Probabilistic Dynamic Mode Primitives

Probabilistische Dynamische Modus Primitiven
Master thesis by Kay Hansel
Date of submission: May 5, 2021

1. Review: Prof. Jan Peters, Ph.D.
2. Review: M.Sc. Hany Abdulsamad
3. Review: M.Sc. Svenja Stark
Darmstadt

TECHNISCHE
UNIVERSITÄT
DARMSTADT

Erklärung zur Abschlussarbeit gemäß §22 Abs. 7 und §23 Abs. 7 APB der TU Darmstadt

Hiermit versichere ich, Kay Hansel, die vorliegende Masterarbeit ohne Hilfe Dritter und nur mit den angegebenen Quellen und Hilfsmitteln angefertigt zu haben. Alle Stellen, die Quellen entnommen wurden, sind als solche kenntlich gemacht worden. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Mir ist bekannt, dass im Fall eines Plagiats (§38 Abs. 2 APB) ein Täuschungsversuch vorliegt, der dazu führt, dass die Arbeit mit 5,0 bewertet und damit ein Prüfungsversuch verbraucht wird. Abschlussarbeiten dürfen nur einmal wiederholt werden.

Bei der abgegebenen Thesis stimmen die schriftliche und die zur Archivierung eingereichte elektronische Fassung gemäß §23 Abs. 7 APB überein.

Bei einer Thesis des Fachbereichs Architektur entspricht die eingereichte elektronische Fassung dem vorgestellten Modell und den vorgelegten Plänen.

Darmstadt, 5. Mai 2021

K. Hansel

Abstract

Movement primitives, a subdomain of imitation learning in robotics, leverage Machine Learning to learn movement sequences from demonstrations. Unfortunately, numerous proposed primitives cannot extract understandable and interpretable physical information from the learned latent space. The Koopman Theory analyzes highly complex dynamical systems and decomposes them into spatio-temporal characteristics and hence in interpretable physical information. In the scope of this work, a new class of movement primitives, the Probabilistic Dynamic Mode Primitives (Pro-DMPs), are introduced based on the underlying concept of Koopman Theory. A probabilistic dual perspective is proposed utilizing Gaussian Process State-Space Model. The transition model is assumed to be a linear stationary Markov sequence, and hence linear dynamics in latent space are considered. A Gaussian Process is applied to define a distribution over possible observation functions defining the dependence of the latent sequence on the given demonstration. The variability within the given demonstrations is represented in the inferred linear trajectories in latent space and captured by a hierarchical structure. This probabilistic framework naturally accounts for uncertainties and noise, leading to a nonparametric Bayesian formalism that allows us to capture time-independent dynamics in latent space. We demonstrated that the framework is capable of learning and reproducing given demonstrations on several benchmarks.

Zusammenfassung

Bewegungsprimitive, ein Teilbereich des Imitationslernens in der Robotik, nutzen maschinelles Lernen, um Bewegungsabläufe aus Demonstrationen zu lernen. Unglücklicherweise sind zahlreiche vorgeschlagene Primitive nicht in der Lage, verständliche und interpretierbare physikalische Informationen aus dem gelernten latenten Raum zu extrahieren. Die Koopman-Theorie analysiert hochkomplexe dynamische Systeme und zerlegt sie in räumlich-zeitliche Eigenschaften und damit in interpretierbare physikalische Informationen. Im Rahmen dieser Arbeit wird, basierend auf dem zugrundeliegenden Konzept der Koopman-Theorie, eine neue Klasse von Bewegungsprimitiven, die Probabilistische Dynamische Modus Primitiven, eingeführt. Es wird eine probabilistische duale Perspektive unter Verwendung von GPSSM erarbeitet. Das Transitionsmodell wird als lineare stationäre Markov-Sequenz angenommen, und daher werden lineare Dynamiken im latenten Raum betrachtet. Mit Hilfe eines Gauß-Prozesses wird eine Verteilung über mögliche Beobachtungsfunktionen definiert, die die Abhängigkeit der latenten Sequenz von der gegebenen Demonstration definiert. Die Variabilität innerhalb der gegebenen Demonstrationen wird in den gefolgerten linearen Trajektorien im latenten Raum dargestellt und durch eine hierarchische Struktur erfasst. Dieser probabilistische Rahmen berücksichtigt auf natürliche Weise Unsicherheiten und Rauschen, was zu einem nichtparametrischen Bayes'schen Formalismus führt, der es erlaubt, zeitunabhängige Dynamik im latenten Raum zu erfassen. Anhand mehrerer Benchmarks haben wir gezeigt, dass das Framework in der Lage ist, gegebene Demonstrationen zu erlernen und zu reproduzieren.

Contents

Figures, Tables and Algorithms

List of Figures

List of Tables

List of Algorithms

Abbreviations

List of Abbreviations

Notation	Description
ARD	Automatic Relevance Detection
Bayesian DMD	Bayesian Dynamic Mode Decomposition
Bayesian GP-DMD	Bayesian Gaussian Process Dynamic Mode Decomposition
Bayesian-DMP	Bayesian Dynamic Mode Primitive
CSD	Circle-Shape Dataset
DMD	Dynamic Mode Decomposition
DMP	Dynamic Movement Primitives
ELBO	Evidence Lower Bound

EM	Expectation Maximization algorithm
ESD	Eight-Shape Dataset
extended DMD	extended Dynamic Mode Decomposition
FA	Factor Analysis
GMM	Gaussian Mixture Modelling
GMR	Gaussian Mixture Regression
GP	Gaussian Process
GP-DMD	Gaussian Process Dynamic Mode Decomposition
GP-LVM	Gaussian Process Latent Variable Model
GP-SSM	Gaussian Process State-Space Model
HMM	Hidden Markov Model
ICA	Independent Component Analysis
i.i.d.	independently and identically distributed
kernel DMD	kernel Dynamic Mode Decomposition
KL	Kullback–Leibler divergence
LVM	Latent Variable Model
MAP	Maximum A Posteriori probability estimate

MCMC	Markov Chain Monte-Carlo
MJD	Minimum-Jerk Dataset
MLE	Maximum Likelihood estimate
PCA	Principle Component Analysis
Pro-DMP	Probabilistic Dynamic Mode Primitive
ProMP	Probabilistic Movement Primitive
RMSE	Root Mean Square Error
SIR	Sequential Importance Resampling
SIS	Sequential Importance Sampling
SMC	Sequential Monte Carlo
SSM	State-Space Model
SVD	Singular Value Decomposition
VB	Variational Bayes
VBEM	Variational Bayes Expectation Maximization algorithm
VI	Variational Inference

1. Introduction

With the deluge of data and the increasing computational power of technologies, Machine Learning attracts increasing attention to analyze and understand given data from highly complex systems [1–6]. With the help of mathematics, statistics, computer science, and engineering techniques, Machine Learning uses an interdisciplinary approach to conclude, make decisions, and predict future outcomes [2, 4–6]. In robotics, imitation learning is a central area aiming to learn behavior from given demonstrations [7–10]. The concepts of movement primitives, a subdomain of imitation learning, leverages Machine Learning to learn movement sequences from demonstrations. In general, there are three distinct types of movement primitives that provide the basis for a large number of other primitives. Among them, the *Dynamic Movement Primitivess (DMPs)* are the first proposed movement primitives and hence laid the foundation for this subdomain in Imitation Learning [7, 11, 12]. They rely on stable linear dynamical systems and learn the nonlinearities by utilizing regression models. The other two primitives are fully data-driven frameworks not requiring an underlying stable linear dynamical system [9, 10]. The first framework builds on *Gaussian Mixture Regression (GMR)* and consequently utilizes *Gaussian Mixture Modelling (GMM)* to identify clusters in the given demonstrations [9, 13]. Subsequently, based on the inferred clusters, regression techniques are applied. The second framework, the *Probabilistic Movement Primitives (ProMPs)*, offers a probabilistic approach, relying fully upon regression models [10, 14, 15]. The regression models learn to recreate the demonstration as a function based on the time as input. A large amount of research has been done on these three concepts, and several promising movement primitives have been proposed [8–10, 16–21]. Unfortunately, many of these proposed frameworks are unable to extract understandable and interpretable physical information within the learned latent space.

The Koopman Theory addresses the analysis of highly complex dynamical systems [22–27]. This theory considers a linear evolution of selected measurement functions on the given data in an infinite-dimensional Hilbert space, rather than the nonlinear evolution of the collected data points themselves [22–24]. As a result of the linear behavior in Hilbert space,

Koopman Theory allows a subsequent decomposition into spatio-temporal characteristics of the given dynamical system and thus into interpretable physical information [24]. The *Dynamic Mode Decomposition (DMD)* family provides frameworks to approximate the infinite-dimensional Hilbert space by a finite-dimensional invariant subspace [25–34]. The origins of DMD lie in fluid mechanics, where it was introduced independently of Koopman Theory and has attracted considerable attention in numerous research areas [32, 33]. Its increasing success results from its simple formulation and its close connection to Koopman Theory [25, 26]. The DMD family provides equation-free, data-driven approaches capable of spatial-temporal decomposition of complex systems without requiring explicit knowledge of the governing dynamics [28–31]. Since the given data can originate from highly nonlinear behavior, techniques based on feature mappings or kernelized functions have been proposed in the literature, hence extending the DMD family [29–31]. Subsequently, the use of spectral decomposition, also known as eigenvalue decomposition, allows the dynamics to be decomposed into spatio-temporal patterns [25, 26]. Based on these spatio-temporal patterns, the dynamics are analyzed and predictions are made.

1.1. Contribution

This thesis formulates a new class of movement primitives based on the underlying concept of Koopman Theory. The different approaches of the DMD family approximate the infinite-dimensional Hilbert space by a finite-dimensional invariant subspace. In the context of this work, a probabilistic *dual perspective* is adopted in the beginning. Based on *Gaussian Process State-Space Model (GP-SSM)*, the evolution in the latent space is assumed to be a stationary linear Markov sequence, and a *Gaussian Process (GP)* is considered to describe the dependence on the given demonstrations. Compared to the DMD family, the inverse mapping from the latent space representing the invariant subspace back to the observation space is considered. The probabilistic formulation naturally accounts for uncertainties and noise and leads to a nonparametric Bayesian formalism for the observation model. Consequently, nonlinear observation functions between latent and observation space are included by the assumed the GP. In the context of this work, the *Gaussian Process Dynamic Mode Decomposition (GP-DMD)* is proposed, a probabilistic dual variant of the DMD family. This framework represents a *Maximum A Posteriori probability estimate (MAP)* of the sequence in the latent space and the corresponding linear operator. The problem that the invariant subspace and thus the latent space can have a higher dimensionality than the observation space involves the risk of overfitting. Accordingly, the *Bayesian Gaussian Process Dynamic Mode Decomposition (Bayesian GP-DMD)*, a fully Bayesian formalism

of the GP-DMD, is introduced. This formulation provides a *Variational Inference (VI)* or *Variational Bayes (VB)* method and thus mitigates overfitting and achieves approximations of the posterior distribution over the linear operator and linear trajectories in the latent space.

While the GP-DMD and the Bayesian GP-DMD represent a dual perspective of the DMD family, they are unable to deal with multiple demonstrations. In the concept of movement primitives, learning variability within multiple given demonstrations is an essential component [8–10]. Therefore, this thesis proposes *Probabilistic Dynamic Mode Primitives (Pro-DMPs)*, a new class of movement primitives. The Pro-DMPs target to express the given variability within the given demonstrations in the derived trajectories in the latent space. Based on the ProMPs [10, 14, 15, 18], a hierarchical structure is provided in the latent space, extending the GP-DMD framework. As a result, the Pro-DMP embodies an *Expectation Maximization algorithm (EM)* and determines optimal point estimates for the sequence in the latent space and the linear operator. In order to reduce the risk of overfitting, the *Bayesian Dynamic Mode Primitives (Bayesian-DMPs)* adopt a fully Bayesian formalization of the Pro-DMP. As a result of the global linearization in latent space, the derived frameworks capture spatio-temporal characteristics of underlying time-independent dynamics. The probabilistic dual perspective hence provides frameworks in which the latent space captures physical information.

1.2. Overview

This thesis is organized as follows. First, an overview is given of the basic paradigms of Machine Learning in Chapter 2. Thereby the introduction of the Probability Density Estimation and the related *Maximum Likelihood estimate (MLE)* and MAP takes place [2, 4, 6]. Next, the introduction of the *Latent Variable Models (LVMs)* [3, 4, 35] and the well-known EM [36, 37] is given. On that basis, fully Bayesian learning is motivated, and techniques such as VI and VB are derived [3–6]. Also, the relation between the EM and the VI resp. VB is shown. To introduce nonparametric models, the GP [38, 39], as well as *sparse Gaussian Processes* techniques [40–43], are discussed next. Chapter 2 concludes with *Gaussian Process Latent Variable Models (GP-LVMs)* [44–49] and *Bayesian GP-LVMs* [50] resulting from the combination of LVMs and GPs. Chapter 3 focuses on Koopman Theory [22–24] and the DMD family [25, 26, 28, 32–34]. Hence, the Koopman Theory is first discussed to gain insight into the topic. Subsequently, the DMD, a straightforward approximation, is introduced. With the *extended Dynamic Mode Decomposition (extended*

DMD) [25, 26, 29] and the *kernel Dynamic Mode Decomposition (kernel DMD)* [25, 30, 31], we then consider two extensions of the DMD framework. Eventually, a probabilistic perspective of DMD is adopted in the context of *Bayesian Dynamic Mode Decomposition (Bayesian DMD)* [51]. Chapters 4 and 5 provide the central contribution of this work. First, an overview of *State-Space Models (SSMs)* [52–54] and GP-SSMs [54–60] is given. After that, the GP-DMD and the Bayesian GP-DMD are proposed. Eventually, the new classes of movement primitives, the Pro-DMP and the Bayesian-DMP, are introduced. Benchmarks for the validation of the performance of the GP-DMD, the Bayesian GP-DMD, and the Pro-DMP on different datasets are provided in Chapter 6. In conclusion, Chapter 7 discusses the proposed framework, highlights its advantages and disadvantages, and gives an outlook for future research.

2. Machine Learning

Machine Learning aims at analyzing and interpreting patterns and relationships from given data of an unknown process [1–6]. Based on these learned relationships, Machine Learning interdisciplinarily leverages mathematics, statistics, computer science, and engineering techniques to conclude, make decisions, and predict future outcomes of interest. In particular, the daily growing deluge of data and the increasing technologies' computational power are increasing the interest in Machine Learning in various modern research fields from physics and robotics to economics and beyond [26, 61]. In the following sections, an overview of Machine Learning paradigms necessary to understand the work, is provided.

Sections 2.1, 2.2 and 2.3 give an overview of classical paradigms in Machine Learning to understand the intricacies and effects of Bayesian formalism. Fitting parameterized models to represent given data is discussed. Basic techniques derive optimal estimates of parameters expressing the probabilistic framework as an optimization procedure in logarithmic space [3–6]. However, the resulting parameters correspond to a point estimate responsible for an optimum in the probability log-density function. For this reason, the techniques of the full Bayesian formalism are discussed to circumvent the points estimate by treating the parameters as random variables [3, 4, 6]. Additional variables are introduced to handle fragmentary data and express more complex structures in the probabilistic paradigms [4, 6, 35].

In Sections 2.4 and 2.5, the constraint imposed by parameterizing a model to express possible functions for given data is removed. The supervised learning paradigm discussed proposes a generalization of the Gaussian probability distributions to infinite-dimensional function spaces [4, 6, 38, 39]. Thus, the restriction of the possible functions responsible for the data is done by a probability distribution. Functions are assigned a higher probability value the more they represent the given data.

Eventually, Sections 2.6 and 2.7 present a central framework of this work. It leads to a proposed Bayesian formalism that considers a regression problem with unknown inputs.

Optimal estimates for the inputs are inferred, considering the distribution over all possible functions responsible for the data. Therefore, it appears to combine the techniques proposed in the earlier sections. The ideas and assumptions made in these sections form the basis for the techniques derived in this work.

2.1. Probability Density Estimation

The data used in Machine Learning in general results from underlying unknown stochastic random processes [4]. For this reason, it is useful to adopt a probabilistic perspective in which the given data is considered as a set of random variables drawn from an unknown probability density function. In *Probability Density Estimation*, the central challenge is to find suitable approximations of this density function that captures as many properties describing the given data as possible [2–6,35,61]. However, unknown sources not captured by the data induce further random noise and complicate the search for satisfactory approximations [4].

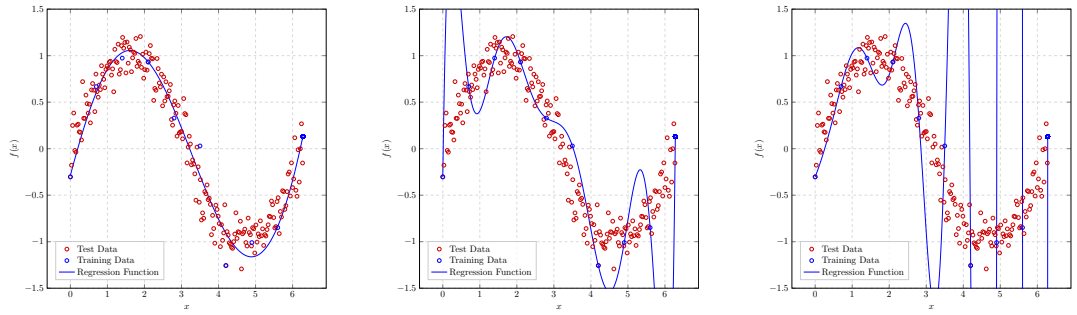
$$\begin{aligned} p(\boldsymbol{\theta} \mid \mathbf{Y}, \mathcal{M}) &= \frac{p(\mathbf{Y} \mid \boldsymbol{\theta}, \mathcal{M}) p(\boldsymbol{\theta} \mid \mathcal{M})}{\int p(\mathbf{Y} \mid \boldsymbol{\theta}, \mathcal{M}) p(\boldsymbol{\theta} \mid \mathcal{M}) \, \mathrm{d}\boldsymbol{\theta}} \\ &= \frac{p(\mathbf{Y} \mid \boldsymbol{\theta}, \mathcal{M}) p(\boldsymbol{\theta} \mid \mathcal{M})}{p(\mathbf{Y} \mid \mathcal{M})}, \end{aligned}$$

a core quantity in Machine Learning [2,4–6]. Using *Bayes' Rule*, also called *Bayes' Theorem*, the posterior distribution is composed of a likelihood and a prior distribution. In general, the likelihood does not integrate to one. Therefore, the denominator, called *marginal likelihood* or *evidence*, ensures that the posterior distribution is normalized and represents a valid probability density function [4, 6].

It is straightforward to take the „best guess“ as the optimal solution for the parameters

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} p(\mathbf{Y} \mid \boldsymbol{\theta}, \mathcal{M}),$$

known as *Maximum Likelihood estimate (MLE)* [2–6,61]. The MLE is a popular probabilistic paradigm in Machine Learning, which is characterized by its simplicity. It considers an optimization problem on the observed data's log-likelihood, assuming that each data point was drawn independently. The prior $p(\boldsymbol{\theta})$ is uninformative, constant, and thus equal for



(a) (b) (c)

Figure 2.1.: This figure illustrates the disadvantage of overfitting on a data set from a sinusoidal stochastic process with additional random noise. While (a) accurately represents the underlying sinusoidal process, (b) and (c)'s more complex models overfit the data. This behavior is a well-known drawback in maximizing the likelihood of parametric models in Machine Learning [4, 6].

all parameters. Therefore, the only relevant quantity for the optimization problem is the likelihood function. This optimization can achieve nearly perfect results if enough data $\mathbf{Y}$ is available. However, Figure 2.1 illustrates a major drawback of using Maximum Likelihood estimate (MLE). Considering only the likelihood and the selected parametric model $\mathcal{M}$ carries the risk of *overfitting* [4, 6]. Increasing the model's complexity leads to an increase in maximum likelihood and a perfect fit to the data points $\mathbf{Y}$. However, the increase in complexity comes with the disadvantage that the functional forms are highly fluctuating and extreme, and the model does not capture the underlying stochastic process of interest. From a mathematical perspective, this occurs due to a resulting underdetermined system with fewer observations than parameters [4]. This is a well-known drawback of MLE [4, 6].

One appealing way to address the risk of overfitting consists of reducing confidence in the model's MLE solution. For this purpose, uncertainty and variability in the parameters $\boldsymbol{\theta}$ and hence in the model's estimates are considered. From a Bayesian perspective, this consideration is done by incorporating a priori knowledge about the parameters into the posterior estimate

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} p(\mathbf{Y} \mid \boldsymbol{\theta}, \mathcal{M}) p(\boldsymbol{\theta} \mid \mathcal{M}),$$

resulting in the well-known *Maximum A Posteriori probability estimate (MAP)* [2–6, 61]. The use of a prior leads to a regularization term in the optimization problem that imposes

additional constraints on the parameter values. Depending on the amount of data, the model balances the best estimate between the prior probability and the likelihood. The uncertainty about the optimal estimate increases, and the prior solution is preferred when the data set becomes smaller. Conversely, as the number of data increases, the confidence in the likelihood estimate increases. As a result, the uncertainty and hence the prior is neglected, and the MAP solution converges to the MLE solution [4–6]. Consequently, properly estimating the prior leads to sophisticated solutions and mitigates the aforementioned problem of overfitting [4, 6].

2.2. Latent Variable Models

One important feature in unsupervised learning and thus in Probability Density Estimation is the consideration of *Latent Variable Models (LVMs)* [3–6, 35]. In these models, the joint density is assumed to be augmented by some unobserved or hidden random variables called *latent variables*. The likelihood expands to a marginal likelihood over these latent variables

$$p(\mathbf{Y} \mid \boldsymbol{\theta}, \mathcal{M}) = \int p(\mathbf{Y}, \mathbf{X} \mid \boldsymbol{\theta}, \mathcal{M}) \, \mathrm{d}\mathbf{X},$$

considered as data set $\mathbf{X} = \{\mathbf{x}_0, \cdots, \mathbf{x}_T\}$. The number of latent variables usually grows with the number of observations $\mathbf{Y}$ [4,6]. For discrete instead of continuous latent variables, the integral is replaced by a sum. On the one hand, the use of LVMs follows from incomplete or fragmentary data. On the other hand, it offers the possibility of taking dependencies between observations into account without increasing the parameter size significantly. In this way, it is possible to model correlations and hierarchical structures and thus handle more complex distributions [3, 6, 35]. However, these advantages come together with the cost of inducing dependencies between the parameters. These dependencies are expressed by the log-likelihood of the data forming the logarithm over integrals

$$\begin{aligned} \log p(\mathbf{Y} \mid \boldsymbol{\theta}, \mathcal{M}) &= \log \int p(\mathbf{Y}, \mathbf{X} \mid \boldsymbol{\theta}, \mathcal{M}) \, \mathrm{d}\mathbf{X} \\ &= \underbrace{\sum_{t=1}^{T} \log \int p(\mathbf{y}_t, \mathbf{x}_t \mid \boldsymbol{\theta}, \mathcal{M}) \, \mathrm{d}\mathbf{x}_t}_{\stackrel{\text{def}}{=} \mathcal{L}(\mathbf{X}, \boldsymbol{\theta})}, \end{aligned} \tag{2.1}$$

which is generally intractable [3, 4, 6, 35]. This intractability complicates the use of the aforementioned probabilistic paradigms MLE or MAP and causes difficulties training generative models.

Instead of working directly on Equation (2.1), a convenient and deterministic way is moving to a more tractable approach by introducing a lower bound [3–6]. This introduction involves extending the joint density model and the likelihood with an auxiliary distribution called the *variational distribution* $q(\mathbf{X})$. A variational distribution either has a parametric form or expresses a family of functions. Thus, using Jensen's inequality [62], the data's log-likelihood changes to

$$\begin{aligned}
\mathcal{L}(\mathbf{X}, \boldsymbol{\theta}) &= \sum_{t=1}^{T} \log \int p(\mathbf{y}_t, \mathbf{x}_t \mid \boldsymbol{\theta}, \mathcal{M}) \, \mathrm{d}\mathbf{x}_t \\
&= \sum_{t=1}^{T} \log \int q(\mathbf{x}_t) \frac{p(\mathbf{y}_t, \mathbf{x}_t \mid \boldsymbol{\theta}, \mathcal{M})}{q(\mathbf{x}_t)} \, \mathrm{d}\mathbf{X} \\
&\geq \sum_{t=1}^{T} \int q(\mathbf{x}_t) \log \frac{p(\mathbf{y}_t, \mathbf{x}_t \mid \boldsymbol{\theta}, \mathcal{M})}{q(\mathbf{x}_t)} \, \mathrm{d}\mathbf{X} \\
&= \sum_{t=1}^{T} \int q(\mathbf{x}_t) \log \frac{p(\mathbf{x}_t \mid \mathbf{y}_t, \boldsymbol{\theta}, \mathcal{M}) p(\mathbf{y}_t \mid \boldsymbol{\theta}, \mathcal{M})}{q(\mathbf{x}_t)} \, \mathrm{d}\mathbf{X} \\
&= \sum_{t=1}^{T} \left(\log p(\mathbf{y}_t \mid \boldsymbol{\theta}, \mathcal{M}) - \mathrm{KL}(p(\mathbf{x}_t \mid \mathbf{y}_t, \boldsymbol{\theta}, \mathcal{M}) \parallel q(\mathbf{x}_t)) \right), \\
&\stackrel{\text{def}}{=} \mathcal{L}_{\text{ELBO}}(q(\mathbf{x}_t), \theta),
\end{aligned} \tag{2.2}$$

resulting in the tractable lower bound known as *Evidence Lower Bound (ELBO)* or *free energy* [3–6, 63, 64]. The optimization of this lower bound leads to the well-known iterative *Expectation Maximization algorithm (EM)*

$$\begin{aligned}
\textbf{E step:} \quad q(\mathbf{x}_i)^* &\leftarrow \arg\max_{q(\mathbf{x}_i)} \mathcal{L}_{\text{ELBO}}(q(\mathbf{x}_i), \boldsymbol{\theta}) \quad \forall i, \\
\textbf{M step:} \quad \boldsymbol{\theta}^* &\leftarrow \arg\max_{\boldsymbol{\theta}} \mathcal{L}_{\text{ELBO}}(q(\mathbf{x}_i)^*, \boldsymbol{\theta}),
\end{aligned}$$

where the variational distribution $q(\mathbf{X})$ and the model parameters $\boldsymbol{\theta}$ are subsequently optimized, as shown in Figure 2.2 [2–6, 36, 37]. The E step tightens the ELBO by minimizing

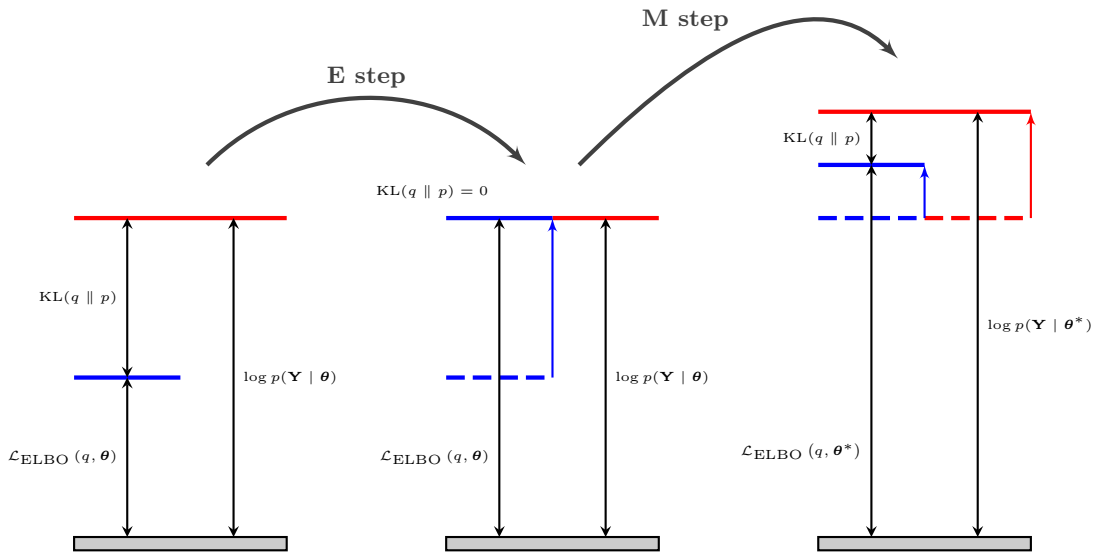


Figure 2.2.: This figure shows the cyclical process of the well-known Expectation Maximization algorithm. The EM maximizes the actual marginal log-likelihood $\log p(\mathbf{Y} \mid \boldsymbol{\theta})$ taking unobserved latent variables into account. The Expectation step (E step) tightens the lower bound by minimizing the Kullback–Leibler divergence between the variational distribution q and the true posterior distribution p. Classical Maximum Likelihood estimate is then applied in the Maximization step (M step), and the parameters $\boldsymbol{\theta}$ are optimized. This optimization increases the lower bound $\mathcal{L}_{\text{ELBO}}$ and thus maximizes the marginal likelihood [3, 4].

the KL divergence (see Appendix B) between the variational distribution $q(\mathbf{X})$ and the latent variables' posterior distribution $p(\mathbf{x}_t \mid \mathbf{y}_t, \boldsymbol{\theta}, \mathcal{M})$. The parameters of the model are updated in the M step. Working with exponential families such as Gaussian distributions (see Appendix A) allows the posterior $p(\mathbf{x}_t \mid \mathbf{y}_t, \boldsymbol{\theta}, \mathcal{M})$ to be analytically determined in closed form. Jensen's inequality (see Equation (2.2)) vanishes completly and the ELBO is equal to the marginal log-likelihood [3]. Consequently, the M step directly optimizes the marginal log-likelihood. In general, the vanishing of Jensen's inequality is not always the case [3]. The KL divergence, for instance, does not become zero, resulting in a constant deviation between the variational and posterior distributions. This deviation results in a constant offset between the true marginal log-likelihood and the calculated ELBO. Thus, the introduction of the ELBO entails the risk of an offset in the estimation of the optimal parameter values $\boldsymbol{\theta}^*$ [3]. The EM algorithm nevertheless provides a convenient way to estimate the parameters' optimal values for a selected model. Additionally, it provides an approximation of the posterior distribution of the latent variables.

2.3. Bayesian Learning

The extension of MLE by the Expectation Maximization algorithm achieves the handling of Latent Variable Models but still maximizes the log of a probability density function. It searches for an optimal parameter value for a selected model, considering only the density. Even if a prior $p(\boldsymbol{\theta})$ is considered and thus the probability density function is reshaped, MAP again provides only an optimal point estimate of $\boldsymbol{\theta}^*$. A suitable and intuitive way is to consider the probability density and the probability mass and thus the entire posterior distribution. Indeed, this consideration is a part of *Bayesian Learning* or *Bayesian Statistics* by treating the unknown parameters as random variables [3–6]. This treatment changes the previous optimization problem of parameters into a Probabilistic Inference problem.

The quantity of interest becomes the marginal likelihood over possible latent variables $\mathbf{X}$ and the parameters $\boldsymbol{\theta}$, represented as

$$\begin{aligned} p(\mathbf{Y} \mid \mathcal{M}) &= \iint p(\mathbf{Y}, \mathbf{X}, \boldsymbol{\theta} \mid \mathcal{M}) \, \mathrm{d}\mathbf{X} \, \mathrm{d}\boldsymbol{\theta} \\ &= \iint p(\mathbf{X}, \boldsymbol{\theta} \mid \mathbf{Y}, \mathcal{M}) p(\mathbf{Y} \mid \mathcal{M}) \, \mathrm{d}\mathbf{X} \, \mathrm{d}\boldsymbol{\theta}. \end{aligned} \tag{2.3}$$

This evidence of a selected model $\mathcal{M}$ can be seen as the expectation of the likelihood given the posterior distribution of all latent variables. The consideration of the entire

posterior distribution due to the Bayesian modeling fully addresses uncertainty and variability in estimating the parameters $\boldsymbol{\theta}$. Also, the *Occam's Razor effect* resulting from marginalization across parameters penalizes overly complex models [6]. However, for most probabilistic models, the corresponding likelihood in Equation (2.3) is intractable, and exact Probabilistic Inference is daunting [3, 4, 6].

Instead of directly inferring the marginal likelihood, two well-known families of techniques in the research community consider the approximation of Equation (2.3). First, numerical sampling techniques such as *Markov Chain Monte-Carlo (MCMC)* are used [4–6, 65–67]. These stochastic approximation techniques converge to the true distribution given a sufficient number of samples [4, 6]. However, the computational complexity grows exponentially with the number of dimensions of the latent space. Therefore, these algorithms suffer from the well-known *curse of dimensionality* [4, 6].

In contrast, a deterministic way is to consider the approximation inference methods known as *Variational Inference (VI)* or *Variational Bayes (VB)* [3–6, 63, 64, 68–73]. The family of these methods results when we extend the Expectation Maximization algorithm introduced previously. Here, the Probabilistic Inference problem is again considered as an optimization function of the data's log-likelihood. Like in section 2.2, they introduce a variational distribution $q(\mathbf{X}, \boldsymbol{\theta})$ over all latent variables and apply Jensen's inequality resulting in an ELBO

$$\begin{aligned}
\mathcal{L}(\mathbf{X}, \boldsymbol{\theta}) &= \sum_{t=1}^{T} \log \iint p(\mathbf{y}_t, \mathbf{x}_t, \boldsymbol{\theta} \mid \mathcal{M}) \, \mathrm{d}\mathbf{x}_t \, \mathrm{d}\boldsymbol{\theta} \\
&\geq \sum_{t=1}^{T} \iint q(\mathbf{X}, \boldsymbol{\theta}) \log \frac{p(\mathbf{y}_t, \mathbf{x}_t, \boldsymbol{\theta} \mid \mathcal{M})}{q(\mathbf{X}, \boldsymbol{\theta})} \, \mathrm{d}\mathbf{X} \, \mathrm{d}\boldsymbol{\theta} \\
&= \sum_{t=1}^{T} \left(\log p(\mathbf{y}_t \mid \boldsymbol{\theta}, \mathcal{M}) - \mathrm{KL}(p(\mathbf{x}_t, \boldsymbol{\theta} \mid \mathbf{y}_t, \mathcal{M}) \parallel q(\mathbf{X}, \boldsymbol{\theta})) \right) \\
&\stackrel{\text{def}}{=} \mathcal{L}_{\text{ELBO}}(q(\mathbf{X}, \theta)).
\end{aligned}$$

However, allowing every possible choice of $q(\mathbf{X}, \boldsymbol{\theta})$ entails intractability due to the dependencies between all latent variables $\mathbf{X}$ and $\boldsymbol{\theta}$ [3, 4, 6]. There are, in general, two ways to restrict the shape of the distribution. One way assumes that the variational distribution has a particular parametric form $q_{\boldsymbol{\omega}}(\mathbf{X}, \boldsymbol{\theta})$ parameterized by $\boldsymbol{\omega}$. This transforms the ELBO's

optimization into a nonlinear optimization problem

$$\boldsymbol{\omega}^* = \arg\max_{\boldsymbol{\omega}} \mathcal{L}_{\text{ELBO}}(q_{\boldsymbol{\omega}}(\mathbf{X}, \boldsymbol{\theta})),$$

and thus can be solved by applying classical methods of nonlinear optimization [4, 6]. An alternative is to constrain the variational distribution $q(\mathbf{X}, \boldsymbol{\theta})$ to represent a particular family of possible distributions. This restriction requires a pre-assumed factorization of the variational distribution

$$q(\mathbf{X}, \boldsymbol{\theta}) = \prod_{t=0}^{T} q(\mathbf{x}_t) q(\boldsymbol{\theta}),$$

the so-called *mean field assumption*. The mean field assumption makes the optimization problem tractable. Applying Calculus of Variations and the Euler-Lagrange formalism leads to optimal parameters for each factor of the variational distribution [3]. Like EM, a cyclic procedure results

$$\begin{aligned}
\textbf{VBE step:} \quad & q(\mathbf{x}_i)^* \quad \leftarrow \arg\max_{q(\mathbf{x}_i)} \mathcal{L}_{\text{ELBO}}(q(\mathbf{x}_i), q(\boldsymbol{\theta})) \quad \forall i, \\
\textbf{VBM step:} \quad & q(\boldsymbol{\theta})^* \quad \leftarrow \arg\max_{q(\boldsymbol{\theta})} \mathcal{L}_{\text{ELBO}}(q(\mathbf{x}_i)^*, q(\boldsymbol{\theta})),
\end{aligned}$$

in which each factor is successively updated while the others are kept fixed. In the research community this procedure is popularly known as *Variational Bayes Expectation Maximization algorithm (VBEM)* [3]. Due to the maximization of a convex lower bound, an optimal solution is guaranteed [4, 6, 74]. However, the accuracy of the optimal solutions depends on the assumption made about the variational distribution. If the true posterior distribution is representable by a functional form or lies in the family of possible distributions, the KL divergence vanishes. Otherwise, the resulting variational distribution gives an approximation and an offset between the marginal log-likelihood and the ELBO [3, 4, 6]. The choice of the best setting is therefore crucial.

Central parameters in the previously mentioned algorithms are the so-called *hyperparameters* of the probability density model. For instance, the hyperparameters are the natural parameters of the prior distribution $p(\boldsymbol{\theta})$. In a Gaussian distribution, this corresponds to the mean and the covariance or precision matrix. While the EM algorithm iteratively infers over the unknown latent variables $\mathbf{X}$ and maximizes the model's parameters $\boldsymbol{\theta}$, the VBEM algorithm transforms this into a purely Probabilistic Inference problem [3–6]. However, in the methods presented so far, the hyperparameters are always assumed to be

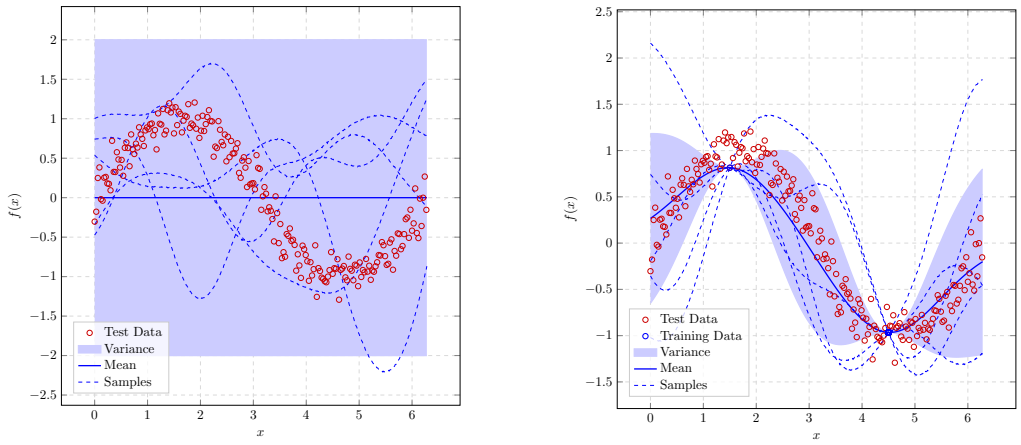
fixed. Therefore, it is a reasonable extension to maximize them as well [3, 6]. The VBEM algorithm first infers over the latent variables $\mathbf{X}$ and $\boldsymbol{\theta}$. Subsequently, a maximization step (M step) is performed to optimize the hyperparameters while keeping the variational distributions fixed. This extension is called *Empirical Bayesian learning*. This extension abandons the fully Bayesian approach due to maximizing over the hyperparameters. In other words, only optimal point estimates for the hyperparameters are considered again, without taking uncertainties and variabilities into account [3, 6].

