta}}(x)\right\}. \tag{7.11}$$

Each of the plots in Figure 7.1 corresponds to a representation of the approximate posterior $\mathbb{Q}_{\mathcal{S}^1}^{(\boldsymbol{\mu}^*_{\boldsymbol{\Theta}}(x),\boldsymbol{t}^*_{\boldsymbol{\Theta}}(x))}$ for datapoint $x \in \mathcal{X}_{\text{Vis}}$ obtained by training a diffusion variational autoencoder for each pair of dataset $\mathcal{X}_\sigma$ and parameter σ_X. In each plot it can be noticed that the recovered latent variables have a periodic latent structure since datapoints with consecutive associated phases are encoded next to each other in the unit circle. This can be clearly identified by noticing the continuous change in color hue for each latent variable representation.

Even though the recovered latent variables are apparently periodic for each trained diffusion variational autoencoder it is important to notice that there are some differences in the structure with respect to the dataset used for training $\mathcal{X}_\sigma$ and parameter σ_X.

For a fixed dataset $\mathcal{X}_\sigma$ we can see the effects of varying the value of the parameter σ within a row of Figure 7.1. As it was mentioned in previous chapters, the effect of changing σ_X is that of weighting the contribution of the reconstruction error with respect to the Kullback-Leibler regularization. Increasing the value of σ_X decreases the contribution of reconstruction error within the negative ELBO with respect to the Kullback-Leibler regularization.

Minimizing the Kullback-Leibler divergence forces the approximate posterior to resemble the prior over the latent space which is a uniform distribution over the unit circle. Therefore, we can notice that higher values of σ_X result in approximate posterior distributions with a larger scale value $\boldsymbol{t}_{\boldsymbol{\Theta}}(x)$ which results in longer arc portions of the unit circle associated to each datapoint $x \in \mathcal{X}_{\text{Vis}}$. In the case of $\sigma_X = 1$, the complete circle is covered with overlapping high probability regions.

For the parameter $\sigma_X = 0.1$ the high probability arcs of the posterior tend to cover less area of the unit circle and the scaling parameter $\boldsymbol{t}_{\boldsymbol{\Theta}}(x)$ calculated for each datapoint is still noticeable. When further reducing $\sigma_X = 0.01$ the latent variables do not span the complete circle and the scaling value makes the resulting arcs small and appear to be lines perpendicular to the unit circle.

For small values of σ_X, the reconstruction error dominates the negative ELBO forcing the variational autoencoder to favor better reconstructions without restricting the shape of the encoding distribution. Therefore in this cases the diffusion variational autoencoder for $\sigma_X = 0.01$ creates latent representations of the datapoints that only span a small section of the unit circle.

Finally if we analyze the effects of increasing the noise of a dataset by looking at the plots in the columns of Figure 7.1 a similar effect can be observed as in the baseline variational autoencoder. There is no apparent difference in the structures recovered for datasets $\mathcal{X}_{0.1}$ and $\mathcal{X}_{0.01}$.

The biggest difference in the results is observed for dataset $\mathcal{X}_1$ where the learned structure for the diffusion variational autoencoder with $\sigma_X \in \{0.01, 0.1\}$ is different to that of the dataset $\mathcal{X}_{0.01}$ and $\mathcal{X}_{0.1}$. In these cases the latent structure appears fragmented and only certain regions of the circle are occupied. This can be due to the variability of the dataset that does not allow the variational autoencoder to identify similarities in the underlying latent variables that generated the dataset. Nonetheless it is interesting to note that the order of datapoints with consecutive phases is maintained (the color hue increases gradually within the circle).

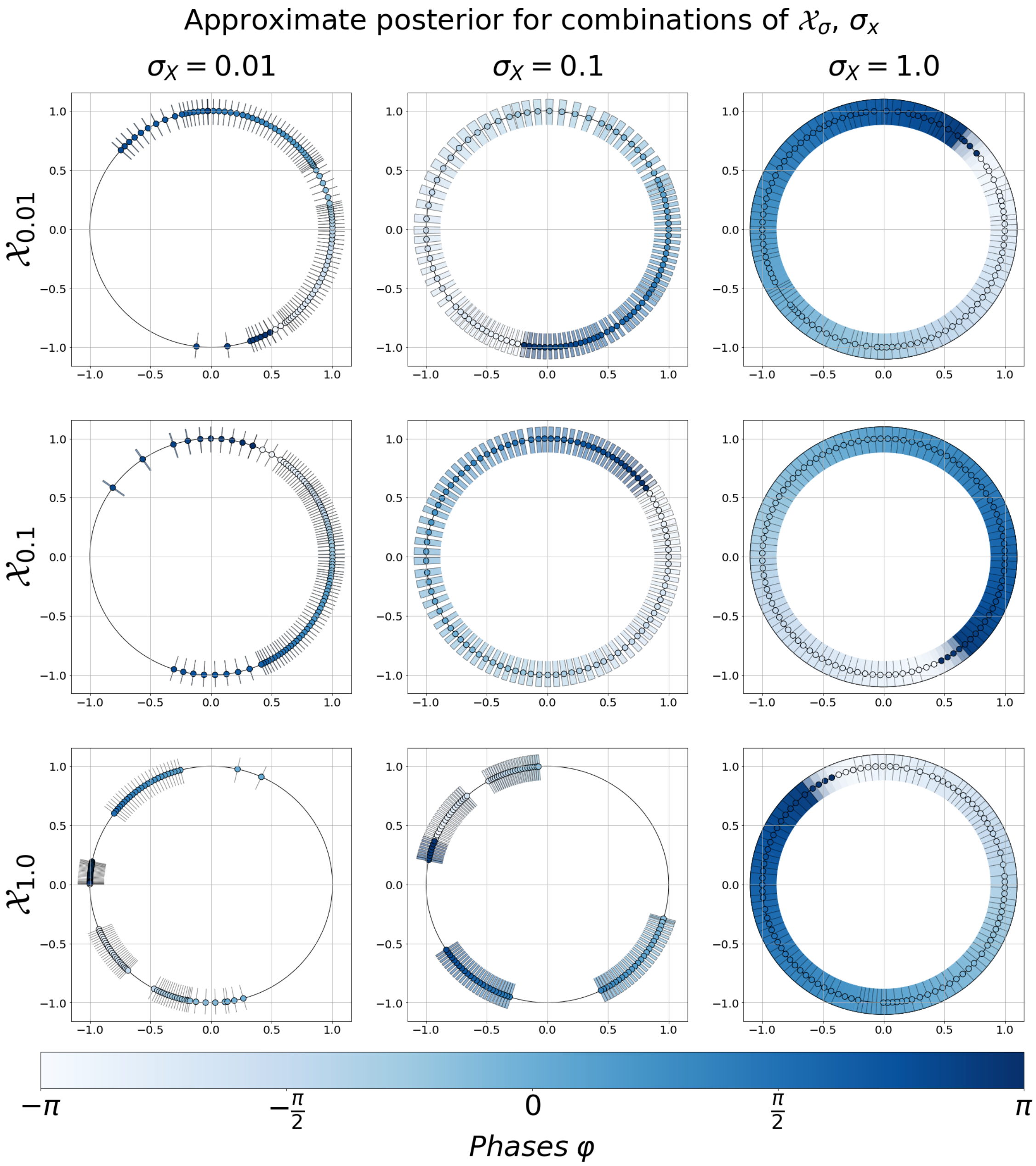

Figure 7.1: Representation of the approximate posterior $\mathbb{Q}_{S^1}^{(\boldsymbol{\mu}^*_{\boldsymbol{\Theta}}(x),\boldsymbol{t}^*_{\boldsymbol{\Theta}}(x))}$ obtained by training the baseline variational autoencoder for the different combinations of input dataset $\mathcal{X}_\sigma$ (Rows) and decoding distribution parameter σ_X (Columns). The markers in each plot represent the calculated values for the encoding distribution's mean $\boldsymbol{\mu}_{\boldsymbol{Z}}(x)$ for each datapoint $x \in \mathcal{X}_{\text{Vis}}$. The hue of each marker represents the underlying phase $\varphi \in \Phi_{\text{Vis}}$ corresponding to each datapoint $x \in \mathcal{X}_{\text{Vis}}$ and helps identify the relationships between datapoints with consecutive phases. The arc portions of the unit circle in each plot represent the high probability regions of the encoding distribution determined by $\boldsymbol{t}_{\boldsymbol{\Theta}}(x)$. In some cases these arcs have a smaller size compared to the size of the mean markers and are observed as lines perpendicular to the circle.

7.4 Quantitative results

The quantities used to evaluate the diffusion variational autoencoder are the same as the ones used for the baseline variational autoencoder discussed in Section 5.5, they are presented in Table 7.1.

Table 7.1: Values for the average negative ELBO, Kullback-Leibler regularization, reconstruction error and the mean square error. Each value is obtained by averaging the results of five repetitions for each corresponding variational autoencoder trained with dataset $\mathcal{X}_\sigma$ and parameter σ_X. The calculated quantities are obtained with respect to dataset $\mathcal{X}_{\text{Vis}}$. For the reconstruction error $L = 100$ samples are taken from latent space according to the trained posterior distribution.

Dataset	σ_X	**-ELBO**	**KL Regularization**	**Reconstruction**	**MSE**
$\mathcal{X}_{0.01}$	0.01	-84.99 ± 1.49	3.49 ± 0.21	-88.48 ± 1.30	$(6.78 \pm 2.52) \times 10^{-4}$
	0.1	-32.16 ± 0.29	2.14 ± 0.29	-34.30 ± 0.03	$(8.40 \pm 7.02) \times 10^{-4}$
	1	24.04 ± 0.06	$(7.69 \pm 0.01) \times 10^{-1}$	23.27 ± 0.06	$(7.26 \pm 12.4) \times 10^{-1}$
$\mathcal{X}_{0.1}$	0.01	-80.96 ± 3.39	3.47 ± 0.07	-84.43 ± 3.32	$(1.49 \pm 0.66) \times 10^{-3}$
	0.1	-31.45 ± 1.77	1.98 ± 0.01	-33.43 ± 1.77	$(1.83 \pm 3.57) \times 10^{-2}$
	1	24.05 ± 0.09	$(7.68 \pm 0.01) \times 10^{-1}$	23.28 ± 0.09	$(9.51 \pm 17.5) \times 10^{-2}$
$\mathcal{X}_{1}$	0.01	1849.45 ± 943.38	5.03 ± 0.33	1844.42 ± 943.40	$(3.87 \pm 1.89) \times 10^{-1}$
	0.1	1.32 ± 23.47	3.20 ± 0.86	-1.88 ± 22.71	$(6.50 \pm 4.54) \times 10^{-1}$
	1	24.15 ± 0.09	$(7.78 \pm 0.06) \times 10^{-1}$	23.37 ± 0.09	$(2.86 \pm 1.88) \times 10^{-1}$

A very interesting result that can be observed in Table 7.1 is the noticeable increase in the mean squared error when the noise in the data is increased. Higher noise produces worse reconstructions with bigger mean squared error. One possible explanation to this effect is that since the latent variables of the model are restricted to the unit circle which is low dimensional, the variability of the input dataset with respect to the noise can not be captured .

It is interesting to notice that as with the baseline variational autoencoder, the results obtained with dataset $\mathcal{X}_{0.1}$ and $\mathcal{X}_{0.01}$ are similar and the only changes are with respect to the value of the mean squared error where the order of magnitude can be up to two orders of magnitude different.

Another result is the high variability of the measured quantities with the models that have $\sigma_X = 0.01$ with the most extreme changes observed for the noisy dataset $\mathcal{X}_1$ and the lowest variability is within models that have $\sigma_X = 1$. These results are similar to the obtained with the baseline variational autoencoder.

We have already presented two variational autoencoders that attempt to recover the underlying latent variable structure of a particular dataset. These variational autoencoders create a representation of the input osberved data and impose a structure on the latent variables that can explain these observations. Moreover this learned structure can be used to produce new datapoints through the learned generative model defined by the variational autoencoder.

Therefore for a datapoint x we have obtained a latent representation in latent space $Z = \mathbb{R}^2$ and $Z = \mathcal{S}^1$. One interesting question that arises is whether these latent variable representations can be connected in such a way that the corresponding generative models are equivalent. This question is addressed in the next chapter.

Chapter 8

Generative Model Reduction

In the previous chapters we have presented the structure for two different variational autoencoders: The baseline variational autoencoder of Chapter 4 which assumes an Euclidean latent space and the diffusion variational autoencoder of Chapter 7 which posits a circular latent space. These variational autoencoders have been employed to recover the underlying structure of the latent variables that generated the benchmark dataset of Chapter 5.

For each variational autoencoder a decoding distribution is obtained which relates the corresponding latent variables with datapoints in dataspace X. This decoding distribution together with the proposed prior over the latent space defines a generative model that attempts to explain the process that generated the data (See Chapter 2). In this chapter we will propose a method for reducing the generative models obtained from training different variational autoencoders with a dataset $\mathcal{X}$.

Consider the case in which K variational autoencoders with different preliminary assumptions have been trained from dataset $\mathcal{X}$. Each of them is entirely characterized by the choice of the latent space Z_k, the prior $\mathbb{P}_{Z_k}$, the trained encoding distribution $\mathbb{Q}_{Z_k}^{\boldsymbol{\alpha}^*(\cdot)}$ and trained decoding distribution $\mathbb{P}_X^{\boldsymbol{\beta}^*(\cdot)}$. Therefore the k-th trained variational autoencoder is specified as the quadruple of elements $(Z_k, \mathbb{P}_{Z_k}, \mathbb{Q}_{Z_k}^{\boldsymbol{\alpha}_k^*(\cdot)}, \mathbb{P}_X^{\boldsymbol{\beta}_k^*(\cdot)})$. Moreover, the choice of the prior together with the trained decoder distribution determines a generative model with respect to the corresponding latent variables.

Definition 8.0.1. ***Generative model of a variational autoencoder***
Given a variational autoencoder represented by the quadruple $(Z_k, \mathbb{P}_{Z_k}, \mathbb{Q}_{Z_k}^{\boldsymbol{\alpha}_k^*(\cdot)}, \mathbb{P}_X^{\boldsymbol{\beta}_k^*(\cdot)})$ the generative model associated to the variational autoencoder $\mathbb{P}_{X\times Z_k}$ is defined in terms of the probability density function $P_{X\times Z_k}$ with respect to the $\mathcal{L}^d \otimes \mathcal{P}_Z$ measure given by:

$$P_{X\times Z_k}(x, z) = P_X^{\boldsymbol{\beta}_k^*(z)}(x) P_{Z_k}(z) \tag{8.1}$$

As it was described in Chapter 2, the process of generating a datapoint according to the k-th variational autoencoder generative model is comprised of the process of first sampling a latent variable z according to the prior $\mathbb{P}_{Z_k}$ and then sampling a datapoint x from the corresponding distribution $\mathbb{P}_X^{\boldsymbol{\beta}_k^*(z)}(x)$.

One question that arises is whether for a given dataset $\mathcal{X}$ the generative models obtained from two variational autoencoders can be expressed in terms of one-another. In this chapter we will discuss a condition for obtaining an approximate reduction of one generative model in terms of a map function between the latent spaces of each variational autoencoder.