2.4. Gaussian Processes

The previous sections consider Probability Density Estimation. The presented methods are interested in an approximation of the probability density model responsible for the given observations. For this purpose, parametric models $\mathcal{M}$ are utilized, which are parameterized by $\boldsymbol{\theta}$. Due to this parameterization, these Machine Learning methods belong to the so-called *parametric methods* [4, 6]. In parametric methods, training and prediction seem to form two completely independent phases [4, 6, 39]. The predictive density distribution, given by

$$\begin{aligned} p(\mathbf{y}^* \mid \mathbf{Y}, \mathcal{M}) &= \int p(\mathbf{y}^* \mid \mathbf{Y}, \boldsymbol{\theta}, \mathcal{M}) p(\boldsymbol{\theta} \mid \mathbf{Y}, \mathcal{M}) \, \mathrm{d}\boldsymbol{\theta} \\ &= \int p(\mathbf{y}^* \mid \boldsymbol{\theta}, \mathcal{M}) p(\boldsymbol{\theta} \mid \mathbf{Y}, \mathcal{M}) \, \mathrm{d}\boldsymbol{\theta}, \end{aligned}$$

clearly shows this separation. The probability of a new data point $\mathbf{y}^*$ depends only on the model $\mathcal{M}$ and the parameters learned $\boldsymbol{\theta}$. In a sense, parametric methods rely solely on the trained model and seem to neglect the given data $\mathbf{Y}$. Thus, the choice of parameterization is an essential component [4–6]. On the one hand, a too strongly parameterized model limits the representability of arbitrary functions. On the other hand, low restriction leads to complex models with many parameters and thus to the danger of overfitting [4, 5, 39]. While the discussed Bayesian learning (see Section 2.3) is one way to find a satisfying tradeoff, another attractive way is to consider nonparametric Bayesian methods such as *Gaussian Processes (GPs)*. In Machine Learning, Gaussian Processess are traditionally used for regression and classification problems and are thus a topic in supervised learning [4, 6, 38, 39, 61, 75]. This paradigm combines the learning and prediction phases attractively. Instead of assuming a parametric model and considering a distribution over the parameters, GPs infer directly over functions [38, 39]. Loosely



(a) (b)

Figure 2.3.: These figures show the behavior of Gaussian process regression when attempting to capture the underlying sinusoidal stochastic process. On the left side (a), only a zero-mean Gaussian process prior is given. Therefore, the resulting trajectories evolve around the zero-mean without following the sinusoidal shape. On the right side (b), two data points are given. The trajectory of the mean function captures parts of the sinusoidal shape from the given data. The sampled trajectories tend to follow the mean and thus the sinusoidal the closer the distance to the observations. As the distance increases, the trajectories become less constrained and take on arbitrary smooth forms [39].

speaking, the idea is to constrain the functions using only a prior distribution over all possible functions. The probability assigned to a function increases the more it fits the observed data. Thus, the inference problem is transformed into an infinite-dimensional functional space [4, 6, 39]. In general, however, dealing with infinite-dimensional objects is challenging.

A convincing prior distribution over possible function values $f(\cdot)$ is the generalization of a Gaussian distribution called Gaussian Process Prior. Given some set of input variables $\mathbf{X} = \{\mathbf{x}_0, \cdots, \mathbf{x}_T\}$, the Prior forms a multivariate Gaussian distribution over corresponding

function values $\mathbf{F} = \{\mathbf{f}_0 = f(\mathbf{x}_0), \cdots, \mathbf{f}_T = f(\mathbf{x}_T)\}$ by

$$\mathbf{F} \sim \mathcal{GP}\left(\mathbf{m}_\mathbf{X}, \mathbf{K}_{\mathbf{XX}}\right),$$

$$\begin{bmatrix} \mathbf{f}_0 \\ \vdots \\ \mathbf{f}_T \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mathbf{m}(\mathbf{x}_0) \\ \vdots \\ \mathbf{m}(\mathbf{x}_T) \end{bmatrix}, \begin{bmatrix} \mathbf{k}(\mathbf{x}_0, \mathbf{x}_0, \boldsymbol{\theta}) & \cdots & \mathbf{k}(\mathbf{x}_0, \mathbf{x}_T, \boldsymbol{\theta}) \\ \vdots & \ddots & \vdots \\ \mathbf{k}(\mathbf{x}_T, \mathbf{x}_0, \boldsymbol{\theta}) & \cdots & \mathbf{k}(\mathbf{x}_T, \mathbf{x}_T, \boldsymbol{\theta}) \end{bmatrix} \right),$$

where $\mathbf{m}_\mathbf{X}$ and $\mathbf{K}_{\mathbf{XX}}$ correspond to a mean vector and a covariance or kernel matrix [38]. Due to this generalization of Gaussian distributions, the handling with infinite-dimensional objects becomes feasible [39]. It is very common to set the mean function $\mathbf{m}(\cdot)$ to zero, hence, it is also done in this work. The entries of the kernel matrix corresponding to some kernel functions $\mathbf{k}(\cdot, \cdot, \boldsymbol{\theta})$ depending on adjustable parameters. In the left part of Figure 2.3, samples drawn from a zero-mean Gaussian Process prior are shown. They exemplify the resulting smooth trajectories and evolution around the zero-mean function. Since no data of the underlying process is given, these trajectories are not sinusoidal.

In a regression problem, the outcoming function should match given observations $\mathbf{Y} = \{\mathbf{y}_0, \cdots, \mathbf{y}_T\}$. Assuming a Gaussian likelihood measuring the uncertainty between the observations and the functions with some precision value λ_y the posterior distribution is expressed by

$$p(\mathbf{F} \mid \mathbf{Y}, \mathbf{X}) \propto \mathcal{N}(\mathbf{Y} \mid \mathbf{F}, \lambda_y^{-1}\mathbf{I})\mathcal{GP}(\mathbf{F} \mid \mathbf{0}, \mathbf{K}_{\mathbf{XX}}),$$

which is analytically computable in closed form [4]. The most probable function values $\mathbf{F}$ given the inputs $\mathbf{X}$ and observations $\mathbf{Y}$ correspond to

$$\widetilde{\boldsymbol{\mu}} = \mathbf{K}_{\mathbf{XX}}\left(\mathbf{K}_{\mathbf{XX}} + \lambda_y^{-1}\right)^{-1}\mathbf{Y}, \tag{2.4}$$

$$\widetilde{\boldsymbol{\Sigma}} = \mathbf{K}_{\mathbf{XX}} - \mathbf{K}_{\mathbf{XX}}\left(\mathbf{K}_{\mathbf{XX}} + \lambda_y^{-1}\right)^{-1}\mathbf{K}_{\mathbf{XX}}. \tag{2.5}$$

The mean value of the function $\widetilde{\boldsymbol{\mu}}$ corresponds to the observed data $\mathbf{Y}$ if $\lambda_y \to \infty$ or, equivalently, the variance converges to zero. The predictive density distribution of an unknown observation $\mathbf{F}^*$ for a new input variable $\mathbf{X}^*$ is given by

$$p(\mathbf{F}^* \mid \mathbf{X}^*, \boldsymbol{\mathcal{D}}) = \int p(\mathbf{F}^* \mid \mathbf{X}^*, \mathbf{F}) p(\mathbf{F} \mid \mathbf{Y}, \mathbf{X}) \, \mathrm{d}\mathbf{F},$$

and represents another interesting query. The integrand corresponds to a multivariate Gaussian distribution represented by

$$\begin{bmatrix} \mathbf{Y} \\ \mathbf{F}^* \end{bmatrix} \sim \mathcal{GP}\left(\begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{K}_{\mathbf{XX}} & \mathbf{K}_{\mathbf{XX}^*} \\ \mathbf{K}_{\mathbf{X}^*\mathbf{X}} & \mathbf{K}_{\mathbf{X}^*\mathbf{X}^*} \end{bmatrix} \right),$$

due to its assumed Gaussian nature [4, 6]. Similar to predicting the value function for known observations $\mathbf{Y}$, the marginalization is analytically solvable and forms a Gaussian distribution [4]. In this way, the new observation's predictive probability results in

$$p(\mathbf{F}^* \mid \mathbf{X}^*, \boldsymbol{\mathcal{D}}) \propto \mathcal{N}(\mathbf{F}^* \mid \widetilde{\boldsymbol{\mu}}^*, \widetilde{\boldsymbol{\Sigma}}^*),$$

with the mean vector and covariance matrix

$$\widetilde{\boldsymbol{\mu}}^* = \mathbf{K}_{\mathbf{X}^*\mathbf{X}} \left(\mathbf{K}_{\mathbf{XX}} + \lambda_y^{-1} \right)^{-1} \mathbf{Y}, \tag{2.6}$$

$$\widetilde{\boldsymbol{\Sigma}}^* = \mathbf{K}_{\mathbf{X}^*\mathbf{X}^*} - \mathbf{K}_{\mathbf{X}^*\mathbf{X}} \left(\mathbf{K}_{\mathbf{XX}} + \lambda_y^{-1} \right)^{-1} \mathbf{K}_{\mathbf{XX}^*}. \tag{2.7}$$

The predicted function values differ from the predicted mean value the greater the distance to the given values $\mathbf{Y}$. The closer the distance, the more they are constrained. The right part of Figure 2.3 shows this behavior for two observations of the data $\mathbf{Y}$. The curve of the mean function already captures parts of the sinusoidal shape. The samples drawn from the resulting distribution tend to follow the mean close to the observed data points. With an increasing number of data points, the model provides a closer fit to the data [4, 38, 39].

The properties of the sampled trajectories, such as smoothness, are determined by an appropriate choice of the covariance or kernel function [39]. An overview of the kernels relevant to the context of this work is given in the Appendix C. A kernel usually represents a similarity measure between two points [4]. In case when the two points are close to each other and show strong similarities, they are assigned a high value. Consequently, the resulting function values similarly exhibits high similarities. Further, the use of nonlinear kernels enables a mapping into a nonlinear feature space. This mapping extends Gaussian processes' capabilities and is also known as the *Kernel Trick* in Machine Learning [4, 6]. Learning the kernel parameters $\boldsymbol{\theta}$ is done by maximizing the marginal log-likelihood given

by

$$\begin{aligned}\mathcal{L}(\boldsymbol{\theta}) &= \log p(\mathbf{y}) \\ &= \log \int p(\mathbf{Y} \mid \mathbf{F}) p(\mathbf{F} \mid \mathbf{X}) \, \mathrm{d}\mathbf{F} \\ &= \log \mathcal{N}\left(\mathbf{Y} \mid \mathbf{0}, \mathbf{K}_{\mathbf{XX}}(\boldsymbol{\theta}) + \lambda_y^{-1}\right) \\ &= -\frac{T}{2}\log 2\pi - \frac{1}{2}\log|\mathbf{K}_{\mathbf{XX}}(\boldsymbol{\theta})| - \frac{1}{2}\mathrm{Tr}\left(\mathbf{K}_{\mathbf{XX}}^{-1}(\boldsymbol{\theta})\mathbf{Y}\mathbf{Y}^T\right), \end{aligned} \tag{2.8}$$

which is computable in closed form due to its Gaussian nature [4, 6, 38, 39].

In summary, the Gaussian process framework does not rely on parametric methods and leverages the seen data to predict future outcomes. Therefore, the complexity of the method depends on the size of the given data [38,39,43,54]. For this reason, the Bayesian Gaussian process belongs to the nonparametric methods. In these methods, the number of parameters and the complexity increase with the size of the data [4, 6]. The automatic adjustment of the complexity seems attractive in itself. However, it is associated with a high computational cost. The computation time increases cubically $\left(\mathcal{O}(T^3)\right)$ with the number of given data due to the inversion of the kernel matrix (see Equations (2.4) to (2.7) and (2.8)) [38, 39, 43]. In the next chapter, techniques that mitigate this problem are proposed and discussed.

2.5. Sparse Gaussian Processes

As mentioned in the previous chapter, the cubic complexity associated with increasing data size is a challenge for Gaussian Processes (GPs). Numerous works focused on this problem and have proposed *sparse Gaussian Process methods* [40–43, 75–79]. These methods allow scaling of Gaussian process models to large data sets without unfavorably affecting the prediction quality [80]. A subset of sparse Gaussian process techniques considers global approximation techniques. These techniques apply an approximation of the $T \times T$ kernel matrix $\mathbf{K}_{\mathbf{X},\mathbf{X}}$. Such techniques use low-rank matrix approximation to decompose the kernel into the following product

$$\mathbf{K}_{\mathbf{XX}} = \mathbf{K}_{\mathbf{X}\tilde{\mathbf{X}}}\mathbf{K}_{\tilde{\mathbf{X}}\tilde{\mathbf{X}}}^{-1}\mathbf{K}_{\tilde{\mathbf{X}}\mathbf{X}},$$

where the middle term corresponds to a $D \times D$ Kernel over $D \ll T$ data points $\widetilde{\mathbf{X}}$ [80–82]. The use of deterministic techniques from this domain results in approximations that cause high prediction variance and are consequently unfavorable [80, 83]. Probabilistic techniques, therefore, model this decomposition in the probabilistic generative model. They consider the so-called *inducing inputs* $\widetilde{\mathbf{X}}$ as parameters living in the same space as the function inputs $\mathbf{X}$ [77]. The prior distribution over all possible functions, expressed by a Gaussian Process prior, expands to

$$\begin{aligned} p(\mathbf{F} \mid \mathbf{X}) &= \int p(\mathbf{F}, \widetilde{\mathbf{F}} \mid \mathbf{X}) \, \mathrm{d}\widetilde{\mathbf{F}} \\ &= \int \mathcal{GP}\left(\begin{bmatrix} \mathbf{F} \\ \widetilde{\mathbf{F}} \end{bmatrix} \;\middle|\; \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{K}_{\mathbf{X}\mathbf{X}} & \mathbf{K}_{\mathbf{X}\widetilde{\mathbf{X}}} \\ \mathbf{K}_{\widetilde{\mathbf{X}}\mathbf{X}} & \mathbf{K}_{\widetilde{\mathbf{X}}\widetilde{\mathbf{X}}} \end{bmatrix} \right) \mathrm{d}\widetilde{\mathbf{F}} \\ &= \mathcal{GP}\left(\mathbf{F} \mid \widetilde{\boldsymbol{\mu}}, \widetilde{\boldsymbol{\Sigma}} \right), \end{aligned}$$

taking into account the inducing inputs and corresponding outputs $\widetilde{\mathbf{F}}$, known as *inducing variables* [40–43, 76, 77]. The resulting prior distribution again forms a Gaussian process prior with corresponding mean and covariance

$$\begin{aligned} \widetilde{\boldsymbol{\mu}} &= \mathbf{K}_{\mathbf{X}\widetilde{\mathbf{X}}} \mathbf{K}_{\widetilde{\mathbf{X}}\widetilde{\mathbf{X}}}^{-1} \widetilde{\mathbf{F}}, \\ \widetilde{\boldsymbol{\Sigma}} &= \mathbf{K}_{\mathbf{X}\mathbf{X}} - \mathbf{K}_{\mathbf{X}\widetilde{\mathbf{X}}} \mathbf{K}_{\widetilde{\mathbf{X}}\widetilde{\mathbf{X}}}^{-1} \mathbf{K}_{\widetilde{\mathbf{X}}\mathbf{X}}. \end{aligned}$$

These equations already show the advantage of using inducing points, which only require an inversion over a $D \times D$ matrix, thus reducing the complexity to $\mathcal{O}(TD^2)$. In order to avoid running into high prediction variances, we extend the predictive density distribution to

$$p(\mathbf{F}^* \mid \mathbf{X}^*, \mathbf{Y}, \mathbf{X}) = \iint p(\mathbf{F}^* \mid \mathbf{X}^*, \mathbf{F}, \widetilde{\mathbf{F}}) p(\mathbf{F}, \widetilde{\mathbf{F}} \mid \mathbf{Y}, \mathbf{X}) \, \mathrm{d}\mathbf{F} \, \mathrm{d}\widetilde{\mathbf{F}},$$

considering the inducing variables and inducing inputs. Instead of inferring exactly this distribution, the assumption is made that the inducing variables form sufficient statistics for the predictions [77]. Consequently, the likelihood of a new data point $\mathbf{F}^*$ depends solely on the inducing variables and not on the tuple $(\mathbf{F}, \widetilde{\mathbf{F}})$. This reason implies the name inducing variable and inducing inputs. Under this assumption, the predictive probability density is formed

$$q(\mathbf{F}^* \mid \mathbf{X}^*) = \iint p(\mathbf{F}^* \mid \mathbf{X}^*, \widetilde{\mathbf{F}}) \underbrace{p(\mathbf{F} \mid \mathbf{X}\widetilde{\mathbf{F}}) q(\widetilde{\mathbf{F}})}_{q(\mathbf{F}, \widetilde{\mathbf{F}})} \, \mathrm{d}\mathbf{F} \, \mathrm{d}\widetilde{\mathbf{F}},$$

where $q(\mathbf{F}, \widetilde{\mathbf{F}})$ is a variational distribution that approximates the true posterior distribution $p(\mathbf{F}, \widetilde{\mathbf{F}} \mid \mathbf{Y}, \mathbf{X}, \widetilde{\mathbf{X}})$. The goal is to optimize the inducing inputs $\widetilde{\mathbf{X}}$ to minimize the KL divergence between the two distributions. This minimization is similar to maximizing an Evidence Lower Bound, using the variational method discussed in Section 2.3 [3, 4, 6]. Thus, if the marginal log-likelihood is assumed, the ELBO is as follows

$$\begin{aligned}
\log p(\mathbf{Y} \mid \mathbf{X}) &= \log \int p(\mathbf{Y} \mid \mathbf{F}) p(\mathbf{F} \mid \mathbf{X}) \, \mathrm{d}\mathbf{F} \\
&= \log \iint p(\mathbf{Y} \mid \mathbf{F}) p(\mathbf{F} \mid \mathbf{X}, \widetilde{\mathbf{F}}) p(\widetilde{\mathbf{F}} \mid \widetilde{\mathbf{X}}) \, \mathrm{d}\mathbf{F} \, \mathrm{d}\widetilde{\mathbf{F}} \\
&\geq \iint q(\mathbf{F}, \widetilde{\mathbf{F}}) \log \frac{p(\mathbf{Y} \mid \mathbf{F}) p(\mathbf{F} \mid \mathbf{X}, \widetilde{\mathbf{F}}) p(\widetilde{\mathbf{F}} \mid \widetilde{\mathbf{X}})}{q(\mathbf{F}, \widetilde{\mathbf{F}})} \, \mathrm{d}\mathbf{F} \, \mathrm{d}\widetilde{\mathbf{F}} \\
&= \iint p(\mathbf{F} \mid \widetilde{\mathbf{F}}) q(\widetilde{\mathbf{F}}) \log \frac{p(\mathbf{Y} \mid \mathbf{F}) p(\widetilde{\mathbf{F}} \mid \widetilde{\mathbf{X}})}{q(\widetilde{\mathbf{F}})} \, \mathrm{d}\mathbf{F} \, \mathrm{d}\widetilde{\mathbf{F}} \\
&\stackrel{\text{def}}{=} \mathcal{L}_{\text{ELBO}}(\widetilde{\mathbf{X}}, q(\widetilde{\mathbf{F}}), \boldsymbol{\theta}),
\end{aligned}$$

depending on the inducing inputs $\widetilde{\mathbf{X}}$, the kernel's hyperparameters $\boldsymbol{\theta}$, and the variational distribution $q(\widetilde{\mathbf{F}})$. An optimal Gaussian distribution in closed form is obtained from the derivative w.r.t. $q(\widetilde{\mathbf{F}})$ using Euler-Lagrange and Variational Calculus. Substituting this optimal distribution into the lower bound leads to a new objective given by

$$\begin{aligned}
\mathcal{L}_{\text{ELBO}}(\widetilde{\mathbf{X}}, \boldsymbol{\theta}) = \; & \log \mathcal{N}(\mathbf{Y} \mid \mathbf{0}, \mathbf{K}_{\mathbf{X},\widetilde{\mathbf{X}}}(\boldsymbol{\theta}) \mathbf{K}^{-1}_{\widetilde{\mathbf{X}}\widetilde{\mathbf{X}}}(\boldsymbol{\theta}) \mathbf{K}_{\widetilde{\mathbf{X}}\mathbf{X}}(\boldsymbol{\theta})) \\
& - \frac{1\lambda^2}{2} \operatorname{Tr}\left(\mathbf{K}_{\mathbf{X}\mathbf{X}}(\boldsymbol{\theta}) - \mathbf{K}_{\mathbf{X}\widetilde{\mathbf{X}}}(\boldsymbol{\theta}) \mathbf{K}^{-1}_{\widetilde{\mathbf{X}}\widetilde{\mathbf{X}}}(\boldsymbol{\theta}) \mathbf{K}_{\widetilde{\mathbf{X}}\mathbf{X}}(\boldsymbol{\theta}) \right),
\end{aligned}$$

where gradient descent based optimization techniques are used to optimize the corresponding hyperparameters and inducing inputs [77]. The trace term accounts for the total variance between the true function $\mathbf{F}$ and the inducing variables $\widetilde{\mathbf{F}}$ [50, 77]. While maximizing the likelihood, the variance between $\mathbf{F}$ and $\widetilde{\mathbf{F}}$ is minimized. If the inducing inputs $\widetilde{\mathbf{X}}$ match the given inputs $\mathbf{X}$, the trace vanishes, and the full Gaussian process is obtained (see Equation (2.8)) [77]. A closer look at the augmentation of the marginal log-likelihood reveals further that the inducing inputs only affect the inducing variables and thus the variational distribution. The marginalization of the inducing variables renders the influence of the inducing inputs vanishes as well. For this reason, the inducing variables are referred to as *variational parameters* [50, 54, 77]. They only influence the variational approximation $q(\widetilde{\mathbf{F}})$ and not the original joint density model $p(\mathbf{Y}, \mathbf{F})$. Thus, they only change how tightly the Lower Bound corresponds to the marginal log-likelihood [50, 54, 77].

2.6. Gaussian Process Latent Variable Models

In numerous fields of research, high-dimensional data $\mathbf{Y}$ is used. For instance, fluid mechanics or computer vision analyze given data in images or video frames [4, 6, 25, 26]. Thus, the dimensionality of a single data point is sometimes enormous. However, in these high-dimensional spaces, the data points are close to each other and form a much less-dimensional manifold [35]. The continuous LVMs (see Section 2.2) used in Machine Learning aim to determine a mapping into such a low-dimensional manifold while preserving the data's inherent structure [4, 6, 35]. Subsequently, analysis and interpretation of the data take place in this low-dimensional spaces. The mapping of a point $\mathbf{x}_t$ in the latent space to a single scalar observation in the observation space is expressed by

$$y_{nt} = f(\mathbf{x}_t, \mathbf{c}_n) + \epsilon_{nt}, \tag{2.9}$$

where $\mathbf{c}_n$ is a vector describing the parameters of the mapping $f(\cdot, \cdot)$ [35, 44–46]. The associated y_{nt} corresponds to the nth row and the tth column of the given observation matrix $\mathbf{Y}$. N and T denote the dimension and the number of given observations, respectively. Assuming an independently drawn noise $\epsilon_{nt} \sim \mathcal{N}(0, \lambda_{\mathbf{Y}}^{-1}\mathbf{I})$ the mapping in Equation (2.9) is equivalent to

$$p(\mathbf{Y} \mid \mathbf{X}, \mathbf{C}) = \prod_{n=1}^{N} \prod_{t=0}^{T} \mathcal{N}(y_{nt} \mid f(\mathbf{x}_t, \mathbf{c}_n), \lambda_{\mathbf{Y}}^{-1}), \tag{2.10}$$

a Gaussian likelihood function [4, 35, 44]. The variables in the latent space resulting from the dimensionality reduction are collected in a matrix $\mathbf{X} \in \mathbb{R}^{M \times T}$. In dimensionality reduction frameworks, usually, the latent space dimension is smaller than that of the observations, i.e., $M < N$. Several frameworks in these fields like *Principle Component Analysis (PCA)*, *Factor Analysis (FA)* and *Independent Component Analysis (ICA)* treat the latent variables as random variables [4, 6]. They assume an appropriate prior over the variables and perform marginalization over them. Subsequently, the parameters collected in $\mathbf{C}$ are optimized. First, marginalizing the latent variables and then optimizing the parameters formulates the primal formalism for dimensionality reduction [44].

The dual formalism accordingly follows the opposite way. The parameters of the mapping $\mathbf{C}$ are considered random variables and are assigned a prior distribution $p(\mathbf{C})\mathcal{N}(\mathbf{c}_n \mid 0, \lambda_{\mathbf{C}}^{-1}\mathbf{I})$. They are then marginalized, and the latent variables $\mathbf{X}$ are optimized [44, 45]. For

simplicity, a linear mapping is assumed, and thus the function becomes $f(\mathbf{x}_t, \mathbf{c}_n) = \mathbf{x}_t^T \mathbf{c}_n$. The marginal likelihood is represented in closed form by

$$\begin{aligned}
p(\mathbf{Y} \mid \mathbf{X}, \lambda_{\mathbf{Y}}, \lambda_{\mathbf{C}}) &= \int \prod_{t=0}^{T} p(\mathbf{y}_t \mid \mathbf{C}, \mathbf{x}_t, \lambda_{\mathbf{Y}}) p(\mathbf{C}) \, \mathrm{d}\mathbf{C} \\
&= \int \cdots \int \prod_{t=0}^{T} \prod_{n=1}^{N} \mathcal{N}(\mathbf{y}_{n,t} \mid \mathbf{x}_t^T \mathbf{c}_n, \lambda_{\mathbf{Y}} \mathbf{I}) \mathcal{N}(\mathbf{c}_n \mid 0, \lambda_{\mathbf{C}}^{-1} \mathbf{I}) \, \mathrm{d}\mathbf{c}_1 \cdots \mathrm{d}\mathbf{c}_N \\
&= \prod_{n=1}^{N} \int \mathcal{N}(\mathbf{Y}_n^T \mid \mathbf{X}^T \mathbf{c}_n, \lambda_{\mathbf{Y}} \mathbf{I}) \mathcal{N}(\mathbf{c}_n \mid 0, \lambda_{\mathbf{C}}^{-1} \mathbf{I}) \, \mathrm{d}\mathbf{c}_n \\
&= \prod_{n=1}^{N} \mathcal{N}\left(\mathbf{Y}_n^T \mid 0, \lambda_{\mathbf{Y}}^{-1} \mathbf{I} + \lambda_{\mathbf{C}}^{-1} \mathbf{X}^T \mathbf{X}\right) \\
&= \prod_{n=1}^{N} \mathcal{GP}\left(\mathbf{Y}_n^T \mid 0, \mathbf{K}_{\mathbf{X},\mathbf{X}}(\boldsymbol{\theta})\right),
\end{aligned}$$

expressed by N independent Gaussian processes [44]. The vector $\mathbf{Y}_n$ corresponds to the nth row of the given observation matrix. The subsequent application of Maximum Likelihood estimate results in the new objective

$$\mathcal{L}(\mathbf{X}, \boldsymbol{\theta}) = -\frac{NT}{2} \log(2\pi) - \frac{D}{2} \log|\mathbf{K}_{\mathbf{X},\mathbf{X}}(\boldsymbol{\theta})| - \frac{1}{2} \mathrm{Tr}\left(\mathbf{K}_{\mathbf{X},\mathbf{X}}(\boldsymbol{\theta})^{-1} \mathbf{Y}^T \mathbf{Y}\right),$$

and forms the Gaussian Process Latent Variable Models. For a linear kernel, eigendecomposition results in a closed-form analytic optimal solution [44, 45]. The Kernel Trick (see Section 2.4) achieves a generalization to nonlinear mappings from the latent space to the data space [44, 45]. A suitable nonlinear kernel function replaces the linear kernel. This generalization has the consequence that there is no closed-form solution available, and thus nonlinear optimization techniques are applied [44].

The GP-LVMs seem to form a combination of the GPs and the LVMs frameworks. It considers a regression problem with known outputs and unknown inputs. The optimal inputs $\mathbf{X}^*$ most likely responsible for the seen data $\mathbf{Y}$ are estimated using Maximum Likelihood estimate (MLE) [44, 45]. However, as discussed in the previous chapter, the resulting latent variable estimate is only a point estimate [4]. Consequently, it does not consider uncertainty and variability. The use of point estimates also carries the risk of overfitting (see Section 2.1) if overly complex models are assumed [6]. In the GP-LVMs

framework, overly complex models result when the latent variables exhibit a higher dimension than the observations, i.e., $M > N$. The majority of problems considered in the literature address dimensionality reductions, and thus such a situation does not occur [44–47, 50]. Therefore, the application of GP-LVMs achieves satisfactory results. The Gaussian Processes resulting from marginalization entail another drawback in the GP-LVMs framework. The computational complexity of each evaluation of the optimization problem increases cubically with the amount of given data, resulting in a complexity of $\mathcal{O}(T^3)$ (see Section 2.4). In the following chapter, a method is presented that alleviates the aforementioned problems based on Sparse Gaussian Process techniques (see Section 2.5).

2.7. Bayesian Gaussian Process Latent Variable Models

The GP-LVMs formalism entails the problem of a resulting point estimate of the latent variables. If the dimensions of the latent space M is larger than the dimensions of the observations space N, there is a risk of overfitting overfitting the estimate of the latent variables $\mathbf{X}$ [50]. A suitable extension is to consider uncertainties and variabilities in the latent variables' estimates utilizing a prior distribution

$$p(\mathbf{X}) = \prod_{t=0}^{T} \mathcal{N}(\mathbf{x}_t \mid 0, \mathbf{I}).$$

This consideration extends the GP-LVMs formalism. Optimization of the resulting objective remains, obviously, a point estimate and results in the Maximum A Posteriori probability estimate [4, 6]. The better way is to treat the latent variables $\mathbf{X}$ as random variables and marginalize them in the marginal log-likelihood. This consideration extends the GPLVM framework to a Bayesian learning paradigm [50]. Simultaneously, the computational complexity is to be reduced by sparse Gaussian process techniques [46, 77]. As discussed in Section 2.5, inducing variables $\widetilde{\mathbf{F}}$ are introduced with their corresponding inducing inputs $\widetilde{\mathbf{X}}$. On the one hand, these inducing pairs reduce the computational complexity to $\mathcal{O}(TD^2)$ [77]. On the other hand, the inducing variables are assumed to provide sufficient statistics. Thus, the function values $\mathbf{F}$ are conditional independent given $\widetilde{\mathbf{F}}$ [77]. The extension of the GP-LVMs is expressed by the following marginal likelihood

$$\begin{aligned}\mathcal{GP}\left(\mathbf{Y}_n^T \mid 0, \mathbf{K}_{\mathbf{X},\mathbf{X}}\right) &= \int \mathcal{N}(\mathbf{Y}_n \mid \mathbf{F}_n, \lambda_{\mathbf{Y}}^{-1}\mathbf{I})\mathcal{GP}(\mathbf{F}_n \mid 0, \mathbf{K}_{\mathbf{X},\mathbf{X}})\,\mathrm{d}\mathbf{F}_n \\ &= \iint \mathcal{N}(\mathbf{Y}_n \mid \mathbf{F}_n, \lambda_{\mathbf{Y}}^{-1}\mathbf{I})\mathcal{N}(\mathbf{F}_n \mid \boldsymbol{\mu}_n, \boldsymbol{\Sigma})\mathcal{GP}(\widetilde{\mathbf{F}}_n \mid \mathbf{0}, \mathbf{K}_{\widetilde{\mathbf{X}}\widetilde{\mathbf{X}}})\,\mathrm{d}\mathbf{F}_n\,\mathrm{d}\widetilde{\mathbf{F}}_n,\end{aligned}$$

with the quantities $\boldsymbol{\mu}_n = \mathbf{K}_{\mathbf{X}\widetilde{\mathbf{X}}}\mathbf{K}_{\widetilde{\mathbf{X}}\widetilde{\mathbf{X}}}^{-1}\widetilde{\mathbf{F}}_n$ and $\boldsymbol{\Sigma} = \mathbf{K}_{\mathbf{X}\mathbf{X}} - \mathbf{K}_{\mathbf{X}\widetilde{\mathbf{X}}}\mathbf{K}_{\widetilde{\mathbf{X}}\widetilde{\mathbf{X}}}^{-1}\mathbf{K}_{\widetilde{\mathbf{X}}\mathbf{X}}$. To achieve the fully Bayesian approach, the function values $\mathbf{F}$ and the inducing variables $\widetilde{\mathbf{F}}$ are considered random variables. Using VI (see Section 2.3), an Evidence Lower Bound for the marginal log-likelihood is formalized by

$$\begin{aligned}\sum_{n=1}^{N}\log p(\mathbf{Y}_n) &= \sum_{n=1}^{N}\log\iiint p(\mathbf{Y}_n,\mathbf{F}_n,\widetilde{\mathbf{F}}_n,\mathbf{X})\,\mathrm{d}\mathbf{F}_n\,\mathrm{d}\widetilde{\mathbf{F}}_n\,\mathrm{d}\mathbf{X}\\ &\geq \sum_{n=1}^{N}\log\iiint q(\mathbf{X})p(\mathbf{F}_n\mid\widetilde{\mathbf{F}}_n)q(\widetilde{\mathbf{F}}_n)\frac{p(\mathbf{Y}_n\mid\mathbf{F}_n)p(\widetilde{\mathbf{F}}_n)p(\mathbf{X})}{q(\mathbf{X})q(\widetilde{\mathbf{F}}_n)}\,\mathrm{d}\mathbf{F}_n\,\mathrm{d}\widetilde{\mathbf{F}}_n\,\mathrm{d}\mathbf{X}\\ &\stackrel{\text{def}}{=} \mathcal{L}_{\text{ELBO}}(q(\mathbf{X}),q(\widetilde{\mathbf{F}}),\widetilde{\boldsymbol{\theta}}),\end{aligned}$$

where the variational distribution is factorized accordingly

$$q(\mathbf{X},\widetilde{\mathbf{F}},\mathbf{F}) = q(\mathbf{X})\prod_{n=1}^{N}\mathcal{N}(\mathbf{F}_n\mid\boldsymbol{\mu},\boldsymbol{\Sigma})q(\widetilde{\mathbf{F}}_n),$$

based on a mean field assumption. For clarity, all hyperparameters and variational parameters are represented by $\widetilde{\boldsymbol{\theta}}$. Few reformulation steps give the ELBO the following form

$$\begin{aligned}\mathcal{L}_{\text{ELBO}}(q(\mathbf{X}),q(\widetilde{\mathbf{F}}),\widetilde{\boldsymbol{\theta}}) = \sum_{n=1}^{N}\Bigg(&-\mathrm{KL}(q(\mathbf{X})\parallel p(\mathbf{X})) - \mathrm{KL}\Big(q(\widetilde{\mathbf{F}}_n)\,\Big\|\,p(\widetilde{\mathbf{F}}_n)\Big)\\ &\big\langle\log\mathcal{N}(\mathbf{Y}_n\mid\mathbf{F}_n,\lambda_{\mathbf{Y}}^{-1}\mathbf{I})\big\rangle_{q(\mathbf{X})q(\widetilde{\mathbf{F}}_n)} -\\ &\frac{\lambda_{\mathbf{Y}}}{2}\mathrm{Tr}\left(\langle\mathbf{K}_{\mathbf{X}\mathbf{X}}\rangle_{q(\mathbf{X})} - \mathbf{K}_{\widetilde{\mathbf{X}}\widetilde{\mathbf{X}}}^{-1}\left\langle\mathbf{K}_{\widetilde{\mathbf{X}}\mathbf{X}}\mathbf{K}_{\mathbf{X}\widetilde{\mathbf{X}}}\right\rangle_{q(\mathbf{X})}\right)\Bigg),\end{aligned}$$

where the operator $\langle\cdot\rangle_{q(\cdot)}$ defines the expected value w.r.t. the probability distribution $q(\cdot)$ [50]. The hyperparameters and variational parameters are jointly optimized using nonlinear optimization techniques [50]. As in Section 2.5, the trace's quantity represents the total variance between the true function values $\mathbf{F}$ and the inducing variables $\widetilde{\mathbf{F}}$. Thus, the maximization of the Evidence Lower Bound simultaneously minimizes the variance between the true functions and the inducing variables [50].

In summary, the extension to a variational method provides a Bayesian learning formalism that approximates the fully marginalized GP-LVMs [50]. If, on the one hand, the hyperparameters and variational parameters are optimized, the result is an Empirical

Bayesian approach. Keeping them fixed, on the other hand, results in a full Bayesian formalism [6]. Furthermore, approximating the true posterior distribution of the latent variables $\mathbf{X}$ with given data $\mathbf{Y}$ leads to an Automatic Relevance Detection (ARD) [50]. An ARD automatically determines the latent space's dimensionality by giving low probabilities to dimensions not relevant to the data [4]. Moreover, the Occam's Razor effect (see Section 2.3) in fully Bayesian learning mitigates the risk of overfitting [6]. This mitigation is beneficial when the latent space's dimensionality is higher relative to the dimensionality of the observation space. The reduced computational complexity $\mathcal{O}(TD^2)$ increases linearly with the size of the observations $\mathbf{Y}$ and quadratically with the size of the inducing pairs. Finally, the concept of taking induced variables as sufficient statistics and achieving conditional independence is particularly emphasized. In the further process of the work, this assumption is essential.

3. Dynamic Mode Decomposition

The analysis and understanding of dynamics of highly complex systems have been a central focus in research for decades [25, 26]. Dynamical systems provide mathematical frameworks to describe the evolution and behavior of quantities in a system over time [84]. Traditionally, highly complex dynamical systems are modeled analytically based on partial or ordinary differential equations [25]. However, due to the deluge of data and Machine Learning techniques, data-driven approaches increasingly gain attention [26]. These approaches learn to understand, and eventually control and predict, the underlying dynamics of an unknown complex system from data [25, 26]. The following sections present a modern perspective of data-driven dynamical systems and form the basis for the rest of the work.

The growing availability of measured data from complex systems enables the use of data-driven frameworks. The data, however, originates from highly nonlinear dynamical behaviors. In Section 3.1, *Koopman Theory* is introduced that considers a linear evolution of measurement functions of the data instead of the nonlinear evolution of the data points themselves [22–24]. In simple terms, this concept forms a latent variable model, where the seen data are mapped into a latent space, where they evolve linearly.

Section 3.2 introduces a data-driven regression framework that performs global linearization directly on the given observations. The framework employs spectral decomposition, also known as eigenvalue decomposition, to decompose the dynamics into spatio-temporal patterns [28, 32, 33]. Based on these spatio-temporal patterns, the dynamics are analyzed, and predictions are made. This framework formalizes an approximation of the concept discussed in Section 3.1, considering linear measurement functions [25, 26].

However, the assumption of inferring linear dynamics directly from the given data is restrictive since these data may arise from highly nonlinear behaviors. Section 3.3, therefore, deals with techniques that generalize the procedure discussed in Section 3.2. These techniques achieve generalization utilizing feature mappings or kernelized functions [25, 29–31].

Data-driven techniques, however, suffer from inaccuracies and uncertainties in the given data [51]. The majority of the concepts in Sections 3.1 to 3.4 are based on deterministic techniques. Thus, representing uncertainties in the given data is a challenge. Section 3.4 therefore adopts a probabilistic perspective to incorporate the advantage of a Bayesian formalism into the data-driven formulation.

3.1. Koopman Theory

As a result of the vast amount of available data, data-driven modeling of dynamic systems is becoming increasingly important in modern times. A variety of techniques such as Machine Learning and Optimization are applied to understand the complexity of the given data and analyze the behavior over time. Due to the discrete nature of the given data and the digital aspects of modern-days technology, the consideration of discrete dynamical systems seems appealing [25, 26]. In this context, the objective is to analyze a discrete dynamical system given by

$$\mathbf{y}_{t+1} = f(\mathbf{y}_t),$$

where f describes an unknown mapping between the given observations collected in $\mathbf{Y} = [\mathbf{y}_0, \cdots, \mathbf{y}_T]$.

In *Koopman Spectral Analysis*, instead of considering the nonlinear evolution of a state $\mathbf{y}_t \in \mathbb{R}^n$, an alternative perspective is taken in the form of the evolution of measurement functions $h : \mathbb{R}^n \to \mathbb{R}$ [22–26]. This alternative representation is visualized in Figure 3.1. In this approach, the infinite number of possible measurement functions h form an infinite-dimensional function space $\mathcal{H}$ known as *Hilbert space*. The measurements h evolve linearly on this infinite-dimensional space, defined by an infinite-dimensional linear operator $\mathcal{K} : \mathcal{H} \to \mathcal{H}$, called the *Koopman operator*. Thus, the nonlinear evolution of the states $\mathbf{Y}$ in $\mathbf{R}^n$ are expressible by

$$\mathcal{K}h(\mathbf{x}_t) = h(f(\mathbf{x}_t)),$$

a linear evolution in the Hilbert space $\mathcal{H}$. While the linear behavior is appealing, the infinite dimensionality of the Hilbert space still poses a problem.

For this reason, specific key measurement functions are sought to provide a basis for the Hilbert space. The linearity associated with the Koopman operator $\mathcal{K}$ provides Spectral

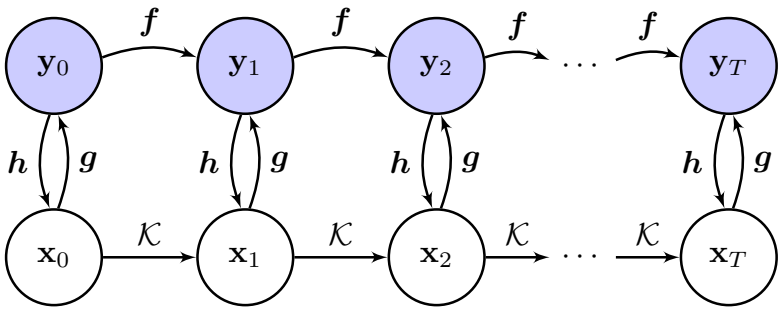


Figure 3.1.: This figure sketches an illustration of the perspective taken in Koopman Theory. Observations of the current nonlinear dynamical system $\mathbf{y}_0$, $\mathbf{y}_1$, $\cdots$, $\mathbf{y}_T$ are given. A measurement function h maps these finite-dimensional observations into an infinite-dimensional function space. In this Hilbert space, these infinite-dimensional measurements $\mathbf{x}_0 = h(\mathbf{y}_0)$, $\mathbf{x}_1 = h(\mathbf{y}_1)$, $\cdots$, $\mathbf{x}_T = h(\mathbf{y}_T)$ evolve linearly. Thus, Koopman Theory circumvents the analysis of nonlinear finite-dimensional dynamical systems by analyzing linear infinite-dimensional systems [25].

Analysis, and the spectral decomposition results in

$$\mathcal{K}\phi_k(\mathbf{y}) = \lambda_k \phi_k(\mathbf{y}),$$

where λ_k and ϕ_k correspond to the kth eigenvalue and eigenfunction, respectively. The eigenfunctions ϕ_k are such central measurements spanning an inherent measurement coordinate system and forming a basis of the Hilbert space $\mathcal{H}$. Consequently, any vector $\mathbf{h}$ of possible measurement functions is expressable by

$$\mathbf{h}(\mathbf{y}) = \begin{bmatrix} h_1(\mathbf{y}) \\ \vdots \\ h_p(\mathbf{y}) \end{bmatrix} = \sum_{k=1}^{\infty} \phi_k(\mathbf{y})\mathbf{v}_k,$$

where the vector $\mathbf{v}_k$ represents the kth *koopman mode*. These modes correspond to the superposition values of the eigenfunctions to obtain the measurement vector. The decomposition from the Spectral Analysis provides a linear representation of the change

of the measurement function in the Hilbert space by

$$\begin{aligned}\mathcal{K}\mathbf{h}(\mathbf{y}_t) &= \mathcal{K}\sum_{k=1}^{\infty}\phi_k(\mathbf{y}_t)\mathbf{v}_k \\ &= \sum_{k=1}^{\infty}\lambda_k\phi_k(\mathbf{y}_t)\mathbf{v}_k \\ &= \sum_{k=1}^{\infty}\lambda_k^t\phi_k(\mathbf{y}_0)\mathbf{v}_k.\end{aligned}$$

The system behaves linearly in this space. Thus, the dynamics of the system are decomposed into a spatio-temporal coherent structures [24, 26]. The eigenvalues represent the temporal progress of the measurement functions, while the eigenfunctions in combination with the Koopman modes describe the coherent spatial characteristics. Temporal changes in the coherent spatial characteristics are computed by multiplying them with the Koopman eigenvalues. In this way, the behavior of the nonlinear dynamical system is fully characterized by spectral decomposition. However, it is also infinite-dimensional. As a result, infinitely many degrees of freedom are necessary to describe the space of all possible measurement functions $\mathbf{h}$ of the state [25, 26].

The handling of an infinite-dimensional measurement space is challenging in practice [24–26]. For this reason, the *invariant Koopman subspace* has been proposed. A finite number of measurement functions $h_1,\ h_2,\ \cdots,\ h_p$ spans the base of the subspace $\widetilde{\mathcal{H}} \subset \mathcal{H}$. In this way, arbitrary measurement function within the space $\widetilde{h} \in \widetilde{\mathcal{H}}$ are expressed by

$$\widetilde{h} = \alpha_1 h_1 + \alpha_2 h_2 + \cdots + \alpha_p h_p,$$

the superposition of the basis functions. The invariant subspace restricts the dynamical behavior of measurement functions to evolve in the subspace $\widetilde{\mathcal{H}}$. Hence, the application of the Koopman operator results in measurement functions

$$\widetilde{h}' = \mathcal{K}h = \beta_1 h_1 + \beta_2 h_2 + \cdots + \beta_p h_p,$$

which are themselves in subspace $\widetilde{\mathcal{H}}$. The finite number of p measurement functions generate a finite-dimensional Koopman operator matrix $\mathbf{K} \in \mathbb{R}^{p\times p}$. It acts on a vector space $\mathbb{R}^p$ whose coordinates are given by the values of the measurement functions $h_i(\mathbf{y})$. Therefore an invariant Koopman subspace induces a finite-dimensional linear system [25, 26]. Any finite set of eigenfunctions establishes a basis for an invariant subspace.

Therefore, a key task in Koopman Spectral Analysis is the search for or an appropriate approximation of the eigenfunctions.

In summary, the Koopman representation circumvents the nonlinear, finite-dimensional dynamics of the dynamical system by considering linear, infinite-dimensional dynamics [22, 24–26]. The advantage of this approach is the resulting linear differential equation system, which is solvable with Spectral Theory [25]. Thereby, it is possible to decompose the dynamics into spatio-temporal patterns [24]. Invariant subspaces are considered to induce linear, finite-dimensional systems and deal with Koopman Theory in practice [26]. However, the correct choice of the bases of these subspaces is a major challenge in Koopman Analysis.

3.2. Dynamic Mode Decomposition

From the perspective of Koopman Theory, a straightforward restriction to the invariant subspace is the use of direct linear measurement functions $\mathbf{x} = \mathbf{h}(\mathbf{y}) = \mathbf{y}$. In this way, we work directly with the given observations $\mathbf{x}_0 = \mathbf{y}_0, \cdots, \mathbf{x}_T = \mathbf{y}_T \in \mathbb{R}^N$, as shown in Figure 3.2. The linear measurements span an invariant subspace, thereby inducing a finite-dimensional Koopman operator $\mathbf{K} \in \mathbb{R}^{N \times N}$ acting on a vector space $\mathbb{R}^N$. The simple assumption of linear measurements and direct use of the observations leads to the data-driven regression framework known as *Dynamic Mode Decomposition (DMD)* [32]. DMD has been introduced independently of Koopman Theory in fluid mechanics and has attracted significant attention in numerous research areas [25,26,28,32–34]. Its increasing success results from its simple formulation in terms of a linear regression problem and its close connection to Koopman Theory. DMD is an equation-free, data-driven approach capable of spatio-temporal decompositions of complex systems without requiring explicit knowledge of the governing dynamics [25, 26].

In the DMD community, the given observations are considered snapshots of an unknown underlying dynamical system. The evolution of these snapshots describes the temporal evolution of the underlying dynamical system. DMD aims at providing the best possible representation of the temporal evolution through a linear dynamical system. First, the

collected observations are arranged into two snapshot matrices

$$\mathbf{X}^0 = \begin{bmatrix} | & | & & | \\ \mathbf{x}_1 & \mathbf{x}_2 & \dots & \mathbf{x}_{T-1} \\ | & | & & | \end{bmatrix}, \qquad \mathbf{X}^1 = \begin{bmatrix} | & | & & | \\ \mathbf{x}_2 & \mathbf{x}_3 & \dots & \mathbf{x}_{T} \\ | & | & & | \end{bmatrix}.$$

Assuming uniform sampling in time, a minimization problem can be formulated as

$$\begin{aligned} \mathbf{A} &= \arg\min_{\mathbf{K}} \left|\left|\mathbf{X}^1 - \mathbf{K}\mathbf{X}^0\right|\right|_F \\ &= \mathbf{X}^1 \underbrace{\mathbf{X}^{0^T}\left(\mathbf{X}^0\mathbf{X}^{0^T}\right)^{-1}}_{\mathbf{X}^{0^+}}, \end{aligned}$$

where $||\cdot||_F$ is the Frobenius norm and $\mathbf{X}^{0^+}$ denotes the pseudo-inverse of the first snapshot matrix. The resulting linear operator $\mathbf{A}$ is closely related to the Koopman operator [25, 26, 32]. Spatio-temporal coherent structures are possible to determine from the linear operator $\mathbf{A}$ using spectral decomposition. Given the assumption of linear dynamics of the data, these structures describe the underlying dynamics of the data entirely. The eigenvectors represent coherent spatial modes that oscillate over time. The frequency and/or the growth or decay rate of the oscillation is described by the eigenvalues [25, 26]. The eigenvectors and eigenvalues resulting from the spectral decomposition of the linear operator $\mathbf{A}$ are therefore an approximation to the Koopman modes and the Koopman eigenvalues [32]. However, high-dimensional data, e.g., from fluid mechanics, lead to high-dimensional snapshot matrices $\mathbf{X}^0, \mathbf{X}^1 \in \mathbb{R}^{N\times T}$. The calculation of the pseudo-inverse and the spectral decomposition of these matrices with a large number of rows becomes intractable. However, many high-dimensional complex systems are based on low-dimensional linear dynamics [28]. Therefore, instead of considering the whole system, DMD focuses primarily on the dominant eigenvalues and eigenvectors of the matrix.

For this reason, DMD applies a dimensionality reduction to the first snapshot matrix $\mathbf{X}^0 \propto \widetilde{\mathbf{U}}\widetilde{\Sigma}^{-1}\widetilde{\mathbf{V}}^*$. Thus, the linear operator of the original space is represented as

$$\begin{aligned} \mathbf{A} &= \mathbf{X}^1\mathbf{X}^{0^+} \\ &\propto \mathbf{X}^1\widetilde{\mathbf{V}}\widetilde{\mathbf{\Sigma}}^{-1}\widetilde{\mathbf{U}}^*, \end{aligned}$$

where $\widetilde{\mathbf{U}} \in \mathbb{R}^{N\times M}$, $\widetilde{\mathbf{\Sigma}} \in \mathbb{R}^{N\times M}$ and $\widetilde{\mathbf{V}} \in \mathbb{R}^{T\times M}$ result from the Singular Value Decomposition (SVD). By appropriate choice of the dimension of the resulting latent space, a

lower dimension results, i.e., $M \leq N$. However, the application of Spectral Analysis on the high-dimensional linear operator of interest $\mathbf{A}$ remains a challenge. For this reason, the dynamics of the low-dimensional space induced by SVD are considered. The spectral decomposition takes place on the linear operator projected into the low-dimensional space given by

$$\begin{aligned}\widetilde{\mathbf{A}} &= \widetilde{\mathbf{U}}^{*}\mathbf{A}\widetilde{\mathbf{U}} \\ &= \widetilde{\mathbf{U}}^{*}\mathbf{X}^{1}\widetilde{\mathbf{V}}\widetilde{\mathbf{\Sigma}}^{-1}.\end{aligned}$$

Note that the reduced linear operator matrix $\widetilde{\mathbf{A}}$ has the same nonzero eigenvalues as the full matrix $\mathbf{A}$, which is a key property [26].

The application of dimensionality reduction in DMD shows parallels to Latent Variable Models (see Sections 2.2 and 2.6). While LVM frameworks, such as PCA, only decompose the spatially correlated structure in the given data and ignore temporal information, DMD also considers temporal information due to the spectral decomposition [25, 26]. The mapping to reconstruct the full state $\mathbf{x}$ from the lower dimensional state is done by

$$\mathbf{x} = \widetilde{\mathbf{U}}\widetilde{\mathbf{x}}.$$

The spectral decomposition of the linear operator results in

$$\widetilde{\mathbf{A}}\mathbf{W} = \mathbf{W}\mathbf{\Lambda},$$

where the entries of the diagonal matrix $\mathbf{\Lambda}$ correspond to the eigenvalues, and the column vectors $\mathbf{W}$ correspond to the eigenvectors. The eigenvalues from the low-dimensional space correspond to the eigenvalues of the original linear operator and represent the DMD eigenvalues [28]. They describe the time behavior of the linear dynamics behind the given data. On the other hand, the eigenvectors describe the modes and therefore the coherent spatial structures of the low-dimensional space. One way to obtain the DMD modes of the original linear operator $\mathbf{A}$ is to straightforwardly apply the left singular matrix

$$\widetilde{\mathbf{\Phi}} = \widetilde{\mathbf{U}}\mathbf{W},$$

where $\widetilde{\mathbf{\Phi}}$ corresponds to the *projected modes* [32, 33]. However, it is not guaranteed that these modes correspond to exact eigenvectors of the original linear operator $\mathbf{A}$ and thus to true DMD modes. Instead, one can reconstruct the eigenvectors exactly by

$$\mathbf{\Phi} = \mathbf{X}^{1}\widetilde{\mathbf{V}}\widetilde{\Sigma}^{-1}\widetilde{\mathbf{U}}^{*}\mathbf{W},$$

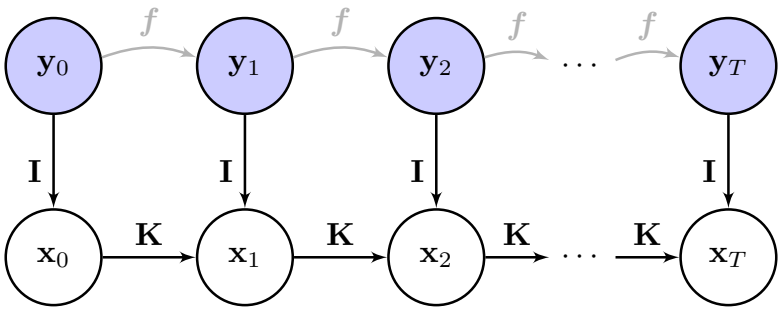


Figure 3.2.: In this figure, the Dynamic Mode Decomposition is represented from the Koopman theory point of view. The observations $\mathbf{y}_0,\ \mathbf{y}_1,\ \cdots,\ \mathbf{y}_T$ is the data of the current unknown dynamical system. The invariant subspace is spanned by the use of linear measurement functions $\mathbf{x} = \mathbf{h}(\mathbf{y}) = \mathbf{y}$. The coordinates of the measurement functions on which the Koopman operator acts are the given measurement data itself. Thus, the Dynamic Mode Decomposition operates directly on the given data under the assumption of a linear dynamic behavior and represents a straightforward global linearization [25].

resulting in true DMD modes of the original data. It has been shown that these projected eigenvectors in high-dimensional space are eigenvectors of the high-dimensional matrix $\mathbf{A}$ [28]. Therefore, they represent true DMD modes of the original high-dimensional system and represent the coherent spatial structures. The resulting spatio-temporal coherent structures depicted by $\mathbf{\Phi}$ and $\mathbf{\Lambda}$ provide the means to represent the system state as a data-driven spectral decomposition

$$\mathbf{x}_t = \sum_{m=1}^{M} \boldsymbol{\psi}_m \lambda_m^{t-1} b_m = \mathbf{\Psi}\mathbf{\Lambda}^{t-1}\mathbf{b},$$

where $\boldsymbol{\psi}_m$ and b_m refer to the mth DMD mode in $\mathbf{\Phi}$ and a mode amplitude, respectively. The vector $\mathbf{b}$ corresponding to all mode amplitudes is generally computed by a

$$\mathbf{b} = \mathbf{\Psi}^{+}\mathbf{x}_0,$$

where $\mathbf{\Psi}^{+}$ represents the pseudo-inverse of the DMD modes. The decomposition enables the reconstruction of the given observations, the prediction of future outcomes, and also the control of the dynamical systems [25–27].

The Dynamic Mode Decomposition framework provides a simple and effective data-driven regression framework. It allows inferring spatio-temporal structures from given data only, without any knowledge about governing equations or properties of the underlying system [25, 26]. Many modern systems exhibit high-dimensional, nonlinear dynamic behavior and thus require nonlinear methods for modeling. However, they are based on a low-dimensional, linear dynamical behavior describing the spatio-temporal change of the high-dimensional data [26, 28]. The use of DMD provides an important feature to analyze and understand these dynamical systems, attracting much attention in various research fields.

3.3. Dynamic Mode Decomposition on Nonlinear Observables

Given a sufficient amount of data, the DMD, on the one hand, provides accurate characteristics of the underlying dynamical system, even for some nonlinear systems [25, 26]. On the other hand, the assumption of linear measurement functions, and thus working directly with the given data $\mathbf{Y}$, is restrictive. In many cases, the data given are not rich enough to properly characterize the underlying system dynamics [29]. In Machine Learning, feature mapping is usually applied to transform given data into a higher dimensional space where Linear Algebra techniques are applicable [4, 6]. From the Koopman perspective, measurement functions $h(\mathbf{y})$ are indeed closely related to feature mappings. The measurement functions can be seen as a mapping from the physical space in which the dynamical system evolves nonlinearly to a linear feature space. The use of linear measurement functions in DMD is therefore very restrictive, and the extension to more complex feature mappings seems reasonable. A more extensive set of measurement functions, such as polynomials or radial basis functions, results in an expansion of the invariant subspace and a better approximation of the Koopman operator. The approximation of the operator gives a better representation of the nonlinear characteristics of the underlying dynamical system [29]

The *extended Dynamic Mode Decomposition (extended DMD)* provides a regression framework that incorporates a broader set of measurement functions and thus generalizes the classical DMD [25, 26, 29]. A set of M measurement functions $h : \mathbb{R}^N \to \mathbb{R}$ is selected to achieve a finite-dimensional approximation of the Koopman operator $\mathcal{K}$. These measurement functions span an invariant subspace $\widetilde{\mathcal{H}}_M \subset \mathcal{H}$ in Hilbert Space (see Section 3.1). The coordinates of the measurement functions in the induced M-dimensional vector space

are defined by

$$\mathbf{h}(\mathbf{y}) = \begin{bmatrix} h_1(\mathbf{y}) \\ h_2(\mathbf{y}) \\ \vdots \\ h_M(\mathbf{y}) \end{bmatrix},$$

where the measurement functions take the form of polynomials, radial basis functions, etc. [29]. The resulting snapshot matrices, given by

$$\mathbf{X}^0 = \begin{bmatrix} | & | & & | \\ \mathbf{h}(\mathbf{y}_0) & \mathbf{h}(\mathbf{y}_1) & \dots & \mathbf{h}(\mathbf{y}_{T-1}) \\ | & | & & | \end{bmatrix}, \quad \mathbf{X}^1 = \begin{bmatrix} | & | & & | \\ \mathbf{h}(\mathbf{y}_1) & \mathbf{h}(\mathbf{y}_2) & \dots & \mathbf{h}(\mathbf{y}_T) \\ | & | & & | \end{bmatrix},$$

further emphasize the extension of the observations. Like classical DMD, extended DMD aims at minimizing the residual error defined by

$$\begin{aligned} \mathbf{A} &= \arg\min_{\mathbf{K}} \frac{1}{2} \sum_{t=1}^{T} \left|\left| \mathbf{h}(\mathbf{y}_t)^T - \mathbf{K}\mathbf{h}(\mathbf{y}_{t-1})^T \right|\right| \\ &= \mathbf{X}^1 \mathbf{X}^{0^T} \left(\mathbf{X}^0 \mathbf{X}^{0^T} \right)^{-1} \\ &= \mathbf{G}_1 \mathbf{G}_0^{-1}, \end{aligned}$$

where $\mathbf{G}_0, \mathbf{G}_1 \in \mathbb{R}^{M \times M}$. The resulting approximation of the Koopman operator is closely related to DMD under the assumption of simple linear measurement functions. The extended DMD represents a finite-dimensional approximation of the Koopman operator $\mathcal{K}$, which describes the linear mapping in the invariant subspace $\widetilde{\mathcal{H}}_M$. In this way, the properties of the underlying nonlinear dynamical system are approximated, which is described by the triple of Koopman eigenvalues, eigenvectors, and modes [29]. The extended DMD offers two main advantages. The matrices $\mathbf{G}_0$ and $\mathbf{G}_1$ are embedded in $\mathbb{R}^{M \times M}$, and consequently, the computational cost is determined by the number of features [25, 29]. It is particularly useful when a large amount of data $\mathbf{Y}$ and thus snapshots of a dynamical system are given. Moreover, the computational cost of the pseudo-inverse of size $M \times M$ is also determined by the resulting dimension of the feature

space. For dynamical systems with a large number of given data, extended DMD thus provides, on the one hand, the possibility to reduce the computational effort and, on the other hand, to extend the spanned invariant subspace by a suitable selection of the feature mapping. Due to this selection, the approximation of the Koopman operator improves and hence the determination of the spatio-temporal characteristics of the dynamical system of interest [29].

However, in dynamical systems with large state dimensions, the approximation of the Koopman operator is challenging [25,26,30,31]. The use of DMD and especially extended DMD, which increases the state dimension by selecting the feature mapping, becomes impractical for such systems. Both methods suffer accordingly from the curse of dimensionality [25]. Like Machine Learning, *kernel Dynamic Mode Decomposition (kernel DMD)* considers a dual representation that achieves a decomposition of data from dynamical systems with high-dimensional state spaces [30,31]. The kernel DMD framework circumvents the dimensionality problem by defining a kernel function that implicitly computes inner products in the high-dimensional space of given observations. Assuming that the dimensions of the data are larger than the number of given data, i.e., $T-1 \ll M$, Singular Value Decomposition can be used to decompose the initial snapshot into

$$\mathbf{X}^0 = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^*,$$

where $\boldsymbol{\Sigma}, \mathbf{V} \in \mathbb{R}^{(T-1)\times(T-1)}$ and $\mathbf{U} \in \mathbb{R}^{M\times T-1}$. Given that $T-1 \ll M$ the range of $\mathbf{X}^0$ contains the range of $\mathbf{A}$, and the eigenvalue problem takes place in a latent space. The projection of the eigenvalues into the original space is given by $\mathbf{w} = \mathbf{U}\widetilde{\mathbf{w}}$. The dual formulation is based on the reformulation of the eigenvalue problem, resulting in

$$\begin{aligned}
0 &= (\mathbf{A} - \lambda\mathbf{I})\,\mathbf{w} \\
&= \left(\mathbf{G}_1\mathbf{G}_0^{-1} - \lambda\mathbf{I}\right)\mathbf{U}\widetilde{\mathbf{w}} \\
&= \left(\mathbf{X}^1\mathbf{V}\boldsymbol{\Sigma}^{-1} - \lambda\mathbf{I}\mathbf{U}\right)\widetilde{\mathbf{w}} \\
&= \left(\mathbf{X}^0\mathbf{X}^{0^T}\mathbf{X}^1\mathbf{V}\boldsymbol{\Sigma}^{-1} - \lambda\mathbf{I}\mathbf{U}\right)\widetilde{\mathbf{w}} \\
&= \left(\mathbf{U}\boldsymbol{\Sigma}^{-1}\mathbf{V}^*\mathbf{X}^{0^T}\mathbf{X}^1\mathbf{V}\boldsymbol{\Sigma}^{-1} - \lambda\mathbf{I}\mathbf{U}\right)\widetilde{\mathbf{w}} \\
&= \mathbf{U}\left(\underbrace{\left(\boldsymbol{\Sigma}^{-1}\mathbf{V}^*\right)\mathbf{X}^{0^T}\mathbf{X}^1\left(\mathbf{V}\boldsymbol{\Sigma}^{-1}\right)}_{=\widetilde{\mathbf{A}}} - \lambda\mathbf{I}\right)\widetilde{\mathbf{w}},
\end{aligned}$$

where the approximation of the Koopman operator is now calculable by $\widetilde{\mathbf{A}}$ [25, 30]. In this way, the eigenvalue problem has been transformed into a low-dimensional latent space. The computational cost is now defined by the number of the given data instead of the feature space dimension [30]. The necessary quantities for the computation are $\widetilde{\mathbf{G}}_0 = \mathbf{X}^{0^T}\mathbf{X}^0 \in \mathbb{R}^{T-1,T-1}$ and $\widetilde{\mathbf{G}}_1 = \mathbf{X}^{0^T}\mathbf{X}^1 \in \mathbb{R}^{T-1,T-1}$, with $\mathbf{V}$ and $\mathbf{\Sigma}$ obtainable from a spectral decomposition of $\widetilde{\mathbf{G}}_1$ given by

$$\widetilde{\mathbf{G}}_1 = \mathbf{V}\mathbf{\Sigma}^2\mathbf{V}^*.$$

The main advantage of the method is the possibility to apply the kernel trick [4, 6]. A closer look at the two outer product matrices, given by

$$\widetilde{\mathbf{G}}_0 = \begin{bmatrix} \mathbf{h}(\mathbf{y}_0)^T\mathbf{h}(\mathbf{y}_0) & \cdots & \mathbf{h}(\mathbf{y}_0)^T\mathbf{h}(\mathbf{y}_{T-1}) \\ \vdots & \ddots & \vdots \\ \mathbf{h}(\mathbf{y}_{T-1})^T\mathbf{h}(\mathbf{y}_0) & \cdots & \mathbf{h}(\mathbf{y}_{T-1})^T\mathbf{h}(\mathbf{y}_{T-1}) \end{bmatrix},$$

$$\widetilde{\mathbf{G}}_1 = \begin{bmatrix} \mathbf{h}(\mathbf{y}_0)^T\mathbf{h}(\mathbf{y}_1) & \cdots & \mathbf{h}(\mathbf{y}_1)^T\mathbf{h}(\mathbf{y}_T) \\ \vdots & \ddots & \vdots \\ \mathbf{h}(\mathbf{y}_{T-1})^T\mathbf{h}(\mathbf{y}_1) & \cdots & \mathbf{h}(\mathbf{y}_{T-1})^T\mathbf{h}(\mathbf{y}_T) \end{bmatrix},$$

reveals that each element in both matrices corresponds to an inner product in the feature space. Consequently, using the kernel trick, each inner product is replaceable by any kernel feature. In this way, the kernel DMD framework presents itself as a nonparametric method, where the complexity of the model increases with the number of given data [30, 31].

In summary, the search for suitable approximations of the Koopman operator and the associated Koopman triples, which represent the spatio-temporal characteristics of given dynamical systems, is a challenge in modern research [25–27, 85, 86]. DMD provided a simple and efficient regression framework which was generalized by extended DMD. An appropriate choice of feature mappings extends the invariant subspace and improves the approximation [29]. However, these methods suffer from the curse of dimensionality, which is an enormous disadvantage in fields such as fluid mechanics. Therefore, kernel DMD provides a framework formulated in dual space and spans the invariant subspace through kernel functions [30, 31]. The complexity of this function increases with the number of given data and not with the dimensionality. The search for suitable kernel functions and feature mappings is an essential part of the current research. Especially modern computational capabilities combined with Deep Learning provide a variety of techniques

and possibilities to develop satisfying expansions and thus improve the approximations of the Koopman operator [85, 86].

3.4. A Probabailistic Interpretation of DMD

In the previous sections, the discussed DMD frameworks are purely data-driven regression techniques. They entirely rely on experimental and numerical data. However, there is a risk that the data is affected by sensor noise and other stochastic disturbances. These disturbances potentially lead to errors in the approximations of the Koopman operator and thus result in incorrectly characterized spatio-temporal structures of the data. In the literature, several approaches have been proposed to address noisy data [51, 87–89]. However, the majority of these proposed frameworks are based on deterministic models. A more appropriate way to incorporate uncertainty and stochastic processes is to adopt a probabilistic view (See Chapter 2). A probabilistic model offers the advantage of treating the data statistically and hence explicitly accounting for noisy observations [51].

Let the following snapshot matrices, defined by

$$\mathbf{X}^0 = \begin{bmatrix} | & | & & | \\ \mathbf{h}(\mathbf{y}_0) & \mathbf{h}(\mathbf{y}_1) & \dots & \mathbf{h}(\mathbf{y}_{T-1}) \\ | & | & & | \end{bmatrix}, \quad \mathbf{X}^1 = \begin{bmatrix} | & | & & | \\ \mathbf{h}(\mathbf{y}_1) & \mathbf{h}(\mathbf{y}_2) & \dots & \mathbf{h}(\mathbf{y}_T) \\ | & | & & | \end{bmatrix},$$

be considered where, similar to Section 3.3, M selected measurement functions $h : \mathbb{R}^N \to \mathbb{R}$ induce an M-dimensional vector space through the coordinates, given by

$$\mathbf{h}(\mathbf{y}) = \begin{bmatrix} h_1(\mathbf{y}) \\ h_2(\mathbf{y}) \\ \vdots \\ h_M(\mathbf{y}) \end{bmatrix}.$$

Any snapshot, e.g., $\mathbf{h}(\mathbf{y}_t)$ and $\mathbf{h}(\mathbf{y}_t)$, defined by

$$\begin{aligned}\mathbf{h}(\mathbf{y}_t) &= \sum_{k=1}^{\infty} \phi_k(\mathbf{y}_t)\mathbf{v}_k, \\ \mathbf{h}(\mathbf{y}_{t+1}) &= \mathcal{K} \sum_{k=1}^{\infty} \phi_k(\mathbf{y}_t)\mathbf{v}_k \\ &= \sum_{k=1}^{\infty} \lambda_k \phi_k(\mathbf{y}_t)\mathbf{v}_k,\end{aligned}$$

is representable by the Koopman triple $(\lambda_k, \phi_k(\mathbf{y}_t), \mathbf{v}_k)$ written [24]. The eigenmodes $\mathbf{v}_k$ and eigenfunctions ϕ_k denote the coherent spatial structures. The eigenvalues λ_k characterize the evolution of a snapshot over time. A concatenation of the two snapshot matrices

$$\widetilde{\mathbf{X}} = \begin{bmatrix}\mathbf{X}^0 \\ \mathbf{X}^1\end{bmatrix} = \begin{bmatrix} | & | & & | \\ \widetilde{\mathbf{x}}_0 & \widetilde{\mathbf{x}}_1 & \dots & \widetilde{\mathbf{x}}_{T-1} \\ | & | & & | \end{bmatrix},$$

implies an extension of the vector space in $\mathbb{R}^{2M}$. Let $K < \infty$ define the dimensionality of a finite-dimensional invariant subspace in which all snapshots of $\widetilde{\mathbf{X}}$ are contained. In this way, the following matrix product takes the form

$$\widetilde{\mathbf{X}} = \underbrace{\begin{bmatrix}\mathbf{V} \\ \mathbf{V}\Lambda\end{bmatrix}}_{\mathbf{B}} \Phi(\mathbf{Y}), \tag{3.1}$$

where $\Lambda = \mathrm{diag}(\lambda_1, \cdots, \lambda_K)$ represents a diagonal matrix. $\mathbf{V} = [\mathbf{v}_1, \cdots, \mathbf{v}_K]$ is a matrix connecting all eigenmodes, and $\Phi(\mathbf{Y})$, given by,

$$\Phi(\mathbf{Y}) = \begin{bmatrix} \phi_1(\mathbf{y}_0) & \cdots & \phi_1(\mathbf{y}_{T-1}) \\ \vdots & & \vdots \\ \phi_K(\mathbf{y}_0) & \cdots & \phi_K(\mathbf{y}_{T-1}) \end{bmatrix} = \begin{bmatrix} | & | & & | \\ \boldsymbol{\phi}(\mathbf{y}_0) & \boldsymbol{\phi}(\mathbf{y}_1) & \dots & \boldsymbol{\phi}(\mathbf{y}_{T-1}) \\ | & | & & | \end{bmatrix},$$

consists of discrete-time snapshots of the eigenfunctions. Thus, the formulation of Equation (3.1) provides a way to represent the given snapshots of the extended snapshot matrix by the Koopman eigenvalues, eigenmodes, and eigenfunctions [51].

Assuming that the reformulated Equation (3.1) is affected by independently and identically distributed drawn additive noise $\boldsymbol{\epsilon}_t \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I})$, the deterministic framework becomes a probabilistic one. The assumption of i.i.d.-drawn noise for a single snapshot is expressible by

$$\widetilde{\mathbf{x}}_t = \mathbf{B}\boldsymbol{\phi}(\mathbf{y}_t) + \boldsymbol{\epsilon}_t.$$

In this way, a likelihood function is obtained

$$p(\widetilde{\mathbf{X}}) = \prod_{t=0}^{T-1} \mathcal{CN}\left(\widetilde{\mathbf{x}}_t \mid \mathbf{B}\boldsymbol{\phi}(\mathbf{y}_t), \sigma^2\mathbf{I}\right), \tag{3.2}$$

which forms the basis for the *Bayesian Dynamic Mode Decomposition (Bayesian DMD)* [51]. This formulation provides a natural view of observational noise and thus explicitly accounts for it. Since the Koopman eigenvalues and eigenfunctions can have complex values, a complex normal distribution $\mathcal{CN}(\cdot)$ is generally assumed. Based on the likelihood of Equation (3.2) and an appropriate choice of prior distributions, Bayesian DMD formalizes a Probability Density Model. This Probability Density model brings the advantages of the Bayesian formulation to DMD [51]. In order to infer the best possible values, Gibbs sampling is applied [51]. The Gibbs sampling results in satisfactory samples from the true probability density model. For noisy data, the maximum likelihood solution of the probabilistic model is equivalent to the solution from noise-aware deterministic methods [51]. In the case of noiseless data and hence where $\sigma^2 \to 0$, the MLE solution is equal to the optimal solution from classical DMD [51]. The probabilistic formulation results in satisfactory approximations of the Koopman eigenvalues, eigenfunctions, and eigenmodes, and thus achieves a good decomposition into coherent spatio-temporal characteristics.

Although this approach seems appealing, it also has its drawbacks similar to extended DMD. On the one hand, an appropriate set of measurement functions has to be selected. The performance of the approximation is related to the choice of this set of measurement functions. Like extended DMD, the probabilistic framework also suffers from the curse of dimensionality. On the other hand, the method's optimal solutions include complex eigenvalues that do not necessarily have a complex conjugate couple. In linear dynamical systems, however, complex eigenvalues always appear as a pair and thus are always

conjugate complex. This issue is why in this work, the Koopman operator is approximated and subsequently a spectral decomposition is performed. In this way, the resulting complex eigenvalues are guaranteed to appear as complex conjugate pairs.

4. Gaussian Process Dynamic Mode Decomposition

In the following sections, two central components of this thesis *Gaussian Process Dynamic Mode Decompositions (GP-DMDs)* and *Bayesian Gaussian Process Dynamic Mode Decomposition (Bayesian GP-DMD)* are presented. Koopman Theory, explained in chapter 3, aims to decompose arbitrary dynamical systems into spatio-temporal coherent structures [24]. Methods such as DMD, extended DMD and kernel DMD approximate these spatiotemporal characteristics based on selected measurement or kernel functions [28–31]. The underlying principles of these techniques show strong similarities to continuous Latent Variable Models, e.g., Principle Component Analysis and Factor Analysis [25, 26]. The contribution made in this chapter is the introduction of a probabilistic *dual prespective* to the DMD family. The advantages of probabilistic methods are combined with those of the kernelized methods. Similar to how Gaussian Process Latent Variable Models (see Section 2.6) formalize a dual approach to various continuous LVMs, a dual perspective to the DMD family is proposed.

Section 4.1 introduces *State-Space Models (SSMs)*, which are continuous LVMs where a Markov sequence is assumed in latent space [6]. These frameworks enable the modeling of a variety of time series data and belong to Time Series Modeling, also known as System Identification [6, 52–54]. They enjoy great popularity and have been extensively applied in many disciplines, from science and engineering to finance and economics and beyond [53, 54]. Special variants of this family are the *Gaussian Process State-Space Models (GP-SSMs)*. These models assume a Gaussian Process over the possible transition functions and/or observation functions [54–60]. They provide the essential framework needed for modeling the dual approach to the DMD family.

Based on the GP-SSM framework, the dual perspective on the DMD family is presented in Section 4.2. Due to this combination of GP-SSM and DMD, the introduced algorithm is referred to as Gaussian Process Dynamic Mode Decompositions (GP-DMDs). It aims at

estimating a stationary linear Markov sequence in the latent space, while a GP describes the relation to the given observations. On the one hand, it provides a probabilistic view of the Koopman Theory while naturally accounting for uncertainties and noise. On the other hand, the Gaussian Process formalization leads to a kernelized method and hence to a nonparametric Bayesian method [4, 6]. From the Koopman perspective, the approach does not rely on mapping from the observations to the invariant subspace. Instead, the GP describes the inverse mapping from the invariant subspace back to the observations. Eventually, a Maximum A Posteriori probability estimate is proposed to estimate suitable parameters for the Gaussian Process, the linear trajectories in the latent space, and the corresponding linear operator. The estimated linear operator can be used to describe the spatio-temporal properties in the latent space.

According to Koopman Theory, an invariant subspace, in which the system evolves linearly, can have a higher dimensionality than the original space from which the observation originates [24, 25]. Unfortunately, as discussed in Section 2.2, continuous LVMs suffer from the effect of overfitting [4, 6]. Section 4.3, therefore, extends the probabilistic generative model using a fully Bayesian formalism introducing Bayesian Gaussian Process Dynamic Mode Decomposition (Bayesian GP-DMD). The Bayesian GP-DMD formalization aims at mitigating the effect of overfitting, thus tackling a central drawback of Gaussian Process Dynamic Mode Decompositions (GP-DMDs). The formalization of an Evidence Lower Bound allows the application of VI or VB. Therefore, using Probabilistic Inference on the derived fully Bayesian model results in approximated posterior distributions over the linear operator and the linear trajectories in the latent space.

4.1. Gaussian Process State Space Models

The family of *State-Space Models (SSMs)* from Time Series Modeling is an established paradigm that provides model-based frameworks for analyzing and studying time series data [4, 6, 52–54, 56, 57]. They are an extension of continuous LVMs (see Section 2.2), assuming dynamic dependence in latent space. SSMs are closely related to the family of *Hidden Markov Models* that assume a discrete latent space [4, 6]. The *Markov property* is a central assumption in this context, where the latent variables form a *Markov sequence* or *Markov Chain* [4, 6, 53]. This property states that the prediction of a state x_{t+1} at time t solely depends on the current state x_t [2, 4, 6]. Therefore, it induces conditional independence which embodies a form of memorylessness. The entire past and thus the history up to x_0 is irrelevant for predicting the next state x_{t+1} if x_t is given [4, 6, 53].

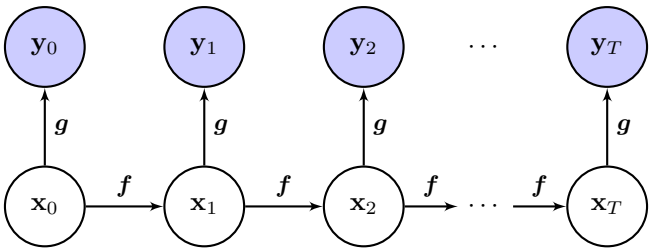


Figure 4.1.: The given graphical models shows a classical State-Space Models. SSMs are primarily used to derive a recursive estimate of the underlying Markov sequence. The blue shaded nodes represent the given time-series observations $\mathbf{y}_0, \cdots, \mathbf{y}_T$ from a stochastic process. The white nodes represent the unknown latent states of interest $\mathbf{x}_0, \cdots, \mathbf{x}_T$. The functions $f(\cdot)$ and $g(\cdot)$ are defined as transition and observation functions, respectively, and show the dependencies between the states through the edges. The figure illustrates the well-known Markov property from which the Markov sequence in the latent space is formed. This property indicates that each latent variable depends solely on the previous state and not on the entire history. [6]

The time sequence $\mathbf{y}_0, \cdots, \mathbf{y}_T \in \mathbb{R}^N$ denotes some given observations, and $\mathbf{x}_0, \cdots, \mathbf{x}_T \in \mathbb{R}^M$ are the corresponding latent states. The underlying generic model for SSMs formally corresponds to

$$\mathbf{x}_{t+1} = f\left(\mathbf{x}_t, \mathbf{u}_t, \boldsymbol{\epsilon}_t\right),$$

$$\mathbf{y}_t = g\left(\mathbf{x}_t, \mathbf{u}_t, \boldsymbol{\delta}_t\right),$$

where $f(\cdot)$ and $g(\cdot)$ are the predefined transition and observation models, respectively. The observation model is occasionally referred to as the emission model [54, 58–60]. To achieve a probabilistic representation, the parameters $\boldsymbol{\epsilon}$ and $\boldsymbol{\delta}$ correspond to some system and observation noise. Input values occurring during the process, e.g., representing control signals, are accounted for by $\mathbf{u}_0, \cdots, \mathbf{u}_T \in \mathbb{R}^N$. However, the focus of this work does not lie in controlling a system, but rather in learning an underlying stationary dynamical system solely from observations, which makes these variables irrelevant for this work. Therefore, they will be left out for convenience in the rest of this thesis.