8.1 Δ-Reduction of generative models

Consider the case in which two variational autoencoders have been trained with dataset $\mathcal{X}$ and are characterized by the corresponding quadruples $(Z_1, \mathbb{P}_{Z_1}, \mathbb{Q}_{Z_1}^{\boldsymbol{\alpha_1^*}(\cdot)}, \mathbb{P}_X^{\boldsymbol{\beta_1^*}(\cdot)})$ and $(Z_2, \mathbb{P}_{Z_2}, \mathbb{Q}_{Z_2}^{\boldsymbol{\alpha_2^*}(\cdot)}, \mathbb{P}_X^{\boldsymbol{\beta_2^*}(\cdot)})$. We will define a condition for approximately reducing the corresponding generative models in terms of a mapping function between latent spaces.

Intuitively, a datapoint $x \in X$ has a representation in terms of the latent variables in Z_1 and Z_2 learned by the corresponding variational autoencoders. In order to reduce the generative model in terms of another we would like to find a mapping function that relates the latent variables in Z_1 and Z_2 that are associated to a same datapoint $x \in X$ on average.

Definition 8.1.1. ***Δ- Reduction of generative models***
Let $(Z_1, \mathbb{P}_{Z_1}, \mathbb{Q}_{Z_1}^{\boldsymbol{\alpha_1^*}(\cdot)}, \mathbb{P}_X^{\boldsymbol{\beta_1^*}(\cdot)})$ and $(Z_2, \mathbb{P}_{Z_2}, \mathbb{Q}_{Z_2}^{\boldsymbol{\alpha_2^*}(\cdot)}, \mathbb{P}_X^{\boldsymbol{\beta_2^*}(\cdot)})$ represent two variational autoencoders trained with a dataset $\mathcal{X}$. Let $\mathbb{P}_{X\times Z_1}$ and $\mathbb{P}_{X\times Z_2}$ denote the generative models associated to the corresponding variational autoencoders.

Let $M_{1;2} : Z_1 \mapsto Z_2$ be a continuous function that maps elements from latent space Z_1 to Z_2. We say that the function $M_{1;2}$ is a Δ-reductor of the generative model $P_{X\times Z_1}$ from Z_1 into Z_2 if for the value $\Delta \in \mathbb{R}^+$, then

$$\mathrm{KL}(\mathbb{P}_{X\times Z_1}||\mathbb{P}_{X\times M_{1;2}(Z_1)}) \leq \Delta. \tag{8.2}$$

Here the generative model $\mathbb{P}_{X\times M_{1;2}(Z_1)}$ is defined in terms of the probability density function $P_{X\times M_{1;2}(Z_1)} : X \times Z_1$ given by

$$P_{X\times M_{1;2}(Z_1)}(x, z) = P_X^{\boldsymbol{\beta_2^*}(M_{1;2}(z))}(x) P_{Z_1}(z). \tag{8.3}$$

By symmetry, this definition also applies to a Δ-reductor $M_{2;1} : Z_2 \mapsto Z_1$.

The condition presented in Equation (8.2) determines that a generative model with a latent space Z_1 obtained by training a variational autoencoder can be expressed in terms of another over the latent space Z_2 by using a suitable mapping function $M_{1;2}$. This mapping $M_{1;2}$ is such that these generative models are close to one another up to a tolerance level Δ with respect to the Kullback-Leibler divergence.

In the next section we will discuss a method for estimating the value of the reduction condition of Equation (8.2) for two trained variational autoencoders.

8.2 Equivalent Δ-reduction condition

The integral over the latent space Z_1 involved in the Kullback-Leibler divergence condition of Equation (8.2) is intractable due to the use of the neural networks to calculate the parameters of the decoding distributions since it involves an integral over the complete latent space Z_1. Thus, we will show an alternative form of the condition for Δ-reduction that can be approximated via Monte Carlo sampling. This condition will result useful for the construction of the interpretation mappings as it is presented in the following sections.

Lemma 4. ***Equivalent Δ-reduction condition***
Consider two trained variational autoencoders represented by the quadruples $(Z_1, \mathbb{P}_{Z_1}, \mathbb{Q}_{Z_1}^{\boldsymbol{\alpha_1^*}(\cdot)}, \mathbb{P}_X^{\boldsymbol{\beta_1^*}(\cdot)})$ and $(Z_2, \mathbb{P}_{Z_2}, \mathbb{Q}_{Z_2}^{\boldsymbol{\alpha_2^*}(\cdot)}, \mathbb{P}_X^{\boldsymbol{\beta_2^*}(\cdot)})$. For a constant $\Delta \in \mathbb{R}^+$ the Δ-reductor $M_{1;2} : Z_1 \mapsto Z_2$ fulfills,

$$\mathbb{E}_{\mathbb{P}_{Z_1}}\left[\mathrm{KL}(\mathbb{P}_X^{\boldsymbol{\beta_1}(\cdot)}||\mathbb{P}_X^{\boldsymbol{\beta_2}(M_{1;2}(\cdot))})\right] \leq \Delta. \tag{8.4}$$

Proof. In the Δ-reduction definition the left hand side of the condition in Equation (8.2) corresponds to

$$\mathrm{KL}\left(\mathbb{P}_{X\times Z_1}||\mathbb{P}_{X\times M_{1;2}(Z_1)}\right) = \int_{Z_1}\int_X P_{X\times Z_1}(x,z)\log\left(\frac{P_{X\times Z_1}(x,z)}{P_{X\times M_{1;2}(Z_1)}(x,z)}\right)d\mathcal{L}^D(x)d\mathbb{P}_{Z_1}(z) \quad (8.5)$$

By substituting the value of the probability densities for the variational autoencoder generative models into the logarithm we obtain the following equivalence

$$\frac{P_{X\times Z_1}(x,z)}{P_{X\times M_{1;2}(Z_1)}(x,z)} = \frac{P_X^{\boldsymbol{\beta_1}(z)}(x)P_{Z_1}(z)}{P_X^{\boldsymbol{\beta_2}(M_{1;2}(z))}(x)P_{Z_1}(z)} = \frac{P_X^{\boldsymbol{\beta_1}(z)}(x)}{P_X^{\boldsymbol{\beta_2}(M_{1;2}(z))}(x)} \quad (8.6)$$

Thus, after substituting the previous result into Equation (8.5) we obtain the formula

$$\mathrm{KL}\left(\mathbb{P}_{X\times Z_1}||\mathbb{P}_{X\times M_{1;2}(Z_1)}\right) = \int_{Z_1} P_{Z_1}(z)\int_X P_X^{\boldsymbol{\beta_1}(z)}(x)\log\left(\frac{P_X^{\boldsymbol{\beta_1}(z)}(x)}{P_X^{\boldsymbol{\beta_2}(M_{1;2}(z))}(x)}\right)d\mathcal{L}^D(x)d\mathbb{P}_{Z_1}(z) \quad (8.7)$$

Notice that right hand side of Equation (8.7) corresponds to the left hand side for the condition in Equation (8.4), therefore for a Δ-reductor $M_{1;2}$,

$$\mathbb{E}_{\mathbb{P}_{Z_1}}\left[KL(\mathbb{P}_X^{\boldsymbol{\beta_1}(\cdot)}||\mathbb{P}_X^{\boldsymbol{\beta_2}(M_{1;2}(\cdot))})\right] = \mathrm{KL}(\mathbb{P}_{X\times Z_1}||\mathbb{P}_{X\times M_{1;2}(Z_1)}) \leq \Delta. \quad (8.8)$$

□

For a given Δ-reductor mapping $M_{1;2} : Z_1 \mapsto Z_2$ we can calculate the condition for Δ-reduction by approximating its value via the Monte Carlo method presented in Section 3.2 through the sampling of elements in Z_1. The equivalent Δ-reduction condition presented in Lemma 4 is therefore approximated by taking L samples $\{z^{(l)}\}_{l=1}^L$ according to the prior $\mathbb{P}_{Z_1}$ and averaging the Kullback-Leibler divergence with respect to these samples as

$$\mathbb{E}_{\mathbb{P}_{Z_1}}\left[\mathrm{KL}(\mathbb{P}_X^{\boldsymbol{\beta_1}(\cdot)}||\mathbb{P}_X^{\boldsymbol{\beta_2}(M_{1;2}(\cdot))})\right] \approx \frac{1}{L}\sum_{l=1}^{L}\mathrm{KL}(\mathbb{P}_X^{\boldsymbol{\beta_1}(z^{(l)})}||\mathbb{P}_X^{\boldsymbol{\beta_2}(M_{1;2}(z^{(l)})})). \quad (8.9)$$

In the next section we will present the value of the Kullback-Leibler divergence between the decoder distributions of two variational autoencoders with parametric families of normal distributions. This result will be used to approximate the Δ-reduction condition of Equation (8.9) between the baseline variational autoencoder and the diffusion autoencoder.

8.3 Normal parametric decoding distributions reduction

Consider the case in which the decoding probability distributions of the two variational autoencoders $\mathbb{P}_X^{\boldsymbol{\beta_1}(\cdot)}$ and $\mathbb{P}_X^{\boldsymbol{\beta_2}(\cdot)}$ are members of a parametric family of normal distributions. In this section we will analyze the Δ-reduction condition for this particular case.

Lemma 5. ***Kullback-Leibler divergence between normal distributions***

Consider two normal probability distributions over $X = \mathbb{R}^D$ given by $\mathbb{P}_X^{(\mu_1,\Sigma_1)}$ and $\mathbb{P}_X^{(\mu_2,\Sigma_2)}$. The Kullback-Leibler divergence of $\mathbb{P}_X^{(\mu_1,\Sigma_1)}$ with respect to $\mathbb{P}_X^{(\mu_2,\Sigma_2)}$ corresponds to

$$\mathrm{KL}(P_X^{(\mu_1,\Sigma_1)}||P_X^{(\mu_2,\Sigma_2)}) = \frac{1}{2}\left(\mathrm{tr}(\Sigma_2^{-1}\Sigma_1) + (\mu_2-\mu_1)^T\Sigma_2^{-1}(\mu_2-\mu_1) - D + \log\left(\frac{\det\Sigma_2}{\det\Sigma_1}\right)\right) \quad (8.10)$$

Corollary 6. *The Kullback-Leibler divergence between two normal probability distributions* $\mathbb{P}_X^{(\mu_1,\sigma_1^2)}$ *and* $\mathbb{P}_X^{(\mu_2,\sigma_2^2)}$ *corresponds to*

$$\mathrm{KL}(\mathbb{P}_X^{(\mu_1,\sigma_1^2)}||\mathbb{P}_X^{(\mu_2,\sigma_2^2)}) = \frac{1}{2\sigma_2^2}\|\mu_1-\mu_2\|_2^2 + \frac{D}{2}\left(\log\left(\frac{\sigma_2^2}{\sigma_1^2}\right)+\frac{\sigma_1^2}{\sigma_2^2}-1\right). \tag{8.11}$$

Now, consider the trained neural network functions $\boldsymbol{\mu}^*_{\mathbf{X,1}} : Z_1 \mapsto \mathbb{R}^2$ and $\boldsymbol{\mu}^*_{\mathbf{X,2}} : Z_2 \mapsto \mathbb{R}^2$ that calculate the parameters for the decoding distributions of two variational autoencoders. The Δ-reduction condition approximation presented in Equation (8.9) is calculated with respect to L sampled latent variables according to $\mathbb{P}_{Z_1}$ and becomes

$$\begin{aligned}\mathbb{E}_{\mathbb{P}_{Z_1}}\left[KL(\mathbb{P}_X^{(\boldsymbol{\mu}^*_{\mathbf{X;1}}(\cdot),\sigma_1^2)}||\mathbb{P}_X^{(\boldsymbol{\mu}^*_{\mathbf{X;2}}(M_{1;2}(\cdot)),\sigma_2^2)})\right] &\approx \frac{1}{2L\sigma_2^2}\sum_{l=1}^{L}\|\boldsymbol{\mu}^*_{\mathbf{X;1}}(z^{(l)}) - \boldsymbol{\mu}^*_{\mathbf{X;2}}(M_{1;2}(z^{(l)}))\|_2^2 \\ &+ \frac{D}{2}\left(\log\left(\frac{\sigma_2^2}{\sigma_1^2}\right)+\frac{\sigma_1^2}{\sigma_2^2}-1\right) \leq \Delta. \end{aligned} \tag{8.12}$$

In this chapter we have established the definition of Δ-reduction for the variational autoencoders that have normal decoder distributions. In the next chapter we will focus in determining an algorithm for the construction of the Δ-reductors between latent spaces.

Chapter 9

Construction of Simple Reduction Mappings

Consider the case in which given two variational autoencoders with normal parametric decoding distributions we want to reduce the corresponding generative model $\mathbb{P}_{X\times Z_1}$ from Z_1 to Z_2 via a suitable Δ-reductor map $M_{1;2}$ that fulfills the conditions of Δ-reducibility. Intuitively we can think that we can construct an arbitrarily complex mapping that identifies the structure of latent space Z_1 with the structure of Z_2, thus achieving an arbitrarily low tolerance level Δ. We are interested in answering the question on whether we can find *simple* Δ-reductor functions that achieve a low Δ-reduction tolerance.

With the purpose of constructing an appropriate reduction mapping, we posit a parametric family of low complexity linear mappings between latent space Z_1 and Z_2 such that each member is a Δ-reductor denoted by $M_{1;2}^{(\lambda)} : Z_2 \mapsto Z_1$ with parameters $\lambda \in \Lambda$. Our main goal is to identify the member of such family that produces the lowest tolerance level Δ. For the decoding distributions with fixed values σ_1, σ_2, the parameters λ^* that determine the lowest tolerance Δ-reductor fullfill the condition

$$\lambda^* = \arg\min_{\lambda\in\Lambda} \mathbb{E}_{\mathbb{P}_{Z_1}} \left[\|\boldsymbol{\mu}^*_{\boldsymbol{X;2}}(M_{1;2}^{(\lambda)}(\cdot)) - \boldsymbol{\mu}^*_{\boldsymbol{X;1}}(\cdot)\|_2^2 \right]. \tag{9.1}$$

The functions $\boldsymbol{\mu}^*_{\boldsymbol{X;1}} : Z_1 \mapsto X$ and $\boldsymbol{\mu}^*_{\boldsymbol{X;2}} : Z_2 \mapsto X$ correspond to the neural networks that calculate the location parameters for the corresponding variational autoencoder. The corresponding mapping $M_{1;2}^{(\lambda^*)}$ satisfies the Δ-reduction condition with a tolerance level given by the formula

$$\Delta = \mathbb{E}_{\mathbb{P}_{Z_1}} \left[\frac{1}{2\sigma_2^2} \|\boldsymbol{\mu}_{\boldsymbol{X;2}}(M_{1;2}^{(\lambda^*)}(z)) - \boldsymbol{\mu}_{\boldsymbol{X;1}}(z)\|_2^2 \right] + \frac{D}{2}\left(\log\left(\frac{\sigma_2^2}{\sigma_1^2}\right) + \frac{\sigma_1^2}{\sigma_2^2} - 1 \right). \tag{9.2}$$

The first term can be approximated via the Monte Carlo method as it is presented in Equation (8.12). Thus, we will restrict ourselves to simple parametric families of functions.

The condition of Equation (9.1) is visualized in Figure 9.1 and states that if we sample latent variables from Z_1 according to $\mathbb{P}_{Z_1}$, the data obtained by following the red path and the blue path are close up to a certain tolerance level Δ on average.

$$\begin{tikzcd}
Z_1 \arrow[dr, twoheadrightarrow, "\boldsymbol{\mu}^*_{\mathbf{X};\mathbf{1}}"] \arrow[dd, bend right, "M_{1;2}^{(\lambda)}"'] & \\
& X \\
Z_2 \arrow[ur, twoheadrightarrow, "\boldsymbol{\mu}^*_{\mathbf{X};\mathbf{2}}"'] &
\end{tikzcd}$$

Figure 9.1: Diagram that represents a generative process in which latent variables are sampled from Z_1 according to $\mathbb{P}_{Z_1}$ and are mapped into Z_2 and X with the corresponding functions by following the arrows. The double arrows represent neural networks. The condition in Equation (9.1) states that the datapoints obtained by following the red path should be close to the results obtained by following the blue one on average up to a certain tolerance Δ.

As it was stated in Section 8.2, the expected value included in the Δ-reduction condition can be estimated via the Monte Carlo method by sampling L elements from Z_1 according to the prior $\mathbb{P}_{Z_1}$. We can optimize the parameters λ by perfoming stochastic gradient descent by minimizing the right hand side of Equation (9.1) via backpropagation through the decoding neural networks $\boldsymbol{\mu}^*_{\mathbf{X};\mathbf{1}}$, $\boldsymbol{\mu}^*_{\mathbf{X};\mathbf{2}}$. Note that the gradient is only estimated with respect to the parameters of the proposed $M_{1;2}^{(\lambda)}$.

```
Input: $\boldsymbol{\mu}^*_{\mathbf{X};\mathbf{1}}$ and $\boldsymbol{\mu}^*_{\mathbf{X};\mathbf{2}}$
Result: $\lambda^*$
Initialize $\lambda$;
repeat
    for $l = 1, 2, \dots, L$ do
        Sample latent variable $z^{(l)}$ from $Z_1$ according to $\mathbb{P}_{Z_1}$;
        Store value of $\|\boldsymbol{\mu}^*_{\mathbf{X};\mathbf{2}}(M_{1;2}^{(\lambda)}(z^{(l)})) - \boldsymbol{\mu}^*_{\mathbf{X};\mathbf{1}}(z^{(l)})\|_2^2$;
    end
    Update $\lambda$ according to $\nabla_\lambda(\frac{1}{L}\sum_{l=1}^{L} \|\boldsymbol{\mu}^*_{\mathbf{X};\mathbf{2}}(M_{1;2}^{(\lambda)}(z^{(l)})) - \boldsymbol{\mu}^*_{\mathbf{X};\mathbf{1}}(z^{(l)})\|_2^2)$
until Convergence of $\lambda$;
```

Algorithm 2: Construction of Δ reduction mapping $M_{1;2}^{(\lambda^*)}$via parameter optimization.

The algorithm for identifying reduction mappings can be used between any generative model that can be described in terms of functions that are differentiable almost everywhere since the gradients of the loss function in the Algorithm 2 can be backpropagated. In particular, the generative model used to produce the benchmark dataset of Chapter 5 is differentiable everywhere. In the next section we will analyze the recovered reductions between the generative model that produced the benchmark dataset together with the obtained from the baseline and diffusion variational autoencoders.

9.1 Reduction of generative models for the benchmark dataset

In the previous chapters we have trained and analyzed the recovered latent structure for the proposed baseline and diffusion variational autoencoder with respect to a benchmark dataset generated from a known latent space $Z = \Phi$ via a proposed generative model.

In this section we will describe the parametric reduction mappings that we propose between the latent spaces of the baseline variational autoencoder $Z = \mathbb{R}^2$ and the diffusion variational autoencoder $Z = \mathcal{S}^1$. Moreover, the described reduction method will also be applied to the known generative model that produced the benchmark dataset from the latent space $Z = \Phi$ shown in Chapter 5. The corresponding reduction mappings and connections between the latent spaces are visualized in the diagram of Figure 9.2.

$$\begin{tikzcd}
 & \mathcal{S}^1 \arrow[dl, bend right, "M^{(\theta')}_{\mathcal{S}^1;\mathbb{R}^2}"'] \arrow[dr, bend left=40, "M^{(\theta')}_{\mathcal{S}^1;\Phi}"] & \\
\mathbb{R}^2 \arrow[ur, bend left=40, "M^{(\theta')}_{\mathbb{R}^2;\mathcal{S}^1}"] \arrow[rr, bend left, "M^{(\theta')}_{\mathbb{R}^2;\Phi}"] & & \Phi \arrow[ul, bend right, "M^{(\theta')}_{\Phi;\mathcal{S}^1}"'] \arrow[ll, bend left, "M^{(\theta')}_{\Phi;\mathbb{R}^2}"]
\end{tikzcd}$$

Figure 9.2: Δ-reductor mappings between latent spaces of the baseline variational autoencoder $Z = \mathbb{R}^2$, diffusion variational autoencoder $Z = \mathcal{S}^1$ and the latent space of the original generative model $Z = \Phi$.

We propose low complexity mappings between the latent spaces, these are enlisted in Table 9.1. These mappings can be classified into two types: the matrix multiplication mappings with parameter $C \in \mathbb{R}^{2\times 2}$ and the angle rotation mappings with parameter $\theta' \in \mathbb{R}$. Some general intuition of the behavior of these mappings is presented in the next sections.

Table 9.1: Functional form of the reduction mappings between the latent spaces of the $\mathbb{R}^2$, $\mathcal{S}^1$ and Φ together with the corresponding parameters.

Mapping	**Parameter**	**Functional Form**
$M^{(C)}_{\mathcal{S}^1;\mathbb{R}^2}$	$C \in \mathbb{R}^{2\times 2}$	$C \cdot z$
$M^{(\theta')}_{\mathcal{S}^1;\Phi}$	$\theta' \in \mathbb{R}$	$(\text{atan2}(z) + \theta' + \pi)\text{mod}(2\pi) - \pi$
$M^{(\theta')}_{\mathbb{R}^2;\mathcal{S}^1}$	$\theta' \in \mathbb{R}$	$\text{Emb}_{\mathbb{R}^2}(\text{atan2}(z) + \theta')$
$M^{(\theta')}_{\mathbb{R}^2;\Phi}$	$\theta' \in \mathbb{R}$	$(\text{atan2}(z) + \theta' + \pi)\text{mod}(2\pi) - \pi$
$M^{(C)}_{\Phi;\mathbb{R}^2}$	$C \in \mathbb{R}^{2\times 2}$	$C \cdot \text{Emb}_{\mathbb{R}^2}(z)$
$M^{(\theta')}_{\Phi;\mathcal{S}^1}$	$\theta' \in \mathbb{R}$	$\text{Emb}_{\mathbb{R}^2}(z + \theta')$

The special function atan2 : $\mathbb{R}^2 \mapsto \Theta$ takes a vector in $\mathbb{R}^2$ and returns its corresponding angle in the range $\Theta = [-\pi, \pi)$. The modulo operation is denoted as $(\cdot)\text{mod}(\cdot)$ with the first entry the dividend and the second entry the divisor. The embedding function corresponds to $\text{Emb}_{\mathbb{R}^2} : \mathbb{R} \mapsto \mathbb{R}^2$,

$$\text{Emb}_{\mathbb{R}^2}(z) = (\cos(z), \sin(z)). \tag{9.3}$$

9.2 Matrix multiplication reductor mapping

The matrix multiplication reductor mappings have a parameter matrix $C \in \mathbb{R}^{2\times 2}$. By performing singular value decomposition of matrix C we obtain the matrices U, S, V^T such that $C = U \cdot S \cdot V^T$. Where $U \in \mathbb{R}^2$ and $V^T \in \mathbb{R}^2$ are orthogonal matrices while $S \in \mathbb{R}^2$ corresponds to a diagonal matrix.

The transformation C can be reinterpreted as the successive rotation/reflection via matrix V^T, a scaling along the canonical directions with the diagonal matrix S and a second rotation/reflection with matrix U. Such process is visualized for the reduction mapping between the latent space $Z = \mathcal{S}^1$ of the diffusion variational autoencoder into the latent space $Z = \mathbb{R}^2$ of the baseline variational autoencoder in Figure 9.3.

Figure 9.3 shows the effects of the reduction mapping between the latent space $Z = \mathcal{S}^1$ of the diffusion variational autoencoder into the latent space $Z = \mathbb{R}^2$ of the baseline variational autoencoder. The input latent space $\mathcal{S}^1$ (top left) is transformed via the succesive transformations with matrices V^T, S and U. These transformations form the resulting matrix multiplication with C. The final transformed latent variable representation is shown in the bottom left plot.

$$M_{\mathcal{S}^1;\mathbb{R}^2}(z) = C \cdot z. \tag{9.4}$$

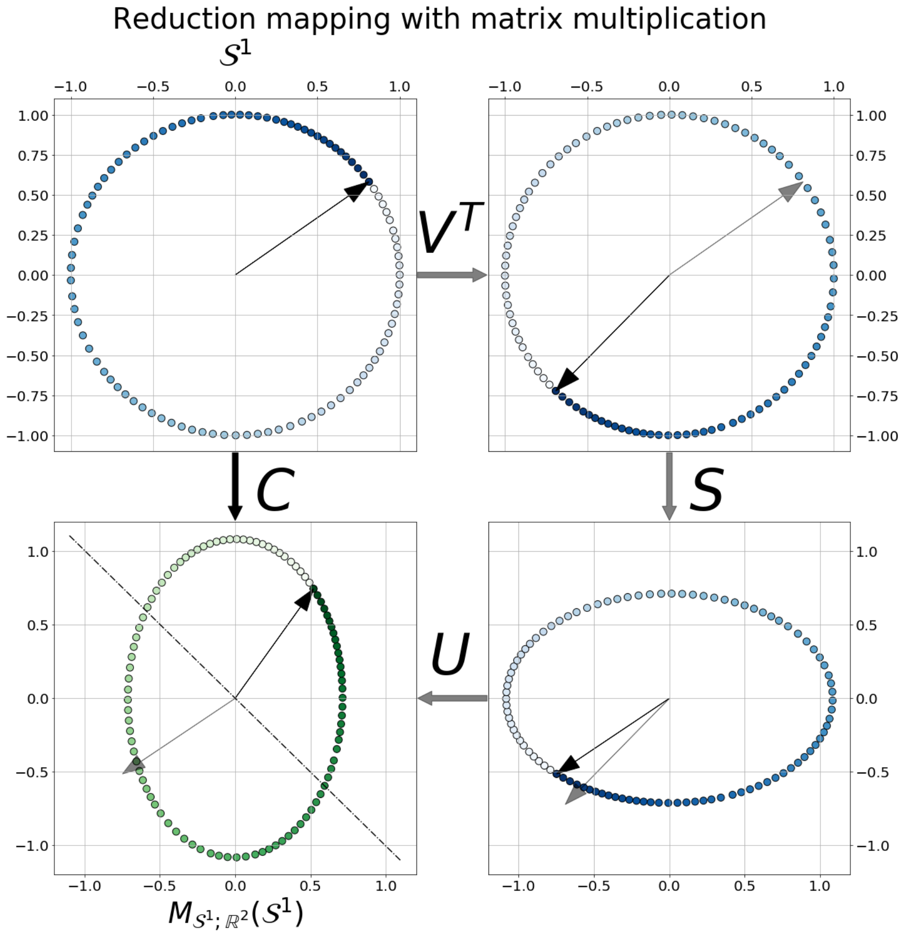

Figure 9.3: Reduction mapping $M_{\mathcal{S}^1;\mathbb{R}^2}$ between the encoded latent representation in $\mathcal{S}^1$ of dataset $\mathcal{X}_{\text{Vis}}$ into $\mathbb{R}^2$ via matrix multiplication with $C \in \mathbb{R}^{2\times2}$. The encoded latent representation for each datapoint $x \in \mathcal{X}$ calculated as $\boldsymbol{\mu}_{\mathcal{S}^1}(x)$ is transformed via the optimal matrix $C = USV^T$. The matrix multiplication is the result of first a rotation with matrix V^T, then a scaling with respect to the principal axes with S and a further reflection with matrix U. The effect of the matrix components that forms C is visualized. The black arrow points at the datapoint corresponding to phase 2π helps visualize the effects of transforming the latent variables.

9.3 Angle rotation reductor mappings

The rotation reductor mappings have a parameter $\theta' \in \mathbb{R}$ that represents an angle rotation of the latent variables in the input latent space.

Figure 9.4 shows the effects of the reduction mapping between the latent space $\mathbb{R}^2$ into Φ. For the sake of visualization, the resulting latent variables in Φ are embedded into $\mathbb{R}^2$ via the embedding function $\mathrm{Emb}_{\mathbb{R}^2}$. The input latent variable representation of dataset $\mathcal{X}_{\mathrm{Vis}}$ (top left) is first projected into the unit circle and then rotated by an angle θ' into the plot in the bottom right

$$M_{\mathbb{R}^2;\Phi}(z) = (\mathrm{atan2}(z) + \phi' + \pi)\mathrm{mod}(2\pi) - \pi. \tag{9.5}$$

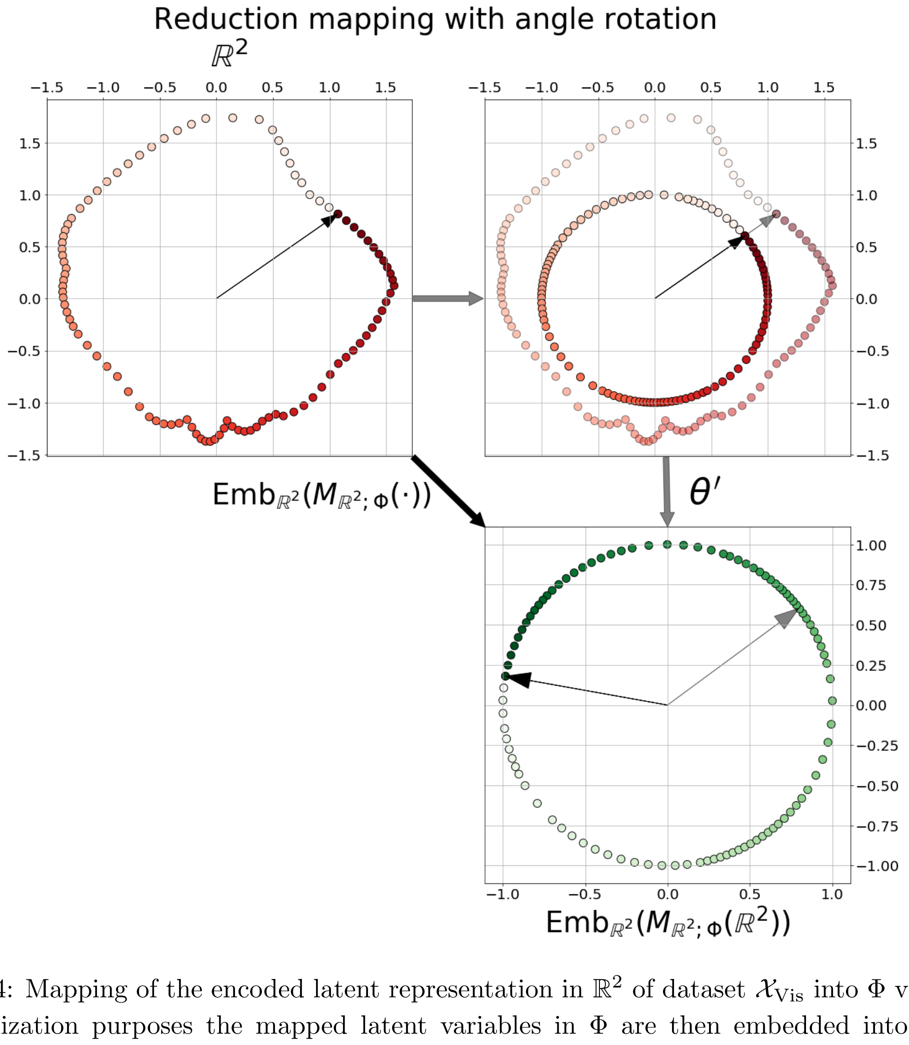

Figure 9.4: Mapping of the encoded latent representation in $\mathbb{R}^2$ of dataset $\mathcal{X}_{\mathrm{Vis}}$ into Φ via $M^{\theta'}_{\mathbb{R}^2;\Phi}$, for visualization purposes the mapped latent variables in Φ are then embedded into $\mathbb{R}^2$ with $\mathrm{Emb}_{\mathbb{R}^2}$. The encoded latent representation for each datapoint $x \in \mathcal{X}_{N'}$ calculated as $\boldsymbol{\mu}^*_{\boldsymbol{Z}_1}(x)$ is transformed by first projecting to the unit circle in $\mathbb{R}^2$ and then rotating the projection within the circle by an angle θ'. The black arrow which points at the datapoint corresponding to phase 2π helps visualize the effects of transforming the latent variables

9.4 Reduction mappings for benchmark dataset

In this section we present the results obtained for the trained reduction maps between the latent spaces of the baseline and the diffusion variational autoencoders trained with dataset $\mathcal{X}_{0.1}$ and parameter $\sigma_X = 0.01$. We visualize the effects of transforming the latent spaces with respect to the encoded datapoints from the dataset $\mathcal{X}_{\text{Vis}}$.