Likewise, the generative Probabilistic Density Model of an SSM is formalized by

$$p(\mathbf{X}, \mathbf{Y}) = p(\mathbf{x}_0)p(\mathbf{y}_0 \mid \mathbf{x}_0) \prod_{t=1}^{T} p(\mathbf{x}_t \mid \mathbf{x}_{t-1})p(\mathbf{y}_t \mid \mathbf{x}_t), \tag{4.1}$$

where the latent states and observations are summarized in $\mathbf{X} \in \mathbb{R}^{M \times T}$ and $\mathbf{Y}^{N \times T}$, respectively. A graphical model of the Probability Density Model (see Equation (4.1)) is represented in Figure 4.1. Such graphical models provide a straightforward way to visualize the underlying structures of probability models [2, 4, 6, 69]. Hence, the Figure illustrates the dependencies and relationships of the generative model in Equation (4.1). The blue shaded nodes represent the given time series data $\mathbf{Y}$, while the white ones represent the latent states $\mathbf{X}$. Edges represent the dependencies induced by the observation model $g(\cdot)$ and the transition model $f(\cdot)$. A variety of techniques based on SSMs have been proposed [2, 4, 6, 52–54, 56, 90, 91]. On the one hand, to primarily infer estimates of the state sequence in the latent space. On the other hand, to also learn the observation and the transition model.

The family of the *Gaussian Process State-Space Models (GP-SSMs)* is a subset of the SSMs [54–60]. These models integrate the ideas of GPs (see Section 2.4) into the family of SSMs. Instead of defining a parameterized transition or observation model, these techniques utilize the concepts of GP and hence perform nonparametric modeling [54]. For this reason, two Gaussian Process priors are adopted

$$f(\mathbf{x}) \sim \mathcal{GP}(\mathbf{0}, \mathbf{k}_f(\mathbf{x}, \mathbf{x}')),$$
$$g(\mathbf{x}) \sim \mathcal{GP}(\mathbf{0}, \mathbf{k}_g(\mathbf{x}, \mathbf{x}')),$$

either over the transition model or the observation model, or even both. For example, the Gaussian Process Latent Variable Model discussed in Section 2.6 with a corresponding Markov sequence in latent space belongs to the family of *Gaussian Process State-Space Models (GP-SSMs)*. The resulting probability density model considering GP priors over the transition and observation models is

$$\begin{aligned} p(\mathbf{Y}, \mathbf{G}, \mathbf{X}, \mathbf{F}) = p(\mathbf{y}_0 \mid \mathbf{g}_0) p(\mathbf{f}_0 \mid \mathbf{x}_0) p(\mathbf{x}_0) p(\mathbf{G} \mid \mathbf{X}) \\ \prod_{t=1}^{T} p(\mathbf{x}_t \mid \mathbf{f}_t) p(\mathbf{f}_t \mid \mathbf{x}_{t-1,\cdots,0}) p(\mathbf{y}_t \mid \mathbf{g}_t), \end{aligned}$$

where $\mathbf{F} = [\mathbf{f}_0, \cdots, \mathbf{f}_T] \in \mathbb{R}^{M \times T}$ and $\mathbf{G} = [\mathbf{g}_0, \cdots, \mathbf{g}_T] \in \mathbb{R}^{N \times T}$ denote the function outputs. For clarity, the expression $\mathbf{x}_{t-1,\cdots,0}$ refers to the collection of $\mathbf{x}_{t-1}, \cdots, \mathbf{x}_0$. The outputs of the observation model $\mathbf{G}$ depend on the entire latent state sequence due to GP. In addition, the output of the transition model $\mathbf{f}_t$ at time t depends on the entire past $\mathbf{x}_{t-1,\cdots,0}$. Thus, using GP with SSMs generally violates the Markov property [54].

Nevertheless, much research has been done addressing various types of GP-SSMs. On one side, they have been focusing on the modeling of the observation models using GP.

Here, the formulation of the transition function and thus the latent space differs. Markov chains [55], linear Gaussian Processes [56, 57, 92], or time-dependent parametric models are considered [47–49]. The latter methods simplify the modeling of the latent space due to the dependence on time, however, with the consequence that stationary state transitions cannot be modeled [54]. On the other hand, the interest of many methods is the modeling of the transition function [54, 58–60, 90, 91, 93, 94]. Those methods focus on a GP prior over the transition function where the observation function is usually defined as a parametric model.

4.2. Gaussian Process Dynamic Mode Decomposition

The *Gaussian Process Dynamic Mode Decomposition (GP-DMD)* is a mathematical framework that adopts a dual perspective on Koopman Theory and the DMD family. It combines the GP-SSM framework and the DMD. On the one hand, it enables an intuitive way to incorporate noise and uncertainty due to the probabilistic formulation [4, 6]. On the other hand, it takes advantage of the kernelized formulation and the kernel trick, resulting in a nonparametric Bayesian method [4, 6]. GP-DMD aims to estimate a stationary linear dynamical system in a latent space represented by a Markov sequence. Instead of considering a mapping from the given observations into a latent space as in Koopman Theory, the inverse mapping from the latent space back to the observations is modeled. Therefore, a Gaussian Process prior

$$g(\mathbf{x}) \sim \mathcal{GP}(\mathbf{0}, \mathbf{k}(\mathbf{x}, \mathbf{x}')),$$

over the observation model of the system is assumed. The transition model is formulated as a stationary linear operator $f(\cdot) = \mathbf{A}$. The generative Probability Density Model is accordingly formalized as

$$p(\mathbf{Y}, \mathbf{G}, \mathbf{X}) = \underbrace{p(\mathbf{Y} \mid \mathbf{G}) p(\mathbf{G} \mid \mathbf{X})}_{p(\mathbf{Y}, \mathbf{G} \mid \mathbf{X})} p(\mathbf{x}_0) \prod_{t=1}^{T} p(\mathbf{x}_t \mid \mathbf{x}_{t-1}, \mathbf{A}), \tag{4.2}$$

where $\mathbf{G} = [\mathbf{g}_0, \cdots, \mathbf{g}_T] \in \mathbb{R}^{N \times T}$ denotes the outputs of the GP. The observations are collected in the matrix $\mathbf{Y} = [\mathbf{y}_0, \cdots, \mathbf{y}_T] \in \mathbb{R}^{N \times T}$. Accounting for some noise, the dependence between given observations and the outputs of the observation model is described by a Gaussian likelihood function $p(\mathbf{G}, \mathbf{Y})$. The matrix $\mathbf{A}$ corresponds to the linear operator in the latent space. This operator approximates the Koopman operator

and thus the spatio-temporal characteristics in the latent space [24]. Figure 4.2 visualizes this generative Probability Density Model. The blue shaded and the white nodes represent the given time series data $\mathbf{y}_0, \cdots, \mathbf{y}_T$ and all unknown latent variables, respectively. An auto-regressive structure resulting from the Gaussian process is denoted as a thick black line, implying that $\mathbf{g}_t$ depends on $\mathbf{g}_0, \cdots, \mathbf{g}_{t-1}$ and $\mathbf{x}_0, \cdots, \mathbf{x}_t$. The small black nodes correspond to the hyperparameters of the given Probability Density Model.

Learning from Demonstration

In order to infer estimates for the unknown random variables and compute optimal hyperparameters, the inference problem of the Probability Density Model is transformed into log-space and hence into an optimization problem. For this purpose, the observation model represented by GP is marginalized

$$\begin{aligned} p(\mathbf{Y} \mid \mathbf{X}) &= \int \prod_{n=1}^{N} p(\mathbf{Y}_n, \mathbf{G}_n \mid \mathbf{X}) \, \mathrm{d}\mathbf{G}_n \\ &= \prod_{n=1}^{N} \int p(\mathbf{Y}_n \mid \mathbf{G}_n) \mathcal{GP}(\mathbf{G}_n \mid \mathbf{0}, \mathbf{K}_{\mathbf{XX}}) \, \mathrm{d}\mathbf{G}_n \\ &= \prod_{n=1}^{N} \int \mathcal{N}(\mathbf{Y}_n \mid \mathbf{G}_n, \lambda_{\mathbf{y}}^{-1}\mathbf{I}) \mathcal{N}(\mathbf{G}_n \mid \mathbf{0}, \mathbf{K}_{\mathbf{XX}}) \, \mathrm{d}\mathbf{G}_n \\ &= \prod_{n=1}^{N} \mathcal{N}(\mathbf{Y}_n \mid \mathbf{0}, \mathbf{K}_{\mathbf{XX}} + \lambda_{\mathbf{y}}^{-1}\mathbf{I}), \end{aligned}$$

and leads to a direct dependence between the given observations $\mathbf{Y}$ and the latent states $\mathbf{X}$ [56, 57, 92]. Due to its Gaussian nature, this marginalization is done analytically in closed form [2, 4, 6]. N independent Gaussian processes represent the dependence between $\mathbf{Y}$ and $\mathbf{X}$. Each dimension n is considered independently, where $\mathbf{Y}_n$ corresponds to the nth row of the matrix $\mathbf{Y}$. The marginalization achieves an optimization considering the uncertainty and variability of the possible functions $g(\cdot)$ [6]. Hence, the probability

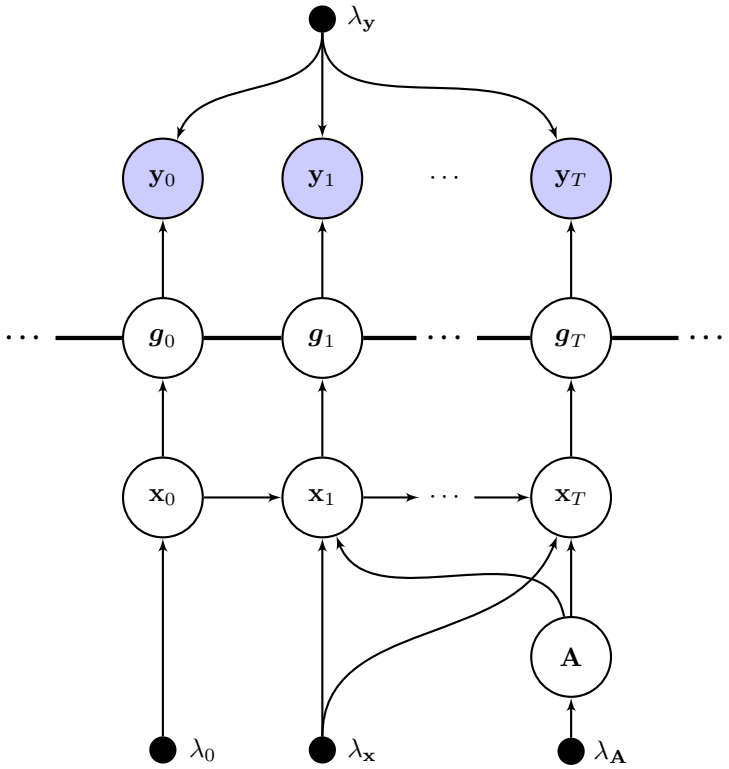


Figure 4.2.: The graphical model visualizes the Probability Density Model of the derived Gaussian Process Dynamic Mode Decomposition. The blue shaded nodes represent the given observations $\mathbf{y}_0, \cdots, \mathbf{y}_T$. White nodes correspond to the unknown latent variables. The sequence of states of interest in the latent space is given by $\mathbf{x}_0, \cdots, \mathbf{x}_T$. $\mathbf{g}_0, \cdots, \mathbf{g}_T$ describe the outputs of the Gaussian process function corresponding to the observation model. The linear operator is given by $\mathbf{A}$. The thick black line denotes an auto-regressive structure within the Gaussian process, implying that $\mathbf{g}_t$ depends on $\mathbf{g}_0, \cdots, \mathbf{g}_{t-1}$ and $\mathbf{x}_0, \cdots, \mathbf{x}_t$. The small black nodes correspond to the hyperparameters of the given Probability Density Model. In this case, they represent the precision values of the Gaussian distributions.

density model changes to

$$
\begin{aligned}
p(\mathbf{Y}, \mathbf{X}, \mathbf{A}) &= p(\mathbf{Y} \mid \mathbf{X}) p(\mathbf{X}, \mathbf{A}) p(\mathbf{A}) \\
&= \prod_{n=1}^{N} \mathcal{N}(\mathbf{Y}_n \mid \mathbf{0}, \mathbf{K}_{\mathbf{XX}} + \lambda_{\mathbf{y}}^{-1}\mathbf{I}) \prod_{m=1}^{M} \mathcal{N}(\mathbf{a}_m \mid 0, \lambda_{\mathbf{a}}^{-1}\mathbf{I}) \\
&\quad \mathcal{N}(\mathbf{x}_0 \mid 0, \lambda_{\mathbf{0}}^{-1}\mathbf{I}) \prod_{t=1}^{T} \mathcal{N}(\mathbf{x}_t \mid \mathbf{A}^T \mathbf{x}_{t-1}, \lambda_{\mathbf{x}}^{-1}\mathbf{I}),
\end{aligned} \tag{4.3}
$$

where $\lambda_{\mathbf{y}}, \lambda_{\mathbf{0}}, \lambda_{\mathbf{x}}, \lambda_{\mathbf{a}} \in \mathbb{R}^+$ correspond to the precision values. A prior distribution $p(\mathbf{A})$ over each column of $\mathbf{A}$ is introduced to incorporate prior knowledge of the linear operator into the system naturally. For conciseness, the parameters of interest are written as $\boldsymbol{\Theta} = \{\mathbf{X}, \mathbf{A}, \lambda_{\mathbf{y}}, \lambda_{\mathbf{0}}, \lambda_{\mathbf{x}}, \lambda_{\mathbf{a}}\}$. The Probability Density Model is then transformed into the log-space

$$
\begin{aligned}
\mathcal{L}(\boldsymbol{\Theta}, \boldsymbol{\theta}) &= \log p(\mathbf{Y}, \mathbf{X}, \mathbf{A} \mid \lambda_{\mathbf{y}}, \lambda_{\mathbf{0}}, \lambda_{\mathbf{x}}, \lambda_{\mathbf{a}}, \boldsymbol{\theta}) \\
&= \underbrace{\log p(\mathbf{Y} \mid \mathbf{X}, \lambda_{\mathbf{y}}, \boldsymbol{\theta})}_{=\mathcal{L}_1(\mathbf{Y}, \mathbf{X}, \lambda_{\mathbf{y}}, \boldsymbol{\theta})} + \underbrace{\log p(\mathbf{X} \mid \mathbf{A}, \lambda_{\mathbf{0}}, \lambda_{\mathbf{x}}, \boldsymbol{\theta})}_{=\mathcal{L}_2(\mathbf{X}, \mathbf{A}, \lambda_{\mathbf{0}}, \lambda_{\mathbf{x}}, \boldsymbol{\theta})} + \underbrace{\log p(\mathbf{A}, \lambda_{\mathbf{a}}, \boldsymbol{\theta})}_{=\mathcal{L}_3(\mathbf{A}, \lambda_{\mathbf{a}}, \boldsymbol{\theta})},
\end{aligned}
$$

where the resulting loss function $\mathcal{L}(\boldsymbol{\Theta}, \boldsymbol{\theta})$ breaks down into three separate parts. The first part on the r.h.s. $\mathcal{L}_1(\mathbf{Y}, \mathbf{X}, \lambda_{\mathbf{y}}, \boldsymbol{\theta})$ describes the observation model and thus depends on the observations seen $\mathbf{Y}$, the latent state $\mathbf{X}$, the corresponding precision value $\lambda_{\mathbf{y}}$, and the hyperparameters for the kernel matrix $\boldsymbol{\theta}$. It formalizes

$$
\begin{aligned}
\mathcal{L}_1(\mathbf{Y}, \mathbf{X}, \lambda_{\mathbf{y}}, \boldsymbol{\theta}) &= \log p(\mathbf{Y} \mid \mathbf{X}, \boldsymbol{\theta}) \\
&= \log \prod_{n=1}^{N} \mathcal{N}(\mathbf{Y}_n \mid \mathbf{0}, \mathbf{K}_{\mathbf{XX}}(\boldsymbol{\theta}) + \lambda_{\mathbf{y}}^{-1}\mathbf{I}) \\
&= c - \frac{N}{2} \log \left| \widetilde{\mathbf{K}}_{\mathbf{XX}}(\boldsymbol{\theta}) \right| - \frac{1}{2} \mathrm{Tr} \left(\widetilde{\mathbf{K}}_{\mathbf{XX}}(\boldsymbol{\theta})^{-1} \mathbf{Y}\mathbf{Y}^T \right),
\end{aligned}
$$

where all values that do not depend on the parameters of interest are represented by a constant value c. The sum of the kernel matrix and the inverse of the precision value form

$\widetilde{\mathbf{K}}_{\mathbf{XX}}(\boldsymbol{\theta}) = \mathbf{K}_{\mathbf{XX}}(\boldsymbol{\theta}) + \lambda_{\mathbf{y}}^{-1}\mathbf{I}$. Then the second part of the loss term resolves to

$$\begin{aligned}\mathcal{L}_2(\mathbf{X}, \mathbf{A}, \lambda_{\mathbf{0}}, \lambda_{\mathbf{x}}, \boldsymbol{\theta}) &= \log p(\mathbf{X} \mid \mathbf{A}, \boldsymbol{\theta}) \\ &= c - \frac{M}{2}\log\left|\lambda_{\mathbf{0}}^{-1}\right| - \frac{(T-1)M}{2}\log\left|\lambda_{\mathbf{x}}^{-1}\right| - \frac{\lambda_{\mathbf{0}}}{2}\mathbf{x}_0^T\mathbf{x}_0 \\ &\quad - \frac{\lambda_{\mathbf{x}}}{2}\mathrm{Tr}\left(\mathbf{X}^1\mathbf{X}^{1^T} - 2\mathbf{A}^T\mathbf{X}^0\mathbf{X}^{1^T} + \mathbf{X}^0\mathbf{X}^{0^T}\mathbf{A}\mathbf{A}^T\right),\end{aligned}$$

where again, c corresponds to all constant terms. $\mathcal{L}_2(\mathbf{X}, \mathbf{A}, \lambda_{\mathbf{0}}, \lambda_{\mathbf{x}}, \boldsymbol{\theta})$ represents the linear dynamical system in latent space and depends on latent states $\mathbf{X}$, the stationary linear operator $\mathbf{A}$, and the precision values $\lambda_{\mathbf{0}}, \lambda_{\mathbf{x}}$. Similar to Chapter 3, $\mathbf{X}^0 = [\mathbf{x}_0, \cdots, \mathbf{x}_{T-1}] \in \mathbb{R}^{M\times T-1}$ and $\mathbf{X}^1 = [\mathbf{x}_1, \cdots, \mathbf{x}_T] \in \mathbb{R}^{M\times T-1}$ denote the snapshot matrices. The last part of the loss term corresponds to the

$$\begin{aligned}\mathcal{L}_3(\mathbf{A}, \lambda_{\mathbf{a}}, \boldsymbol{\theta}) &= \log p(\mathbf{A}, \boldsymbol{\theta}) \\ &= \log \prod_{m=1}^{M} \mathcal{N}(\mathbf{a}_m \mid 0, \lambda_{\mathbf{a}}^{-1}\mathbf{I}) \\ &= c - \frac{M^2}{2}\log\left|\lambda_{\mathbf{a}}^{-1}\right| - \frac{\lambda_{\mathbf{a}}}{2}\mathrm{Tr}\left(\mathbf{A}\mathbf{A}^T\right),\end{aligned}$$

induced by the prior distribution $p(\mathbf{A})$. Thus, it depends solely on $\mathbf{A}$ and the precision value $\lambda_{\mathbf{a}}$. Eventually, the logarithm of the Probability Density Model transforms into

$$\begin{aligned}\mathcal{L}(\boldsymbol{\Theta}, \boldsymbol{\theta}) &= \mathcal{L}_1(\mathbf{Y}, \mathbf{X}, \lambda_{\mathbf{y}}, \boldsymbol{\theta}) + \mathcal{L}_2(\mathbf{X}, \mathbf{A}, \lambda_{\mathbf{0}}, \lambda_{\mathbf{x}}, \boldsymbol{\theta}) + \mathcal{L}_3(\mathbf{A}, \lambda_{\mathbf{a}}, \boldsymbol{\theta}) \\ &= c - \frac{1}{2}\Bigg[- M\log|\lambda_{\mathbf{0}}| - (T-1)M\log|\lambda_{\mathbf{x}}| - M^2\log|\lambda_{\mathbf{a}}| \\ &\quad + N\log\left|\widetilde{\mathbf{K}}_{\mathbf{XX}}(\boldsymbol{\theta})\right| + \mathrm{Tr}\left(\widetilde{\mathbf{K}}_{\mathbf{XX}}(\boldsymbol{\theta})^{-1}\mathbf{Y}\mathbf{Y}^T\right) \\ &\quad + \lambda_{\mathbf{x}}\mathrm{Tr}\left(\mathbf{X}^1\mathbf{X}^{1^T} - 2\mathbf{A}^T\mathbf{X}^0\mathbf{X}^{1^T} + \mathbf{X}^0\mathbf{X}^{0^T}\mathbf{A}\mathbf{A}^T\right) \\ &\quad + \lambda_{\mathbf{0}}\mathbf{x}_0^T\mathbf{x}_0 + \lambda_{\mathbf{a}}\mathrm{Tr}\left(\mathbf{A}\mathbf{A}^T\right)\Bigg],\end{aligned} \tag{4.4}$$

which corresponds to an optimization problem. The constant c collects all terms not depending on $\boldsymbol{\Theta}$ and $\boldsymbol{\theta}$. Maximizing $\mathcal{L}(\boldsymbol{\Theta}, \boldsymbol{\theta})$ is equivalent to maximizing the Probability Density Model, as discussed in Equation (4.3) [2, 4, 6]. However, it is important to keep

in mind that the GP-DMD framework only achieves point estimates of $\mathcal{L}(\boldsymbol{\Theta}, \boldsymbol{\theta})$. The loss function $\mathcal{L}(\boldsymbol{\Theta}, \boldsymbol{\theta})$ enables the use of vanilla gradient descent methods to

$$\boldsymbol{\Theta}^*, \boldsymbol{\theta}^* = \underset{\boldsymbol{\Theta}, \boldsymbol{\theta}}{\arg\max}\, \mathcal{L}(\boldsymbol{\Theta}, \boldsymbol{\theta}),$$

to achieve optimal estimates for $\boldsymbol{\Theta}^*$ and $\boldsymbol{\theta}^*$ [4, 6]. This optimization problem can be grouped into two parts based on gradient calculations. On the one hand, in gradient calculations, which are analytically and in closed-form calculable. On the other hand, in gradient calculations, which need numerical differentiation frameworks since no closed-form generally exists. Similar to the linear operator $\mathbf{A}$, prior distributions in the form of Gamma distributions $\mathrm{Gam}(\cdot)$ (see Appendix A) for the precision values $\lambda_{\mathbf{y}}, \lambda_{\mathbf{0}}, \lambda_{\mathbf{x}}$ and $\lambda_{\mathbf{a}}$ are assumed. With the use of Gamma distributions, additional regularization parameters are introduced. For example, a Gamma distribution prevents division by 0 and thus stabilizes the optimization. In the following, the closed-form solution for the optimal linear operator $\mathbf{A}$ is derived first, followed by closed-form solutions for the precision values $\lambda_{\mathbf{0}}, \lambda_{\mathbf{x}}$ and $\lambda_{\mathbf{a}}$. In the end, the final loss function is given depending on the parameters for which there is no closed-form solution. These parameters include the latent state sequence $\mathbf{X}$, the observation precision value $\lambda_{\mathbf{y}}$ and the kernel parameters $\boldsymbol{\theta}$ provided.

Optimal Linear Operator $\mathbf{A}^*$

In order to estimate the optimal linear operator $\mathbf{A}^*$ corresponding to the stationary linear dynamics of the trajectories $\mathbf{X}$ in the latent state, the gradient w.r.t. $\mathbf{A}$ is taken as

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial \mathbf{A}} &= -\lambda_{\mathbf{x}} \mathbf{X}^0 \mathbf{X}^{1^T} + \lambda_{\mathbf{x}} \mathbf{X}^0 \mathbf{X}^{0^T} \mathbf{A} + \lambda_{\mathbf{a}} \mathbf{A} \mathbf{A}^T \\ &\overset{!}{=} 0.\end{aligned}$$

The gradient is then used as a basis for the estimation of the optimal linear operator $\mathbf{A}^*$. By setting the gradient equal to zero and reformulating the given equation

$$\mathbf{A}^* = \left(\mathbf{X}^0 \mathbf{X}^{0^T} + \frac{\lambda_{\mathbf{a}}}{\lambda_{\mathbf{x}}} \right)^{-1} \mathbf{X}^0 \mathbf{X}^{1^T}, \tag{4.5}$$

the closed-form solution is obtained, which corresponds to a *ridge regression* with the ridge parameter $\lambda_{\mathbf{a}}/\lambda_{\mathbf{x}}$ [4, 6]. The given solution in Equation (4.5) shows great similarities to the classical DMD framework (see Section 3.2). However, the complementary prior distribution $p(\mathbf{A})$ leads to a MAP estimation and thus to an additional regularization parameter expressed by the ridge parameter $\lambda_{\mathbf{a}}/\lambda_{\mathbf{x}}$.

Optimal Precision Value $\lambda_{\mathbf{0}}^*$

For optimizing the precision value $\lambda_{\mathbf{0}}$ associated with the noise of the initial latent state $\mathbf{x}_0$, an additional $\mathrm{Gam}\left(\lambda_{\mathbf{0}} \mid \alpha_{\mathbf{0}}, \beta_{\mathbf{0}}\right)$ is applied with the hyperparameters $\alpha_{\mathbf{0}} > 0$ and $\beta_{\mathbf{0}} > 0$. The gradient of the loss function is taken w.r.t. $\lambda_{\mathbf{0}}$ to obtain the optimal precision value, resulting in

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial \lambda_{\mathbf{0}}} &= -\frac{M + 2\alpha_{\mathbf{0}} - 2}{2\lambda_{\mathbf{0}}} + \frac{1}{2}\mathbf{x}_0^T\mathbf{x}_0 + \beta_{\mathbf{0}} \\ &\stackrel{!}{=} 0.\end{aligned}$$

Setting the gradient equal to zero leads to the optimal solution

$$\lambda_{\mathbf{0}}^* = \frac{M + 2\alpha_{\mathbf{0}} - 2}{\mathbf{x}_0^T\mathbf{x}_0 + 2\beta_{\mathbf{0}}}. \tag{4.6}$$

The optimal solution, which depends on the number of samples M and the outer product of the initial state $\mathbf{x}_0$. In addition, due to the prior distribution, it also depends on the predefined hyperparameters $\alpha_{\mathbf{0}}$ and $\beta_{\mathbf{0}}$.

Optimal Precision Value $\lambda_{\mathbf{x}}^*$

To optimize the precision value $\lambda_{\mathbf{x}}$, which describes the noise of the sequence in the latent space $\mathbf{X}$, a Gamma distribution $\mathrm{Gam}\left(\lambda_{\mathbf{x}} \mid \alpha_{\mathbf{x}}, \beta_{\mathbf{x}}\right)$ with the hyperparameters $\alpha_{\mathbf{x}} > 0$ and $\beta_{\mathbf{x}} > 0$ is additionally used. Taking the gradient of the loss function w.r.t. $\lambda_{\mathbf{x}}$

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial \lambda_{\mathbf{x}}} = &-\frac{(T-1)M + 2\alpha_{\mathbf{x}} - 2}{2\lambda_{\mathbf{x}}} + \beta_{\mathbf{x}} \\ &+ \frac{1}{2}\mathrm{Tr}\left(\mathbf{X}^1\mathbf{X}^{1^T} - 2\mathbf{A}^T\mathbf{X}^0\mathbf{X}^{1^T} + \mathbf{X}^0\mathbf{X}^{0^T}\mathbf{A}\mathbf{A}^T\right) \\ &\stackrel{!}{=} 0,\end{aligned}$$

provides the optimal precision value

$$\lambda_{\mathbf{x}}^* = \frac{(T-1)M + 2\alpha_{\mathbf{x}} - 2}{\mathrm{Tr}\left(\mathbf{X}^1\mathbf{X}^{1^T} - 2\mathbf{A}^T\mathbf{X}^0\mathbf{X}^{1^T} + \mathbf{X}^0\mathbf{X}^{0^T}\mathbf{A}\mathbf{A}^T\right) + 2\beta_{\mathbf{x}}}. \tag{4.7}$$

The parameter $\lambda_{\mathbf{x}}^*$ depends on the number of samples M, the linear operator $\mathbf{A}$ and the two snapshot matrices $\mathbf{X}^0$ and $\mathbf{X}^1$. Moreover, due to the prior distribution, $\lambda_{\mathbf{x}}^*$ also depends on the given hyperparameters $\alpha_{\mathbf{x}}$ and $\beta_{\mathbf{x}}$.

Optimal Precision Value $\lambda_{\mathbf{a}}^*$

The optimal precision value $\lambda_{\mathbf{a}}^*$, which represents the variability and uncertainty in the estimation of $\mathbf{A}$, is also optimized using a Gamma distribution $\mathrm{Gam}\left(\lambda_{\mathbf{a}} \mid \alpha_{\mathbf{a}}, \beta_{\mathbf{a}}\right)$ with the hyperparameters $\alpha_{\mathbf{a}} > 0$ and $\beta_{\mathbf{a}} > 0$. Like $\lambda_{\mathbf{0}}^*$ and $\lambda_{\mathbf{x}}^*$, the gradient of the loss function is taken w.r.t. $\lambda_{\mathbf{a}}$

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial \lambda_{\mathbf{a}}} &= -\frac{M^2 + 2\alpha_{\mathbf{a}} - 2}{2\lambda_{\mathbf{a}}} + \frac{1}{2}\mathrm{Tr}\left(\mathbf{A}\mathbf{A}^T\right) + \beta_{\mathbf{a}} \\ &\stackrel{!}{=} 0,\end{aligned}$$

to obtain the optimal precision value

$$\lambda_{\mathbf{a}}^* = \frac{M^2 + 2\alpha_{\mathbf{a}} - 2}{\mathrm{Tr}\left(\mathbf{A}\mathbf{A}^T\right) + 2\beta_{\mathbf{a}}}, \tag{4.8}$$

depending on the number of samples N and the linear operator $\mathbf{A}$. It also depends on the given hyperparameters $\alpha_{\mathbf{a}}$ and $\beta_{\mathbf{a}}$.

Optimal Latent State X^*, Precision Value $\lambda_{\mathbf{y}}^*$ And Kernel Hyperparameter $\boldsymbol{\theta}^*$

For the optimal values of the sequence in the latent space $\mathbf{X}^*$, the precision value $\lambda_{\mathbf{y}}^*$ and the kernel hyperparameters $\boldsymbol{\theta}^*$, no closed-form solutions generally exists. Therefore, the gradients are determined using numerical computational methods. The loss function utilized, subject to all parameters of interest, is given by

$$\begin{aligned}\mathcal{L}(\mathbf{X}, \lambda_{\mathbf{y}}, \boldsymbol{\theta}) = -\frac{1}{2}\Bigg[& N \log\left|\widetilde{\mathbf{K}}_{\mathbf{XX}}(\boldsymbol{\theta})\right| + \mathrm{Tr}\left(\widetilde{\mathbf{K}}_{\mathbf{XX}}(\boldsymbol{\theta})^{-1}\mathbf{Y}\mathbf{Y}^T\right) \\ & + \lambda_{\mathbf{x}}\mathrm{Tr}\left(\mathbf{X}^1\mathbf{X}^{1^T} - 2\mathbf{A}^T\mathbf{X}^0\mathbf{X}^{1^T} + \mathbf{X}^0\mathbf{X}^{0^T}\mathbf{A}\mathbf{A}^T\right) \\ & + \lambda_{\mathbf{0}}\mathbf{x}_0^T\mathbf{x}_0 - (\alpha_{\mathbf{y}} - 1)\log\lambda_{\mathbf{y}} + \beta_{\mathbf{y}}\lambda_{\mathbf{y}}\Bigg] + c,\end{aligned} \tag{4.9}$$

with all irrelevant terms collected in c. Similar to $\lambda_{\mathbf{0}}^*, \lambda_{\mathbf{x}}^*$ and $\lambda_{\mathbf{a}}^*$ the loss function is extended by an additional Gamma distribution $\mathrm{Gam}\left(\lambda_{\mathbf{y}} \mid \alpha_{\mathbf{y}}, \beta_{\mathbf{y}}\right)$ over the parameter $\lambda_{\mathbf{y}}$. On the one hand, the loss function aims to provide the optimal parameters for the given observations. On the other hand, nonlinear dynamical behavior in the latent space is penalized. The precision value $\lambda_{\mathbf{x}}^*$ can be seen as a regularization that adjusts the priority of the linear dynamical behavior.

Reproducing Demonstration

In order to reconstruct demonstrations in the observation space based on the optimized parameters $\boldsymbol{\Theta}^*$ and $\boldsymbol{\theta}$, the trajectories in the latent space $\mathbf{X}^*$ are considered first. The estimation of the optimal linear operator $\mathbf{A}^*$ is used to achieve a decomposition into the spatio-temporal characteristics. Thus, the different trajectories in the latent space can be analyzed and reproduced based on these characteristics. The decomposition is similar to the classical DMD (see Section 3.2) using spectral decomposition and results in corresponding eigenvectors, eigenvalues, and amplification factors. These characteristics enable the modification and prediction of the resulting time-series data in the latent space (see Section 3.2).

Eventually, to construct trajectories in the observation space based on the estimates in the latent space, the predictive distribution is taken into account, as discussed in Section 2.4. This distribution is given by

$$p(\mathbf{G}^* \mid \mathbf{X}^*, \mathbf{X}, \mathbf{Y}) = \mathcal{N}\Big(\mathbf{K}_{\mathbf{X}^*\mathbf{X}}\widetilde{\mathbf{K}}_{\mathbf{X},\mathbf{X}}^{-1}\mathbf{Y}^T, \mathbf{K}_{\mathbf{X}^*\mathbf{X}^*} - \mathbf{K}_{\mathbf{X}^*\mathbf{X}}\widetilde{\mathbf{K}}_{\mathbf{X},\mathbf{X}}^{-1}\mathbf{K}_{\mathbf{X}\mathbf{X}^*}\Big),$$

omitting the kernel hyperparameters $\boldsymbol{\theta}$. Thus, assuming some given observations $\mathbf{Y}$ at certain time points and the corresponding states $\mathbf{X}$ in the latent space, trajectories $\mathbf{G}^*$ in the observation space are predictable for the given inputs states $\mathbf{X}^*$. The calculation of the natural parameters of the predictive distribution is done in Equations (2.6) and (2.7) in Section 2.4.

4.3. Bayesian Gaussian Process Dynamic Mode Decomposition

According to Koopman Theory, an invariant subspace, in which the system evolves linearly, can have a higher dimensionality than the original space from which the observation originates [24, 25]. Unfortunately, continuous LVMs and hence the derived Gaussian Process Dynamic Mode Decomposition suffers from the effect of overfitting [4, 6]. Therefore, *Bayesian Gaussian Process Dynamic Mode Decomposition (Bayesian GP-DMD)* is introduced to extend the Probabilistic Density Model using a fully Bayesian formalism. On the one hand, the Bayesian GP-DMD framework aims at mitigating the effect of overfitting, thus tackling one central drawback of Gaussian Process Dynamic Mode Decompositions (GP-DMDs). On the other hand, the formalization enables the application of VI or VB, leading

to approximations of the posterior distribution over the linear operator $\mathbf{A}$, the trajectories in the latent space $\mathbf{X}$, etc.