The reduction maps are trained over 10000 epochs by taking $L = 10000$ samples over the latent space according to the corresponding prior, see Algorithm 2. For each reduction map, three repetitions are trained in order to assess the repeatability of the maps.

Recall that the latent representation of a dataset $\mathcal{X}_{\text{Vis}}$ corresponds to the set of latent variables obtained from the encoding distribution's mean calculated with the trained neural network $\boldsymbol{\mu}_{\boldsymbol{Z}}(x)$. In the case of the benchmark generative latent space Φ, the latent representations correspond to the phases $\varphi \in \Phi$ associated to each datapoint in $\mathcal{X}_{\text{Vis}}$ which, for the sake of visualization, are embedded into $\mathbb{R}^2$ via $\text{Emb}_{\mathbb{R}^2}$.

Figure 9.5 shows the mapped latent representations from Z_1 to Z_2 via $M_{Z_1;Z_2}$ where Z_1 is an input latent space and Z_2 is the target. In each case the mapped latent representation $M_{Z_1;Z_2}(\boldsymbol{\mu}_{\boldsymbol{Z_1}}(x))$ of each datapoint $x \in \mathcal{X}_{\text{Vis}}$ is shown with a green hue. Baseline variational autoencoder latent representations are shown in red. Diffusion variational autoencoder latent representations are shown in blue.

As a qualitative result we can identify that the learned reduction mappings map latent variable representations with the same underlying phases close to one another. This is seen in the similarity of the color hues of the mapped latent variable representation compared to the target.

We have also calculated the tolerance level Δ corresponding to each of the reduction maps between the corresponding latent spaces. The values are shown in Table 9.2. Three repetitions for each reduction map are trained and the resulting tolerance is averaged. The uncertainty interval is calculated with the standard deviation of the measurements.

Table 9.2: Δ-tolerance for the trained reduction mappings from latent space Z_1 into Z_2.

$M_{Z_1;Z_2}$		Z_2		
		$\mathbb{R}^2$	$\mathcal{S}^1$	Φ
Z_1	$\mathbb{R}^2$	N/A	91.36 ± 2.16	70.62 ± 2.04
	$\mathcal{S}^1$	10.37 ± 0.23	N/A	13.45 ± 0.36
	Φ	6.59 ± 0.22	13.77 ± 0.22	N/A

Notice from Table 9.2 that the tolerance values with an input latent space $Z_1 = \mathbb{R}^2$ have a bigger uncertainty interval, this can be a result of the sampling of latent space $\mathbb{R}^2$ with the wide-spread normal standard prior. Possibly more iterations were needed to obtain more consistent results. Nevertheless we have decided to give all the reduction maps the same training conditions parameters.

The highest values of Δ are obtained for the reduction maps with latent space $Z_1 = \mathbb{R}^2$. The shapes of the recovered latent representation for dataset $\mathcal{X}_{\text{Vis}}$ are more intricate and therefore the projection into the unit circle might be not completely adequate.The Δ values for the remaining reduction maps with $Z_1 = \mathcal{S}^1$ and $Z_1 = \Phi$ are very similar with a smaller value compared to those of $Z_1 = \mathbb{R}^2$.

We have presented a benchmark dataset for datasets with underlying circular geometry. We will test in the next chapter for different example datasets the result obtained by training the baseline and difussion variational autoencoders to test whether the underlying geometrical structure can be recovered.

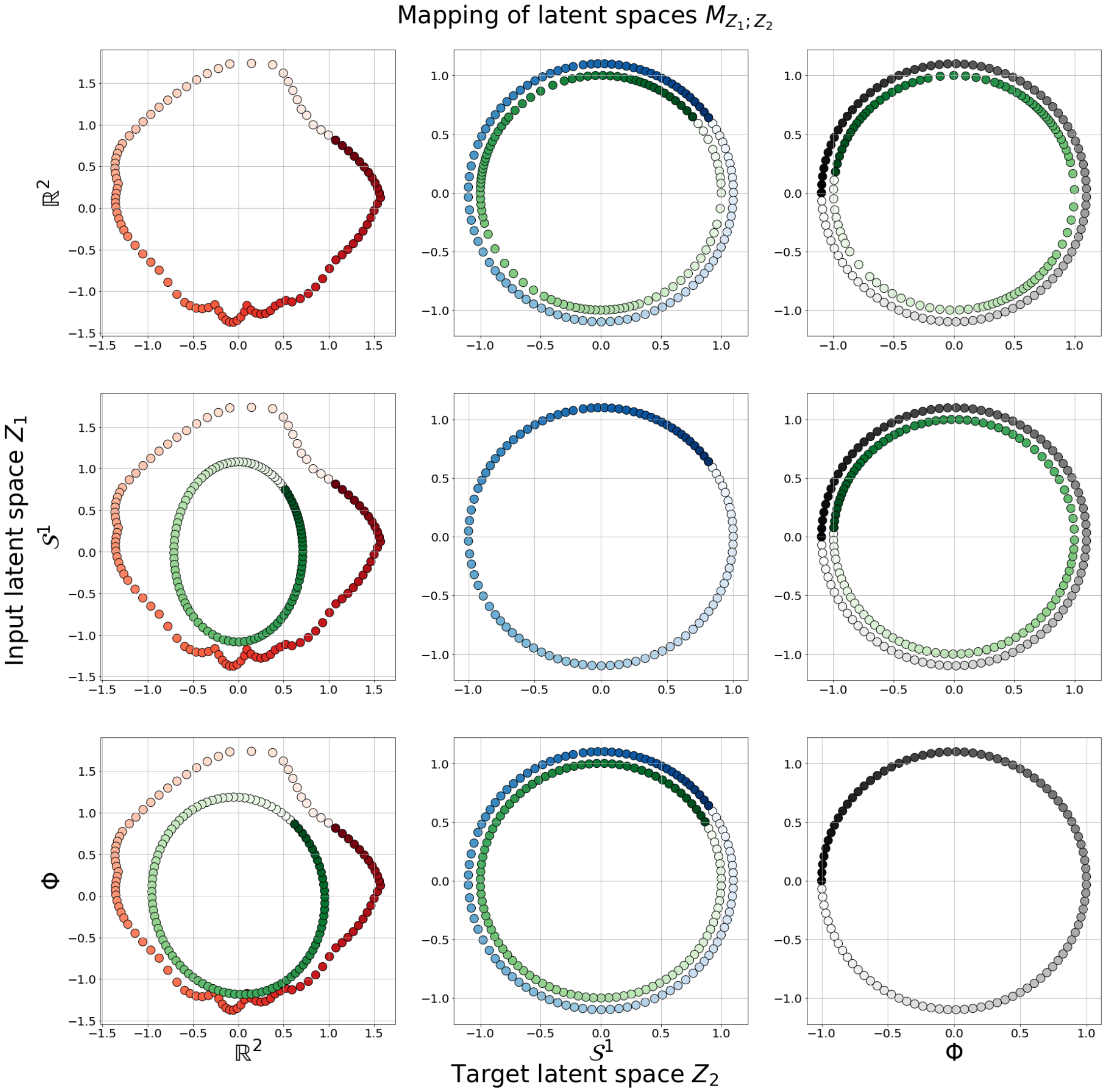

Figure 9.5: Mapping of the encoded latent representation of dataset $\mathcal{X}_{\text{Vis}}$ in Z_1 into Z_2 via the corresponding learned reduction map $M_{Z_1;Z_2}$. The plot in the i-th row and the j-th column shows the mapped latent space representation of the i-th latent space into the j-th latent space together with the original latent representation of the j-th latent space for comparison. Plots are the representations of dataset $\mathcal{X}_{\text{Vis}}$ in the corresponding latent space. Each latent space representation has a corresponding color code. Red: baseline variational autoencoder. Blue: diffusion variational autoencoder. Gray: benchmark generative model. Green: Mapped latent representation. Color hue represents the underlying phases associated to dataset $\mathcal{X}_{\text{Vis}}$. Latent variables in latent space Φ are embedded in $\mathbb{R}^2$ with function $\text{Emb}_{\mathbb{R}^2}$ for the sake of visualization, moreover latent representations in the unit circle are scaled to allow the comparison between representations.

Chapter 10

Circular Dataset Examples

The benchmark dataset used in the previous chapters provided a basis for comparing results with respect to more complex datasets that share a similar underlying circular structure. In this chapter we seek to test whether the proposed baseline and diffusion variational autoencoder is capable of recovering this underlying circular structure for different datasets. Additionally for the trained variational autoencoders recovered in each example, we test the developed method for constructing reduction mappings between the latent spaces of the baseline and diffusion variational autoencoders. The importance of focusing in these types of problems lies in the wide range of applications that can be described with respect to circular variables [22].

10.1 Circular pixel shift

Consider a gray-scale image which can be represented as a $H \times W$ matrix $I \in [0,1]^{H \times W}$ consisting of H pixels of height and W pixels of width with entry values $I_{i,j} \in [0,1]$ for $i \in [H]$ and $j \in [W]$ where $[H] = \{1, 2, \ldots, H\}$ and $[W] = \{1, 2, \ldots, W\}$.

In this section we present a dataset consisting of gray-scale images from dataspace $X = [0,1]^{H \times W}$ with an underlying circular latent structure enforced by a function $F : [W] \mapsto [0,1]^{H \times W}$ that shifts an image by a certain amount of pixels in a circular fashion. Each individual component of the function is determined in terms of the entries of the image. For a pixel shift s, the i, j-th resulting pixel value corresponds to

$$f_{i,j}(s) = I_{i,[(j+s)\mathrm{mod}(W)]+1}$$

A complete dataset is thus a set of W images described by

$$\mathcal{X} = \left\{ F(s) \in [0,1]^{H \times W} \mid s \in [W] \right\} \tag{10.1}$$

Figure 10.1 shows a representation of some example images obtained for different values of $F(s)$ with s the corresponding number of pixels shifted in the horizontal direction.

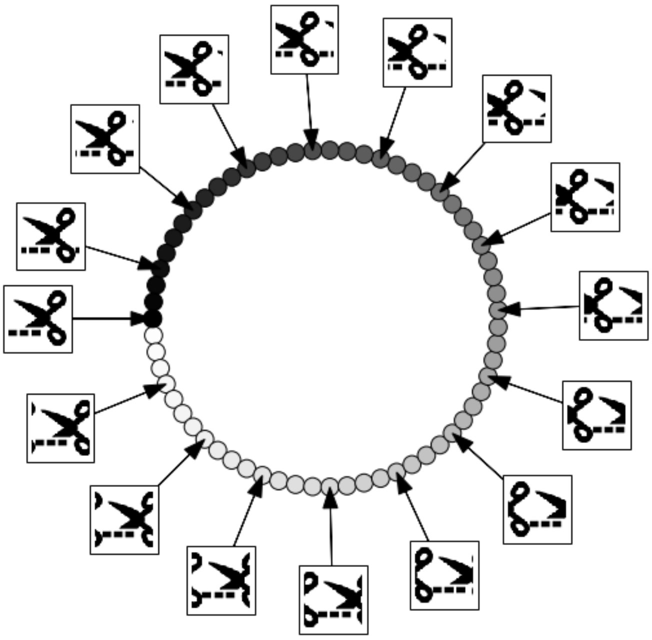

Figure 10.1: Shifted images obtained with function $F(s)$ for different pixel shift values s. The underlying pixel shift value s is represented by embedding it into the unit circle.

10.1.1 Results

Both the baseline and the diffusion variational autoencoders are trained for 100000 epochs with a dataset consisting of 64 pixel shift images of 64×64 pixels as the ones presented in Figure 10.1. The neural network architecture presented in Section 3.5 is used, with three hidden layers for the encoding and decoding neural network. The number of neurons for each hidden dense layer corresponds to $\lfloor 64^2/3 \rfloor$. Three values for the parameter $\sigma_X = 0.01, 0.1, 1.0$ were used. The number of latent samples is chosen as $L = 1$.

Figure 10.2 presents the latent representation for the pixel shift dataset for different values of σ_X recovered with the baseline variational autoencoder. It is noticeable once again that for the parameters $\sigma_X = \{0.01, 0.1\}$ the recovered latent structure is cyclic as it was also seen in the benchmark dataset. For these values of σ_X, the standard deviation calculated from σ_Z is smaller.

For the recovered latent structures corresponding to the variational autoencoders with $\sigma_X = 1$ we obtain latent representations with higher standard deviation represented by the surrounding ellipses of each datapoint representation. Moreover we can notice that for this case the cyclical latent structure is not explicitly apparent. In this sense it appears that higher values of σ_X (which favor the Kullback-Leibler regularization term) push the distribution towards the center of the plane i.e. to the prior distribution corresponding to the standard normal distribution.

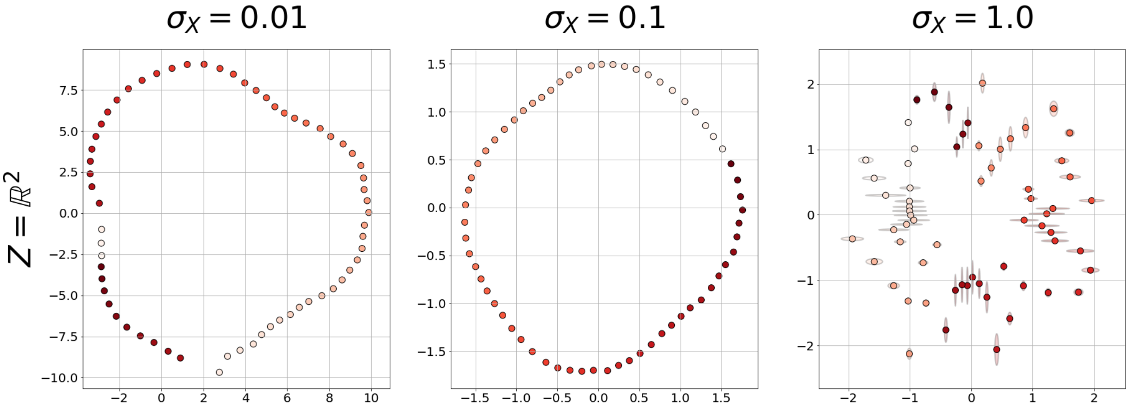

Figure 10.2: Latent representation for baseline variational autoencoder. Each point represents the calculated value for the mean $\boldsymbol{\mu_Z}$. The ellipses represent high probability regions for each data representation defined with respect to the scale parameter $\boldsymbol{\mu_Z}$. The color hue represents the different values for $s \in [W]$ in the pixel shift function.

Figure 10.3 presents the latent representation for the pixel shift dataset for different values of σ_X obtained by the diffusion variational autoencoder. In this case the behavior of the recovered latent structure for different values of σ_X appears to be the same. An important feature of the recovered latent structure is once again the periodicity of the latent variables with respect to the pixel shift corresponding to each datapoint.

One noticeable characteristic of the recovered latent representation for $\sigma_X = 0.01$ is the non-continuous representation, with respect to the phases there is no continuous loop. This type of behavior can be seen for some repetitions of the experiment and shows is possibly related to the initialization of the neural network parameters where a piece of the recovered latent structure is severed from the whole network.

One noticeable difference with respect to the benchmark dataset for both the baseline and diffusion variational autoencoder is the small uncertainty associated to the latent variable representation for each datapoint. This can be due to the lack in variability of the provided datapoints. It can be interesting to investigate for example the changes in this behavior by introducing noise to the input data.

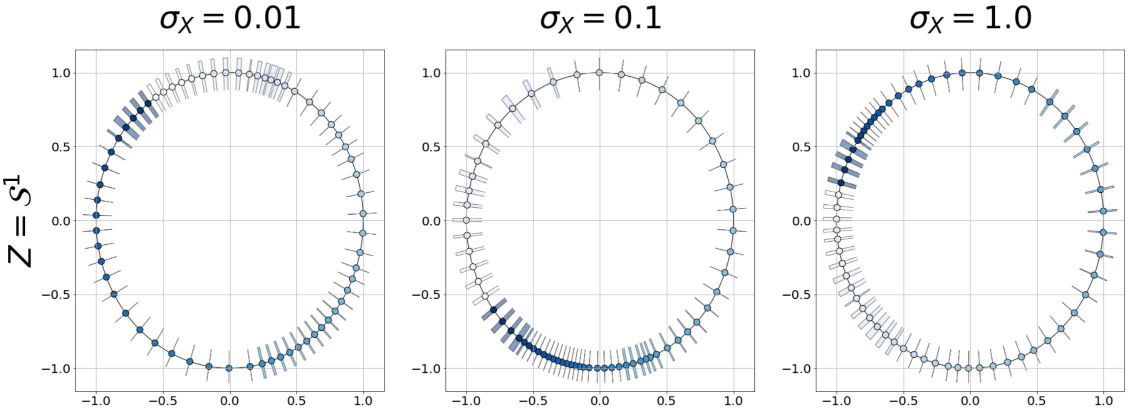

Figure 10.3: Latent representation for the diffusion variational autoencoder. Each point represents the calculated value for the mean $\mathrm{Emb}_{\mathbb{R}^2}\boldsymbol{\mu}_{\boldsymbol{\Theta}}$. The arc sections represent the high probability regions for each data representation defined with respect to the scale parameter t. The color hue represents the different values for $s \in [W]$ in the pixel shift function.

Figure 10.4 shows the recovered maps between the latent spaces of the baseline and diffusion variational autoencoder. In this case, the optimal orientations for each of the mapped structure does not fit as closely as with the benchmark dataset for latent variables with similar datapoints possibly due to the spacing between the recovered latent structure for the diffusion variational autoencoder.

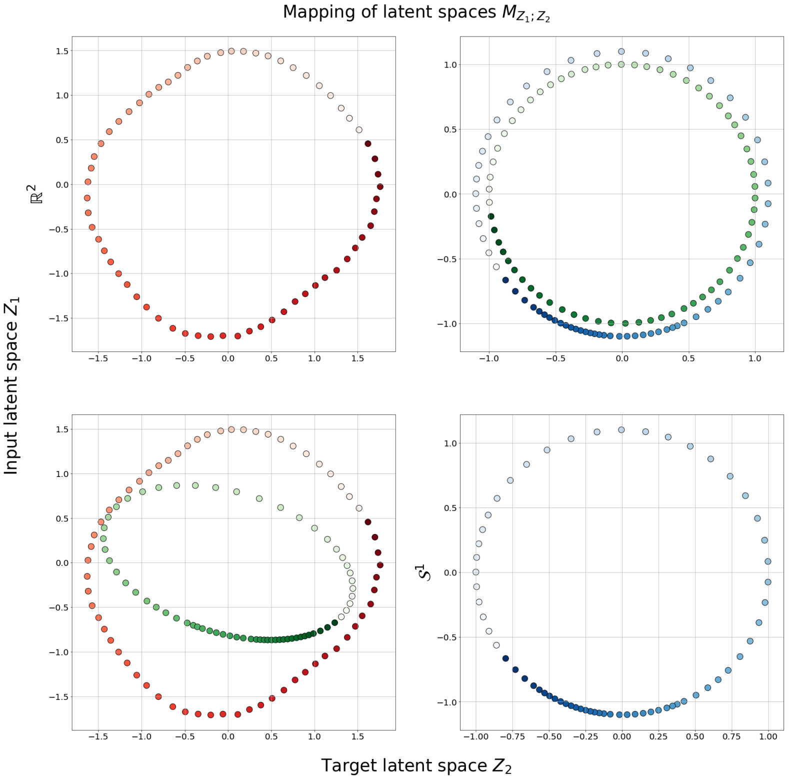

Figure 10.4: Mapping of the encoded latent representation the pixel shift dataset in Z_1 into Z_2 via the corresponding learned reduction map $M_{Z_1;Z_2}$. The plot in the i-th row and the j-th column shows the mapped latent space representation of the i-th latent space into the j-th latent space together with the original latent representation of the j-th latent space for comparison. Each latent space representation has a corresponding color code. Red: baseline variational autoencoder. Blue: diffusion variational autoencoder. Green: Mapped latent representation. Color hue represents the underlying shift associated to each image.

10.2 Objects observed from multiple angles

The Columbia University Image Library (COIL-20) [23] is a database consisting of processed grayscale photographs of 20 common objects taken at 72 different angles between 0 and 360 with 5 between each capture. Each image is a 128 × 128 pixel image.

Even though the true underlying generative process is unknown, we as humans can identify possible latent variables that explain the observations in the dataset in terms of the camera angle used to capture each photograph. Therefore it is intuitive to propose the existence of an underlying circular structure for this dataset.

Figure 10.5 shows some example images from this dataset together with the embedded angles that correspond to each image.

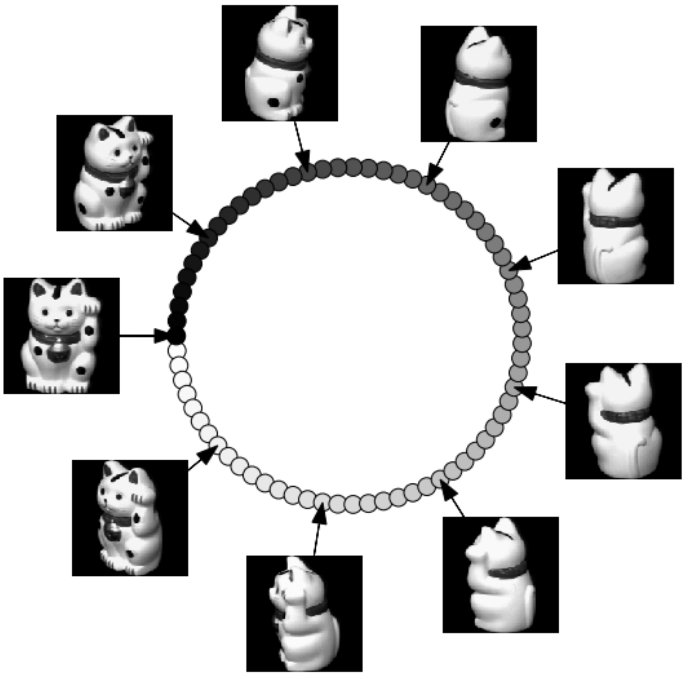

Figure 10.5: Example object observed from different angles between 0 and 360 varying the angle 5 between each capture. The underlying camera angle is represented by an embedding it into the unit circle with function $\mathrm{Emb}_{\mathbb{R}^2}$

10.2.1 Results

Both the baseline and the diffusion variational autoencoders are trained for 100000 epochs with a dataset consisting of 72 a single object images from COIL-20 dataset of 128×128 pixels as the ones presented in Figure 10.5. The neural network architecture presented in Section 3.5 is used, with two hidden layers for the encoding and decoding neural network. The number of neurons for each hidden dense layer corresponds to $\lfloor 128^2/3 \rfloor$. The number of latent samples is chosen as $L = 1$.

In this case both, the baseline and the diffusion variational autoencoder present a similar behavior to the one observed for the pixel shift as shown in Figure 10.6 and Figure 10.7. For the baseline variational autoencoder the bevarior changes from a clear cyclic structure to a more intricate by varying the value of σ_X. The diffusion variational autoencoder is capable of recovering also the periodic latent variables in the three cases.

The mapping of the encoded latent representations for the object dataset is shown in Figure 10.8. It is obtained from the latent representations obtained with the baseline and diffusion variational autoencoder with $\sigma_X = 0.1$. As we have seen in previous examples the learned reduction maps tend to map the latent representations corresponding to similar datapoints together as it can be seen from the hue colors in the representation.

The corresponding datasets presented in this chapter and the qualitative results support the performance of the proposed variational autoencoders for the task of recovering the underlying circular structure of a dataset. For different datasets produced from different underlying generative models we have been able to obtain also the Δ reduction maps that reduce the corresponding generative models by mapping latent variable representations close together if they represent the same datapoint.

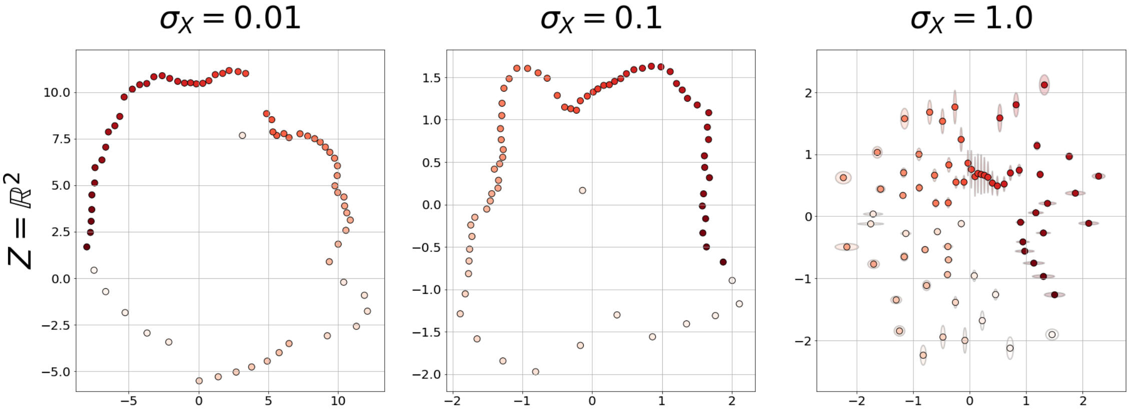

Figure 10.6: Latent representation for baseline variational autoencoder. Each point represents the calculated value for the mean $\boldsymbol{\mu}_{\boldsymbol{Z}}$. The ellipses represent high probability regions for each data representation defined with respect to the scale parameter $\boldsymbol{\mu}_{\boldsymbol{Z}}$. The color hue represents the underlying angles for the object orientation.

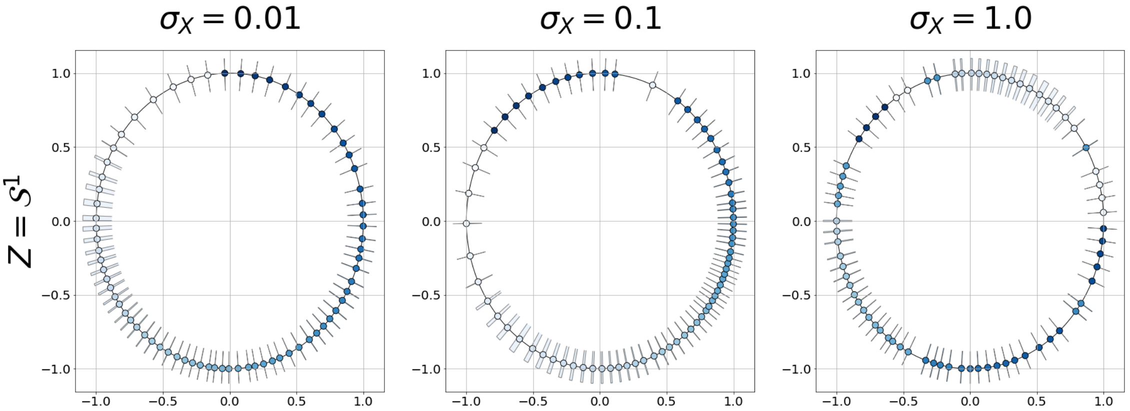

Figure 10.7: Latent representation for the diffusion variational autoencoder. Each point represents the calculated value for the mean $\mathrm{Emb}_{\mathbb{R}^2}()\boldsymbol{\mu}_{\boldsymbol{\Theta}})$. The arc sections represent the high probability regions for each data representation defined with respect to the scale parameter t. The color hue represents the underlying angles for the object orientation.

In the next chapter we will present a different setting in which the dataset has an underlying toroidal structure. For this case we will propose an extension of both the benchmark and the diffusion variational autoencoder with the corresponding simple reduction maps.

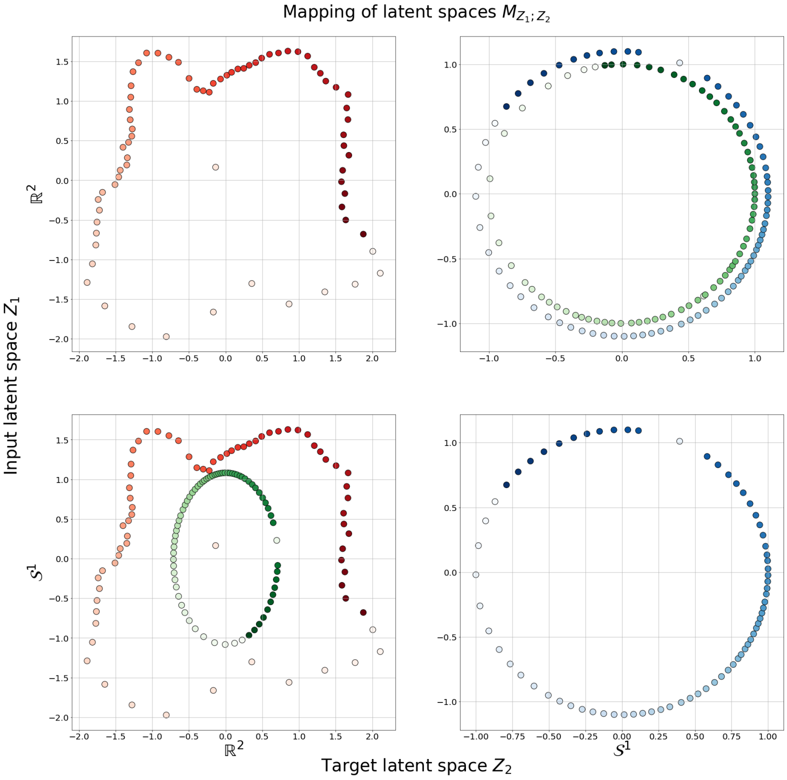

Figure 10.8: Mapping of the encoded latent representation of object rotated dataset in Z_1 into Z_2 via the corresponding learned reduction map $M_{Z_1;Z_2}$. The plot in the i-th row and the j-th column shows the mapped latent space representation of the i-th latent space into the j-th latent space together with the original latent representation of the j-th latent space for comparison. Each latent space representation has a corresponding color code. Red: baseline variational autoencoder. Blue: diffusion variational autoencoder. Green: Mapped latent representation. Color hue represents the underlying angles associated to each objects pose.