One central problem of the Probability Density Model used for GP-DMD in Equation (4.3) is the auto-regressive structure induced by the Gaussian Process. Under marginalization, this structure leads to a dependence of the observations $\mathbf{Y}$ on the latent states $\mathbf{X}$ and poses a problem in deriving a fully Bayesian approach [50, 54, 59]. Therefore, based on the Bayesian formalism of GP-LVM discussed in Section 2.7, the literature has proposed convenient ways to use sparse GP techniques to achieve an ELBO and hence approximate a fully Bayesian formalism [43, 46, 50, 54, 59, 77]. On the one hand, the use of inducing pairs $(\mathbf{U}, \mathbf{Z})$ resolves the auto-regressive structure implied by the GP observation model. On the other hand, it reduces the computational complexity, as discussed in Section Section 2.7. The matrices $\mathbf{U} = [\mathbf{u}, \cdots, \mathbf{u}] \in \mathbb{R}^{N \times D}$ and $\mathbf{Z} = [\mathbf{z}, \cdots, \mathbf{z}] \in \mathbb{R}^{M \times D}$ are the inducing variables and the corresponding inducing inputs, respectively. The Probability Density Model introduced in Equation (4.2) leads to

$$\begin{aligned} p(\mathbf{Y}, \mathbf{\Theta}) &= p(\mathbf{Y}, \mathbf{X}, \mathbf{G}, \mathbf{U}, \mathbf{A}, \lambda_{\mathbf{x}}, \lambda_{\mathbf{0}}, \lambda_{\mathbf{y}}, \lambda_{\mathbf{a}}) \\ &= p(\mathbf{Y}, \mathbf{G}, \mathbf{U}, \lambda_{\mathbf{y}} \mid \mathbf{X}) p(\mathbf{X}, \mathbf{A}, \lambda_{\mathbf{x}}, \lambda_{\mathbf{0}}, \lambda_{\mathbf{a}}) \\ &= p(\mathbf{Y}, \mathbf{G}, \mathbf{U}, \lambda_{\mathbf{y}} \mid \mathbf{X}) p(\mathbf{X}, \lambda_{\mathbf{x}}, \lambda_{\mathbf{0}} \mid \mathbf{A}) p(\mathbf{A}, \lambda_{\mathbf{a}}), \end{aligned} \tag{4.10}$$

where the inducing pairs are considered. As discussed in Section 2.7, the inducing inputs form variational parameters. These variational parameters and the hyperparameters, e.g., from the kernel function, are omitted below for conciseness. The Probability Density Model in Equation (4.10) also considers the precision values $\lambda_{\mathbf{x}}, \lambda_{\mathbf{0}}, \lambda_{\mathbf{y}}$ and $\lambda_{\mathbf{a}}$ as random variables. The first term of the loss function corresponds to the observational model and is therefore extended by the inducing inputs. It forms

$$\begin{aligned} p(\mathbf{Y}, \mathbf{G}, \mathbf{U}, \lambda_{\mathbf{y}} \mid \mathbf{X}) &= \prod_{t=0}^{T} p(\mathbf{y}_t, \mathbf{g}_t, \mathbf{U} \mid \mathbf{X}, \lambda_{\mathbf{y}}) p(\lambda_{\mathbf{y}}) \\ &= \prod_{t=0}^{T} p(\mathbf{y}_t \mid \mathbf{g}_t, \lambda_{\mathbf{y}}) p(\mathbf{g}_t \mid \mathbf{U}, \mathbf{X}) p(\mathbf{U}) p(\lambda_{\mathbf{y}}) \\ &= \prod_{t=0}^{T} \mathcal{N}(\mathbf{y}_t \mid \mathbf{g}_t, \lambda_{\mathbf{y}}^{-1} \mathbf{I}) \mathcal{N}(\mathbf{g}_t \mid \mathbf{C}_t \mathbf{U}, \mathbf{D}_t) \\ &\quad \prod_{n=1}^{N} \mathcal{N}(\mathbf{u}_n \mid 0, \mathbf{K}_{\mathbf{ZZ}}) \mathrm{Gam}(\lambda_{\mathbf{y}} \mid \alpha_{\mathbf{y}}, \beta_{\mathbf{y}}), \end{aligned}$$

where two operatores are given by $\mathbf{C}_t = \mathbf{K}_{t\mathbf{Z}}\mathbf{K}_{\mathbf{ZZ}}^{-1}$ and $\mathbf{D}_t = \mathbf{K}_{tt} - \mathbf{K}_{t\mathbf{Z}}\mathbf{K}_{\mathbf{ZZ}}^{-1}\mathbf{K}_{\mathbf{Z}t}$. The abbreviations for the kernels are $\mathbf{K}_{\mathbf{ZZ}} = \mathbf{K}(\mathbf{Z}, \mathbf{Z})$, $\mathbf{K}_{tt} = \mathbf{K}(\mathbf{x}_t, \mathbf{x}_t)$, $\mathbf{K}_{t\mathbf{Z}} = \mathbf{K}(\mathbf{x}_t, \mathbf{Z})$, and $\mathbf{K}_{\mathbf{Z}t} = \mathbf{K}_{t\mathbf{Z}}^T$. Because of the inducing inputs and the associated conditional independence, an output $\mathbf{g}_t$ of the observational model depends on the inducing inputs $\mathbf{U}$ and solely on the latent state $\mathbf{x}_t$ at time t. The operator $\mathbf{D}_t$ represent the covariance matrix describing the posterior variance between the output $\mathbf{g}_t$ of the observation model and the inducing variables $\mathbf{U}$. The remaining terms on the r.h.s of the probability density model correspond, on the one hand, to the linear dynamical system in the latent space,

$$\begin{aligned} p(\mathbf{X}, \lambda_{\mathbf{x}}, \lambda_{\mathbf{0}} \mid \mathbf{A}) &= p(\mathbf{X} \mid \mathbf{A}, \lambda_{\mathbf{x}}, \lambda_{\mathbf{0}}) p(\lambda_{\mathbf{x}}, \lambda_{\mathbf{0}}) \\ &= p(\mathbf{x}_0 \mid \lambda_{\mathbf{0}}) \prod_{t=1}^{T} p(\mathbf{x}_t \mid \mathbf{x}_{t-1}, \mathbf{A}, \lambda_{\mathbf{x}}) p(\lambda_{\mathbf{x}}, \lambda_{\mathbf{0}}) \\ &= \mathcal{N}(\mathbf{x}_0 \mid 0, \lambda_{\mathbf{0}}^{-1}\mathbf{I}) \prod_{t=1}^{T} \mathcal{N}(\mathbf{x}_t \mid \mathbf{A}^T\mathbf{x}_{t-1}, \lambda_{\mathbf{x}}^{-1}\mathbf{I}) \\ &\quad \mathrm{Gam}(\lambda_{\mathbf{0}} \mid \alpha_{\mathbf{0}}, \beta_{\mathbf{0}}) \mathrm{Gam}(\lambda_{\mathbf{x}} \mid \alpha_{\mathbf{x}}, \beta_{\mathbf{x}}), \end{aligned}$$

and, on the other hand, to the prior distribution over the linear operator

$$\begin{aligned} p(\mathbf{A}, \lambda_{\mathbf{a}}) &= \prod_{m=1}^{M} p(\mathbf{a}_m \mid \lambda_{\mathbf{a}}) p(\lambda_{\mathbf{a}}) \\ &= \prod_{m=1}^{M} \mathcal{N}(\mathbf{a}_m \mid \mathbf{0}, \lambda_{\mathbf{a}}^{-1}\mathbf{I}) \mathrm{Gam}(\lambda_{\mathbf{a}} \mid \alpha_{\mathbf{a}}, \beta_{\mathbf{a}}). \end{aligned}$$

A graphical model visualizing the initial Probability Density Model of the derived Bayesian Gaussian Process Dynamic Mode Decomposition without the inducing inputs is shown in Figure 4.3. The blue shaded nodes corresponds to the given observations $\mathbf{Y}$ while the white nodes represent the unknown latent variables of interest $\mathbf{\Theta}$. The thick black line denotes the problematic auto-regressive structure, implying that $\mathbf{g}_t$ depends on $\mathbf{g}_0, \cdots, \mathbf{g}_{t-1}$ and $\mathbf{x}_0, \cdots, \mathbf{x}_t$. However, the use of the inducing pairs resolves this auto-regressive dependency, as discussed previously. In the following, the learning procedure for the latent variables $\mathbf{\Theta}$ and the variational parameters and hyperparameter collected in $\boldsymbol{\theta}$ is considered.

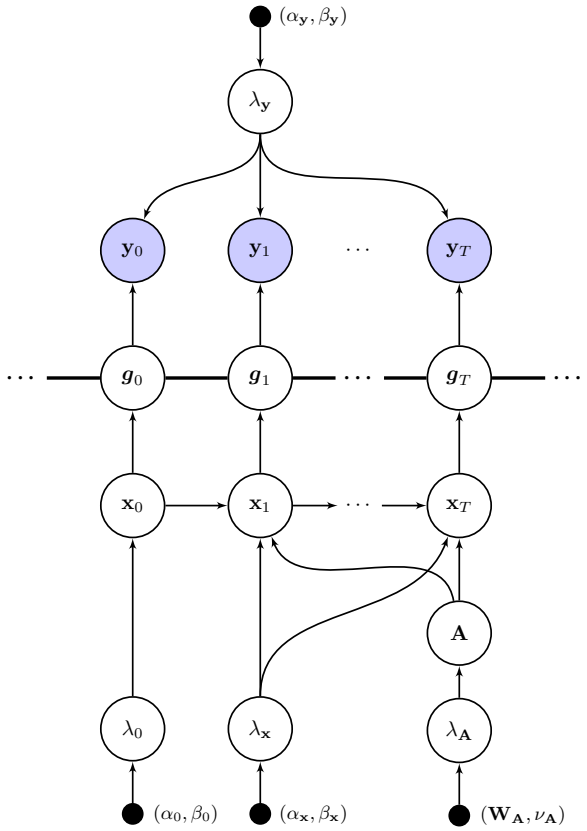


Figure 4.3.: The graphical model visualizes the Probability Density Model of the derived Bayesian Gaussian Process Dynamic Mode Decomposition. The blue shaded nodes represent the given observations $\mathbf{y}_0, \cdots, \mathbf{y}_T$. White nodes correspond to the unknown latent variables. The sequence of states of interest in the latent space is given by $\mathbf{x}_0, \cdots, \mathbf{x}_T$. $\mathbf{g}_0, \cdots, \mathbf{g}_T$ describe the outputs of the Gaussian process function corresponding to the observation model. The linear operator is given by $\mathbf{A}$. The precision values of the assumed Gaussian distributions are depicted by $\lambda_{\mathbf{y}}, \lambda_{\mathbf{0}}, \lambda_{\mathbf{x}}, \lambda_{\mathbf{a}}$. The thick black line denotes an auto-regressive structure within the Gaussian process, implying that $\mathbf{g}_t$ depends on $\mathbf{g}_0, \cdots, \mathbf{g}_{t-1}$ and $\mathbf{x}_0, \cdots, \mathbf{x}_t$. The small black nodes correspond to the hyperparameters of the given Probability Density Model. In this case, they represent shape and rate parameters for Gamma distribution $\mathrm{Gam}(\cdot)$ and the degrees of freedom and the scale matrix for a Wishart distribution $\mathcal{W}(\cdot)$.

The optimization of the fully Bayesian approach utilizes Variational Inference or Variational Bayes techniques, and consequently, an Evidence Lower Bound is first formulated [50, 54, 59]. In analogy to Section 2.3, the ELBO is given by

$$\begin{aligned}\log p(\mathbf{Y} \mid \boldsymbol{\theta}) &\geq \int q(\boldsymbol{\Theta}) \log \frac{p(\mathbf{Y}, \boldsymbol{\Theta} \mid \boldsymbol{\theta})}{q(\boldsymbol{\Theta})} \, \mathrm{d}\boldsymbol{\Theta} \\ &= \left\langle \log \frac{p(\mathbf{Y}, \boldsymbol{\Theta} \mid \boldsymbol{\theta})}{q(\boldsymbol{\Theta})} \right\rangle_{q(\boldsymbol{\Theta})} \\ &\stackrel{\text{def}}{=} \mathcal{L}_{\text{ELBO}}(q(\boldsymbol{\Theta}), \boldsymbol{\theta}),\end{aligned}$$

extending the marginal log-likelihood with a variational distribution $q(\boldsymbol{\Theta})$. The operator $\langle f(\cdot) \rangle$ corresponds to the expected value w.r.t. this variational distribution. For the variational distribution concerning the outputs of the observational model, the specific form of the original GP from the Probability Density Model (see Equation (4.10)) is assumed. As a result, the following factorization is obtained under the mean field assumption

$$q(\boldsymbol{\Theta}) = \prod_{\boldsymbol{\Theta}_i \in \boldsymbol{\Theta}} q(\boldsymbol{\Theta}_i) = \prod_{t=0}^{T} p(\mathbf{g}_t \mid \mathbf{U}, \mathbf{X}) \prod_{\boldsymbol{\Theta}_i \in \boldsymbol{\Theta} \setminus \{\mathbf{G}\}} q(\boldsymbol{\Theta}_i).$$

Hence, no specific functional forms are chosen for the remaining latent variables. Those functional forms will arise naturally based on the chosen structure of the likelihood functions and the corresponding conjugate priors during this section [4, 6].

The ELBO leads to

$$\begin{aligned}\mathcal{L}_{\text{ELBO}}(q(\boldsymbol{\Theta}), \boldsymbol{\theta}) = \; & - \mathrm{KL}(q(\lambda_{\mathbf{y}}) \parallel p(\lambda_{\mathbf{y}})) - \mathrm{KL}(q(\lambda_{\mathbf{0}}) \parallel p(\lambda_{\mathbf{0}})) - \mathrm{KL}(q(\lambda_{\mathbf{x}}) \parallel p(\lambda_{\mathbf{x}})) \\ & - \mathrm{KL}(q(\mathbf{U}) \parallel p(\mathbf{U})) - \mathrm{KL}(q(\mathbf{A})q(\lambda_{\mathbf{a}}) \parallel p(\mathbf{A} \mid \lambda_{\mathbf{a}})p(\lambda_{\mathbf{a}})) \\ & - \mathrm{H}(q(\mathbf{X})q(\lambda_{\mathbf{0}}) \parallel p(\mathbf{x}_0 \mid \lambda_{\mathbf{0}})) + \mathrm{H}(q(\mathbf{X}) \parallel q(\mathbf{X})) \\ & + \left\langle \log \prod_{t=0}^{T} p(\mathbf{y}_t \mid \mathbf{g}_t, \lambda_{\mathbf{y}}) \right\rangle_{q(\mathbf{G}, \mathbf{U}, \mathbf{X}, \lambda_{\mathbf{y}})} \\ & + \left\langle \log \prod_{t=1}^{T} p(\mathbf{x}_t \mid \mathbf{x}_{t-1}, \mathbf{A}, \lambda_{\mathbf{x}}) \right\rangle_{q(\mathbf{X}, \mathbf{A}, \lambda_{\mathbf{x}})},\end{aligned}$$

where $\mathrm{KL}(\cdot \parallel \cdot)$ and $\mathrm{H}(\cdot \parallel \cdot)$ describe Kullback–Leibler divergence and cross-entropy, respectively (see Appendix B). A closer look at the expectation value over the log-likelihood of the seen data

$$\left\langle \log \prod_{t=0}^{T} p(\mathbf{y}_t \mid \mathbf{g}_t, \lambda_{\mathbf{y}}) \right\rangle_{q(\mathbf{G},\mathbf{U},\mathbf{X},\lambda_{\mathbf{y}})} = \left\langle \sum_{t=0}^{T} \underbrace{\langle \log p(\mathbf{y}_t \mid \mathbf{g}_t, \lambda_{\mathbf{y}}) \rangle_{p(\mathbf{g}_t \mid \mathbf{U},\mathbf{X})}}_{=\zeta_1} \right\rangle_{q(\mathbf{U},\mathbf{X},\lambda_{\mathbf{y}})},$$

w.r.t. the variational distribution $q(\mathbf{G})$ results in an analytic closed-form solution due to its Gaussian nature. This analytic calculation of

$$\begin{aligned}
\zeta_1 &= \langle \log p(\mathbf{y}_t \mid \mathbf{g}_t, \lambda_{\mathbf{y}}) \rangle_{p(\mathbf{g_t} \mid \mathbf{U},\mathbf{X})} \\
&= \int p(\mathbf{g}_t \mid \mathbf{U}, \mathbf{X}) \log p(\mathbf{y}_t \mid \mathbf{g}_t, \lambda_{\mathbf{y}}) \, \mathrm{d}\mathbf{g}_t \\
&= \int \mathcal{N}(\mathbf{g}_t \mid \mathbf{C}_t\mathbf{U}, \mathbf{D}_t) \log \mathcal{N}(\mathbf{y}_t \mid \mathbf{g}_t, \lambda_{\mathbf{y}}^{-1}\mathbf{I}) \, \mathrm{d}\mathbf{g}_t \\
&= \log \mathcal{N}(\mathbf{y}_t \mid \mathbf{C}_t\mathbf{U}, \lambda_{\mathbf{y}}^{-1}\mathbf{I}) - \frac{N\lambda_{\mathbf{y}}}{2}\mathrm{Tr}(\mathbf{D}_t),
\end{aligned}$$

leads to an additional penalty term $-N\lambda_{\mathbf{y}}/2\mathrm{Tr}(\mathbf{D}_t)$ induced by the inducing pairs, similar to Sections 2.5 and 2.7. This term corresponds to the posterior variance between the inducing variables $\mathbf{U}$ and the given observations $\mathbf{Y}$ [54, 77, 80]. Eventually, the ELBO results in

$$\begin{aligned}
\mathcal{L}_{\text{ELBO}}(q(\boldsymbol{\Theta}), \boldsymbol{\theta}) = \; & - \mathrm{KL}(q(\lambda_{\mathbf{y}}) \parallel p(\lambda_{\mathbf{y}})) - \mathrm{KL}(q(\lambda_{\mathbf{0}}) \parallel p(\lambda_{\mathbf{0}})) - \mathrm{KL}(q(\lambda_{\mathbf{x}}) \parallel p(\lambda_{\mathbf{x}})) \\
& - \mathrm{KL}(q(\mathbf{U}) \parallel p(\mathbf{U})) - \mathrm{KL}(q(\mathbf{A})q(\lambda_{\mathbf{a}}) \parallel p(\mathbf{A} \mid \lambda_{\mathbf{a}})p(\lambda_{\mathbf{a}})) \\
& - \mathrm{H}(q(\mathbf{X})q(\lambda_{\mathbf{0}}) \parallel p(\mathbf{x}_0 \mid \lambda_{\mathbf{0}})) + \mathrm{H}(q(\mathbf{X}) \parallel q(\mathbf{X})) \\
& + \left\langle \log \prod_{t=0}^{T} \mathcal{N}(\mathbf{y}_t \mid \mathbf{C}_t\mathbf{U}, \lambda_{\mathbf{y}}^{-1}\mathbf{I}) \right\rangle_{q(\mathbf{X},\mathbf{U},\lambda_{\mathbf{y}})} \\
& + \left\langle \log \prod_{t=1}^{T} p(\mathbf{x}_t \mid \mathbf{x}_{t-1}, \mathbf{A}, \lambda_{\mathbf{x}}) \right\rangle_{q(\mathbf{X},\mathbf{A},\lambda_{\mathbf{x}})} \\
& - \left\langle \frac{N\lambda_{\mathbf{y}}}{2}\mathrm{Tr}(\mathbf{D}_{:}) \right\rangle_{q(\mathbf{X},\lambda_{\mathbf{y}})},
\end{aligned}$$

where $\mathbf{D}_{:} = \mathbf{K}_{::} - \mathbf{K}_{:\mathbf{Z}}\mathbf{K}_{\mathbf{ZZ}}\mathbf{K}_{\mathbf{Z}:}$ generalizes the expression of $\mathbf{D}_t$ for all time steps. The KL divergences provide regularization parameters ensuring that the variational distributions stay close to the pre-assumed prior distributions. The cross-entropy term

$\mathrm{H}(q(\mathbf{X})q(\lambda_{\mathbf{0}}) \parallel p(\mathbf{x}_0 \mid \lambda_{\mathbf{0}}))$ prioritizes initial states $\mathbf{x}_0$ in the latent space, which are very likely according to the selected prior. The subsequent entropy term $\mathrm{H}(q(\mathbf{X}) \parallel q(\mathbf{X}))$ penalizes too narrow variational distributions $q(\mathbf{X})$ [54]. The latter two terms, not discussed earlier, represent expectation values over two likelihood functions. On the one hand, the former guarantees that observations $\mathbf{Y}$ are taken into account during optimization. On the other hand, the second one ensures linear dynamical behavior in the latent space. Consequently, the ELBO results in an optimization procedure

$$\begin{aligned} q^*(\boldsymbol{\Theta}) = &\quad \arg\max_{q(\boldsymbol{\Theta})} \mathcal{L}_{\text{ELBO}}(q(\boldsymbol{\Theta}), \boldsymbol{\theta}), \\ \boldsymbol{\theta}^* = &\quad \arg\max_{\boldsymbol{\theta}} \mathcal{L}_{\text{ELBO}}(q^*(\boldsymbol{\Theta}), \boldsymbol{\theta}), \end{aligned}$$

similar to Section 2.3. In the following, the optimal variational distributions are first derived utilizing the *Calculus of Variation* and *Euler Lagrange* [3, 4, 6, 54, 59]. Then, the optimization of the variational parameters and hyperparameters is discussed.

Optimal Variational Distribtuion $q^*(\lambda_{\mathbf{y}})$

In order to obtain an optimal estimate for $q(\lambda_{\mathbf{y}})$, the derivative of $\mathcal{L}_{\text{ELBO}}(q(\boldsymbol{\Theta}), \boldsymbol{\theta})$, is taken w.r.t. $q(\lambda_{\mathbf{y}})$ and set equal to zero, leading to

$$\begin{aligned} \frac{\partial \mathcal{L}_{\text{ELBO}}}{\partial q(\lambda_{\mathbf{y}})} &= \log \frac{p(\lambda_{\mathbf{y}})}{q(\lambda_{\mathbf{y}})} - 1 - \left\langle \frac{N\lambda_{\mathbf{y}}}{2} \mathrm{Tr}(\mathbf{D}_{:}) \right\rangle_{q(\mathbf{X})} + \left\langle \log \prod_{t=0}^{T} \mathcal{N}(\mathbf{y}_t \mid \mathbf{C}_t \mathbf{U}, \lambda_{\mathbf{y}}^{-1} \mathbf{I}) \right\rangle_{q(\mathbf{X},\mathbf{U})} \\ &= \log \frac{p(\lambda_{\mathbf{y}})}{q(\lambda_{\mathbf{y}})} - 1 - \left\langle \frac{N\lambda_{\mathbf{y}}}{2} \mathrm{Tr}(\mathbf{D}_{:}) \right\rangle_{q(\mathbf{X})} \\ &\quad + \left\langle \log \mathcal{N}(\mathbf{Y} \mid \mathbf{C}_{:} \widetilde{\boldsymbol{\mu}}_{\mathbf{U}}, \lambda_{\mathbf{y}}^{-1} \mathbf{I}) \right\rangle_{q(\mathbf{X})} - \frac{N\lambda_{\mathbf{y}}}{2} \left\langle \mathrm{Tr}\left(\mathbf{C}_{:}^T \mathbf{C}_{:} \widetilde{\boldsymbol{\Lambda}}_{\mathbf{U}}^{-1} \right) \right\rangle_{q(\mathbf{X})} \\ &\stackrel{!}{=} 0, \end{aligned}$$

where $\mathbf{C}_{:} = \mathbf{K}_{:\mathbf{Z}} \mathbf{K}_{\mathbf{ZZ}}^{-1}$ generalizes $\mathbf{C}_t$ over all time steps. The mean matrix $\widetilde{\boldsymbol{\mu}}_{\mathbf{U}}$ (see Equation (4.21)) and a precision matrix $\widetilde{\boldsymbol{\Lambda}}_{\mathbf{U}}$ (see Equation (4.22)) are the natural parameters of the optimal variational distribution $q^*(\mathbf{U})$. Considering the exponential space, the

optimal variational distribution

$$
\begin{aligned}
q^*(\lambda_{\mathbf{y}}) \propto \exp\Bigg(& \log \mathrm{Gam}\left(\lambda_{\mathbf{y}} \mid \alpha_{\mathbf{y}}, \beta_{\mathbf{y}}\right) - \left\langle \frac{N\lambda_{\mathbf{y}}}{2} \mathrm{Tr}(\mathbf{D}_{:}) \right\rangle_{q(\mathbf{X})} \\
& + \left\langle \log \mathcal{N}(\mathbf{Y} \mid \mathbf{C}_{:}\widetilde{\boldsymbol{\mu}}_{\mathbf{U}}, \lambda_{\mathbf{y}}^{-1}\mathbf{I}) \right\rangle_{q(\mathbf{X})} - \frac{N\lambda_{\mathbf{y}}}{2} \left\langle \sum_{s=1} \mathrm{Tr}\left(\mathbf{C}_{:}^T \mathbf{C}_{:} \widetilde{\boldsymbol{\Lambda}}_{\mathbf{U}}^{-1}\right) \right\rangle_{q(\mathbf{X})} \Bigg) \\
\propto \exp\Bigg(& \frac{2\alpha_{\mathbf{y}} - 2}{2} \log|\lambda_{\mathbf{y}}| - \frac{2\beta_{\mathbf{y}}\lambda_{\mathbf{y}}}{2} - \left\langle \frac{N\lambda_{\mathbf{y}}}{2} \mathrm{Tr}(\mathbf{D}_{:}) \right\rangle_{q(\mathbf{X})} \\
& + \frac{TN}{2} \log|\lambda_{\mathbf{y}}| - \frac{\lambda_{\mathbf{y}}}{2} \left\langle (\mathbf{Y} - \mathbf{C}_{:}\widetilde{\boldsymbol{\mu}}_{\mathbf{U}}) (\mathbf{Y} - \mathbf{C}_{:}\widetilde{\boldsymbol{\mu}}_{\mathbf{U}})^T \right\rangle_{q(\mathbf{X})} \\
& - \frac{N\lambda_{\mathbf{y}}}{2} \left\langle \mathrm{Tr}\left(\mathbf{C}_{:}^T \mathbf{C}_{:} \widetilde{\boldsymbol{\Lambda}}_{\mathbf{U}}^{-1}\right) \right\rangle_{q(\mathbf{X})} \Bigg),
\end{aligned}
$$

takes on the form of a Gamma distribution

$$
q^*(\lambda_{\mathbf{y}}) \propto \mathrm{Gam}\left(\lambda_{\mathbf{y}} \mid \widetilde{\alpha}_{\mathbf{y}}, \widetilde{\beta}_{\mathbf{y}}\right),
$$

with natural parameters

$$
\widetilde{\alpha}_{\mathbf{y}} = \frac{TN + 2\alpha_{\mathbf{y}}}{2}, \tag{4.11}
$$

$$
\widetilde{\beta}_{\mathbf{y}} = 2\beta_{\mathbf{y}} + \mathrm{Tr}\left((\mathbf{Y} - \mathbf{C}_{:}\widetilde{\boldsymbol{\mu}}_{\mathbf{U}}) (\mathbf{Y} - \mathbf{C}_{:}\widetilde{\boldsymbol{\mu}}_{\mathbf{U}})^T + N \left(\mathbf{D}_{:} + \mathbf{C}_{:} \widetilde{\boldsymbol{\Lambda}}_{\mathbf{U}}^{-1} \mathbf{C}_{:}^T \right) \right), \tag{4.12}
$$

corresponding to the shape and the rate parameter, respectively. This resulting form of a Gamma distribution confirms the conjugacy property [4, 6].

Optimal Variational Distribution $q^*(\lambda_{\mathbf{0}})$

Next, the optimal variational distribution $q^*(\lambda_{\mathbf{0}})$ results from the derivative of the ELBO $\mathcal{L}_{\text{ELBO}}(q(\boldsymbol{\Theta}), \boldsymbol{\theta})$ w.r.t. to the distribution $q(\lambda_{\mathbf{0}})$ resulting in

$$
\begin{aligned}
\frac{\partial \mathcal{L}_{\text{ELBO}}}{\partial q(\lambda_{\mathbf{0}})} &= \log \frac{p(\lambda_{\mathbf{0}})}{q(\lambda_{\mathbf{0}})} - 1 + \left\langle \log p(\mathbf{x}_0 \mid \lambda_{\mathbf{0}}) \right\rangle_{q(\mathbf{X})} \\
&\stackrel{!}{=} 0,
\end{aligned}
$$

where the resulting gradient is set equal to zero. The exponential space leads to the relation

$$q^*(\lambda_{\mathbf{0}}) \propto \exp\left(\log \mathrm{Gam}\left(\lambda_{\mathbf{0}} \mid \alpha_{\mathbf{0}}, \beta_{\mathbf{0}}\right) + \left\langle \log \mathcal{N}(\mathbf{x}_0 \mid 0, \lambda_{\mathbf{0}}^{-1}\mathbf{I})\right\rangle_{q(\mathbf{X})}\right)$$

$$\propto \exp\left(\frac{2\alpha_{\mathbf{0}} - 2}{2}\log|\lambda_{\mathbf{0}}| - \frac{2\beta_{\mathbf{0}}\lambda_{\mathbf{0}}}{2} + \frac{M}{2}\log|\lambda_{\mathbf{0}}| - \frac{\lambda_{\mathbf{0}}}{2}\mathrm{Tr}\left(\left\langle \mathbf{x}_0\mathbf{x}_0^T\right\rangle_{q(\mathbf{X})}\right)\right),$$

where $q^*(\lambda_{\mathbf{0}})$ takes the form of a Gamma distribution

$$q^*(\lambda_{\mathbf{0}}) \propto \mathrm{Gam}\left(\lambda_{\mathbf{0}} \mid \widetilde{\alpha}_{\mathbf{0}}, \widetilde{\beta}_{\mathbf{0}}\right),$$

with the natural parameters

$$\widetilde{\alpha}_{\mathbf{0}} = \frac{M + 2\alpha_{\mathbf{0}}}{2}, \tag{4.13}$$

$$\widetilde{\beta}_{\mathbf{0}} = 2\beta_{\mathbf{0}} + \mathrm{Tr}\left(\left\langle \mathbf{x}_0\mathbf{x}_0^T\right\rangle_{q(\mathbf{X})}\right), \tag{4.14}$$

representing the shape and the rate parameter, respectively. Again, the resulting Gamma distribution is a natural consequence of the conjugacy property [4, 6].

Optimal Variational Distribution $q^*(\lambda_{\mathbf{x}})$

The gradient is taken w.r.t. $q(\lambda_{\mathbf{x}})$ and set equal to zero to estimate the optimal distribution $q^*(\lambda_{\mathbf{x}})$

$$\begin{aligned}
\frac{\partial \mathcal{L}_{\text{ELBO}}}{\partial q(\lambda_{\mathbf{x}})} &= \log\frac{p(\lambda_{\mathbf{0}})}{q(\lambda_{\mathbf{0}})} - 1 + \left\langle \log\prod_{t=1}^{T} p(\mathbf{x}_t \mid \mathbf{x}_{t-1}, \mathbf{A}, \lambda_{\mathbf{x}})\right\rangle_{q(\mathbf{X})q(\mathbf{A})} \\
&= \log\frac{p(\lambda_{\mathbf{0}})}{q(\lambda_{\mathbf{0}})} - 1 + \left\langle \log\prod_{t=1}^{T}\prod_{m=1}^{M} p(\mathbf{x}_{m,t} \mid \mathbf{x}_{t-1}, \widetilde{\boldsymbol{\mu}}_{\mathbf{a}_m}, \lambda_{\mathbf{x}})\right\rangle_{q(\mathbf{X})} \\
&\quad - \sum_{t=1}^{T}\frac{\lambda_{\mathbf{x}}}{2}\left\langle \mathrm{Tr}\left(\mathbf{x}_{t-1}^T\left(\sum_{m=1}^{M}\widetilde{\boldsymbol{\Lambda}}_{\mathbf{a}_m}^{-1}\right)\mathbf{x}_{t-1}\right)\right\rangle_{q(\mathbf{X})}, \\
&\stackrel{!}{=} 0,
\end{aligned}$$

where $\widetilde{\boldsymbol{\mu}}_{\mathbf{a}_m}$ (see Equation (4.19)) and $\widetilde{\boldsymbol{\Lambda}}_{\mathbf{a}_m}$ for $m = 1, \cdots, M$ (see Equation (4.20)) are natural parameters coming from the optimal variational distribution $q^*(\mathbf{A})$. The transformation to exponential space, given by

$$
\begin{aligned}
q^*(\lambda_{\mathbf{x}}) \propto \exp\Bigg(& \log \mathrm{Gam}\left(\lambda_{\mathbf{x}} \mid \alpha_{\mathbf{x}}, \beta_{\mathbf{x}}\right) + \left\langle \log \prod_{t=1}^{T} \prod_{m=1}^{M} \mathcal{N}(\mathbf{x}_{m,t} \mid \mathbf{x}_{t-1}^T \widetilde{\boldsymbol{\mu}}_{\mathbf{a}_m}, \lambda_{\mathbf{x}}^{-1}) \right\rangle_{q(\mathbf{X})} \\
& - \sum_{t=1}^{T} \frac{\lambda_{\mathbf{x}}}{2} \left\langle \mathrm{Tr}\left(\mathbf{x}_{t-1}^T \left(\sum_{m=1}^{M} \widetilde{\boldsymbol{\Lambda}}_{\mathbf{a}_m}^{-1} \right) \mathbf{x}_{t-1} \right) \right\rangle_{q(\mathbf{X})} \Bigg) \\
\propto \exp\Bigg(& \frac{2\alpha_{\mathbf{x}} - 2}{2} \log|\lambda_{\mathbf{x}}| - \frac{2\widetilde{\beta}_{\mathbf{x}} \lambda_{\mathbf{x}}}{2} + \frac{M(T-1)}{2} \log|\lambda_{\mathbf{x}}| \\
& - \frac{\lambda_{\mathbf{x}}}{2} \mathrm{Tr} \left\langle \left(\left(\mathbf{X}^{1^T} - \mathbf{X}^{0^T} \widetilde{\boldsymbol{\mu}}_{\mathbf{a}} \right) \left(\mathbf{X}^{1^T} - \mathbf{X}^{0^T} \widetilde{\boldsymbol{\mu}}_{\mathbf{a}} \right)^T \right\rangle_{q(\mathbf{X})} \right) \\
& - \frac{\lambda_{\mathbf{x}}}{2} \left\langle \mathrm{Tr}\left(\sum_{t=1}^{T} \mathbf{X}^{0^T} \left(\sum_{m=1}^{M} \widetilde{\boldsymbol{\Lambda}}_{\mathbf{a}_m}^{-1} \right) \mathbf{X}^0 \right) \right\rangle_{q(\mathbf{X})} \Bigg),
\end{aligned}
$$

leads to an optimal $q^*(\lambda_{\mathbf{x}})$ that takes the form of a Gamma distribution

$$
q^*(\lambda_{\mathbf{x}}) \propto \mathrm{Gam}\left(\lambda_{\mathbf{x}} \mid \widetilde{\alpha}_{\mathbf{x}}, \widetilde{\beta}_{\mathbf{x}}\right).
$$

The shape and rate are given by

$$
\widetilde{\alpha}_{\mathbf{x}} = \frac{M(T-1) + 2\alpha_{\mathbf{x}}}{2}, \tag{4.15}
$$

$$
\begin{aligned}
\widetilde{\beta}_{\mathbf{x}} = 2\beta_{\mathbf{x}} + \mathrm{Tr}\Bigg(& \left\langle \mathbf{X}^{0^T} \left(\sum_{m=1}^{M} \widetilde{\boldsymbol{\Lambda}}_{\mathbf{a}_m}^{-1} \right) \mathbf{X}^0 \right\rangle_{q(\mathbf{X})} \\
& + \left\langle \left(\mathbf{X}^{1^T} - \mathbf{X}^{0^T} \widetilde{\boldsymbol{\mu}}_{\mathbf{a}} \right) \left(\mathbf{X}^{1^T} - \mathbf{X}^{0^T} \widetilde{\boldsymbol{\mu}}_{\mathbf{a}} \right)^T \right\rangle_{q(\mathbf{X})} \Bigg),
\end{aligned} \tag{4.16}
$$

corresponding to the natural parameters of $q^*(\lambda_{\mathbf{x}})$.

Optimal Variational Distribution $q^*(\lambda_{\mathbf{a}})$

The optimal variational distribution $q^*(\lambda_{\mathbf{a}})$ for the last precision value $\lambda_{\mathbf{a}}$ is obtained similarly. First, the derivative is performed

$$\begin{aligned}\frac{\partial \mathcal{L}_{\text{ELBO}}}{\partial q(\lambda_{\mathbf{a}})} &= \left\langle \log \frac{p(\mathbf{A} \mid \lambda_{\mathbf{a}}) p(\lambda_{\mathbf{a}})}{q(\lambda_{\mathbf{a}})} \right\rangle_{q(\mathbf{A})} - 1 \\ &\stackrel{!}{=} 0,\end{aligned}$$

which should be equal to zero. In exponential space, it follows

$$\begin{aligned}q^*(\lambda_{\mathbf{a}}) &\propto \exp\left(\left\langle \log \mathcal{N}(\prod_{m=1}^{M} \mathbf{a}_m \mid \mathbf{0}, \lambda_{\mathbf{a}}^{-1}\mathbf{I}) \right\rangle_{q(\mathbf{A})} + \log \operatorname{Gam}(\lambda_{\mathbf{a}} \mid \alpha_{\mathbf{a}}, \beta_{\mathbf{a}}) \right) \\ &\propto \exp\left(-\frac{\lambda_{\mathbf{a}}}{2} \operatorname{Tr}\left(\left\langle \mathbf{A}\mathbf{A}^T \right\rangle_{q(\mathbf{A})} \right) - \frac{2\beta_{\mathbf{a}}}{2}\lambda_{\mathbf{a}} + \frac{M^2}{2}\log|\lambda_{\mathbf{a}}| + \frac{2\alpha_{\mathbf{a}} - 2)}{2}\log|\lambda_{\mathbf{a}}| \right),\end{aligned}$$

where the resulting optimal distribution $q^*(\lambda_{\mathbf{a}})$ adopts the form of the Gamma distribution

$$q^*(\lambda_{\mathbf{a}}) \propto \operatorname{Gam}(\lambda_{\mathbf{a}} \mid \widetilde{\alpha}_{\mathbf{a}}, \widetilde{\beta}_{\mathbf{a}}),$$

with natural parameters

$$\widetilde{\alpha}_{\mathbf{a}} = 2\alpha_{\mathbf{a}} + M^2, \tag{4.17}$$

$$\begin{aligned}\widetilde{\beta}_{\mathbf{a}} &= 2\beta_{\mathbf{a}} + \operatorname{Tr}\left(\left\langle \mathbf{A}\mathbf{A}^T \right\rangle_{q(\mathbf{A})} \right) \\ &= 2\beta_{\mathbf{a}} + \operatorname{Tr}\left(\widetilde{\boldsymbol{\mu}}_{\mathbf{a}} \widetilde{\boldsymbol{\mu}}_{\mathbf{a}}^T + \sum_{m=1}^{M} \widetilde{\boldsymbol{\Lambda}}_{\mathbf{a}_m}^{-1} \right),\end{aligned} \tag{4.18}$$

corresponding to the shape and rate, respectively. The mean matrix $\widetilde{\boldsymbol{\mu}}_{\mathbf{a}}$ (see Equation (4.19)) and the precision matricies $\widetilde{\boldsymbol{\Lambda}}_{\mathbf{a}_m}$ for $m = 1, \cdots, M$ (see Equation (4.20)) are the natural parameters of the optimal variational distribution $q^*(\mathbf{A})$ corresponding to the linear operator $\mathbf{A}$.

Optimal Variational Distribution $q^*(\mathbf{A})$

The estimation of the optimal variational distribution $q^*(\mathbf{A})$ associated with the linear operator $\mathbf{A}$ is based on the gradient of the ELBO $\mathcal{L}_{\text{ELBO}}(q(\boldsymbol{\Theta}), \boldsymbol{\theta})$ w.r.t. $q(\mathbf{A})$, resulting in

$$\begin{aligned}\frac{\partial \mathcal{L}_{\text{ELBO}}}{\partial q(\mathbf{A})} &= \left\langle \log \frac{\prod_{m=1}^{M} p(\mathbf{a}_m \mid \lambda_{\mathbf{a}}) p(\lambda_{\mathbf{a}})}{q(\mathbf{A}) q(\lambda_{\mathbf{a}})} \right\rangle_{q(\lambda_{\mathbf{a}})} - 1 \\ &\quad + \left\langle \log \prod_{t=1}^{T} p(\mathbf{x}_{m,t} \mid \mathbf{x}_{t-1}, \mathbf{a}_m, \lambda_{\mathbf{x}}) \right\rangle_{q(\mathbf{X}) q(\lambda_{\mathbf{x}})} \\ &\stackrel{!}{=} 0,\end{aligned}$$

which is then set equal to zero. The transformation to the exponential space leads to

$$\begin{aligned}q^*(\mathbf{A}) &\propto \prod_{m=1}^{M} \exp\Bigg(\left\langle \log \mathcal{N}(\mathbf{a}_m \mid \mathbf{0}, \lambda_{\mathbf{a}}^{-1} \mathbf{I}) \right\rangle_{q(\lambda_{\mathbf{a}})} \\ &\qquad \left\langle \log \prod_{t=1}^{T} \mathcal{N}(\mathbf{x}_{m,t} \mid \mathbf{x}_{t-1}^T \mathbf{a}_m, \lambda_{\mathbf{x}}^{-1} \mathbf{I}) \right\rangle_{q(\mathbf{X}) q(\lambda_{\mathbf{x}})} \Bigg) \\ &\propto \prod_{m=1}^{M} \exp\Bigg(-1/2 \Bigg(\left\langle \lambda_{\mathbf{x}} \mathbf{X}_m^1 \mathbf{X}_m^{1^T} \right\rangle_{q(\lambda_{\mathbf{x}}) q(\mathbf{X})} \\ &\qquad - 2 \mathbf{a}_m^T \left(\left\langle \lambda_{\mathbf{x}} \mathbf{X}^0 \mathbf{X}_m^{1^T} \right\rangle_{q(\lambda_{\mathbf{x}}) q(\mathbf{X})} \right) \\ &\qquad + \mathbf{a}_m^T \left(\left\langle \lambda_{\mathbf{x}} \mathbf{X}^0 \mathbf{X}^{0^T} \right\rangle_{q(\lambda_{\mathbf{x}}) q(\mathbf{X})} + \langle \lambda_{\mathbf{a}} \rangle_{q(\lambda_{\mathbf{a}})} \right) \mathbf{a}_m \Bigg) \Bigg),\end{aligned}$$

where completing the squares produces the factorization into M independent Gaussian distributions

$$\begin{aligned}q^*(\mathbf{A}) &\propto \prod_{m=1}^{M} q^*(\mathbf{a}_m) \\ &= \prod_{m=1}^{M} \mathcal{N}(\mathbf{a}_m \mid \widetilde{\boldsymbol{\mu}}_{\mathbf{a}_m}, \widetilde{\boldsymbol{\Lambda}}_{\mathbf{a}}^{-1}),\end{aligned}$$

with natural parameters

$$\begin{aligned}\widetilde{\boldsymbol{\mu}}_{\mathbf{a}_m} &= \left(\left\langle \frac{\lambda_{\mathbf{a}}}{\lambda_{\mathbf{x}}} \right\rangle_{q(\lambda_{\mathbf{x}})q(\boldsymbol{\Lambda}_{\mathbf{a}_m})} + \Psi^2\right)^{-1} \Psi_{1_m} \\ &= \left(\frac{\widetilde{\beta}_{\mathbf{x}}\widetilde{\alpha}_{\mathbf{a}}}{\widetilde{\alpha}_{\mathbf{x}}\widetilde{\beta}_{\mathbf{a}}} + \Psi_2\right)^{-1} \Psi_{1_m}, \end{aligned} \tag{4.19}$$

$$\begin{aligned}\widetilde{\boldsymbol{\Lambda}}_{\mathbf{a}} &= \langle \lambda_{\mathbf{a}} \rangle_{q(\lambda_{\mathbf{a}})} + \langle \lambda_{\mathbf{x}} \rangle_{q(\lambda_{\mathbf{x}})} \Psi_2 \\ &= \frac{\widetilde{\alpha}_{\mathbf{a}}}{\widetilde{\beta}_{\mathbf{a}}} + \frac{\widetilde{\alpha}_{\mathbf{x}}}{\widetilde{\beta}_{\mathbf{x}}} \Psi_2. \end{aligned} \tag{4.20}$$

The two terms Ψ_{1_m} and Ψ_2 correspond to sufficient statistics and are defined by

$$\Psi_{1_m} = \left\langle \mathbf{X}^0 \mathbf{X}_m^{1^T} \right\rangle_{q(\mathbf{X})}, \quad \Psi_2 = \left\langle \mathbf{X}^0 \mathbf{X}^{0^T} \right\rangle_{q(\mathbf{X})}.$$

Similar to the Maximum A Posteriori probability estimate of the linear operator $\mathbf{A}^*$ in Equation (4.5), the mean $\widetilde{\boldsymbol{\mu}}_{\mathbf{a}_m}$ is an expression of a ridge regression. However, it now depends on several expectations. On the one hand, the ridge coefficient is obtained from the natural parameters of the optimal variational distributions $q^*(\lambda_{\mathbf{x}})$ (see Equations (4.15) and (4.16)) and $q^*(\lambda_{\mathbf{a}})$ (see Equations (4.17) and (4.18)). On the other hand, the sufficient statistics Ψ_{1_m} and Ψ_2 correspond to estimates of expected values w.r.t. the variational distribution $q(\mathbf{X})$.