Chapter 11

Toroidal Latent Space Structure

In previous chapters we have focused completely on datasets that have an underlying circular structure. In this chapter we will analyze datasets with a different underlying geometry which corresponds to a simple extension to the already studied circular geometry. We will study the cases in which the underlying structure is toroidal. These type of models have an ample range of applications associated to motion capture, electroencephalograms and audiosignals [22].

In this chapter we introduced the benchmark dataset of $D \times D$-dimensional datapoints that can be identified to images. The dataset will be constructed with respect to a function $F : \Phi \times \Phi \mapsto \mathbb{R}^{D\times D}$ that induces a toroidal structure into the data.

Moreover we will present the equivalent version of the baseline and diffusion variational autoencoder for this geometrical setting and the corresponding Δ-reduction maps between the learned latent representations.

11.1 Extended benchmark dataset

We propose a new benchmark dataset based on the one described in Chapter 5. This toroidal benchmark dataset considers an artificial experiment with an observable data space X given by $D \times D$-dimensional datapoints $X = \mathbb{R}^{D\times D}$. Each observation x in dataset $\mathcal{X}$ is generated from the pair of phases from the set $\Phi \times \Phi$ with $\Phi = [-\pi, \pi)$. To generate each datapoint we have used the function $F : \Phi \times \Phi \mapsto X$ which calculates the average datapoint for a given pair of phases $\varphi^{(1)} \times \varphi^{(2)} \in \Phi \times \Phi$. The function F is defined as $F(\varphi^{(1)}, \varphi^{(2)}) = (f_{i,j}(\varphi^{(1)}, \varphi^{(2)}))_{i,j=1}^{D,D}$ where each individual function $f_{i,j} : \Phi^{(1)} \times \Phi^{(2)} \mapsto \mathbb{R}$ is given by the expression

$$f_{i,j}(\varphi^{(1)}, \varphi^{(2)}) = \sin\left(\frac{2\pi i}{D} + \varphi^{(1)}\right) + \sin\left(\frac{4\pi i}{D} + \varphi^{(2)}\right). \tag{11.1}$$

Notice that for each of these individual functions, the pair of values of $\varphi^{(1)} \times \varphi^{(2)} \in \Phi$ can be considered as the phases of two discrete sine functions defined over a discrete meshgrid of $[0, 1] \times [0, 1]$. Each of sine has an angular frequency of 2π and 4π respectively. Hence, for a particular datapoint $x \in X$ we will refer to $(\varphi^{(1)}, \varphi^{(2)}) \in \Phi \times \Phi$ as its corresponding underlying phases. A representation of the function F for the phase $\varphi^{(1)} \times \varphi^{(2)} = (0, 0)$ is shown in Figure 11.1.

$$P_{\Phi\times\Phi}(\varphi^{(1)}, \varphi^{(2)}) = \frac{1}{4\pi^2}. \tag{11.2}$$

For a given pair of phases $(\varphi^{(1)}, \varphi^{(2)}) \in \Phi \times \Phi$, the conditional distribution $\mathbb{P}_{X|(\varphi^{(1)},\varphi^{(2)})}$ from which data is sampled corresponds to the normal distribution with location parameter $F(\varphi^{(1)}, \varphi^{(2)})$

and covariance matrix $\Sigma = \sigma^2 I_D$ determined by the scalar $\sigma \in \mathbb{R}^+$ which can be interpreted as the amount of noise added to the function F. The probability density of this conditional distribution with respect to the $\mathcal{L}^{D\times D}$ measure is given by

$$P_{X|(\varphi^{(1)},\varphi^{(2)})}(x) = \frac{1}{(\sigma\sqrt{2\pi})^{D^2}}\exp\left(-\frac{\|x - F(\varphi^{(1)},\varphi^{(2)})\|_2^2}{2\sigma^2}\right).$$

According to Equation (2.6) presented in Chapter 2, the probability density of the generative model $\mathbb{P}_{X\times\Phi}$ for a given datapoint $x \in X$ and a phase $\varphi \in \Phi$ is given by

$$P_{X\times(\Phi\times\Phi)}(x,\varphi) = P_{X|(\varphi^{(1)},\varphi^{(2)})}(x)\cdot P_{\Phi\times\Phi}(\varphi^{(1)},\varphi^{(2)}) = \frac{1}{4\pi^2(\sigma\sqrt{2\pi})^{D^2}}\exp\left(-\frac{\|x - F(\varphi^{(1)},\varphi^{(2)})\|_2^2}{2\sigma^2}\right).$$

From the generative model that we have described we will produce different datasets that can be characterized in terms of the value σ used for the conditional distribution $\mathbb{P}_{X\times\varphi}$. The dataset with N datapoints generated according to a generative model with parameter σ will be denoted as $\mathcal{X}_\sigma$. In Figure 5.2 we present an example datapoint generated from the latent phase $\varphi = 0$ for the corresponding datasets $\mathcal{X}_1$, $\mathcal{X}_{0.1}$ and $\mathcal{X}_{0.01}$.

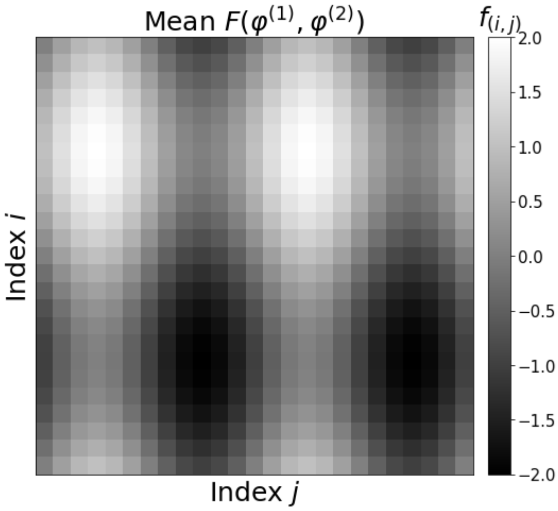

Figure 11.1: Representation of the mean function F with $D = 25$ calculated for the pair of phases $(\varphi^{(1)},\varphi^{(2)})$ with $\varphi^{(1)} = \varphi^{(2)} = 0$.

We will propose a simplified dataset for this setting defined in terms of the regular partition of the phase space $\Phi\times\Phi$ denoted as

$$(\Phi\times\Phi)_{\text{Vis}} = \left\{-\pi + \frac{2\pi i}{50}\right\}_{i=0}^{50} \times \left\{-\pi + \frac{2\pi i}{100}\right\}_{i=0}^{50} \tag{11.3}$$

From the regular phases set $(\Phi\times\Phi)_{\text{Vis}}$ of Equation 11.3 we construct the corresponding visualization dataset

$$\mathcal{X}_{\text{Vis}} = \left\{x \in \mathbb{R}^{D\times D}\middle| x = F(\varphi^{(1)},\varphi^{(2)})\ ;\ (\varphi^{(1)},\varphi^{(2)}) \in (\Phi\times\Phi)_{\text{Vis}}\right\} \tag{11.4}$$

We propose this simplified dataset for both training and visualization with the purpose of simplifying the training and visualization.

11.2 Dataset structure

In this case the chosen function F has the property of being periodic with respect to the two possible phases $(\varphi^{(1)}, \varphi^{(2)})$. As it was seen for the benchmark dataset of the circular geometry of Chapter 5, the minimum dimension needed for retrieving the periodic structure expected for the latent variables was two.

In this case we can notice that the smallest dimension needed to capture the periodic structure of the two phases is three. Take as an intuitive example Figure 11.2. In the three dimensional torus we can identify two perpendicular circular directions that can represent each of the phases.

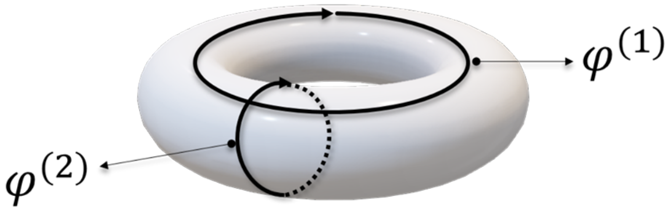

Figure 11.2: Visual intuition for the geometry of a torus embedded in $\mathbb{R}^3$. The surface of the torus can be described in terms of two periodic variables $\varphi^{(1)}, \varphi^{(2)}$.

Even though it is possible to represent the toroidal geometry in three dimensions we will be proposing a baseline variational autoencoder with a latent space corresponding to the 4-dimensional Euclidean space $Z = \mathbb{R}^4$. The reason for this is to provide a smooth connection to the diffusion variational autoencoder extended to the Clifford torus presented in the next sections which is embedded in $\mathbb{R}^4$.

11.3 Baseline variational autoencoder

Both the baseline and the diffusion variational autoencoders are trained for 100000 epochs with a dataset consisting of 2500 images of $D \times D = 25 \times 25$ pixels as the one presented in Figure 11.1. The neural network architecture presented in Section 3.5 is used, with two hidden layers for the encoding and decoding neural network. The number of neurons for each hidden dense layer corresponds to $\lfloor 25^2/3 \rfloor$. The number of latent samples is chosen as $L = 1$.

Figure 11.3 shows the recovered latent representations for the dataset $\mathcal{X}_{\text{Vis}}$. Each recovered representation has a clear structure with respect to the underlying phases of the corresponding datapoints. For instance in the case of the baseline variational autoencoder with parameter σ_X we can identify that phase $\varphi^{(1)}$ increases by following along the torus shape while phase $\varphi^{(2)}$ varies along a perpendicular direction with respect to this projection.

There is an interesting behavior for the recovered structure with respect to the parameter σ_X (rows). We can identify that the value of σ_X appears to either contract or expand the shape of the torus. As we have discussed for the ciruclar benchmark dataset, this can be due to the change in the weight of the Kullback-Leibler regularization term which favors posterior approximations close to the prior.

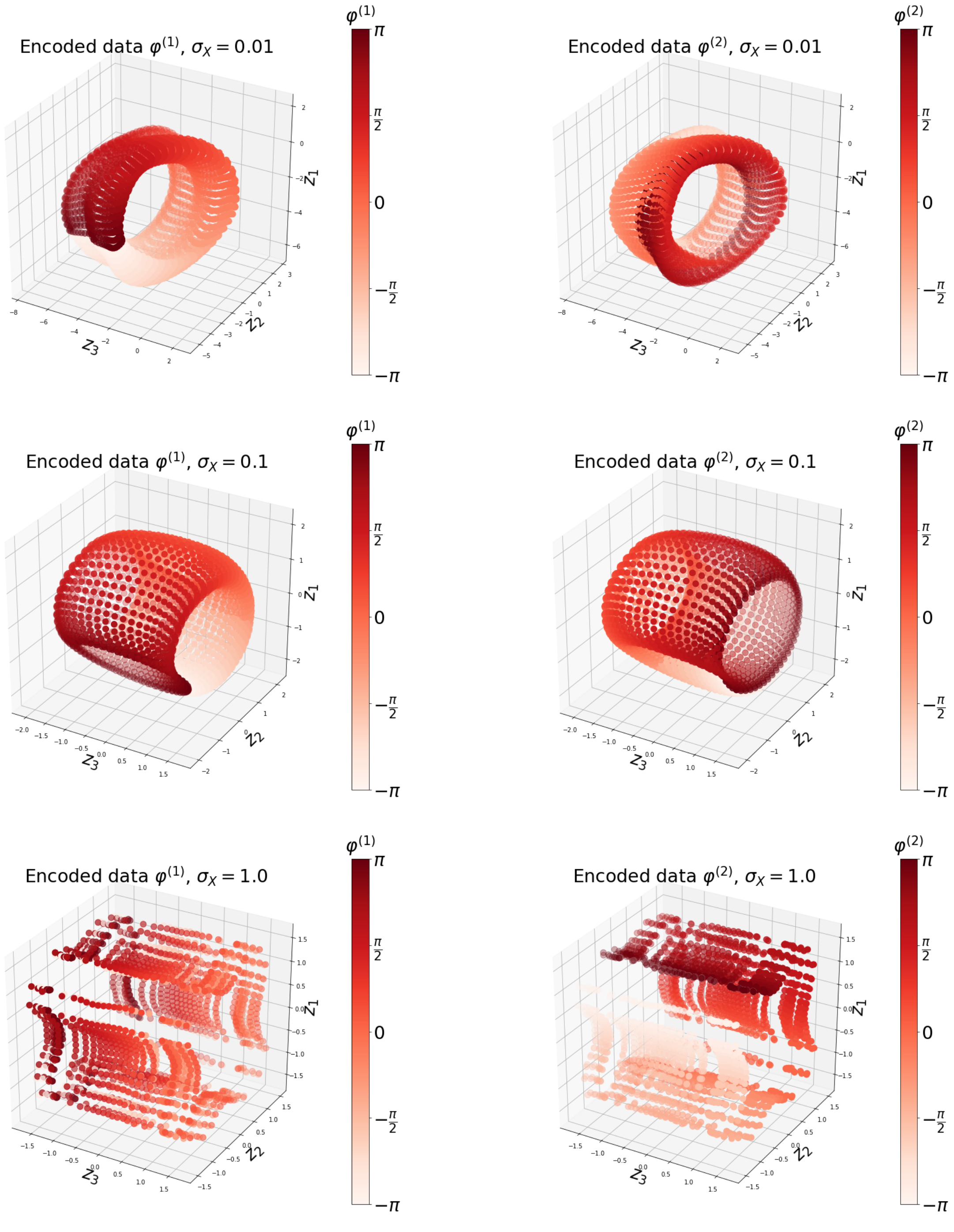

Figure 11.3: Encoded latent representation of the dataset $\mathcal{X}_{\text{Vis}}$ obtained with the mean $\boldsymbol{\mu}(\cdot))$ from the baseline variational autoencoder. Only three projected dimensions are shown corresponding to three entries of the latent representation $1, 2, 3$. Each column represents a fixed phase of interest $\varphi^{(1)}$ or $\varphi^{(1)}$. The rows showed the encoded representations for values of $\sigma_X \in \{0.01, 0.1, 1.0\}$. The color hue represents the values for the phases $\varphi^{(1)}, \varphi^{(2)}$

11.4 Diffusion variational autoencoder

We can extend the diffusion distribution introduced in Chapter 7 over the auxiliary set $\Theta \times \Theta$. We will consider a uniform measure $\mathbb{P}_{\Theta\times\Theta}$ over $\Theta \times \Theta$, the probability density $P_{\Theta\times\Theta}$ with respect to the $\mathcal{L}^2$ measure corresponds to

$$P_{\Theta\times\Theta}(\theta_1, \theta_2) = \frac{1}{4\pi^2} \tag{11.5}$$

We define the latent space Z as the Clifford torus $Z = \mathcal{T}^2$ embedded in $\mathbb{R}^4$ described in terms of the elements of $\Theta \times \Theta$ through the measurable function $\mathrm{Emb}_{\mathbb{R}^4} : \Theta \times \Theta \mapsto \mathbb{R}^4$ given by

$$\mathrm{Emb}_{\mathbb{R}^4}(\theta_1, \theta_2) = \frac{1}{\sqrt{2}}(\cos(\theta_1), \sin(\theta_1), \cos(\theta_2), \sin(\theta_2)). \tag{11.6}$$

Thus the latent space corresponding to the Clifford torus is described by the set

$$\mathcal{T}^2 = \left\{ \mathrm{Emb}_{\mathbb{R}^4}(\varphi^{(1)}, \varphi^{(2)}) \in \mathbb{R}^4 \middle| \varphi^{(1)} \in \Theta \varphi^{(2)} \in \Theta \right\} \tag{11.7}$$

We propose the prior distribution $\mathbb{P}_{\mathcal{T}^2}$ over the latent space $\mathcal{T}^2$ as the pushforward measure of the measurable embedding function $\mathrm{Emb}_{\mathbb{R}^4\#}\mathbb{P}_{\Theta\times\Theta}$ which corresponds to the uniform measure over $\mathcal{T}^2$.

For estimating the posterior distribution we choose the family of parametric distributions $\mathcal{P}^{A\times A}_{\Theta\times\Theta}$ defined in terms of the probability densities obtained from the solutions of the diffusion equation from Appendix A. We assume that each of the entry values from the pair $(\theta_1, \theta_2) \in \Theta \times \Theta$ is independent. Therefore the family is characterized in terms of the product measure between elements of the family $\mathcal{P}^A_\Theta$ as

$$\mathcal{P}^{A\times A}_{\Theta\times\Theta} = \left\{ \mathbb{P}^{(\mu_1,t_1)}_\Theta \otimes \mathbb{P}^{(\mu_2,t_2)}_\Theta \in \mathcal{P}_{\Theta\times\Theta} \;\middle|\; \mathbb{P}^{(\mu_{\Theta_i}, t_{\Theta_i})}_\Theta \in \mathcal{P}^A_\Theta \right\}. \tag{11.8}$$

Here the family of distributions $\mathcal{P}^A_\Theta$ is defined in Chapter 7 and in Equation (7.5) The parameter set A is given by

$$A = \left\{ (\mu, t) \in \Theta \times \mathbb{R}^+_0 \right\} \tag{11.9}$$

From the parametric family over the set $\Theta \times \Theta$ the latent space distribution over $Z = \mathcal{T}^2$ is determined by the pushforward measure of the embedding. The posterior approximation is therefore defined as

$$\mathcal{Q}^{A\times A}_{\mathcal{T}^2} = \left\{ \mathbb{Q}^{(\mu_1,t_1,\mu_2,t_2)}_{\mathcal{T}^2} \middle| \mathbb{Q}^{(\mu_1,t_1,\mu_2,t_2)}_{\mathcal{T}^2} = \mathrm{Emb}_{\mathbb{R}^4\#}(\mathbb{P}^{(\mu_1,t_1)}_\Theta \otimes \mathbb{P}^{(\mu_2,t_2)}_\Theta) \right\} \tag{11.10}$$

Due to the independence of the elements in $\Theta \times \Theta$ the calculation of the Kullback-Leibler divergence regularization term is performed in a similar fashion as in Chapter 7 via Monte Carlo. Moreover, each element in the pair $(\theta_1, \theta_2) \in \Theta \times \Theta$ can be sampled with the same reparametrization function defined in Equation (7.10).

Figure 11.4 shows the obtained latent representations for dataset $\mathcal{X}_{\mathrm{Vis}}$. From the images it can be clearly identified a pattern and a direction for each of the phases $\varphi^{(1)}$ and $\varphi^{(2)}$ with respect to the geometry of the Clifford torus. For example in the diffusion variational autoencoder with parameter $\sigma_X = 0.01$ the phase $\varphi^{(1)}$ increases and decreases along the vertical direction while the second phase $\varphi^{(2)}$ changes horizontally.

Unlike the baseline variational autoencoder, there is no noticeable qualitative differences between the retrieved latent representations. The only changes are with respect to the direction of change for each phase since there should be no preferred direction for a specific phase.

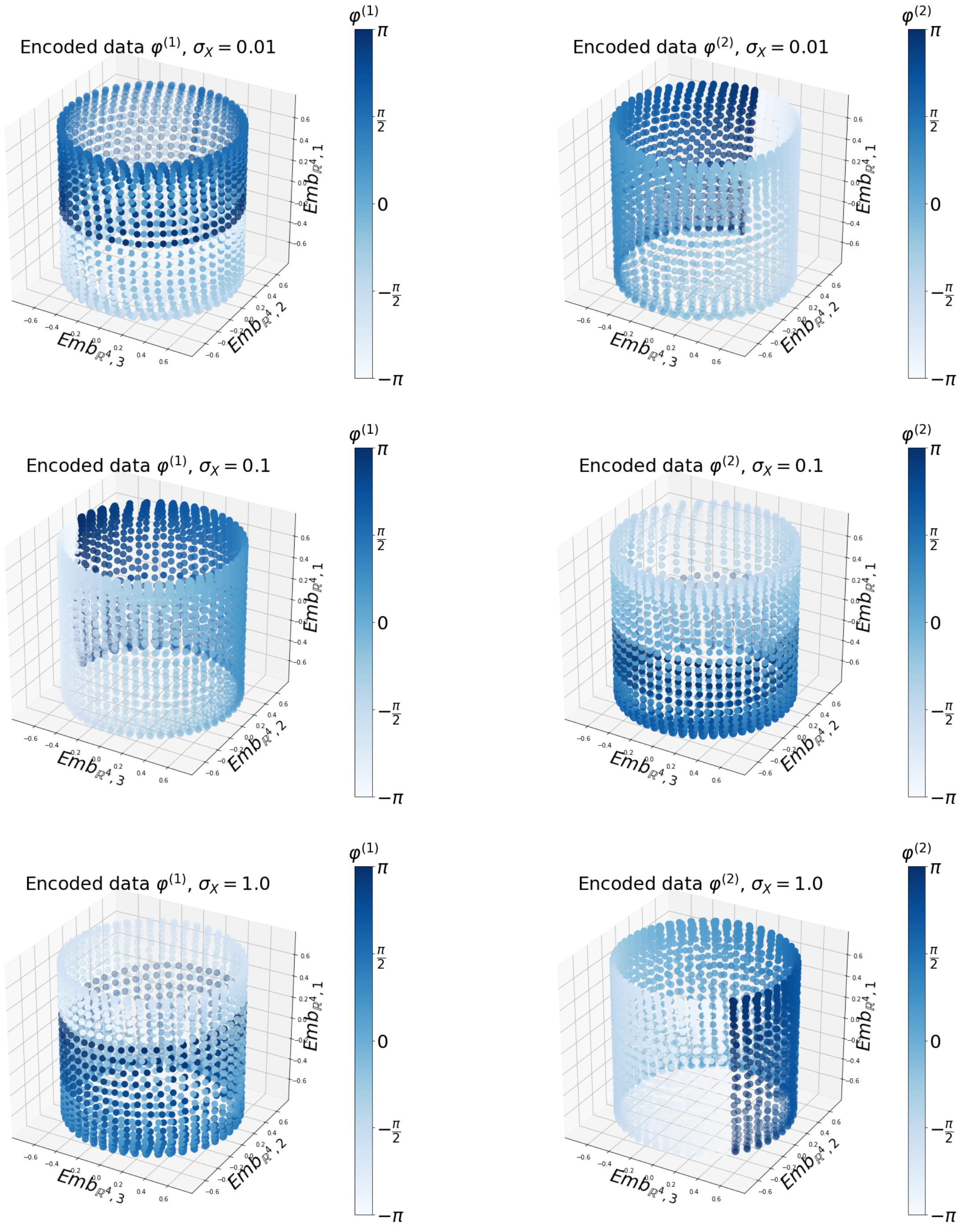

Figure 11.4: Encoded latent representation of the dataset $\mathcal{X}_{\text{Vis}}$ obtained with the mean $\text{Emb}_{\mathbb{R}^4}(\boldsymbol{\mu}(\cdot))$ from the diffusion variational autoencoder. Only three projected dimensions are shown corresponding to three of the entries $1, 2, 3$ of the embedding function $\text{Emb}_{\mathbb{R}^4}$. Each column represents a fixed phase of interest $\varphi^{(1)}$ or $\varphi^{(1)}$. The rows showed the encoded representations for values of $\sigma_X \in \{0.01, 0.1, 1.0\}$. The color hue represents the values for the phases $\varphi^{(1)}, \varphi^{(2)}$

11.5 Δ-reductor maps

We propose simple reductor maps between the latent spaces of the corresponding baseline variational autoencoder $Z = \mathbb{R}^4$ and the diffusional variational autoencoder $Z = \mathcal{T}^2$. These simple mappings are constructed based on the algorithm presented in Chapter 9.

The mapping from the 4-dimensional Euclidean space into the Clifford torus $Z = \mathcal{T}^2$ is performed via first a matrix multiplication with $C \in \mathbb{R}^{4\times 4}$ with a subsequent projection into Clifford torus by taking the . First we define the matrix product of the latent variable $z \in \mathcal{R}^4$ with matrix C as

$$y = C \cdot z, \tag{11.11}$$

such that the reduction map from $\mathcal{T}^2$ into $\mathbb{R}^2$ is given by

$$M^{(C)}_{\mathbb{R}^4;\mathcal{T}^2}(y) = \frac{1}{\sqrt{2}}\left(\frac{y_1}{\sqrt{y_1^2+y_2^2}}, \frac{y_2}{\sqrt{y_1^2+y_2^2}}, \frac{y_3}{\sqrt{y_3^2+y_4^2}}, \frac{y_4}{\sqrt{y_3^2+y_4^2}}\right). \tag{11.12}$$

Here y_i corresponds to the i-th entry of the vector y. The mapping from the latent space of the Clifford torus $Z = \mathcal{T}^2$ into the 4-dimensional Euclidean space is performed via a simple matrix multiplication with $C \in \mathbb{R}^{4\times 4}$ such that for an element $z \in \mathcal{T}^2$ we have

$$M^{(C)}_{\mathcal{T}^2;\mathbb{R}^4}(z) = C \cdot z \tag{11.13}$$

We present the obtained results for the mapped latent spaces corresponding to the models trained with dataset $\mathcal{X}_{\text{Vis}}$ with parameter $\sigma_X^2 = 0.1$. In Figure 11.5 we can observe the mapped latent representations from the baseline and diffusion variational autoencoders. Qualitatively we can identify that the mapped latent spaces tend to align as close as possible similar phases, i.e. similar color hues are mapped together. It is noticeable that the overlap between the mapped latent space and the target Z_2 is not completely perfect due to the shape difference from the original latent representations.

Table 11.1: Δ-tolerance for the trained reduction mappings from latent space Z_1 into Z_2.

$M_{Z_1;Z_2}$		Z_2	
		$\mathbb{R}^4$	$\mathcal{T}^2$
Z_1	$\mathbb{R}^4$	N/A	6216.02 ± 43.05
	$\mathcal{T}^2$	446.84 ± 6.42	N/A

The quantitative results show that the value Δ is higher for the transformation from latent space $\mathbb{R}^4$ into $\mathcal{T}^2$. A similar result was obtained for the benchmark dataset were the reduction maps from the baseline variational autoencoder latent space had a higher variability and larger Δ values. In this case the lower value for Δ is attained for the mapping between the Clifford torus $Z = \mathcal{T}^2$ into $\mathbb{R}^2$.

We have presented in this chapter the extended results for a different geometry for the proposed baseline and diffusion variational autoencoder. As it can be noticed the results are similar to the circular case in which the models are capable of recovering the underlying geometrical structure expected from the dataset.

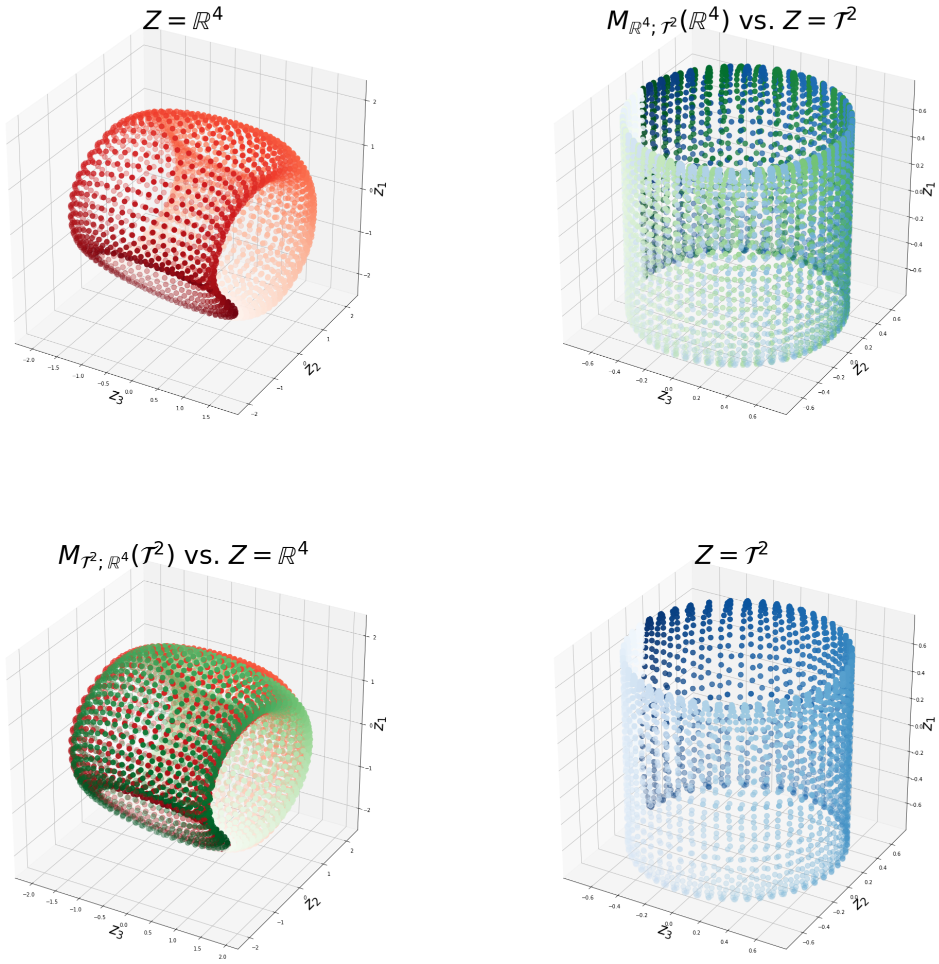

Figure 11.5: Mapping of the encoded latent representation of the $\mathcal{X}_{\text{Vist}}$ dataset from Z_1 into Z_2 via the corresponding learned reduction map $M_{Z_1;Z_2}$. The plot in the i-th row and the j-th column shows the mapped latent space representation of the i-th latent space into the j-th latent space together with the original latent representation of the j-th latent space for comparison.The plots in the diagonal correspond to the original latent space representations. Each latent space representation has a corresponding color code. Red: baseline variational autoencoder. Blue: diffusion variational autoencoder. Green: Mapped latent representation. Color hue represents the underlying angles associated to each objects pose. The hue represents the corresponding phase for $\varphi^{(1)}$, a similar qualitative result is obtained for $\varphi^{(2)}$.

Chapter 12

Conclusions

In this thesis we have focused on the the capabilities of variational autoencoders for latent variable separation by studying datasets with two types of underlying geometrical structures (circular and toroidal) . We have tested the retrieval of the underlying geometrical structure of the input dataset by the latent representations created by the encoding distributions learned by variational autoencoders. For this thesis we have used two variational autoencoders: a baseline VAE that incorporates the elements from [18, 14], and a proposed diffusion VAE which enforces a circular/toroidal geometry over the latent variables. We have found that both variational autoencoders are capable of recovering periodic latent variables associated to the geometry of our datasets.