Optimal Variational Distribution $q^*(\mathbf{U})$

To achieve an optimal estimate for the variational distribution $q^*(\mathbf{U})$ the derivative of the ELBO $\mathcal{L}_{\text{ELBO}}(q(\boldsymbol{\Theta}), \boldsymbol{\theta})$ w.r.t. $q(\mathbf{U})$ is given

$$\begin{aligned}\frac{\partial \mathcal{L}_{\text{ELBO}}}{\partial q(\mathbf{U})} &= \log \frac{p(\mathbf{U})}{q(\mathbf{U})} - 1 + \left\langle \log \prod_{t=0}^{T} \mathcal{N}(\mathbf{y}_t \mid \mathbf{C}_t \mathbf{U}, \lambda_{\mathbf{y}}^{-1} \mathbf{I}) \right\rangle_{q(\mathbf{X}, \lambda_{\mathbf{y}})} \\ &\stackrel{!}{=} 0, \end{aligned}$$

and subsequently set equal to zero. In the exponential space

$$
\begin{aligned}
q^*(\mathbf{U}) &\propto \prod_{n=1}^{N} \mathcal{N}(\mathbf{U}_n \mid, 0, \mathbf{K_{ZZ}}) \left\langle \log \mathcal{N}(\mathbf{Y}_n \mid \mathbf{C}_{:}\mathbf{U}_n, \lambda_{\mathbf{y}}^{-1}\mathbf{I}) \right\rangle_{q(\mathbf{X}),q(\lambda_{\mathbf{y}})} \\
&\propto \prod_{n=1}^{N} \exp\Bigg(-1/2 \Bigg(\left\langle \lambda_{\mathbf{y}} \right\rangle_{q(\lambda_{\mathbf{y}})} \mathbf{Y}_n^T \mathbf{Y}_n \\
&\qquad - 2\mathbf{U}_n^T \mathbf{K}_{\mathbf{ZZ}}^{-1} \left\langle \lambda_{\mathbf{y}} \mathbf{K}_{z:} \right\rangle_{q(\mathbf{X})q(\lambda_{\mathbf{y}})} \mathbf{Y}_n \\
&\qquad + \mathbf{U}_n^T \left(\mathbf{K}_{\mathbf{ZZ}}^{-1} + \mathbf{K}_{\mathbf{ZZ}}^{-1} \left\langle \lambda_{\mathbf{y}} \mathbf{K}_{z:} \mathbf{K}_{:z} \right\rangle_{q(\mathbf{X})q(\lambda_{\mathbf{y}})} \mathbf{K}_{\mathbf{ZZ}}^{-1} \right) \mathbf{U}_n \Bigg) \Bigg),
\end{aligned}
$$

completing the squares results in an optimal $q^*(\mathbf{U})$ which factorizes into N independent Gaussian distributions

$$
q^*(\mathbf{U}) \propto \prod_{n=1}^{N} \mathcal{N}\left(\mathbf{U}_n \mid \widetilde{\boldsymbol{\mu}}_{\mathbf{U}_n}, \widetilde{\boldsymbol{\Lambda}}_{\mathbf{U}}^{-1} \right).
$$

The natural parameters of this distribution are formalized

$$
\begin{aligned}
\widetilde{\boldsymbol{\mu}}_{\mathbf{U}_n} &= \left(\frac{1}{\left\langle \lambda_{\mathbf{y}} \right\rangle_{q(\lambda_{\mathbf{y}})}} \mathbf{K}_{\mathbf{ZZ}}^{-1} + \mathbf{K}_{\mathbf{ZZ}}^{-1} \Psi_4 \mathbf{K}_{\mathbf{ZZ}}^{-1} \right)^{-1} \mathbf{K}_{\mathbf{ZZ}}^{-1} \Psi_{3_n} \\
&= \left(\frac{\widetilde{\beta}_{\mathbf{y}}}{\widetilde{\alpha}_{\mathbf{y}}} \mathbf{K}_{\mathbf{ZZ}}^{-1} + \mathbf{K}_{\mathbf{ZZ}}^{-1} \Psi_4 \mathbf{K}_{\mathbf{ZZ}}^{-1} \right)^{-1} \mathbf{K}_{\mathbf{ZZ}}^{-1} \Psi_{3_n}, \qquad (4.21) \\
\widetilde{\boldsymbol{\Lambda}}_{\mathbf{U}} &= \mathbf{K}_{\mathbf{ZZ}}^{-1} + \left\langle \lambda_{\mathbf{y}} \right\rangle_{q(\lambda_{\mathbf{y}})} \mathbf{K}_{\mathbf{ZZ}}^{-1} \Psi_4 \mathbf{K}_{\mathbf{ZZ}}^{-1} \\
&= \mathbf{K}_{\mathbf{ZZ}}^{-1} + \frac{\widetilde{\alpha}_{\mathbf{y}}}{\widetilde{\beta}_{\mathbf{y}}} \mathbf{K}_{\mathbf{ZZ}}^{-1} \Psi_4 \mathbf{K}_{\mathbf{ZZ}}^{-1}, \qquad (4.22)
\end{aligned}
$$

corresponding to the mean vector $\widetilde{\boldsymbol{\mu}}_{\mathbf{U}_n}$ and precision matrix $\widetilde{\boldsymbol{\Lambda}}_{\mathbf{U}}$, respectively. The sufficient statistics Ψ_{3_n} and Ψ_4 are given

$$
\Psi_{3_n} = \left\langle \mathbf{K}_{z:} \right\rangle_{q(\mathbf{X})} \mathbf{Y}_n, \quad \Psi_4 = \left\langle \mathbf{K}_{z:} \mathbf{K}_{:z} \right\rangle_{q(\mathbf{X})},
$$

depending on the variational distribution $q(\mathbf{X})$ of the latent states $\mathbf{X}$. The shape $\widetilde{\alpha}_{\mathbf{y}}$ (see Equation (4.11)) and the rate $\widetilde{\beta}_{\mathbf{y}}$ (see Equation (4.12)) parameter are the natural parameters of optimal variational distribution $q^*(\lambda_{\mathbf{y}})$.

Optimal Variational Distribution $q^*(\mathbf{X})$

In order to estimate the optimal variation distribution $q^*(\mathbf{X})$, the gradient w.r.t. $q(\mathbf{X})$ is considered and then set equal to zero

$$
\begin{aligned}
\frac{\partial \mathcal{L}_{\text{ELBO}}}{\partial q(\mathbf{X})} = & -\log q(\mathbf{X}) - 1 + \langle \log p(\mathbf{x}_0 \mid \lambda_{\mathbf{0}}) \rangle_{q(\lambda_{\mathbf{0}})} \\
& + \left\langle \log \prod_{t=0}^{T} \mathcal{N}(\mathbf{y}_t \mid \mathbf{C}_t \mathbf{U}, \lambda_{\mathbf{y}}^{-1} \mathbf{I}) \right\rangle_{q(\mathbf{U}, \lambda_{\mathbf{y}})} \\
& + \left\langle \log \prod_{t=1}^{T} p(\mathbf{x}_t \mid \mathbf{x}_{t-1}, \mathbf{A}, \lambda_{\mathbf{x}}) \right\rangle_{q(\mathbf{A}) q(\lambda_{\mathbf{x}})} \\
& - \left\langle \frac{N \lambda_{\mathbf{y}}}{2} \operatorname{Tr}(\mathbf{D}_{:}) \right\rangle_{q(\lambda_{\mathbf{y}})} \\
= & -\log q(\mathbf{X}) - 1 + \log \mathcal{N}(\mathbf{x}_0 \mid 0, \frac{\widetilde{\beta}_{\mathbf{0}}}{\widetilde{\alpha}_{\mathbf{0}}} \mathbf{I}) \\
& + \sum_{t=0}^{T} \left(\log \mathcal{N}(\mathbf{y}_t \mid \mathbf{C}_t \widetilde{\boldsymbol{\mu}}_{\mathbf{U}}, \frac{\widetilde{\beta}_{\mathbf{y}}}{\widetilde{\alpha}_{\mathbf{y}}} \mathbf{I}) - \frac{N \widetilde{\alpha}_{\mathbf{y}}}{2 \widetilde{\beta}_{\mathbf{y}}} \operatorname{Tr}(\mathbf{D}_t + \mathbf{C}_t \widetilde{\boldsymbol{\Lambda}}_{\mathbf{U}}^{-1} \mathbf{C}_t^T) \right) \\
& + \sum_{t=1}^{T} \left(\log \mathcal{N}(\mathbf{x}_t \mid \mathbf{x}_{t-1}^T \widetilde{\boldsymbol{\mu}}_{\mathbf{a}}, \frac{\widetilde{\beta}_{\mathbf{x}}}{\widetilde{\alpha}_{\mathbf{x}}} \mathbf{I}) - \frac{\widetilde{\alpha}_{\mathbf{x}}}{2 \widetilde{\beta}_{\mathbf{x}}} \operatorname{Tr}(\mathbf{x}_{t-1}^T \sum_{m=1}^{M} \widetilde{\boldsymbol{\Lambda}}_{\mathbf{a}}^{-1} \mathbf{x}_{t-1}) \right) \\
\stackrel{!}{=} & \, 0.
\end{aligned}
$$

It depends on the natural parameters of the optimal variational distribution $q^*(\mathbf{U})$, $q^*(\mathbf{A})$, $q^*(\lambda_{\mathbf{y}})$, $q^*(\lambda_{\mathbf{0}})$ and $q^*(\lambda_{\mathbf{x}})$ derived in the previous sections (see Equations (4.11) to (4.16) and (4.19) to (4.22)). In exponential space, the optimal variational distribution $q^*(\mathbf{X})$ is

given by

$$
\begin{aligned}
q^*(\mathbf{X}) \propto \mathcal{N}(\mathbf{x}_0 \mid 0, \frac{\widetilde{\beta}_{\mathbf{0}}}{\widetilde{\alpha}_{\mathbf{0}}}\mathbf{I}) \\
&\prod_{t=0}^{T} \underbrace{\mathcal{N}(\mathbf{y}_t \mid \mathbf{C}_t\widetilde{\boldsymbol{\mu}}_{\mathbf{U}}, \frac{\widetilde{\beta}_{\mathbf{y}}}{\widetilde{\alpha}_{\mathbf{y}}}\mathbf{I}) \exp\left(-\frac{N\widetilde{\alpha}_{\mathbf{y}}}{2\widetilde{\beta}_{\mathbf{y}}}\mathrm{Tr}\left(\mathbf{D}_t + \mathbf{C}_t\widetilde{\boldsymbol{\Lambda}}_{\mathbf{U}}^{-1}\mathbf{C}_t^T\right)\right)}_{\widetilde{p}(\mathbf{y}_t|\mathbf{x}_t)} \\
&\prod_{t=1}^{T} \underbrace{\mathcal{N}(\mathbf{x}_t \mid \widetilde{\boldsymbol{\mu}}_{\mathbf{a}}^T\mathbf{x}_{t-1}, \frac{\widetilde{\beta}_{\mathbf{x}}}{\widetilde{\alpha}_{\mathbf{x}}}\mathbf{I}) \exp\left(-\frac{\widetilde{\alpha}_{\mathbf{x}}}{2\widetilde{\beta}_{\mathbf{x}}}\mathrm{Tr}(\mathbf{x}_{t-1}^T\sum_{m=1}^{M}\widetilde{\boldsymbol{\Lambda}}_{\mathbf{a}_m}^{-1}\mathbf{x}_{t-1})\right)}_{\widetilde{p}(\mathbf{x}_t|\mathbf{x}_{t-1})} \\
&\propto p(\mathbf{x}_0)\widetilde{p}\,(\mathbf{y}_0 \mid \mathbf{x}_0)\prod_{t=1}^{T}\widetilde{p}\,(\mathbf{y}_t \mid \mathbf{x}_t)\,\widetilde{p}\,(\mathbf{x}_t \mid \mathbf{x}_{t-1})\,, \qquad (4.23)
\end{aligned}
$$

proportional to several Gaussian distributions and some exponential terms. Thus, the resulting distribution represents a State-Space Model with additional penalty terms expressed by these exponential terms.

The optimal variational distribution $q^*(\mathbf{X})$ expresses a smoothing distribution, which is discussed in Appendix D [53, 54, 59]. Therefore, sampling can be achieved by Probabilistic Inference techniques from Bayesian smoothing or the Sequential Monte Carlo family. While transitions in latent space evolve linearly, the observational model expresses a nonlinear function. Therefore, Probabilistic Inference techniques should be applied which are capable of dealing with nonlinear transitions [52–54]. However, the penalty terms expressed by the exponential functions carry the risk of leading to non-Gaussian distributions. Therefore, techniques from Bayesian Inference based on Gaussian approximations, such as those in Appendix D, lead to inappropriate approximations [53]. For such cases of nonlinear non-Gaussian distributions, techniques from the family of Sequential Monte Carlo are applicable [4, 52, 53].

In this work, however, we assume that the penalty terms represent only zero-mean additive Gaussian noise, and thus the underlying distribution corresponds to a nonlinear Gaussian distribution. Therefore, this nonlinear Gaussian distribution is approximated by a particular functional form of the variational distribution

$$
q^*(\mathbf{X}) = \mathcal{N}(\mathbf{x}_0 \mid 0, \frac{\widetilde{\beta}_{\mathbf{0}}}{\widetilde{\alpha}_{\mathbf{0}}}\mathbf{I}) \quad \prod_{t=0}^{T}\mathcal{N}(\mathbf{y}_t \mid \mathbf{C}_t\widetilde{\boldsymbol{\mu}}_{\mathbf{U}}, \frac{\widetilde{\beta}_{\mathbf{y}}}{\widetilde{\alpha}_{\mathbf{y}}}\mathbf{I})\prod_{t=1}^{T}\mathcal{N}(\mathbf{x}_t \mid \widetilde{\boldsymbol{\mu}}_{\mathbf{a}}^T\mathbf{x}_{t-1}, \frac{\widetilde{\beta}_{\mathbf{x}}}{\widetilde{\alpha}_{\mathbf{x}}}\mathbf{I}), \qquad (4.24)
$$

representing a nonlinear Gaussian State-Space Model. The sampling on this SSM is again performed by techniques from the SMC family, e.g., Sequential Importance Resampling (see Appendix D). In addition, however, Bayesian Inference techniques, such as Cubature Kalman smoothing (see Appendix D), are now applicable.

Optimal Variational Parameters And Hyperparameters

The variational parameters corresponding to the inducing inputs and the hyperparameters representing the kernel parameters are collected in $\boldsymbol{\theta}$. For optimization, both can be jointly maximized using gradient descent techniques [50, 54, 59]. However, optimization of these parameters results in an Empirical Bayesian learning framework, as discussed in Section 2.3.

On the one hand, the ELBO, given by

$$\begin{aligned}\mathcal{L}_{\text{ELBO}}(q(\boldsymbol{\Theta}), \boldsymbol{\theta}) = c - \mathrm{KL}(q(\mathbf{U}) \parallel p(\mathbf{U})) - \left\langle \frac{N\lambda_{\mathbf{y}}}{2} \mathrm{Tr}(\mathbf{D}_{:}) \right\rangle_{q(\mathbf{X}, \lambda_{\mathbf{y}})} \\ + \left\langle \log \prod_{t=0}^{T} \mathcal{N}(\mathbf{y}_t \mid \mathbf{C}_t \mathbf{U}, \lambda_{\mathbf{y}}^{-1} \mathbf{I}) \right\rangle_{q(\mathbf{X}, \mathbf{U}, \lambda_{\mathbf{y}})},\end{aligned}$$

can be optimized directly using numerical calculation techniques. The constant c includes all terms not depending on the parameters $\boldsymbol{\theta}$. On the other hand, it is possible to reverse the applied Jensen inequality. Since in the VBE step (see Section 2.3), the optimal variational distributions $q^*(\boldsymbol{\Theta})$ were found, the distance between the true marginal log-likelihood and the ELBO was minimized. Under the assumption of a satisfying minimization, the

Jensen inequality can then be reversed [50]. The reversion results in

$$\begin{aligned}\mathcal{L}_{\text{ELBO}}(q(\boldsymbol{\Theta}),\boldsymbol{\theta}) &= c - \left\langle \frac{N\lambda_{\mathbf{y}}}{2}\mathrm{Tr}(\mathbf{D}_{:})\right\rangle_{q^*(\mathbf{X},\lambda_{\mathbf{y}})} \\ &\quad + \sum_{n=1}^{N}\left\langle \log p(\mathbf{U}_n)\exp\left(\left\langle \log\mathcal{N}(\mathbf{Y}_n \mid \mathbf{C}_{:}\mathbf{U}_n,\lambda_{\mathbf{y}}^{-1}\mathbf{I})\right\rangle_{q^*(\mathbf{X},\lambda_{\mathbf{y}})}\right)\right\rangle_{q^*(\mathbf{U}_n)} \\ &= c + \frac{N\widetilde{\alpha}_{\mathbf{y}}}{2\widetilde{\beta}_{\mathbf{y}}}\left\langle \mathrm{Tr}(\mathbf{D}_{:})\right\rangle_{q^*(\mathbf{X})} \\ &\quad + \sum_{n=1}^{N}\log \underbrace{\int p(\mathbf{U}_n)\exp\left(\left\langle \log\mathcal{N}(\mathbf{Y}_n \mid \mathbf{C}_{:}\mathbf{U}_n,\lambda_{\mathbf{y}}^{-1}\mathbf{I})\right\rangle_{q^*(\mathbf{X},\lambda_{\mathbf{y}})}\right)\mathrm{d}\mathbf{U}_n}_{=\zeta_2},\end{aligned}$$

where the logarithm function is drawn out of the integral. Hence, the optimal variational distribution $q^*(\mathbf{U})$ has been eliminated. The inner part, represented by ζ_2, corresponds to a quadratic form and is thus analytically solvable

$$\begin{aligned}\zeta_2 &= \int p(\mathbf{U}_n)\exp\left(\left\langle \log\mathcal{N}(\mathbf{Y}_n \mid \mathbf{C}_{:}\mathbf{U}_n,\lambda_{\mathbf{y}}^{-1}\mathbf{I})\right\rangle_{q^*(\mathbf{X},\lambda_{\mathbf{y}})}\right)\mathrm{d}\mathbf{U}_n \\ &= \int (2\pi)^{-\frac{T}{2}}\left|\mathbf{K}_{\mathbf{ZZ}}\right|^{-\frac{1}{2}}(2\pi)^{-\frac{T}{2}}\left\langle \lambda_{\mathbf{y}}^{\frac{T}{2}}\right\rangle_{q^*(\lambda_{\mathbf{y}})} \\ &\quad \exp\Bigg(-\frac{1}{2}\Bigg(\mathbf{U}_n^T\mathbf{K}_{\mathbf{ZZ}}\mathbf{U}_n + \frac{\widetilde{\alpha}_{\mathbf{y}}}{\widetilde{\beta}_{\mathbf{y}}}\mathbf{Y}_n^T\mathbf{Y}_n - \frac{\widetilde{\alpha}_{\mathbf{y}}}{\widetilde{\beta}_{\mathbf{y}}}\mathbf{Y}_n^T\Psi_5\mathbf{K}_{\mathbf{ZZ}}^{-1}\mathbf{U}_n \\ &\quad - \frac{\widetilde{\alpha}_{\mathbf{y}}}{\widetilde{\beta}_{\mathbf{y}}}\mathbf{U}_n^T\mathbf{K}_{\mathbf{ZZ}}^{-1}\Psi_5^T\mathbf{Y}_n + \frac{\widetilde{\alpha}_{\mathbf{y}}}{\widetilde{\beta}_{\mathbf{y}}}\mathbf{U}_n^T\mathbf{K}_{\mathbf{ZZ}}^{-1}\Psi_6\mathbf{K}_{\mathbf{ZZ}}^{-1}\mathbf{U}_n\Bigg)\Bigg)\mathrm{d}\mathbf{U}_n,\end{aligned}$$

where $\Psi_5 = \left\langle \mathbf{K}_{:\mathbf{Z}}\right\rangle_{q^*(\mathbf{X})}$ and $\Psi_6 = \left\langle \mathbf{K}_{\mathbf{Z}:}\mathbf{K}_{:\mathbf{Z}}\right\rangle_{q^*(\mathbf{X})}$ correspond to two sufficient statistics [50]. Due to the quadratic nature and the corresponding Gaussian integral, ζ_2 equals

$$\zeta_2 = \frac{\left\langle \lambda_{\mathbf{y}}^{\frac{T}{2}}\right\rangle_{q^*(\lambda_{\mathbf{y}})}\left|\mathbf{K}_{\mathbf{ZZ}}\right|^{\frac{1}{2}}}{(2\pi)^{-\frac{T}{2}}\left|\frac{\widetilde{\alpha}_{\mathbf{y}}}{\widetilde{\beta}_{\mathbf{y}}}\Psi_6 + \mathbf{K}_{\mathbf{ZZ}}\right|^{\frac{1}{2}}}\exp\left(-\frac{1}{2}\mathbf{Y}_n^T\boldsymbol{\Lambda}\mathbf{Y}_n\right),$$

where $\boldsymbol{\Lambda} = \frac{\widetilde{\alpha}_{\mathbf{y}}}{\widetilde{\beta}_{\mathbf{y}}}\mathbf{I} - \frac{\widetilde{\alpha}_{\mathbf{y}}^2}{\widetilde{\beta}_{\mathbf{y}}^2}\Psi_5\left(\frac{\widetilde{\alpha}_{\mathbf{y}}}{\widetilde{\beta}_{\mathbf{y}}}\Psi_6 + \mathbf{K}_{\mathbf{ZZ}}\right)^{-1}\Psi_5^T$ [50]. Thus, an alternative loss function is

eventually derived, given by

$$\mathcal{L}_{\text{ELBO}}(q^*(\boldsymbol{\Theta}), \boldsymbol{\theta}) = c - \frac{N\widetilde{\alpha}_{\mathbf{y}}}{2\widetilde{\beta}_{\mathbf{y}}} \left\langle \text{Tr}(\mathbf{D}_{:}) \right\rangle_{q^*(\mathbf{X})} + \frac{N}{2} \log|\mathbf{K}_{\mathbf{ZZ}}| \\ - \frac{N}{2} \log \left| \frac{\widetilde{\alpha}_{\mathbf{y}}}{\widetilde{\beta}_{\mathbf{y}}} \Psi_6 + \mathbf{K}_{\mathbf{ZZ}} \right| - \frac{1}{2} \text{Tr}\left(\boldsymbol{\Lambda}\mathbf{Y}\mathbf{Y}^T\right). \tag{4.25}$$

This loss function still represents an ELBO. However, it now depends directly on the variational parameters and hence on the inducing inputs $\mathbf{Z}$ no longer on the induced variables $\mathbf{U}$. Similarly, as above, all terms not relevant for the optimization of the variational parameters and the kernel hyperparameters are summarized in c. Numerical calculation methods then obtain the gradients and gradient descent techniques are applied [50].

Reproducing Demonstration

For the reconstruction of the demonstration in the observation space, the latent space is considered first. The linear state sequences $\mathbf{X}$ in the latent space are inferred based on the assumed variational distribution $q^*(\mathbf{X})$. $q^*(\mathbf{X})$ represents a nonlinear Gaussian SSM, which is defined by the natural parameters of the variational distribution $q^*(\mathbf{U})$, $q^*(\mathbf{A})$, $q^*(\lambda_{\mathbf{y}})$, $q^*(\lambda_{\mathbf{0}}), q^*(\lambda_{\mathbf{x}})$ and the variational parameters and hyperparameters $\boldsymbol{\theta}$. The sequences $\mathbf{X}^* \sim q^*(\mathbf{X})$ are then sampled in latent space using Probabilistic Inference techniques (see Appendix D). These samples and the remaining optimized parameters are applied to reconstruct trajectories in the observation space. Similar to GP-DMD, the predictive distribution (see Section 2.4) is considered

$$p(\mathbf{G}^* \mid \mathbf{X}^*, \mathbf{X}) = \mathcal{N}\Big(\mathbf{K}_{\mathbf{X}^*\mathbf{Z}}\mathbf{K}_{\mathbf{Z},\mathbf{Z}}^{-1}\widetilde{\boldsymbol{\mu}}_{\mathbf{U}}, \mathbf{K}_{\mathbf{X}^*\mathbf{X}^*} - \mathbf{K}_{\mathbf{X}^*\mathbf{Z}}\mathbf{K}_{\mathbf{Z},\mathbf{Z}}^{-1}\mathbf{K}_{\mathbf{Z}\mathbf{X}^*}\Big),$$

predicting new function values $\mathbf{G}^*$ in the observation space for the corresponding inputs $\mathbf{X}^*$. Unlike GP-DMD, however, the prediction of Bayesian GP-DMD shows a dependence on the induced pairs $(\mathbf{U}, \mathbf{Z})$ and thus not on the given observation $\mathbf{Y}$ [54, 59]. This independence on the observations $\mathbf{Y}$ follows from the conditional independence implied by the induced pairs $(\mathbf{U}, \mathbf{Z})$ [50, 77]. As a result, no labels $\mathbf{Y}$ need to be collected in order to reproduce the trajectories. Instead, by optimizing the inducing inputs $\mathbf{Z}$ and inferring the mean $\widetilde{\boldsymbol{\mu}}_{\mathbf{U}}$ (see Equation (4.21)), these required labels are directly obtained for the reconstruction.

5. Probabailistic Dynamic Mode Primitives

GP-DMD and Bayesian GP-DMD, two central concepts of this thesis, were introduced in the previous Chapter 4. These concepts take a dual perspective on Koopman Theory and thus on the DMD family, utilizing Gaussian Process State-Space Models. On the one hand, using an underlying GP-SSM structure provides a probabilistic perspective that naturally accounts for uncertainty and variability. On the other hand, the use of GPs results in a nonparametric Bayesian framework. In both frameworks, the observation model is assumed to be represented by a GP, while a linear stationary operator expresses the transition model. However, both approaches fail to learn from multiple data. Especially in the field of movement primitives, this is a fundamental drawback since multiple trajectories are commonly given [7–10,20]. This chapter, therefore, extends the GP-DMD and Bayesian GP-DMD frameworks with a hierarchical structure that takes multiple trajectories into account. These considerations eventually give rise to the *Probabilistic Dynamic Mode Primitive (Pro-DMP)* and the *Bayesian Dynamic Mode Primitive (Bayesian-DMP)*, two novel movement primitives.

In Section 5.1, the Probabilistic Dynamic Mode Primitive is introduced, an extension of the GP-DMD framework. In the field of movement primitives, the variability in the movements is usually provided by multiple trajectories [7–10]. In the context of GP-DMD, this variability can be incorporated either in the GP-based observational model or as a distribution in latent space. The Pro-DMP framework adopts the latter option since it aims at expressing the given variability in the data in the inferred trajectories in the latent space. Therefore, based on Probabilistic Movement Primitives, a popular movement primitive framework [10, 14, 15, 18], a hierarchical structure is utilized and incorporated into the existing GP-DMD framework. This structure eventually results in an EM-like framework capable of handling multiple trajectories and representing a new type of movement primitive.

Like GP-DMD, however, the approach carries the risk of overfitting in cases where the latent space has a higher dimensionality than the observation space (see Sections 2.2 and 4.2).

In Koopman Theory, the invariant subspace can exhibit much higher dimensionality, providing an essential drawback of Probabilistic Dynamic Mode Primitives. For this reason, Section 5.2 presents a possible combination of Pro-DMP and the fully Bayesian approach Bayesian GP-DMD, leading to the Bayesian Dynamic Mode Primitive (Bayesian-DMP) framework. This framework considers the hierarchical structure in a fully Bayesian manner. Thus, it theoretically combines the advantages of Pro-DMP and Bayesian GP-DMD. On the one hand, the handling of multiple trajectories. On the other hand, the mitigation of overfitting and the provision of approximated posterior distributions.

5.1. Probabailistic Dynamic Mode Primitives

In the field of movement primitives, where multiple trajectories generally provide the variability of a movement of interest [7–10], the use of the GP-DMD framework is limited. Therefore, this section presents *Probabilistic Dynamic Mode Primitive (Pro-DMP)*, a hierarchical extension of the GP-DMD. The hierarchical structure of the Probabilistic Movement Primitives (ProMPs) inspires the proposed framework [10, 14, 15, 18]. In contrast to Section 4.2, S-independent trajectories $\mathbf{Y} = [\mathbf{Y}^0, \cdots, \mathbf{Y}^S]$ are given, with each observation sequence taking the form $\mathbf{Y}^s \in \mathbb{R}^{N \times T}$. Thus, the following Probability Density Model results in

$$p(\mathbf{Y}, \mathbf{X}, \mathbf{A}) = \prod_{s=1}^{S} p(\mathbf{Y}^s, \mathbf{X}^s, \mathbf{A}^s),$$

assuming that the trajectories are independently observed. This independence results in S-independent representations in the latent space $\mathbf{X} = [\mathbf{X}^0 \in \mathbb{R}^{M \times T}, \cdots, \mathbf{X}^S \in \mathbb{R}^{M \times T}]$. A distribution can be modeled over these resulting representations, hence, expressing the variability of the linear dynamics in the latent space. Therefore, the distribution is

transformed into a hierarchical structure

$$
\begin{aligned}
p(\mathbf{Y}, \mathbf{X}) &= \prod_{s=1}^{S} \int p(\mathbf{Y}^s, \mathbf{X}^s, \mathbf{A}^s)\, \mathrm{d}\mathbf{A}^s \\
&= \prod_{s=1}^{S} p(\mathbf{Y}^s \mid \mathbf{X}^s) \prod_{m=1}^{M} \int p(\mathbf{X}^s \mid \mathbf{a}_m) p(\mathbf{a}_m^s)\, \mathrm{d}\mathbf{a}_m^s \\
&= \prod_{s=1}^{S} \Bigg(\mathcal{N}(\mathbf{Y}^s \mid \mathbf{0}, \widetilde{\mathbf{K}}_{\mathbf{X}^s\mathbf{X}^s}(\boldsymbol{\theta})) \mathcal{N}(\mathbf{x}_1^s \mid 0, \lambda_{\mathbf{0}}^{-1}\mathbf{I}) \\
&\qquad \prod_{m=1}^{M} \int \mathcal{N}(\mathbf{X}_m^{1^s} \mid \mathbf{a}_m^{s^T} \mathbf{X}^{0^s}, \lambda_{\mathbf{x}}^{-1}\mathbf{I}) \mathcal{N}(\mathbf{a}_m^s \mid \boldsymbol{\mu}_{\mathbf{a}_m}, \boldsymbol{\Lambda}_{\mathbf{a}_m}^{-1})\, \mathrm{d}\mathbf{a}_m^s \Bigg).
\end{aligned}
\tag{5.1}
$$

Here, the prior distributions of the respective linear operators $\mathbf{A}^0, \cdots, \mathbf{A}^S$ are extended by common mean vectors and precision matrices. These variables express the distribution of the linear operator $\mathbf{A}$ and therefore the linear dynamics variability in the latent space. The marginalization expresses the variability of all possible linear operator and hence achieves a hierarchical structure. The graphical model in fig. 5.1 visualizes the probability density model of Equation (5.1). The blue shaded nodes and the white nodes represent the given trajectories $\mathbf{Y}$ and all unknown latent variables, respectively. In the larger box outlined in blue, the independent consideration of the individual trajectories is shown, where a hierarchical structure is provided due to the common natural parameters in the smaller box outlined in blue.

Learning From Demonstration

Like GP-DMDs (see Section 4.1), estimating the optimal parameters of interest is considered an optimization procedure in logarithmic space. Therefore, the Probability Density Model formalizes from Equation (5.1) to

$$
\begin{aligned}
\mathcal{L}(\boldsymbol{\Theta}, \boldsymbol{\theta}) = &\sum_{s=1}^{S} \log \Big(\mathcal{N}(\mathbf{Y}^s \mid \mathbf{0}, \widetilde{\mathbf{K}}_{\mathbf{X}^s\mathbf{X}^s}(\boldsymbol{\theta})) \mathcal{N}(\mathbf{x}_1^s \mid 0, \lambda_{\mathbf{0}}^{-1}\mathbf{I}) \Big) \\
&+ \underbrace{\sum_{s=1}^{S} \sum_{m=1}^{M} \log \int \mathcal{N}(\mathbf{X}_m^{1^s} \mid \mathbf{a}_m^{s^T} \mathbf{X}^{0^s}, \lambda_{\mathbf{x}}^{-1}\mathbf{I}) \mathcal{N}(\mathbf{a}_m^s \mid \boldsymbol{\mu}_{\mathbf{a}_m}, \boldsymbol{\Lambda}_{\mathbf{a}_m}^{-1})\, \mathrm{d}\mathbf{a}_m^s}_{=\widetilde{\mathcal{L}}(\mathbf{X}, \boldsymbol{\theta})},
\end{aligned}
$$

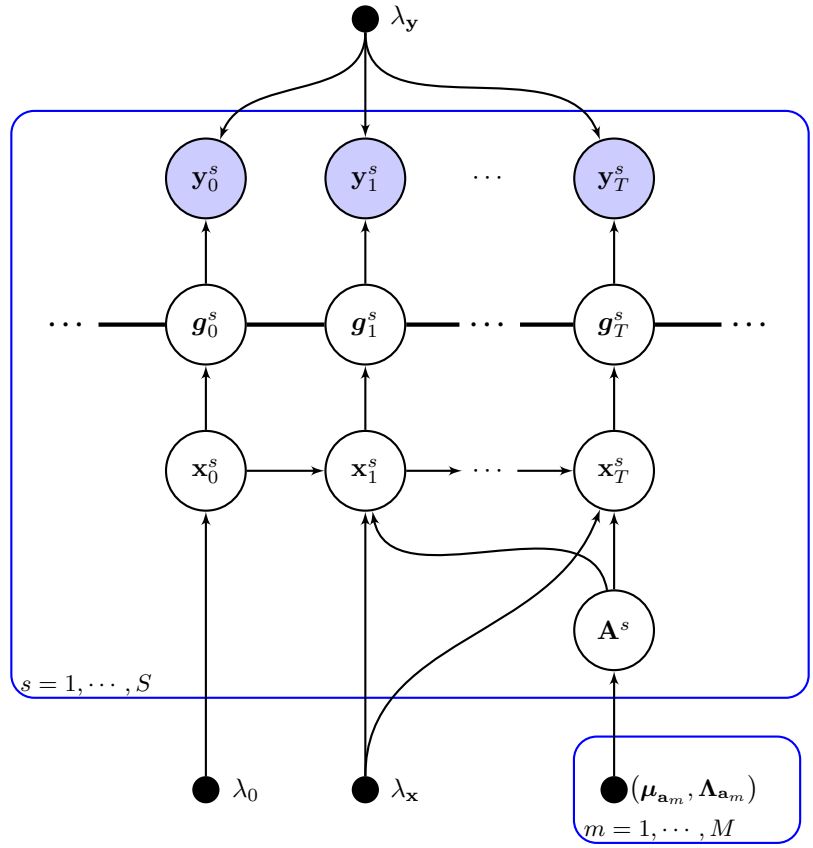


Figure 5.1.: The graphical model visualizes the Probability Density Model of the derived Probabilistic Dynamic Mode Primitive. The hierarchical structure for S given trajectories is represented by the larger blue outlined box. The blue shaded nodes represent the given observations $\mathbf{y}_0^s, \cdots, \mathbf{y}_T^s$ for each trajectory. The white nodes correspond to the unknown latent variables. By $\mathbf{x}_0^s, \cdots, \mathbf{x}_T^s$, the sequence of states of interest in the latent space is given. $\mathbf{g}_0^s, \cdots, \mathbf{g}_T^s$ describe the outputs of the Gaussian process function. The linear operator corresponding to each trajectory is given by $\mathbf{A}^s$. The thick black line denotes an auto-regressive structure within the Gaussian process, which means that $\mathbf{g}_t^s$ depends on $\mathbf{g}_0^s, \cdots, \mathbf{g}_{t-1}^s$ and $\mathbf{x}_0^s, \cdots, \mathbf{x}_t^s$. The small black nodes correspond to the hyperparameters of the given Probability Density Model. In this case, they represent the precision values $\lambda_{\mathbf{0}}, \lambda_{\mathbf{x}}$ and $\lambda_{\mathbf{y}}$, the mean vectors $\boldsymbol{\mu}_{\mathbf{a}_m}$, and precision matrices $\boldsymbol{\Lambda}_{\mathbf{a}_m}$ for $m = 1, \cdots, M$.

where $\boldsymbol{\Theta} = [\mathbf{X}, \lambda_{\mathbf{0}}, \lambda_{\mathbf{x}}, \lambda_{\mathbf{y}}, \boldsymbol{\mu}_{\mathbf{a}_1}, \cdots, \boldsymbol{\mu}_{\mathbf{a}_M}, \boldsymbol{\Lambda}_{\mathbf{a}_1}, \cdots, \boldsymbol{\Lambda}_{\mathbf{a}_M}]$. The former part merely represents S-independent observation models in analogy to GP-DMD. However, the second part results in a logarithm over an integral due to dependencies arising from the hierarchical structure [2–4, 6]. Considering a variational distribution $q(\mathbf{A})$ and applying Jensens' inequality to $\widetilde{\mathcal{L}}(\mathbf{X}, \boldsymbol{\theta})$ leads to

$$\begin{aligned}\widetilde{\mathcal{L}}(\boldsymbol{\Theta}, \boldsymbol{\theta}) &= \sum_{s=1}^{S} \sum_{m=1}^{M} \log \int \mathcal{N}(\mathbf{X}_m^{1^s} \mid \mathbf{a}_m^{s^T} \mathbf{X}^{0^s}, \lambda_{\mathbf{x}}^{-1} \mathbf{I}) \mathcal{N}(\mathbf{a}_m^s \mid \boldsymbol{\mu}_{\mathbf{a}_m}, \boldsymbol{\Lambda}_{\mathbf{a}_m}^{-1}) \, \mathrm{d}\mathbf{a}_m^s \\ &\geq \sum_{s=1}^{S} \sum_{m=1}^{M} \int q(\mathbf{A}) \log \frac{\mathcal{N}(\mathbf{X}_m^{1^s} \mid \mathbf{a}_m^{s^T} \mathbf{X}^{0^s}, \lambda_{\mathbf{x}}^{-1} \mathbf{I}) \mathcal{N}(\mathbf{a}_m^s \mid \boldsymbol{\mu}_{\mathbf{a}_m}, \boldsymbol{\Lambda}_{\mathbf{a}_m}^{-1})}{q(\mathbf{A})} \, \mathrm{d}\mathbf{a}_m^s,\end{aligned}$$

and an Evidence Lower Bound

$$\begin{aligned}\mathcal{L}(\boldsymbol{\Theta}, \boldsymbol{\theta}) \geq \sum_{s=1}^{S} \Bigg(& \log \left(\mathcal{N}(\mathbf{Y}^s \mid \mathbf{0}, \widetilde{\mathbf{K}}_{\mathbf{X}^s \mathbf{X}^s}(\boldsymbol{\theta})) \mathcal{N}(\mathbf{x}_1^s \mid 0, \lambda_{\mathbf{0}}^{-1} \mathbf{I}) \right) \\ & + \sum_{m=1}^{M} \int q(\mathbf{A}) \log \frac{\mathcal{N}(\mathbf{X}_m^{1^s} \mid \mathbf{a}_m^{s^T} \mathbf{X}^{0^s}, \lambda_{\mathbf{x}}^{-1} \mathbf{I}) \mathcal{N}(\mathbf{a}_m^s \mid \boldsymbol{\mu}_{\mathbf{a}_m}, \boldsymbol{\Lambda}_{\mathbf{a}_m}^{-1})}{q(\mathbf{A})} \, \mathrm{d}\mathbf{a}_m^s \Bigg) \\ \stackrel{\text{def}}{=} & \mathcal{L}_{\text{ELBO}}(q(\mathbf{A}), \boldsymbol{\Theta}, \boldsymbol{\theta}). \end{aligned} \tag{5.2}$$

As mentioned in Section 2.2, the distance between the ELBO and the marginal log-likelihood becomes minimal or equal to zero when the KL divergence between the true posterior $p(\mathbf{A} \mid \mathbf{X}, \boldsymbol{\theta})$ and the variational distribution $q(\mathbf{A})$ becomes minimal. Subsequently, the parameters of interest $\boldsymbol{\Theta}$ and $\boldsymbol{\theta}$ are optimized using gradient based methods. Hence, the optimization procedure of the Pro-DMP framework takes the form of an EM algorithm and is expressed as

$$\begin{aligned} q^*(\mathbf{A}) = & \quad \arg\max_{q(\mathbf{A})} \mathcal{L}_{\text{ELBO}}(q(\mathbf{A}), \boldsymbol{\Theta}, \boldsymbol{\theta}), \\ \boldsymbol{\Theta}^*, \boldsymbol{\theta}^* = & \quad \arg\max_{\boldsymbol{\Theta}, \boldsymbol{\theta}} \mathcal{L}_{\text{ELBO}}(q^*(\mathbf{A}), \boldsymbol{\Theta}, \boldsymbol{\theta}), \end{aligned}$$

where the first and second steps correspond to an expectation step and maximization step, respectively [36, 37]. These two steps are performed alternately.