The diffusion variational autoencoder is presented as a latent variable separation method that assumes periodic latent variables restricted to either the unit circle $Z = \mathcal{S}^1 \subseteq \mathbb{R}^2$ or the Clifford torus $Z = \mathcal{T}^2 \subseteq \mathbb{R}^4$. The diffusion VAE introduces a parametric family for the approximation to the posterior based on the solutions to the diffusion equation with periodic boundary conditions. This parametric family provides a different alternative to the distributions presented in the existing literature of [6, 25, 22]. Our results show that the diffusion variational autoencoders are capable of identifying the periodic latent structure assumed for the studied datasets in a natural way by restricting the geometry of the latent space.

For both variational autoencoders we have assessed the effects of the parameter σ_X with respect to the latent variable representations obtained and the data reconstruction. As in [13, 14] we have connected the effects of varying the value of σ_X (β in the literature) with the quality of the reconstructed datapoints and the recovered latent structure. Moreover, we have discussed the effects of modifying σ_X with respect to the structure of the recovered latent representations from a geometrical point of view.

The trained variational autoencoders obtained from the different example datasets have provided each a generative model for producing new datapoints with respect to the learned latent structure determined by the data. We presented a mathematical definition for reducing a generative model obtained with a variational autoencoder in terms of another up to a certain tolerance level Δ.

From our mathematical definition we proposed an algorithm for constructing simple Δ-reductor maps via backpropagation of the gradients through the trained variational autoencoders. These reductor maps were proposed specifically for the data with the expected underlying circular and toroidal structure and we assessed their performance with respect to the tolerance value Δ attained. We conclude from these experiments that we can obtain connections between learned generative models for the cases in which we expect similar underlying geometrical structure for the latent variables.

Further Work

We have presented in this thesis a discussion on the latent variable separation for datasets with an underlying geometrical structure is important together with a new variational autoencoder that enforces geometrical restrictions to the latent space. Moreover we have provided a mathematical definition and a practical algorithm for reducing generative models obtained from variational autoencoders. We propose different lines of study for future work based on the results presented in this thesis.

- Compare the latent variable representations and reconstruction performance obtained from the parametric family of distributions of the diffusion variational autoencoder with respect to other circular/toroidal distributions from literature [6, 25, 22].
- Extend the variational autoencoders to incorporate other geometries. Moreover, test the framework developed in this thesis for practical cases that involve representation learning for datasets with spherical/toroidal underlying geometries.
- Study the reduction maps from the point of view of their complexity. Now we have proposed simple reduction maps between latent spaces with a specific geometry but have not studied any formal measure for the complexity of the reduction map. Try to propose general reduction maps between latent spaces that are not restricted to the underlying geometry of the dataset .
- Explore the concept of interpretability of a latent variable representation within the reduction map context. By finding connections between interpretable and non-interpretable recovered generative models we can find explanations to the decisions performed by deep learning methods.
- Extend the construction of Δ-reduction maps to generative models obtained from other deep learning frameworks for instance to Generative Adversarial Networks (GANs) [10].

Bibliography

[1] Alexander A. Alemi, Ben Poole, Ian Fischer, Joshua V. Dillon, Rif A. Saurous, and Kevin Murphy. Fixing a Broken ELBO. 2017. 4

[2] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1798–1828, 2013. 3

[3] 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. 3, 11

[4] Alexandr A. Borovkov. *Probability Theory*. 2013. 7

[5] Miguel Ángel Carreira-Perpiñán. *Continuous latent variable models for dimensionality reduction and sequential data reconstruction*. PhD thesis, University of Sheffield, 2001. 25

[6] Tim R. Davidson, Luca Falorsi, Nicola De Cao, Thomas Kipf, and Jakub M. Tomczak. Hyperspherical Variational Auto-Encoders. 2018. 4, 73, 74

[7] Yadolah Dodge, editor. *The Oxford Dictionary of Statistical Terms*. Oxford University Press, Oxford, 1 edition, 2003. 6, 10

[8] Carl Doersch. Tutorial on Variational Autoencoders. 7:23, jun 2016. 21

[9] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. Representation Learning. In *Deep Learning Book*, pages 528–559. 2016. 3

[10] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative Adversarial Networks. pages 1–9, 2014. 74

[11] Justin Grimmer. An Introduction to Bayesian Inference via Variational Approximations. *Political Analysis*, 19(01):32–47, jan 2011. 3

[12] Geoffrey Grimmett and Dominic Welsh. *Probability: An Introduction*. Clarendon Press, Oxford, 2nd edition, 1986. 7

[13] Irina Higgins, Loic Matthey, Xavier Glorot, Arka Pal, Benigno Uria, Charles Blundell, Shakir Mohamed, and Alexander Lerchner. Early Visual Concept Learning with Unsupervised Deep Learning. jun 2016. 4, 31, 73

[14] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, Alexander Lerchner, and Google Deepmind. beta-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework. In *ICLR 2017*, number July, pages 1–13, 2017. ii, 4, 21, 31, 73

[15] Matt Hoffman, David M. Blei, Chong Wang, and John Paisley. Stochastic Variational Inference. 14:1303–1347, jun 2012. 3

[16] Diederik P. Kingma, Danilo J. Rezende, Shakir Mohamed, and Max Welling. Semi-Supervised Learning with Deep Generative Models. *ArXiv e-prints*, pages 1–9, jun 2014. ii, 4

[17] Diederik P. Kingma, Tim Salimans, Rafal Jozefowicz, Xi Chen, Ilya Sutskever, and Max Welling. Improving Variational Inference with Inverse Autoregressive Flow. (Nips), jun 2016. 4

[18] Diederik P Kingma and Max Welling. Auto-Encoding Variational Bayes. (Ml):1–14, dec 2013. 3, 4, 13, 14, 16, 19, 73

[19] Ludwig Kohaupt. Introduction to the discrete Fourier series considering both mathematical and engineering aspects - A linear-algebra approach. *Cogent Education*, 2(1):1064560, jul 2015. 32

[20] John A. Lee and Michel Verleysen. *Nonlinear Dimensionality Reduction*. Springer, feb 2007. 3

[21] Daniel J. Levitin. *The Organized Mind*. Penguin Random House, 2014. 2

[22] Alexandre K. W. Navarro, Jes Frellsen, and Richard E. Turner. The Multivariate Generalised von Mises distribution: Inference and applications. feb 2016. 4, 57, 65, 73, 74

[23] S.A. Nene, S.K. Nayar, and H. Murase. Columbia Object Image Library (COIL-20). Technical report, Columbia University, 1996. 61

[24] Jim Portegies. *An introduction to Measure Theory and Integration*. 6

[25] Sergey Prokudin, Peter Gehler, and Sebastian Nowozin. Deep Directional Statistics: Pose Estimation with Uncertainty Quantification. pages 1–18, may 2018. 4, 73, 74

[26] Robert L. Taylor, Marek Capinski, and Ekkehard Kopp. Measure, Integral and Probability. *Journal of the American Statistical Association*, 95(449):348, 2000. 6, 7

[27] Shengjia Zhao, Jiaming Song, and Stefano Ermon. InfoVAE: Information Maximizing Variational Autoencoders. jun 2017. 4

Appendix A

Diffusion Equation

The diffusion equation is a second order parabolic partial differential equation which describes the transport of mass in a medium with diffusion coefficient $\mathcal{D} > 0$ over time . The solutions are described by the function $\rho : [-\pi, \pi] \times \mathbb{R}_0^+ \mapsto \mathbb{R}$ which is the concentration of mass $\rho(r, t)$ for a given point r in one-dimensional interval $r \in [-\pi, \pi]$ at time $t \in \mathbb{R}_0^+$. The diffusion equation is given by:

$$\mathcal{D}\frac{\partial^2}{\partial r^2}\rho(r,t) = \frac{\partial}{\partial t}\rho(r,t). \tag{A.1a}$$

In our setting we assume $\mathcal{D} = 1$ and enforce the periodic boundary conditions in terms of a symmetric interval $[-\pi, \pi]$ centered at zero given by:

$$\rho(-\pi,t) = \rho(\pi,t)\ \forall t \in \mathbb{R}_0^+, \tag{A.1b}$$

$$\frac{\partial}{\partial r}u(-\pi,t) = \frac{\partial}{\partial r}\rho(\pi,t)\ \forall t \in \mathbb{R}_0^+, \tag{A.1c}$$

Due to the conservation of mass within the diffusion equation. For a unitary mass. This assumption is important for using the solutions to the equation as probability distributions. Therefore we have that the solutions to this equation must fulfill.

$$\int_{[-\pi,\pi]} \rho(r,t)d\mathcal{L}^1 = 1 \tag{A.1d}$$

The initial conditions are chosen with respect to the Dirac measure $\delta_\mu : \mathcal{F}_{[-\pi,\pi]} \mapsto \mathbb{R}_0^+$ centered at $\mu \in [-\pi, \pi]$. Such that for an event $\mathcal{E} \in \mathcal{F}_{[-\pi,\pi]}$ at time $t = 0$,

$$\int_{\mathcal{E}} \rho(r,0)d\mathcal{L}^1 = \delta_\mu(\mathcal{E}). \tag{A.1e}$$

We assume that the solutions are separable i.e. they can be explained in terms of the product of a spatial function $R : \mathbb{R} \mapsto \mathbb{R}$ and a temporal $T : \mathbb{R}_0^+ \mapsto \mathbb{R}$

$$\rho(r,t) = R(r) \cdot T(t). \tag{A.2}$$

Substituting the separable solutions assumption into the diffusion equation yields

$$T(t)\frac{d^2}{dr^2}R(r) = R(r)\frac{d}{dt}T(t). \tag{A.3}$$

Equation (A.3) is valid for all values of $r \in \mathbb{R}$ and $t \in \mathbb{R}_0^+$. This implies that the left and right hand side are equal to a constant $c \in \mathbb{R}$.

$$\frac{1}{T(t)}\frac{d}{dt}T(t) = c \tag{A.4}$$

$$\frac{1}{R(r)}\frac{d^2}{dr^2}R(r) = c \tag{A.5}$$

The solutions to the temporal and spatial equation are of the general form

$$T(t) = A\exp(-ct), \tag{A.6}$$

$$R(r) = B\text{cos}(r\sqrt{c}) + C\text{sin}(r\sqrt{c}), \tag{A.7}$$

With A, B, C constant values in $\mathbb{R}$ that are determined by the boundary conditions. In order to obtain solutions that do not diverge in time we choose values $c \geq 0$. The first boundary condition in Equation (A.1b) becomes

$$A\exp(-ct)B\text{cos}(\pi\sqrt{c}) - C\text{sin}(\pi\sqrt{c}) = A\exp(-ct)B\text{cos}(\pi\sqrt{c}) + C\text{sin}(\pi\sqrt{c}) \tag{A.8}$$

Due to the orthogonality of the sine and cosine functions we have the corresponding conditions,

$$B\text{cos}(\pi\sqrt{c}) = B\text{cos}(\pi\sqrt{c}), \tag{A.9}$$

$$C\text{sin}(\pi\sqrt{c}) = -C\text{sin}(\pi\sqrt{c}). \tag{A.10}$$

The second condition states that either $C = 0$ or $\sin(\pi\sqrt{c}) = 0$. If we focus on the condition $\sin(\pi\sqrt{c}) = 0$ we have that for a value $m \in \mathbb{Z}$ then

$$c = m^2. \tag{A.11}$$

Now, for the second boundary condition of Equation (A.1c) we have

$$m(B\text{sin}(\pi m) + C\text{cos}(\pi m)) = m(-B\text{sin}(\pi m) + C\text{cos}(\pi m)) \tag{A.12}$$

Which is already fulfilled since $\sin(\pi m) = 0$ for all $m \in \mathbb{Z}$. Finally to satisfy the initial conditions at time $t = 0$ we will take a linear combination of the spatial solutions for different values of $m \in \mathcal{Z}$

$$\rho(x, 0) = \sum_{m=0}^{\infty} B_m\text{cos}(rm^2) + C_m\text{sin}(rm^2) \tag{A.13}$$

The coefficients are calculated by integrating with respect to the Dirac measure δ_μ

$$B_m = \int_{[-\pi,\pi]} \cos(rm)d\delta_\mu(r) = \cos(\mu m) \tag{A.14}$$

$$C_m = \int_{[-\pi,\pi]} \sin(rm)d\delta_\mu(r) = \sin(\mu m) \tag{A.15}$$

The spatial solutions are given by:

$$\sum_{m=0}^{\infty} \cos(\mu m)\text{cos}(rm) + \sin(\mu m)\text{sin}(rm) = \sum_{m=0}^{\infty} \cos(m(r-\mu)) \tag{A.16}$$

Finally to enforce the conservation of mass we have to comply with

$$\int_{[-\pi,\pi]} \rho(r,t)d\mathcal{L}^1(r) = \int_{[-\pi,\pi]} A\sum_{m=0}^{\infty} \cos(m(r-\mu))d\mathcal{L}^1(r)\exp(-m^2t) = 1 \tag{A.17}$$

The integral of the cosine over $[-\pi, \pi]$ is only non-zero for $m = 0$ therefore

$$\int_{[-\pi,\pi]} A d\mathcal{L}^1 = 2\pi A = 1 \tag{A.18}$$

Therefore $A = 1/2\pi$. The final solution to the diffusion equation with initial conditions centered at μ is therefore

$$\rho_\mu(r,t) = T(t)R(r) = \frac{1}{2\pi}\sum_{m=0}^{\infty} \cos(m(r-\mu))\exp(-m^2 t) \tag{A.19}$$

A visualization of the solutions for different values of t is shown in figure A.1.

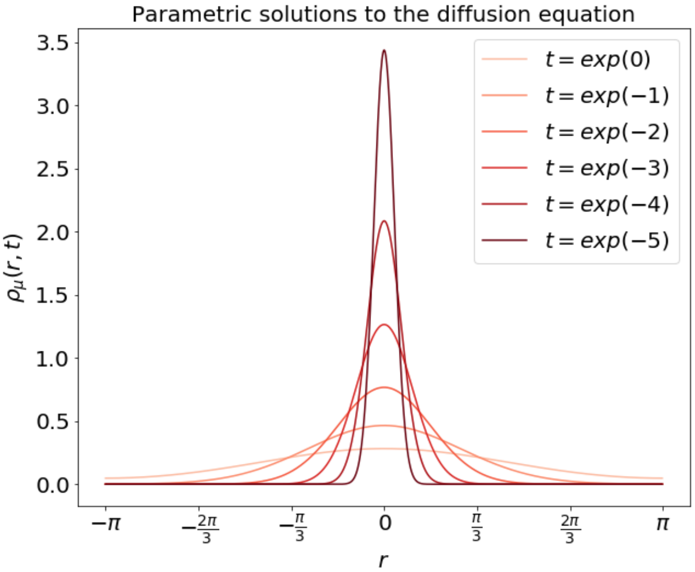

Figure A.1: Obtained solutions to the diffusion equation for a fixed location parameter $\mu = 0$ and variable time parameter t.

The solutions to the diffusion equation for each pair of parameters $(\mu, t) \in [-\pi, \pi] \times \mathbb{R}_0^+$ form the parametric family for the diffusion variational autoencoders. In Figure A.1 we present some solutions of the diffusion equation for different values of t for a fixed parameter $\mu = 0$