Expectation Step

The expectation step finds an estimate for the optimal variational distribution $q^*(\mathbf{A})$ in analogy to Section 4.3. First, the derivative of the ELBO of Equation (5.2) is derived w.r.t. $q(\mathbf{A})$ and set equal to zero. Subsequently, the transformation into exponential space leads to

$$
\begin{aligned}
q^*(\mathbf{A}) &\propto \prod_{s=1}^{S}\prod_{m=1}^{M} \mathcal{N}(\mathbf{X}_m^{1^s} \mid \mathbf{a}_m^{s^T}\mathbf{X}^{0^s}, \lambda_{\mathbf{x}}^{-1}\mathbf{I})\mathcal{N}(\mathbf{a}_m^s \mid \boldsymbol{\mu}_{\mathbf{a}_m}, \boldsymbol{\Lambda}_{\mathbf{a}_m}^{-1}) \\
&\propto \prod_{s=1}^{S}\prod_{m=1}^{M} \exp\Bigg(-\frac{1}{2}\Bigg(2\begin{bmatrix}\mathbf{a}_m^{s^T} & \mathbf{X}_m^{1^s}\end{bmatrix}\begin{bmatrix}-\boldsymbol{\Lambda}_{\mathbf{a}_m}\boldsymbol{\mu}_{\mathbf{a}_m} \\ 0\end{bmatrix} \\
&\qquad + \begin{bmatrix}\mathbf{a}_m^{s^T} & \mathbf{X}_m^{1^s}\end{bmatrix}\begin{bmatrix}\lambda_{\mathbf{x}}\mathbf{X}^{0^s}\mathbf{X}^{0^s{}^T} + \boldsymbol{\Lambda}_{\mathbf{a}_m} & -\mathbf{X}^{0^s}\mathbf{I}\lambda_{\mathbf{x}} \\ -\lambda_{\mathbf{x}}\mathbf{I}\mathbf{X}^{0^s{}^T} & \lambda_{\mathbf{x}}\mathbf{I}\end{bmatrix}\begin{bmatrix}\mathbf{a}_m^s \\ \mathbf{X}_m^{1^s{}^T}\end{bmatrix}\Bigg)\Bigg),
\end{aligned}
$$

where completing the squares leads to an optimal distribution in terms of $S \times M$-independent Gaussian distributions. It factorizes accordingly

$$
q^*(\mathbf{A}) = \prod_{s=1}^{S}\prod_{m=1}^{M} q^*(\mathbf{a}_m^s) \propto \prod_{s=1}^{S}\prod_{m=1}^{M} \mathcal{N}\left(\mathbf{a}_m^s \mid \widetilde{\boldsymbol{\mu}}_{\mathbf{a}_m^s}, \widetilde{\boldsymbol{\Lambda}}_{\mathbf{a}_m^s}^{-1}\right),
$$

where the natural parameters are given by

$$
\widetilde{\boldsymbol{\mu}}_{\mathbf{a}_m^s} = \left(\boldsymbol{\Lambda}_{\mathbf{a}_m} + \lambda_{\mathbf{x}}\mathbf{X}^{0^s}\mathbf{X}^{0^s{}^T}\right)^{-1}\left(\lambda_{\mathbf{x}}\mathbf{X}^{0^s}\mathbf{X}_m^{1^s{}^T} + \boldsymbol{\Lambda}_{\mathbf{a}_m}\boldsymbol{\mu}_{\mathbf{a}_m}\right), \tag{5.3}
$$

$$
\widetilde{\boldsymbol{\Lambda}}_{\mathbf{a}_m^s} = \boldsymbol{\Lambda}_{\mathbf{a}_m} + \lambda_{\mathbf{x}}\mathbf{X}^{0^s}\mathbf{X}^{0^s{}^T}. \tag{5.4}
$$

The resulting form of the SN-independent Gaussian distribution arises naturally from the conjugacy property [4, 6]. The natural parameters for each Gaussian distribution also depend on the current estimate of the mean vector $\boldsymbol{\mu}_{\mathbf{a}_m}$ and the precision matrix $\boldsymbol{\Lambda}_{\mathbf{a}_m}$. The expectation step hence provides an estimate of the current natural parameters most likely responsible for the current sequences $\mathbf{X}$ in the latent space, based on the prior of $\boldsymbol{\mu}_{\mathbf{a}_m}$ and $\boldsymbol{\Lambda}_{\mathbf{a}_m}$.

Maximization Step

Considering the optimal posterior distribution $q^*(\mathbf{A})$ resulting from the E step, the ELBO of Equation (5.2) results in

$$\begin{aligned}\mathcal{L}_{\text{ELBO}}(q^*(\mathbf{A}), \boldsymbol{\Theta}, \boldsymbol{\theta}) = c - \frac{1}{2}\sum_{s=1}^{S}\Bigg[& -M\log|\lambda_{\mathbf{0}}| - (T-1)M\log|\lambda_{\mathbf{x}}| + \lambda_{\mathbf{0}}\mathbf{x}_0^{s^T}\mathbf{x}_0^s \\ & + N\log\left|\widetilde{\mathbf{K}}_{\mathbf{X}^s\mathbf{X}^s}(\boldsymbol{\theta})\right| + \operatorname{Tr}\left(\widetilde{\mathbf{K}}_{\mathbf{X}^s\mathbf{X}^s}(\boldsymbol{\theta})^{-1}\mathbf{Y}^s\mathbf{Y}^{s^T}\right) \\ & + \sum_{m=1}^{M}\Bigg(\operatorname{Tr}\left(\boldsymbol{\Lambda}_{\mathbf{a}_m}\left(\langle\mathbf{a}_m^s\rangle_{q^*(\mathbf{a}_m^s)} - \boldsymbol{\mu}_{\mathbf{a}_m}\right)\left(\langle\mathbf{a}_m^s\rangle_{q^*(\mathbf{a}_m^s)} - \boldsymbol{\mu}_{\mathbf{a}_m}\right)^T\right) \\ & + \lambda_{\mathbf{x}}\operatorname{Tr}\left(\left(\mathbf{X}_m^{1^s} - \langle\mathbf{a}_m^s\rangle_{q^*(\mathbf{a}_m^s)}\mathbf{X}^{0^s}\right)\left(\mathbf{X}_m^{1^s} - \langle\mathbf{a}_m^s\rangle_{q^*(\mathbf{a}_m^s)}\mathbf{X}^{0^s}\right)^T\right) \\ & - \log|\boldsymbol{\Lambda}_{\mathbf{a}_m}|\Bigg)\Bigg],\end{aligned}$$

where c summarizes all terms not depending on $\boldsymbol{\Theta}$ and $\boldsymbol{\theta}$. This ELBO thus extends the loss function of GP-DMD from Equation (4.4) regarding multiple trajectories. In the following, the gradients with respect to $\boldsymbol{\Theta}$ and $\boldsymbol{\theta}$ are considered. In analogy to GP-DMD (see Section 4.2), the optimal parameters are first determined, which provide a closed-form analytical solution. Subsequently, the loss function for the remaining parameters of interest is given, for which numerical gradient-based methods are needed.

For $m = 1, \cdots, M$, the optimal mean vector $\boldsymbol{\mu}_{\mathbf{a}_m}^*$ is given by

$$\boldsymbol{\mu}_{\mathbf{a}_m}^* = \frac{1}{S}\sum_{s=1}^{S}\widetilde{\boldsymbol{\mu}}_{\mathbf{a}_m^s}, \tag{5.5}$$

resulting from the derivative of $\mathcal{L}_{\text{ELBO}}(q^*(\mathbf{A}), \boldsymbol{\Theta}, \boldsymbol{\theta})$ w.r.t. to $\boldsymbol{\mu}_{\mathbf{a}_m}$. It expresses the mean value over all current mean estimates $\widetilde{\boldsymbol{\mu}}_{\mathbf{a}_m^s}$ for each movement $\mathbf{X}^s$. The corresponding optimal precision matrix $\boldsymbol{\Lambda}_{\mathbf{a}_m}^*$ is

$$\boldsymbol{\Lambda}_{\mathbf{a}_m}^* = \left(\sum_{s=1}^{S}\left(\left(\widetilde{\boldsymbol{\mu}}_{\mathbf{a}_m^s} - \boldsymbol{\mu}_{\mathbf{a}_m}\right)\left(\widetilde{\boldsymbol{\mu}}_{\mathbf{a}_m^s} - \boldsymbol{\mu}_{\mathbf{a}_m}\right)^T + \widetilde{\boldsymbol{\Lambda}}_{\mathbf{a}_m^s}^{-1}\right) + \mathbf{W}_m^{-1}\right)^{-1}\Big(S + \nu_m - M - 1\Big). \tag{5.6}$$

The parameters $\nu_m^0 \geq M+1$ and $\mathbf{W}_m$ are the degrees of freedom and the scale matrix, respectively, resulting from an assumed Wishart distribution $\mathcal{W}\left(\mathbf{\Lambda}_{\mathbf{a}_m} \mid \mathbf{W}_m, \nu_m\right)$ (see Appendix A). The optimal precision parameters $\lambda_{\mathbf{x}}^*$ and $\lambda_{\mathbf{0}}^*$ are calculated in a similar way as in Section 4.2. The gradient of the ELBO w.r.t. $\lambda_{\mathbf{x}}$ and $\lambda_{\mathbf{0}}$, respectively, leads to the closed-form solutions

$$\lambda_{\mathbf{0}}^* = \frac{SM + 2\alpha_{\mathbf{0}} - 2}{\sum_{s=1}^{S} \mathbf{x}_0^{s^T} \mathbf{x}_0^s + 2\beta_{\mathbf{0}}}, \tag{5.7}$$

$$\lambda_{\mathbf{x}}^* = \frac{S(T-1)M + 2\alpha_{\mathbf{x}} - 2}{\sum_{s=1}^{S} \sum_{m=1}^{M} \mathrm{Tr}\left(\mathbf{X}_m^{1^s} \mathbf{X}_m^{1^{s^T}} - 2\widetilde{\boldsymbol{\mu}}_{\mathbf{a}}^T \mathbf{X}^{0^s} \mathbf{X}_m^{1^{s^T}} + \mathbf{X}^{0^s} \mathbf{X}^{0^{s^T}} \left(\widetilde{\boldsymbol{\mu}}_{\mathbf{a}} \widetilde{\boldsymbol{\mu}}_{\mathbf{a}}^T + \widetilde{\mathbf{\Lambda}}_{\mathbf{a}}^{-1}\right)\right) + 2\beta_{\mathbf{x}}}, \tag{5.8}$$

assuming two additional Gamma distributions $\mathrm{Gam}\left(\lambda_{\mathbf{0}} \mid \alpha_{\mathbf{0}}, \beta_{\mathbf{0}}\right)$ and $\mathrm{Gam}\left(\lambda_{\mathbf{x}} \mid \alpha_{\mathbf{x}}, \beta_{\mathbf{x}}\right)$ as prior distributions for the precision values.

In order to optimize the remaining parameters $\mathbf{X}$, $\lambda_{\mathbf{y}}$ and $\boldsymbol{\theta}$, where no closed-form solution generally exists, the corresponding loss function is given by

$$\begin{aligned}
\mathcal{L}_{\text{ELBO}}(\mathbf{X}, \lambda_{\mathbf{y}}, \boldsymbol{\theta}) = c - \frac{1}{2} \sum_{s=1}^{S} \Bigg[& \lambda_{\mathbf{0}}^* \mathbf{x}_0^{s^T} \mathbf{x}_0^s + (\alpha_{\mathbf{y}} - 1) \log \lambda_{\mathbf{y}} + \beta_{\mathbf{y}} \lambda_{\mathbf{y}} \\
& + \sum_{m=1}^{M} \Bigg[\mathbf{X}^{0^s} \mathbf{X}^{0^{s^T}} \widetilde{\mathbf{\Lambda}}_{\mathbf{a}}^{-1} + \lambda_{\mathbf{x}}^* \mathrm{Tr}\left(\left(\mathbf{X}_m^{1^s} - \widetilde{\boldsymbol{\mu}}_{\mathbf{a}} \mathbf{X}^{0^s}\right) \left(\mathbf{X}_m^{1^s} - \widetilde{\boldsymbol{\mu}}_{\mathbf{a}} \mathbf{X}^{0^s}\right)^T\right) \Bigg] \\
& + N \log \left| \widetilde{\mathbf{K}}_{\mathbf{X}^s \mathbf{X}^s}(\boldsymbol{\theta}) \right| + \mathrm{Tr}\left(\widetilde{\mathbf{K}}_{\mathbf{X}^s \mathbf{X}^s}(\boldsymbol{\theta})^{-1} \mathbf{Y}^s \mathbf{Y}^{s^T}\right) \Bigg],
\end{aligned} \tag{5.9}$$

where $\widetilde{\mathbf{K}}_{\mathbf{X}^s \mathbf{X}^s}(\boldsymbol{\theta}) = \mathbf{K}_{\mathbf{X}^s \mathbf{X}^s}(\boldsymbol{\theta}) + \lambda_{\mathbf{y}}^{-1} \mathbf{I}$. The gradients are determined using numerical computational methods. Hence, this loss function in Equation (5.9) extends the one of GP-DMD from Equation (4.9) concerning multiple trajectories.

Reproducing Demonstration

The trajectories in the latent space are first considered to reconstruct demonstrations in the observation space subsequently based on the optimized parameters $\mathbf{\Theta}^*$ and $\boldsymbol{\theta}^*$. As

in ProMPs [10, 14, 15, 18], an optimal demonstration $\mathbf{X}^* \in \mathbb{R}^{M \times T}$ in the latent space is sampled from

$$
\begin{aligned}
p(\mathbf{X}^*) &= p(\mathbf{x}_0^*) \prod_{m=1}^{M} p\left(\mathbf{X}_m^*\right) \\
&= p(\mathbf{x}_0^*) \prod_{m=1}^{M} \int p\big(\mathbf{X}_m^{1^*} \mid \mathbf{X}^{0^{*^T}} \mathbf{a}_m^*, \lambda_{\mathbf{x}}^{-1}\mathbf{I}\big) p\big(\mathbf{a}_m^* \mid \boldsymbol{\mu}_{\mathbf{a}_m}, \boldsymbol{\Lambda}_{\mathbf{a}_m}^{-1}\big)\, \mathrm{d}\mathbf{a}_m \\
&= p(\mathbf{x}_0^*) \prod_{m=1}^{M} p\big(\mathbf{X}_m^{1^*} \mid \mathbf{X}^{0^{*^T}} \boldsymbol{\mu}_{\mathbf{a}_m}, \lambda_{\mathbf{x}}^{-1}\mathbf{I} + \mathbf{X}^{0^T} \boldsymbol{\Lambda}_{\mathbf{a}_m}^{-1} \mathbf{X}^0\big), \\
&= p(\mathbf{x}_0^*) \prod_{m=1}^{M} \prod_{t=1}^{T} p\big(\mathbf{x}_{m,t}^* \mid \mathbf{x}_{t-1}^{*^T} \boldsymbol{\mu}_{\mathbf{a}_m}, \lambda_{\mathbf{x}}^{-1}\mathbf{I} + \mathbf{x}_{0,\cdots,t-1}^T \boldsymbol{\Lambda}_{\mathbf{a}_m}^{-1} \mathbf{x}_{0,\cdots,t-1}\big),
\end{aligned}
$$

where $\mathbf{x}_{m,t}^*$ corresponds to the mth element of $\mathbf{x}_t^*$ vector. This distribution accounts for learned variability in the linear operator $\mathbf{A}$ and hence in the resulting state sequences $\mathbf{X}^*$. Like GP-DMD (see Section 4.2), the predictive distribution is utilized to reconstruct demonstrations in the observation space based on drawn state sequences.

5.2. Bayesian Gaussian Process Dynamic Mode Decomposition

The introduced Pro-DMP framework, like GP-DMD, suffers from the problem of overfitting. Therefore, this section proposes a combination of Pro-DMPs and the fully Bayesian approach Bayesian GP-DMDs, resulting in *Bayesian Dynamic Mode Primitives (Bayesian-DMPs)*. Bayesian-DMPs combine the advantages of both frameworks. First, the handling of multiple trajectories and hence the extension of Bayesian GP-DMDs to a framework applicable in the domain of movement primitives. Second, the mitigation of overfitting and the provision of approximated posterior distributions. Like GP-DMD, sparse-GP techniques are applied to circumvent the auto-regressive structure induced by the Gaussian Process observation model. As a result of the multiple given demonstrations, S-inducing pairs are considered $\{(\mathbf{U}_s, \mathbf{Z})\}_{s=1}^{S}$, where all inducing variables have the same inducing input $\mathbf{Z}$. The corresponding outputs of the observational model are represented by $\mathbf{G} = [\mathbf{G}^1 \in \mathbb{R}^{N \times T}, \cdots, \mathbf{G}^S \in \mathbb{R}^{N \times T}]$. The Probability Density Model considered in the

Bayesian-DMP factorizes accordingly

$$\begin{aligned} p(\mathbf{Y}, \mathbf{\Theta}) &= p(\mathbf{Y}, \mathbf{X}, \mathbf{G}, \mathbf{U}, \mathbf{A}^s, \boldsymbol{\mu}_{\mathbf{a}_1}, \cdots, \boldsymbol{\mu}_{\mathbf{a}_M}, \mathbf{\Lambda}_{\mathbf{a}_1}, \cdots, \mathbf{\Lambda}_{\mathbf{a}_M}, \lambda_{\mathbf{x}}, \lambda_{\mathbf{0}}, \lambda_{\mathbf{y}}) \\ &= p(\mathbf{Y}, \mathbf{G}, \mathbf{U}, \lambda_{\mathbf{y}} \mid \mathbf{X}) p(\mathbf{X}, \mathbf{A}, \boldsymbol{\mu}_{\mathbf{a}_1}, \cdots, \boldsymbol{\mu}_{\mathbf{a}_M}, \mathbf{\Lambda}_{\mathbf{a}_1}, \cdots, \mathbf{\Lambda}_{\mathbf{a}_M}, \lambda_{\mathbf{x}}, \lambda_{\mathbf{0}}) \\ &= p(\mathbf{Y}, \mathbf{G}, \mathbf{U}, \lambda_{\mathbf{y}} \mid \mathbf{X}) p(\mathbf{X}, \lambda_{\mathbf{x}}, \lambda_{\mathbf{0}} \mid \mathbf{A}) p(\mathbf{A}, \boldsymbol{\mu}_{\mathbf{a}_1}, \cdots, \boldsymbol{\mu}_{\mathbf{a}_M}, \mathbf{\Lambda}_{\mathbf{a}_1}, \cdots, \mathbf{\Lambda}_{\mathbf{a}_M}), \end{aligned} \tag{5.10}$$

with $\mathbf{\Theta} = (\mathbf{X}, \mathbf{G}, \mathbf{U}, \mathbf{A}, \boldsymbol{\mu}_{\mathbf{a}_1}, \cdots, \boldsymbol{\mu}_{\mathbf{a}_M}, \mathbf{\Lambda}_{\mathbf{a}_1}, \cdots, \mathbf{\Lambda}_{\mathbf{a}_M}, \lambda_{\mathbf{x}}, \lambda_{\mathbf{0}}, \lambda_{\mathbf{y}})$ collecting all latent variables of interest. The variational parameters and hyperparameters are dropped in the following for conciseness. A graphical model is shown in Figure 5.2 to visualize the structure of the Probability Density Model and underlying dependencies. Here, the blue shaded nodes describe the given observations, and the white ones describe the unknown latent variables of interest.

Similar to Bayesian GP-DMDs, the Probability Density Model from Equation (5.10) breaks down into three parts. The right part describes the observation model and is represented by

$$\begin{aligned} p(\mathbf{Y}, \mathbf{G}, \mathbf{U}, \lambda_{\mathbf{y}} \mid \mathbf{X}) &= \prod_{s=1}^{S} \prod_{t=0}^{T} p(\mathbf{y}_t^s, \mathbf{g}_t^s, \mathbf{U}^s \mid \mathbf{X}^s, \lambda_{\mathbf{y}}) p(\lambda_{\mathbf{y}}) \\ &= \prod_{s=1}^{S} \prod_{t=0}^{T} p(\mathbf{y}_t^s \mid \mathbf{g}_t^s, \lambda_{\mathbf{y}}) p(\mathbf{g}_t^s \mid \mathbf{U}^s, \mathbf{X}^s) p(\mathbf{U}^s) p(\lambda_{\mathbf{y}}) \\ &= \prod_{s=1}^{S} \prod_{t=0}^{T} \mathcal{N}(\mathbf{y}_t^s \mid \mathbf{g}_t^s, \lambda_{\mathbf{y}}^{-1} \mathbf{I}) \mathcal{N}(\mathbf{g}_t^s \mid \mathbf{C}_t^s \mathbf{U}^s, \mathbf{D}_t^s) \\ &\quad \prod_{n=1}^{N} \mathcal{N}(\mathbf{u}_n \mid 0, \mathbf{K}_{zz}) \mathrm{Gam}(\lambda_{\mathbf{y}} \mid \alpha_{\mathbf{y}}, \beta_{\mathbf{y}}), \end{aligned}$$

where $\mathbf{C}_t^s = \mathbf{K}_{t\mathbf{Z}}^s \mathbf{K}_{\mathbf{ZZ}}^{-1}$ and $\mathbf{D}_t^s = \mathbf{K}_{tt}^s - \mathbf{K}_{t\mathbf{Z}}^s \mathbf{K}_{\mathbf{ZZ}}^{-1} \mathbf{K}_{\mathbf{Z}t}^s$ describe operators. $\mathbf{K}_{\mathbf{ZZ}}^s = \mathbf{K}(\mathbf{Z}, \mathbf{Z})$, $\mathbf{K}_{tt}^s = \mathbf{K}(\mathbf{x}_t^s, \mathbf{x}_t^s)$, $\mathbf{K}_{t\mathbf{Z}}^s = \mathbf{K}(\mathbf{x}_t^s, \mathbf{Z})$, and $\mathbf{K}_{\mathbf{Z}t}^s = \mathbf{K}_{t\mathbf{Z}}^{s^T}$ correspond to abbreviations for the

required kernels. The middle part denotes the linear dynamics in the latent space

$$
\begin{aligned}
p(\mathbf{X}, \lambda_{\mathbf{x}}, \lambda_{\mathbf{0}} \mid \mathbf{A}) &= \prod_{s=1}^{S} p(\mathbf{X}^s \mid \mathbf{A}^s, \lambda_{\mathbf{x}}, \lambda_{\mathbf{0}}) p(\lambda_{\mathbf{x}}, \lambda_{\mathbf{0}}) \\
&= \prod_{s=1}^{S} p(\mathbf{x}_0^s \mid \lambda_{\mathbf{0}}) \prod_{t=1}^{T} p(\mathbf{x}_t^s \mid \mathbf{x}_{t-1}^s, \mathbf{A}^s, \lambda_{\mathbf{x}}) p(\lambda_{\mathbf{x}}, \lambda_{\mathbf{0}}) \\
&= \prod_{s=1}^{S} \mathcal{N}(\mathbf{x}_0^s \mid 0, \lambda_{\mathbf{0}}^{-1}\mathbf{I}) \prod_{t=1}^{T} \mathcal{N}(\mathbf{x}_t^s \mid \mathbf{A}^T \mathbf{x}_{t-1}^s, \lambda_{\mathbf{x}}^{-1}\mathbf{I}) \\
&\quad \mathrm{Gam}(\lambda_{\mathbf{0}} \mid \alpha_{\mathbf{0}}, \beta_{\mathbf{0}}) \mathrm{Gam}(\lambda_{\mathbf{x}} \mid \alpha_{\mathbf{x}}, \beta_{\mathbf{x}}).
\end{aligned}
$$

In the remaining term, the hierarchical structure of the linear operator is modeled

$$
\begin{aligned}
p(\mathbf{A}, \boldsymbol{\mu}_{\mathbf{a}_1}, \cdots, \boldsymbol{\mu}_{\mathbf{a}_M}, \boldsymbol{\Lambda}_{\mathbf{a}_1}, \cdots, \boldsymbol{\Lambda}_{\mathbf{a}_M}) &= \prod_{s=1}^{S} \prod_{m=1}^{M} p(\mathbf{a}_m^s \mid \boldsymbol{\mu}_{\mathbf{a}_m}, \boldsymbol{\Lambda}_{\mathbf{a}_m}) p(\boldsymbol{\mu}_{\mathbf{a}_m}, \boldsymbol{\Lambda}_{\mathbf{a}_m}) \\
&= \prod_{s=1}^{S} \prod_{m=1}^{M} \mathcal{N}(\mathbf{a}_m^s \mid \boldsymbol{\mu}_{\mathbf{a}_m}, \boldsymbol{\Lambda}_{\mathbf{a}_m}^{-1}) \mathcal{N}(\boldsymbol{\mu}_{\mathbf{a}_m} \mid \mathbf{0}, \lambda_{\boldsymbol{\mu}_{\mathbf{A}}}^{-1}\mathbf{I}) \\
&\quad \mathcal{W}(\boldsymbol{\Lambda}_{\mathbf{a}_m} \mid \mathbf{W}_m, \nu_m).
\end{aligned}
$$

It assumes M Wishart distributions $\mathcal{W}(\boldsymbol{\Lambda}_{\mathbf{a}_m} \mid \mathbf{W}_m, \nu_m)$ (see Appendix A) as conjugate priors for the precision matrices $\boldsymbol{\Lambda}_{\mathbf{a}_m}$. For the mean vectors $\boldsymbol{\mu}_{\mathbf{a}_m}$, multivariate Gaussian distributions $\mathcal{N}(\boldsymbol{\mu}_{\mathbf{a}_m} \mid \mathbf{0}, \lambda_{\boldsymbol{\mu}_{\mathbf{A}}}^{-1}\mathbf{I})$ are used analogously. As in Section 4.3 regarding the Bayesian GP-DMDs, the learning procedure for Bayesian-DMPs is presented in the following. For this purpose, first, an ELBO is derived, and then the optimal variational distributions are given. Finally, similar to Section 4.3, the Jensen inequality is reversed, and a loss function independent of the induced variables $\mathbf{U}_1, \cdots, \mathbf{U}_S$ is obtained. The development of procedures for Bayesian-DMPs to reproduce demonstrations has not been addressed in this thesis and thus represents an open topic for future research.

Learning from Demonstration

Probabilistic Inference on the Bayesian-DMP framework is performed using Variational Inference or Variational Bayes techniques, similar to Bayesian GP-DMDs (see Section 4.3).

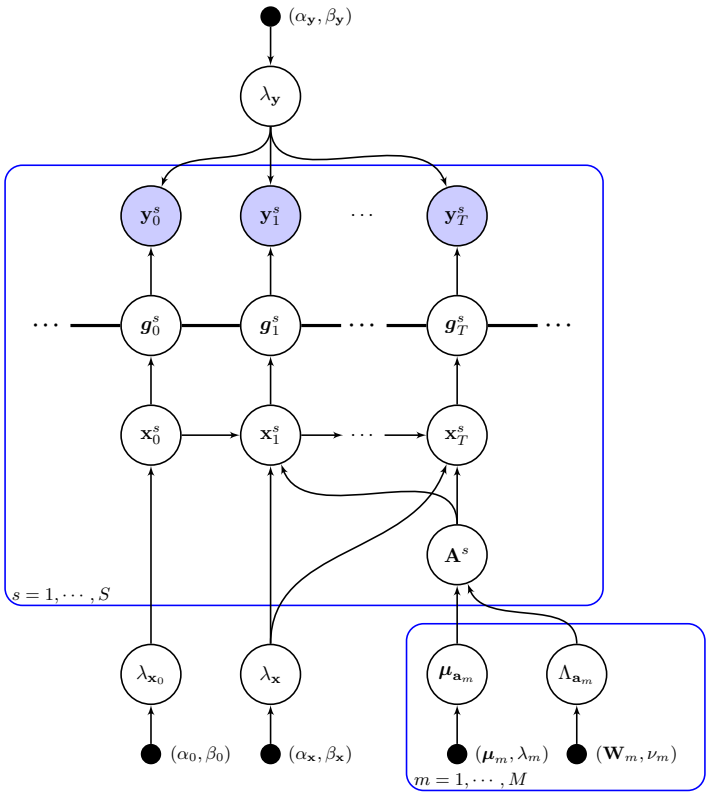


Figure 5.2.: The graphical model visualizes the Probability Density Model of the derived Bayesian Dynamic Mode Primitive. The hierarchical structure for S given trajectories is represented by the larger blue outlined box. The blue shaded nodes represent the given observations $\mathbf{y}_0^s, \cdots, \mathbf{y}_T^s$ for each trajectory. The white nodes correspond to the unknown latent variables. By $\mathbf{x}_0^s, \cdots, \mathbf{x}_T^s$, the sequence of states of interest in the latent space is given. $\mathbf{g}_0^s, \cdots, \mathbf{g}_T^s$ describe the outputs of the Gaussian process function. The linear operator corresponding to each trajectory is given by $\mathbf{A}^s$. The precision values of the assumed Gaussian distributions are depicted by $\lambda_{\mathbf{y}}, \lambda_{\mathbf{0}}, \lambda_{\mathbf{x}}$. $\boldsymbol{\mu}_{\mathbf{a}_m}$ and $\boldsymbol{\Lambda}_{\mathbf{a}_m}$ are the mean vectors and precision matrices representing the hierarchical structure. The thick black line denotes an auto-regressive structure within the Gaussian process, which means that $\mathbf{g}_t^s$ depends on $\mathbf{g}_0^s, \cdots, \mathbf{g}_{t-1}^s$ and $\mathbf{x}_0^s, \cdots, \mathbf{x}_t^s$. The small black nodes correspond to the hyperparameters of the given Probability Density Model.

For this reason, the marginal log-likelihood is extended with a variational distribution $q(\boldsymbol{\Theta})$. Then, using Jensen's inequality results in an ELBO of the following form

$$\begin{aligned}\log p(\mathbf{Y} \mid \boldsymbol{\theta}) &\geq \int q(\boldsymbol{\Theta}) \log \frac{p(\mathbf{Y}, \boldsymbol{\Theta} \mid \boldsymbol{\theta})}{q(\boldsymbol{\Theta})} \, \mathrm{d}\boldsymbol{\Theta} \\ &= \left\langle \log \frac{p(\mathbf{Y}, \boldsymbol{\Theta} \mid \boldsymbol{\theta})}{q(\boldsymbol{\Theta})} \right\rangle_{q(\boldsymbol{\Theta})} \\ &\stackrel{\text{def}}{=} \mathcal{L}_{\text{ELBO}}(q(\boldsymbol{\Theta}), \boldsymbol{\theta}).\end{aligned}$$

In this case, the operator $\langle f(\cdot) \rangle$ corresponds again to the expected value w.r.t. $q(\boldsymbol{\Theta})$. Similar to Bayesian GP-DMDs, a specific form is assumed exclusively for the variational distribution, which describes the outputs of the observation model. Thus, $q(\boldsymbol{\Theta})$ is factorized as follows under the mean field assumption

$$q(\boldsymbol{\Theta}) = \prod_{s=1}^{S} \prod_{t=0}^{T} p(\mathbf{g}_t^s \mid \mathbf{U}^s, \mathbf{X}^s) \prod_{\boldsymbol{\Theta}_i \in \boldsymbol{\Theta} \setminus \{\mathbf{G}\}} q(\boldsymbol{\Theta}_i).$$

The specific functional forms of the remaining variational distributions arise naturally based on the assumed structures of the likelihood functions and the corresponding conjugate prior distributions [3, 4, 6]. The incorporation of the variational distribution $q(\boldsymbol{\Theta})$ and several mathematical reformulations, analogous to Section 4.3, expresses the ELBO as follows

$$\begin{aligned}\mathcal{L}_{\text{ELBO}}(q(\boldsymbol{\Theta}), \boldsymbol{\theta}) = \; & - \mathrm{KL}(q(\lambda_{\mathbf{y}}) \parallel p(\lambda_{\mathbf{y}})) - \mathrm{KL}(q(\lambda_{\mathbf{0}}) \parallel p(\lambda_{\mathbf{0}})) - \mathrm{KL}(q(\lambda_{\mathbf{x}}) \parallel p(\lambda_{\mathbf{x}})) \\ & - \mathrm{KL}\left(q(\mathbf{U}) \,\middle\|\, \prod_{s=1}^{S} p(\mathbf{U}^s) \right) - \mathrm{H}\left(q(\mathbf{X}) q(\lambda_{\mathbf{0}}) \,\middle\|\, \prod_{s=1}^{S} p(\mathbf{x}_0^s \mid \lambda_{\mathbf{0}}) \right) \\ & - \mathrm{KL}\left(q(\mathbf{A}) \prod_{m=1}^{M} q(\boldsymbol{\mu}_{\mathbf{a}_m}) q(\boldsymbol{\Lambda}_{\mathbf{a}_m}) \,\middle\|\, \prod_{m=1}^{M} \prod_{s=1}^{S} p(\mathbf{a}_m^s \mid \boldsymbol{\mu}_{\mathbf{a}_m}, \boldsymbol{\Lambda}_{\mathbf{a}_m}) p(\boldsymbol{\Lambda}_{\mathbf{a}_m}) \right) \\ & + \left\langle \log \prod_{s=1}^{S} \prod_{t=0}^{T} \mathcal{N}(\mathbf{y}_t^s \mid \mathbf{C}_t^s \mathbf{U}^s, \lambda_{\mathbf{y}}^{-1} \mathbf{I}) \right\rangle_{q(\mathbf{X}, \mathbf{U}, \lambda_{\mathbf{y}})} \\ & + \left\langle \log \prod_{s=1}^{S} \prod_{t=1}^{T} p(\mathbf{x}_t^s \mid \mathbf{x}_{t-1}^s, \mathbf{A}^s, \lambda_{\mathbf{x}}) \right\rangle_{q(\mathbf{X}) q(\mathbf{A}) q(\lambda_{\mathbf{x}})} \\ & + \mathrm{H}(q(\mathbf{X}) \parallel q(\mathbf{X})) - \sum_{s=1}^{S} \left\langle \frac{N \lambda_{\mathbf{y}}}{2} \mathrm{Tr}(\mathbf{D}_{:}^s) \right\rangle_{q(\mathbf{X}, \lambda_{\mathbf{y}})},\end{aligned}$$

with $\mathbf{D}_{:}^{s} = \mathbf{K}_{::}^{s} - \mathbf{K}_{:\mathbf{Z}}^{s}\mathbf{K}_{\mathbf{ZZ}}^{-1}\mathbf{K}_{\mathbf{Z}:}^{s}$. This ELBO extends the Bayesian GP-DMD framework regarding multiple trajectories and leads to the following optimization procedure

$$
\begin{aligned}
q^*(\boldsymbol{\Theta}) = &\quad \arg\max_{q(\boldsymbol{\Theta})} \mathcal{L}_{\text{ELBO}}(q(\boldsymbol{\Theta}), \boldsymbol{\theta}), \\
\boldsymbol{\theta}^* = &\quad \arg\max_{\boldsymbol{\theta}} \mathcal{L}_{\text{ELBO}}(q^*(\boldsymbol{\Theta}), \boldsymbol{\theta}).
\end{aligned}
$$

In the next sections, the optimal variational distributions $q^*(\boldsymbol{\Theta})$ are first derived using Calculus of Variations and Euler Lagrange [3, 4, 6, 54, 59]. Then an alternative ELBO is defined, similar to Section 4.3, which results from the reversion of Jensen's Inequality.

Optimal Variational Distribution $q^*(\lambda_{\mathbf{y}})$

Taking the derivative w.r.t. $q(\lambda_{\mathbf{y}})$ and then setting it equal to zero gives

$$
\begin{aligned}
\frac{\partial \mathcal{L}_{\text{ELBO}}}{\partial q(\lambda_{\mathbf{y}})} = &\log \frac{p(\lambda_{\mathbf{y}})}{q(\lambda_{\mathbf{y}})} - 1 - \frac{N\lambda_{\mathbf{y}}}{2} \sum_{s=1}^{S} \langle \operatorname{Tr}(\mathbf{D}_{:}^{s}) \rangle_{q(\mathbf{X})} \\
&+ \left\langle \log \prod_{s=1}^{S} \mathcal{N}(\mathbf{Y}^s \mid \mathbf{C}_{:}^{s} \widetilde{\boldsymbol{\mu}}_{\mathbf{U}^s}, \lambda_{\mathbf{y}}^{-1}\mathbf{I}) \right\rangle_{q(\mathbf{X})} - \frac{N\lambda_{\mathbf{y}}}{2} \left\langle \sum_{s=1}^{S} \operatorname{Tr}\left(\mathbf{C}_{:}^{s^T} \mathbf{C}_{:}^{s} \widetilde{\boldsymbol{\Lambda}}_{\mathbf{U}^s}^{-1} \right) \right\rangle_{q(\mathbf{X})} \\
&\stackrel{!}{=} 0,
\end{aligned}
$$

where $\widetilde{\boldsymbol{\mu}}_{\mathbf{U}^s}$ and $\widetilde{\boldsymbol{\Lambda}}_{\mathbf{U}^s}$ correspond to the natural parameters from the variational distribution $q^*(\mathbf{U})$. In exponential space, the optimal distribution $q^*(\lambda_{\mathbf{y}})$ is proportional to

$$
\begin{aligned}
q^*(\lambda_{\mathbf{y}}) \propto \exp\Bigg(&\log \operatorname{Gam}(\lambda_{\mathbf{y}} \mid \alpha_{\mathbf{y}}, \beta_{\mathbf{y}}) - \frac{N\lambda_{\mathbf{y}}}{2} \sum_{s=1}^{S} \langle \operatorname{Tr}(\mathbf{D}_{:}^{s}) \rangle_{q(\mathbf{X})} \\
&+ \left\langle \log \prod_{s=1}^{S} \mathcal{N}(\mathbf{Y}^s \mid \mathbf{C}_{:}^{s} \widetilde{\boldsymbol{\mu}}_{\mathbf{U}^s}, \lambda_{\mathbf{y}}^{-1}\mathbf{I}) \right\rangle_{q(\mathbf{X})} \\
&- \frac{N\lambda_{\mathbf{y}}}{2} \left\langle \sum_{s=1}^{S} \operatorname{Tr}\left(\mathbf{C}_{:}^{s^T} \mathbf{C}_{:}^{s} \widetilde{\boldsymbol{\Lambda}}_{\mathbf{U}^s}^{-1} \right) \right\rangle_{q(\mathbf{X})} \Bigg),
\end{aligned}
$$

leading to a Gamma distribution

$$q^*(\lambda_{\mathbf{y}}) \propto \mathrm{Gam}\left(\lambda_{\mathbf{y}} \mid \widetilde{\alpha}_{\mathbf{y}}, \widetilde{\beta}_{\mathbf{y}}\right).$$

The natural parameters are given by

$$\widetilde{\alpha}_{\mathbf{y}} = \frac{STN + 2\alpha_{\mathbf{y}}}{2}, \tag{5.11}$$

$$\widetilde{\beta}_{\mathbf{y}} = 2\beta_{\mathbf{y}} + \sum_{s=1}^{S} \mathrm{Tr}\Big(\left(\mathbf{Y}^s - \mathbf{C}^s_{:}\widetilde{\boldsymbol{\mu}}_{\mathbf{U}^s}\right)\left(\mathbf{Y}^s - \mathbf{C}^s_{:}\widetilde{\boldsymbol{\mu}}_{\mathbf{U}^s}\right)^T + N\left(\mathbf{D}^s_{:} + \mathbf{C}^s_{:}\widetilde{\boldsymbol{\Lambda}}^{-1}_{\mathbf{U}^s}\mathbf{C}^{s^T}_{:}\right)\Big), \tag{5.12}$$

where the former corresponds to the shape and the latter to the rate parameter. These natural parameters have strong similarities to the former ones of Bayesian GP-DMD (see Equations (4.11) and (4.12)) with the extension in terms of multiple trajectories.

Optimal Variational Distribution $q^*(\lambda_{\mathbf{0}})$

Subsequently, the derivative w.r.t. $q(\lambda_{\mathbf{0}})$ is considered

$$\begin{aligned}\frac{\partial \mathcal{L}_{\text{ELBO}}}{\partial q(\lambda_{\mathbf{0}})} &= \log\frac{p(\lambda_{\mathbf{0}})}{q(\lambda_{\mathbf{0}})} + \left\langle \log \prod_{s=1}^{S} p(\mathbf{x}^s_0 \mid \lambda_{\mathbf{0}}) \right\rangle_{q(\mathbf{X})} \\ &\stackrel{!}{=} 0,\end{aligned}$$

and set equal to zero. The optimal distribution $q^*(\lambda_{\mathbf{0}})$ shows the following proportionality in exponential space

$$q^*(\lambda_{\mathbf{0}}) \propto \exp\left(\log \mathrm{Gam}\left(\lambda_{\mathbf{0}} \mid \alpha_{\mathbf{0}}, \beta_{\mathbf{0}}\right) + \left\langle \log \prod_{s=1}^{S} \mathcal{N}(\mathbf{x}^s_0 \mid 0, \lambda_{\mathbf{0}}^{-1}\mathbf{I}) \right\rangle_{q(\mathbf{X})} \right),$$

which corresponds to a Gamma distribution

$$q^*(\lambda_{\mathbf{0}}) \propto \mathrm{Gam}\left(\lambda_{\mathbf{0}} \mid \widetilde{\alpha}_{\mathbf{0}}, \widetilde{\beta}_{\mathbf{0}}\right).$$

The shape and rate parameters of the Gamma distribution are given by

$$\widetilde{\alpha}_{\mathbf{0}} = \frac{SM + 2\alpha_{\mathbf{0}}}{2}, \tag{5.13}$$

$$\widetilde{\beta}_{\mathbf{0}} = 2\beta_{\mathbf{0}} + \mathrm{Tr}\left(\sum_{s=1}^{S} \left\langle \mathbf{x}_0^s \mathbf{x}_0^{s^T} \right\rangle_{q(\mathbf{X})}\right), \tag{5.14}$$

which also show strong similarities to the parameters of Bayesian GP-DMD in Equations (4.13) and (4.14) with the extension to multiple trajectories.

Optimal Variational Distribution $q^*(\lambda_{\mathbf{x}})$

Next, setting the derivative w.r.t. $q(\lambda_{\mathbf{x}})$ equal to zero, given by

$$\begin{aligned}\frac{\partial \mathcal{L}_{\text{ELBO}}}{\partial q(\lambda_{\mathbf{x}})} &= \log \frac{p(\lambda_{\mathbf{0}})}{q(\lambda_{\mathbf{0}})} + \left\langle \log \prod_{s=1}^{S}\prod_{t=1}^{T} p(\mathbf{x}_t^s \mid \mathbf{x}_{t-1}^s, \mathbf{A}^s, \lambda_{\mathbf{x}}) \right\rangle_{q(\mathbf{X})q(\mathbf{A})} \\ &\stackrel{!}{=} 0,\end{aligned}$$

results in the following relation in exponential space

$$\begin{aligned}q^*(\lambda_{\mathbf{x}}) \propto \exp\Bigg(&\frac{2\alpha_{\mathbf{x}} - 2}{2}\log|\lambda_{\mathbf{x}}| - \frac{2\widetilde{\beta}_{\mathbf{x}}\lambda_{\mathbf{x}}}{2} + \frac{SM(T-1)}{2}\log|\lambda_{\mathbf{x}}| \\ &- \frac{\lambda_{\mathbf{x}}}{2}\left\langle \mathrm{Tr}\left(\sum_{s=1}^{S}\left(\mathbf{X}^{1^{s^T}} - \mathbf{X}^{0^{s^T}}\widetilde{\boldsymbol{\mu}}_{\mathbf{a}}^s\right)\left(\mathbf{X}^{1^{s^T}} - \mathbf{X}^{0^{s^T}}\widetilde{\boldsymbol{\mu}}_{\mathbf{a}}^s\right)^T\right)\right\rangle_{q(\mathbf{X})} \\ &- \frac{\lambda_{\mathbf{x}}}{2}\left\langle \mathrm{Tr}\left(\sum_{s=1}^{S}\mathbf{X}^{0^{s^T}}\sum_{m=1}^{M}\widetilde{\boldsymbol{\Lambda}}_{\mathbf{a}_m^s}^{-1}\mathbf{X}^{0^s}\right)\right\rangle_{q(\mathbf{X})}\Bigg)\end{aligned}$$

The optimal variational distribution $q^*(\lambda_{\mathbf{x}})$ hence corresponds to a Gamma distribution

$$q^*(\lambda_{\mathbf{x}}) \propto \mathrm{Gam}\left(\lambda_{\mathbf{x}} \mid \widetilde{\alpha}_{\mathbf{x}}, \widetilde{\beta}_{\mathbf{x}}\right),$$

with the natural parameters

$$\widetilde{\alpha}_{\mathbf{x}} = \frac{SM(T-1) + 2\alpha_{\mathbf{x}}}{2}, \tag{5.15}$$

$$\begin{aligned}\widetilde{\beta}_{\mathbf{x}} = 2\beta_{\mathbf{x}} + \operatorname{Tr}\Bigg(\sum_{s=1}^{S}\Bigg(\Bigg\langle \mathbf{X}^{0^s T} \sum_{m=1}^{M} \widetilde{\mathbf{\Lambda}}_{\mathbf{a}_m^s}^{-1} \mathbf{X}^{0^s} \Bigg\rangle_{q(\mathbf{X})} \\ + \Bigg\langle \left(\mathbf{X}^{1^s T} - \mathbf{X}^{0^s T} \widetilde{\boldsymbol{\mu}}_{\mathbf{a}}^{s}\right)\left(\mathbf{X}^{1^s T} - \mathbf{X}^{0^s T} \widetilde{\boldsymbol{\mu}}_{\mathbf{a}}^{s}\right)^T \Bigg\rangle_{q(\mathbf{X})}\Bigg)\Bigg).\end{aligned} \tag{5.16}$$

These parameters are very similar to the shape and rate parameters of Bayesian GP-DMD from eEquations (4.15) and (4.16) extended to multiple trajectories.

Optimal Variational Distribution $q^*(\boldsymbol{\mu}_{\mathbf{a}_m})$

The optimal distribution $q^*(\boldsymbol{\mu}_{\mathbf{a}_m})$ regarding the mth mean vector of the linear operator $\mathbf{A}$ is derived in the following. First, similar to the previous derivations, the derivative w.r.t. $q(\boldsymbol{\mu}_{\mathbf{a}_m})$ is taken and set equal to zero

$$\begin{aligned}\frac{\partial \mathcal{L}_{\text{ELBO}}}{\partial q(\boldsymbol{\mu}_{\mathbf{a}_m})} &= \left\langle \log \frac{\prod_{s=1}^{S} p(\mathbf{a}_m^s \mid \boldsymbol{\mu}_{\mathbf{a}_m}, \mathbf{\Lambda}_{\mathbf{a}_m}) p(\boldsymbol{\mu}_{\mathbf{a}_m})}{q(\mathbf{A}) \prod_{m=1}^{M} q(\boldsymbol{\mu}_{\mathbf{a}_m}) q(\mathbf{\Lambda}_{\mathbf{a}_m})} \right\rangle_{q(\mathbf{A})q(\mathbf{\Lambda}_{\mathbf{a}_m})} - 1 \\ &\stackrel{!}{=} 0.\end{aligned}$$

In exponential space, the following relationship is obtained

$$q^*(\boldsymbol{\mu}_{\mathbf{a}_m}) \propto \exp\left(\left\langle \log \prod_{s=1}^{S} \mathcal{N}(\mathbf{a}_m^s \mid \boldsymbol{\mu}_{\mathbf{a}_m}, \mathbf{\Lambda}_{\mathbf{a}_m}^{-1}) \mathcal{N}(\boldsymbol{\mu}_{\mathbf{a}_m} \mid \mathbf{0}, \lambda_{\boldsymbol{\mu}_{\mathbf{A}}}^{-1} \mathbf{I}) \right\rangle_{q(\mathbf{A})q(\mathbf{\Lambda}_{\mathbf{a}_m})}\right),$$

which corresponds to a Gaussian distribution

$$q^*(\boldsymbol{\mu}_{\mathbf{a}_m}) \propto \mathcal{N}(\boldsymbol{\mu}_{\mathbf{a}_m} \mid \widetilde{\boldsymbol{\mu}}_{\mathbf{a}_m}, \widetilde{\mathbf{\Lambda}}_{\mathbf{a}_m}^{-1}).$$

Thus, the natural parameters are defined by

$$\widetilde{\boldsymbol{\mu}}_{\mathbf{a}_m} = \left(S\widetilde{\nu}_m \widetilde{\mathbf{W}}_m + \lambda_{\boldsymbol{\mu}_{\mathbf{A}}} \mathbf{I}\right)^{-1} \widetilde{\nu}_m \widetilde{\mathbf{W}}_m \sum_{s=1}^{S} \widetilde{\boldsymbol{\mu}}_{\mathbf{a}_m^s}, \tag{5.17}$$

$$\widetilde{\mathbf{\Lambda}}_{\mathbf{a}_m} = S\widetilde{\nu}_m \widetilde{\mathbf{W}}_m + \lambda_{\boldsymbol{\mu}_{\mathbf{A}}} \mathbf{I}, \tag{5.18}$$

where $\widetilde{\boldsymbol{\mu}}_{\mathbf{a}_m}$ and $\widetilde{\boldsymbol{\Lambda}}_{\mathbf{a}_m}$ are the mean vector and the precision matrix, respectively. When the predefined precision value $\lambda_{\boldsymbol{\mu}_\mathbf{A}}$ is set to zero, the resulting expectation value $\widetilde{\boldsymbol{\mu}}_{\mathbf{a}_m}$ is equal to the point estimate of Pro-DMP in Equation (5.5). Thus, the extension to a fully Bayesian approach leads to a MAP estimate and hence to a ridge regression with the ridge coefficient $\lambda_{\boldsymbol{\mu}_\mathbf{A}}$ for the parameter $\widetilde{\boldsymbol{\mu}}_{\mathbf{a}_m}$ [4, 6].

Optimal Variational Distribution $q^*(\boldsymbol{\Lambda}_{\mathbf{a}_m})$

The optimal distribution $q^*(\boldsymbol{\Lambda}_{\mathbf{a}_m})$ for the corresponding precision matrix $\boldsymbol{\Lambda}_{\mathbf{a}_m}$ follows. Setting the derivative w.r.t. $q(\boldsymbol{\Lambda}_{\mathbf{a}_m})$ equal to zero results in

$$\begin{aligned}\frac{\partial \mathcal{L}_{\text{ELBO}}}{\partial q(\boldsymbol{\Lambda}_{\mathbf{a}_m})} &= \left\langle \log \frac{\prod_{s=1}^{S} p(\mathbf{a}_m^s \mid \boldsymbol{\mu}_{\mathbf{a}_m}, \boldsymbol{\Lambda}_{\mathbf{a}_m}) p(\boldsymbol{\Lambda}_{\mathbf{a}_m})}{q(\mathbf{A}) \prod_{m=1}^{M} q(\boldsymbol{\mu}_{\mathbf{a}_m}) q(\boldsymbol{\Lambda}_{\mathbf{a}_m})} \right\rangle_{q(\mathbf{A})q(\boldsymbol{\Lambda}_{\mathbf{a}_m})} - 1 \\ &\stackrel{!}{=} 0.\end{aligned}$$

In the exponential space, the optimal distribution $q^*(\boldsymbol{\Lambda}_{\mathbf{a}_m})$ has the following relation

$$q^*(\boldsymbol{\Lambda}_{\mathbf{a}_m}) \propto \exp\left(\left\langle \log \prod_{s=1}^{S} \mathcal{N}(\mathbf{a}_m^s \mid \boldsymbol{\mu}_{\mathbf{a}_m}, \boldsymbol{\Lambda}_{\mathbf{a}_m}^{-1}) \right\rangle_{q(\mathbf{A})q(\boldsymbol{\mu}_{\mathbf{a}_m})} + \log \mathcal{W}\left(\boldsymbol{\Lambda}_{\mathbf{a}_m} \mid \mathbf{W}_m, \nu_m\right) \right),$$

leading to a Wishart distribution

$$q^*(\boldsymbol{\Lambda}_{\mathbf{a}_m}) \propto \mathcal{W}(\boldsymbol{\Lambda}_{\mathbf{a}_m} \mid \widetilde{\mathbf{W}}_m, \widetilde{\nu}_m).$$

The corresponding scale matrix and degrees of freedom parameter are

$$\widetilde{\mathbf{W}}_m^{-1} = \mathbf{W}_m^{-1} + \sum_{s=1}^{S} \left(\left(\widetilde{\boldsymbol{\mu}}_{\mathbf{a}_m^s} - \widetilde{\boldsymbol{\mu}}_{\mathbf{a}_m} \right) \left(\widetilde{\boldsymbol{\mu}}_{\mathbf{a}_m^s} - \widetilde{\boldsymbol{\mu}}_{\mathbf{a}_m} \right)^T + \widetilde{\boldsymbol{\Lambda}}_{\mathbf{a}_m^s}^{-1} \right), \tag{5.19}$$

$$\widetilde{\nu}_m = S + \nu_m. \tag{5.20}$$

The expectation value of this Wishart distribution $\widetilde{\nu}_m \widetilde{\mathbf{W}}$ exhibits similarities to the resulting point (see Equation (5.6)) estimate of Pro-DMP.

Optimal Variational Distribution $q^*(\mathbf{A})$

Then, taking the derivative w.r.t. $q(\mathbf{A})$ and setting it equal to zero gives

$$\begin{aligned}\frac{\partial \mathcal{L}_{\text{ELBO}}}{\partial q(\mathbf{A})} = &\left\langle \log \frac{\prod_{m=1}^{M}\prod_{s=1}^{S} p(\mathbf{a}_m^s \mid \boldsymbol{\mu}_{\mathbf{a}_m}, \boldsymbol{\Lambda}_{\mathbf{a}_m}) p(\boldsymbol{\Lambda}_{\mathbf{a}_m})}{q(\mathbf{A}) \prod_{m=1}^{M} q(\boldsymbol{\mu}_{\mathbf{a}_m}) q(\boldsymbol{\Lambda}_{\mathbf{a}_m})} \right\rangle_{\prod_{m=1}^{M} q(\boldsymbol{\mu}_{\mathbf{a}_m}, \boldsymbol{\Lambda}_{\mathbf{a}_m})} - 1 \\ &+ \left\langle \log \prod_{t=1}^{T} p(\mathbf{x}_{m,t}^s \mid \mathbf{x}_{t-1}^s, \mathbf{a}_m, \lambda_{\mathbf{x}}) \right\rangle_{q(\mathbf{X}) q(\lambda_{\mathbf{x}})} \\ &\stackrel{!}{=} 0,\end{aligned}$$

where a transformation to exponential space returns the following relation

$$\begin{aligned}q^*(\mathbf{A}) \propto \prod_{m=1}^{M} \exp\Bigg(&\left\langle \log \prod_{s=1}^{S} \mathcal{N}(\mathbf{a}_m^s \mid \boldsymbol{\mu}_{\mathbf{a}_m}, \boldsymbol{\Lambda}_{\mathbf{a}_m}^{-1}) \right\rangle_{q(\boldsymbol{\mu}_{\mathbf{a}_m}) q(\boldsymbol{\Lambda}_{\mathbf{a}_m})} \\ &\left\langle \log \prod_{t=1}^{T} \mathcal{N}(\mathbf{x}_{m,t}^s \mid \mathbf{x}_{t-1}^{s^T} \mathbf{a}_m^s, \lambda_{\mathbf{x}}^{-1} \mathbf{I}) \right\rangle_{q(\mathbf{X}) q(\lambda_{\mathbf{x}})} \Bigg).\end{aligned}$$

Accordingly, the optimal variational distribution $q^*(\mathbf{A})$ factorizes into SM-independent Gaussian distributions

$$q^*(\mathbf{A}) \propto \prod_{s=1}^{S}\prod_{m=1}^{M} q^*(\mathbf{a}_m^s) = \prod_{s=1}^{S}\prod_{m=1}^{M} \mathcal{N}(\mathbf{a}_m^s \mid \widetilde{\boldsymbol{\mu}}_{\mathbf{a}_m^s}, \widetilde{\boldsymbol{\Lambda}}_{\mathbf{a}_m^s}^{-1}),$$

where the natural parameters of each distribution are given by

$$\widetilde{\boldsymbol{\mu}}_{\mathbf{a}_m^s} = \left(\widetilde{\nu}_m \widetilde{\mathbf{W}}_m + \frac{\widetilde{\alpha}_{\mathbf{x}}}{\widetilde{\beta}_{\mathbf{x}}} \Psi_2^s \right)^{-1} \left(\frac{\widetilde{\alpha}_{\mathbf{x}}}{\widetilde{\beta}_{\mathbf{x}}} \Psi_{1_m}^s + \widetilde{\nu}_m \widetilde{\mathbf{W}}_m \widetilde{\boldsymbol{\mu}}_{\mathbf{a}_m} \right), \tag{5.21}$$

$$\widetilde{\boldsymbol{\Lambda}}_{\mathbf{a}_m^s} = \widetilde{\nu}_m \widetilde{\mathbf{W}}_m + \frac{\widetilde{\alpha}_{\mathbf{x}}}{\widetilde{\beta}_{\mathbf{x}}} \Psi_2^s. \tag{5.22}$$

The corresponding sufficient statistics $\Psi_{1_m}^s$ and Ψ_2^s are defined by

$$\Psi_{1_m}^s = \left\langle \mathbf{X}^{0^s} \mathbf{X}_m^{1^{s^T}} \right\rangle_{q(\mathbf{X})}, \quad \Psi_2^s = \left\langle \mathbf{X}^{0^s} \mathbf{X}^{0^{s^T}} \right\rangle_{q(\mathbf{X})},$$

depending on the variational distribution $q(\mathbf{X})$. These estimates thus extend the point estimates from the expectation step of Pro-DMPs (see Equations (5.3) and (5.4)). The natural parameters $\widetilde{\nu}_m, \widetilde{\mathbf{W}}_m, \widetilde{\alpha}_{\mathbf{x}}, \widetilde{\beta}_{\mathbf{x}}$, and $\widetilde{\boldsymbol{\mu}}_{\mathbf{a}_m}$ belong to the optimal variational distributions $q^*(\boldsymbol{\Lambda}_{\mathbf{a}_m}), q^*(\lambda_{\mathbf{x}})$ and $q^*(\mathbf{X})$, respectively.

Optimal Variational Distribution $q^*(\mathbf{U})$

Next, the derivative to $q(\mathbf{U})$ is taken and set equal to zero

$$\frac{\partial \mathcal{L}_{\text{ELBO}}}{\partial q(\mathbf{U})} = \log \frac{\prod_{s=1}^{S} p(\mathbf{U}^s)}{q(\mathbf{U})} - 1 + \left\langle \log \prod_{s=1}^{S} \prod_{t=0}^{T} \mathcal{N}(\mathbf{y}_t^s \mid \mathbf{C}_t^s \mathbf{U}^s, \lambda_{\mathbf{y}}^{-1} \mathbf{I}) \right\rangle_{q(\mathbf{X}, \lambda_{\mathbf{y}})} \stackrel{!}{=} 0.$$

The transformation into exponential space gives

$$q^*(\mathbf{U}) \propto \prod_{s=1}^{S} \prod_{n=1}^{N} \exp\Bigg(-\frac{1}{2} \Bigg(\langle \lambda_{\mathbf{y}} \rangle_{q(\lambda_{\mathbf{y}})} \mathbf{Y}_n^{s^T} \mathbf{Y}_n^s - 2 \mathbf{U}_n^{s^T} \mathbf{K}_{zz}^{-1} \langle \lambda_{\mathbf{y}} \mathbf{K}_{z:} \rangle_{q(\mathbf{X}) q(\lambda_{\mathbf{y}})} \mathbf{Y}_n^s + \mathbf{U}_n^{s^T} \left(\mathbf{K}_{zz}^{-1} + \mathbf{K}_{zz}^{-1} \langle \lambda_{\mathbf{y}} \mathbf{K}_{z:} \mathbf{K}_{:z} \rangle_{q(\mathbf{X}) q(\lambda_{\mathbf{y}})} \mathbf{K}_{zz}^{-1} \right) \mathbf{U}_n^s \Bigg) \Bigg).$$

The optimal variational distribution $q^*(\mathbf{U})$ thus factorizes into SN-independent Gaussian distributions

$$q^*(\mathbf{U}) \propto \prod_{s=1}^{S} \prod_{n=1}^{N} \mathcal{N}\left(\mathbf{U}_n^s \mid \widetilde{\boldsymbol{\mu}}_{\mathbf{U}_n^s}, \widetilde{\boldsymbol{\Lambda}}_{\mathbf{U}^s}^{-1} \right),$$

with the natural parameters of each distribution

$$\widetilde{\boldsymbol{\mu}}_{\mathbf{U}_n^s} = \left(\frac{\widetilde{\beta}_{\mathbf{y}}}{\widetilde{\alpha}_{\mathbf{y}}} \mathbf{K}_{zz}^{-1} + \mathbf{K}_{zz}^{-1} \Psi_4^s \mathbf{K}_{zz}^{-1} \right)^{-1} \mathbf{K}_{zz}^{-1} \Psi_{3_n}^s, \tag{5.23}$$

$$\widetilde{\boldsymbol{\Lambda}}_{\mathbf{U}^s} = \mathbf{K}_{zz}^{-1} + \frac{\widetilde{\alpha}_{\mathbf{y}}}{\widetilde{\beta}_{\mathbf{y}}} \mathbf{K}_{zz}^{-1} \Psi_4^s \mathbf{K}_{zz}^{-1}. \tag{5.24}$$

In this case, the sufficient statistics $\Psi_{3_n}^s$ and Ψ_4^s are given by

$$\Psi_{3_n}^s = \langle \mathbf{K}_{z:} \rangle_{q(\mathbf{X})} \mathbf{Y}_n^s, \quad \Psi_4^s = \langle \mathbf{K}_{z:} \mathbf{K}_{:z} \rangle_{q(\mathbf{X})},$$

depending on the variational distribution $q(\mathbf{X})$. These natural parameters expand the mean vector and precision matrix from Equations (4.21) and (4.22) of Bayesian GP-DMD. The natural parameters $\widetilde{\alpha}_{\mathbf{y}}$ and $\widetilde{\beta}_{\mathbf{y}}$ correspond to the optimal variational distributions $q^*(\lambda_{\mathbf{y}})$.

Optimal Variational Distribution $q^*(\mathbf{X})$

The derivative w.r.t. $q(\mathbf{X})$ results in

$$\begin{aligned}
\frac{\partial \mathcal{L}_{\text{ELBO}}}{\partial q(\mathbf{X})} = & -\log q(\mathbf{X}) - 1 + \sum_{s=1}^{S} \log \mathcal{N}(\mathbf{x}_0^s \mid 0, \frac{\widetilde{\beta}_{\mathbf{0}}}{\widetilde{\alpha}_{\mathbf{0}}} \mathbf{I}) \\
& + \sum_{s=1}^{S} \sum_{t=0}^{T} \left(\log \mathcal{N}(\mathbf{y}_t^s \mid \mathbf{C}_t^s \widetilde{\boldsymbol{\mu}}_{\mathbf{U}^s}, \frac{\widetilde{\beta}_{\mathbf{y}}}{\widetilde{\alpha}_{\mathbf{y}}} \mathbf{I}) - \frac{N \widetilde{\alpha}_{\mathbf{y}}}{2 \widetilde{\beta}_{\mathbf{y}}} \operatorname{Tr}(\mathbf{D}_t^s + \mathbf{C}_t^s \widetilde{\boldsymbol{\Lambda}}_{\mathbf{U}^s}^{-1} \mathbf{C}_t^{s^T}) \right) \\
& + \sum_{s=1}^{S} \sum_{t=1}^{T} \left(\log \mathcal{N}(\mathbf{x}_t^s \mid \mathbf{x}_{t-1}^{s^T} \widetilde{\boldsymbol{\mu}}_{\mathbf{a}^s}, \frac{\widetilde{\beta}_{\mathbf{x}}}{\widetilde{\alpha}_{\mathbf{x}}} \mathbf{I}) - \frac{\widetilde{\alpha}_{\mathbf{x}}}{2 \widetilde{\beta}_{\mathbf{x}}} \operatorname{Tr}(\mathbf{x}_{t-1}^{s^T} \sum_{m=1}^{M} \widetilde{\boldsymbol{\Lambda}}_{\mathbf{a}_m^s}^{-1} \mathbf{x}_{t-1}^s) \right) \\
& \stackrel{!}{=} 0.
\end{aligned}$$

with $\widetilde{\boldsymbol{\mu}}_{\mathbf{a}^s} = [\widetilde{\boldsymbol{\mu}}_{\mathbf{a}^s_1}, \cdots, \widetilde{\boldsymbol{\mu}}_{\mathbf{a}^s_M}]$ and $\widetilde{\boldsymbol{\mu}}_{\mathbf{U}^s} = [\widetilde{\boldsymbol{\mu}}_{\mathbf{U}^s_1}, \cdots, \widetilde{\boldsymbol{\mu}}_{\mathbf{U}^s_N}]$. In exponential space, this derivative leads to the following relation

$$
\begin{aligned}
q(\mathbf{X})^* \propto & \prod_{s=1}^{S} \mathcal{N}(\mathbf{x}_0^s \mid 0, \frac{\widetilde{\beta}_{\mathbf{0}}}{\widetilde{\alpha}_{\mathbf{0}}}\mathbf{I}) \\
& \underbrace{\prod_{t=0}^{T} \mathcal{N}(\mathbf{y}_t^s \mid \mathbf{C}_t^s \widetilde{\boldsymbol{\mu}}_{\mathbf{U}^s}, \frac{\widetilde{\beta}_{\mathbf{y}}}{\widetilde{\alpha}_{\mathbf{y}}}\mathbf{I}) \exp\left(-\frac{N\widetilde{\alpha}_{\mathbf{y}}}{2\widetilde{\beta}_{\mathbf{y}}} \mathrm{Tr}\left(\mathbf{D}_t^s + \mathbf{C}_t^c \widetilde{\boldsymbol{\Lambda}}_{\mathbf{U}^s}^{-1} \mathbf{C}_t^{c^T}\right)\right)}_{\widetilde{p}(\mathbf{y}_t^s \mid \mathbf{x}_t^s)} \\
& \underbrace{\prod_{t=1}^{T} \mathcal{N}(\mathbf{x}_t^s \mid \widetilde{\boldsymbol{\mu}}_{\mathbf{a}^s}^T \mathbf{x}_{t-1}^s, \frac{\widetilde{\beta}_{\mathbf{x}}}{\widetilde{\alpha}_{\mathbf{x}}}\mathbf{I}) \exp\left(-\frac{\widetilde{\alpha}_{\mathbf{x}}}{2\widetilde{\beta}_{\mathbf{x}}} \mathrm{Tr}(\mathbf{x}_{t-1}^{s^T} \sum_{m=1}^{M} \widetilde{\boldsymbol{\Lambda}}_{\mathbf{a}_m^s}^{-1} \mathbf{x}_{t-1}^s)\right)}_{\widetilde{p}\left(\mathbf{x}_t^s \mid \mathbf{x}_{t-1}^s\right)} \\
\propto & \prod_{s=1}^{S} p(\mathbf{x}_0^s) \widetilde{p}\left(\mathbf{y}_0^s \mid \mathbf{x}_0^s\right) \prod_{t=1}^{T} \widetilde{p}\left(\mathbf{y}_t^s \mid \mathbf{x}_t^s\right) \widetilde{p}\left(\mathbf{x}_t^s \mid \mathbf{x}_{t-1}^s\right),
\end{aligned}
$$

taking the form of a Markovian nonlinear SSM that is potentially non-Gaussian, similar to Equation (4.23) of Bayesian GP-DMD. The risk of a non-Gaussian distribution comes from the additional exponential terms. Like Bayesian GP-DMD, the exponential terms are assumed to correspond to additive Gaussian noise with zero-mean, and hence an approximation is made by the following specific functional form

$$
q(\mathbf{X})^* \propto \prod_{s=1}^{S} \mathcal{N}(\mathbf{x}_0^s \mid 0, \frac{\widetilde{\beta}_{\mathbf{0}}}{\widetilde{\alpha}_{\mathbf{0}}}\mathbf{I}) \prod_{t=0}^{T} \mathcal{N}(\mathbf{y}_t^s \mid \mathbf{C}_t^s \widetilde{\boldsymbol{\mu}}_{\mathbf{U}^s}, \frac{\widetilde{\beta}_{\mathbf{y}}}{\widetilde{\alpha}_{\mathbf{y}}}\mathbf{I}) \prod_{t=1}^{T} \mathcal{N}(\mathbf{x}_t^s \mid \widetilde{\boldsymbol{\mu}}_{\mathbf{a}^s}^T \mathbf{x}_{t-1}^s, \frac{\widetilde{\beta}_{\mathbf{x}}}{\widetilde{\alpha}_{\mathbf{x}}}\mathbf{I}).
$$

This approximation corresponds to a Markovian nonlinear Gaussian State-Space Model. Therefore, Probabilistic Inference techniques, as described in the Appendix D, are applicable [4, 52, 53].

Optimal Variational Parameters And Hyperparameters

In the following, similar to Bayesian GP-DMD in Section 4.3, an alternative formulation of an ELBO is given

$$\begin{aligned}\mathcal{L}_{\text{ELBO}}(q^*(\boldsymbol{\Theta}), \boldsymbol{\theta}) = c - \frac{N\widetilde{\alpha}_{\mathbf{y}}}{2\widetilde{\beta}_{\mathbf{y}}} \sum_{s=1}^{S} \langle \mathrm{Tr}(\mathbf{D}_{:}^{s}) \rangle_{q^*(\mathbf{X})} + \frac{SN}{2} \log|\mathbf{K}_{zz}| \\ - \sum_{s=1}^{S} \frac{N}{2} \log\left| \frac{\widetilde{\alpha}_{\mathbf{y}}}{\widetilde{\beta}_{\mathbf{y}}} \Psi_6^s + \mathbf{K}_{zz} \right| - \frac{1}{2} \mathrm{Tr}\left(\sum_{s=1}^{S} \boldsymbol{\Lambda}^s \mathbf{Y}^s \mathbf{Y}^{s^T} \right),\end{aligned}$$

where $\boldsymbol{\Lambda}^s = \frac{\widetilde{\alpha}_{\mathbf{y}}}{\widetilde{\beta}_{\mathbf{y}}} \mathbf{I} - \frac{\widetilde{\alpha}_{\mathbf{y}}^2}{\widetilde{\beta}_{\mathbf{y}}^2} \Psi_5^s \left(\frac{\widetilde{\alpha}_{\mathbf{y}}}{\widetilde{\beta}_{\mathbf{y}}} \Psi_6^s + \mathbf{K}_{zz} \right)^{-1} \Psi_5^{s^T}$. The variational parameters corresponding to the inducing inputs and the hyperparameters representing the kernel parameters are collected in $\boldsymbol{\theta}$. The corresponding sufficient statistics are defined as

$$\Psi_5^s = \langle \mathbf{K}_{:z}^s \rangle_{q^*(\mathbf{X})} ,$$

$$\Psi_6^s = \langle \mathbf{K}_{z:}^s \mathbf{K}_{:z}^s \rangle_{q^*(\mathbf{X})} .$$

This alternative ELBO follows from the reversion of Jensen's inequality [50]. It depends directly on the variational parameters and hence on the inducing inputs $\mathbf{Z}$ and no longer on the induced variables $\mathbf{U}$. All terms not relevant for the optimization of the variational parameters and the kernel hyperparameters are summarized in c. Numerical calculation methods then obtain the gradients and gradient descent techniques are applied [50].

6. Experiments and Results

Chapter 4 introduced the GP-DMD and the Bayesian GP-DMD. The former framework is a Maximum A Posteriori probability estimate that combines the GP-SSM family with Koopman Theory and the DMD family. The latter framework formulates GP-DMD as a fully Bayesian approach. This formulation leads to a mitigation of overfitting and learning approximations of the posterior distributions. However, both frameworks, the GP-DMD and the Bayesian GP-DMD are unable to address multiple given demonstrations. Therefore, the Pro-DMP and the Bayesian-DMP were introduced in Chapter 5. The former extends the GP-DMD utilizing a hierarchical structure to achieve valid movement primitives able to handle multiple demonstrations. The second framework is inspired by Bayesian GP-DMD and formulates a fully Bayesian formalism of Pro-DMP. However, the final development of this framework is ongoing and forms an open research topic. Therefore, Bayesian-DMP is not discussed throughout this chapter. The results obtained from testing and validating the GP-DMD, the Bayesian GP-DMD, and the Pro-DMP on three different datasets are presented in the following. The primary goal here is to determine whether the frameworks are capable of learning and reproducing given demonstrations. In the case of GP-DMD and Bayesian GP-DMD, only one demonstration is considered. For Pro-DMP, five demonstrations are given.

All the given demonstrations considered throughout this chapter are preprocessed by the application of PCA [4]. Subsequently, uncorrelated, Gaussian-distributed data with zero mean and unit variance along with each dimension result [4,6]. The implementation of the numerical differentiation algorithms relies on the JAX library, which provides automatic differentiation and just-in-time compilation either on CPU or GPU [95]. As a result of the work with Probabilistic Inference techniques, the NumPyro library is used, which provides a probabilistic programming library [96]. Algorithms sketching the learning procedure of the GP-DMD, the Bayesian GP-DMD, and the Pro-DMP are presented in the appendix. In the case of the parameters which exhibit a closed-form solution, vanilla gradient descent with learning rate 1.0 is applied. For the remaining parameters where a closed-form solution does not exist, the automatic differentiation provided by JAX is applied in combination

with *Adam*, a first-order gradient-based optimization method [97]. Adam has been chosen due to its computational efficiency and low memory requirements [97]. Detailed lists of the parameter settings are provided in Appendix F.

The Root Mean Square Error (RMSE) metric is taken to measure the performance of the reconstructed movements [6]. A list summarizing the performance is shown in Table 6.1. The RMSE is taken between the predicted mean of the trained model and the given demonstration. In the case of Pro-DMPs, where multiple demonstrations are given, the average is considered. In order to obtain a measure of variability in the results, the standard deviation is used w.r.t. the estimated RMSE. For this purpose, one hundred samples are taken from each framework. Then, the RMSE between those samples and the corresponding given demonstration is calculated. Based on these RMSE results from the samples, a standard deviation of the RMSE means is estimated. Thus, the former value in each cell represents the RMSE mean, and the latter value represents the standard deviation. The thick values indicate the best RMSE mean performance on the corresponding data set.

6.1. The Circle-Shape Dataset

The *Circle-Shape Dataset (CSD)* provides rhythm-based trajectories that follow circular motions in a three-dimensional space. The left side of Figure 6.1 displays three benchmarks of the GP-DMD, the Bayesian GP-DMD, and the Pro-DMP on this dataset. The red dashed lines show the respective demonstrations. In the top two left plots, the GP-DMD and the Bayesian GP-DMD are represented, and consequently, only one demonstration is considered. For the benchmark with the Pro-DMP, five demonstrations are given in the remaining left plot. The thick blue line depicts the mean estimate of each trained framework. Samples drawn from these trained frameworks are shown as thin blue lines.

The GP-DMD and the Bayesian GP-DMD both achieve satisfactory results in estimating the mean for the respective demonstration. Both,Figure 6.1 and Table 6.1, show that the Bayesian GP-DMD has a lower variance than the GP-DMD. However, one reason for this issue is that the Bayesian GP-DMD takes 25 inducing variables while the GP-DMD only takes four labels from the given demonstration. On the one hand, a major advantage of the GP-DMD in comparison to the Bayesian GP-DMD is the computational efficiency. Even though the inducing pairs used in the Bayesian GP-DMD reduce the computational complexity of the kernel inversion compared to the GP-DMD, the Probabilistic Inference Techniques (see Appendix D) required to infer the variational distribution over the latent states drastically

slow down the Bayesian GP-DMD framework. On the other hand, a major disadvantage of the GP-DMD is keeping several labels to reproduce the given demonstration. Based on the choice of these labels, the performance of the resulting demonstration varies. The Bayesian GP-DMD circumvents this problem by utilizing the inducing variables. These variables directly infer knowledge about the given demonstrations to provide labels to reproduce the given demonstration indirectly. However, the use of inducing pairs entails the drawback of adjusting additional parameters for the optimization. The number and initial values of inducing pairs affect the performance of the Bayesian GP-DMD significantly. The Pro-DMP in the lower right plot in Figure 6.1 achieves satisfactory results based on the given demonstrations. Like the GP-DMD, the Pro-DMP requires a fixed number of labels in the observation space to reproduce the demonstrations. In the scope of this experiment, only the mean estimates in the latent space were considered. Therefore, it is interesting for future research to take samples in the latent space and reproduce the corresponding demonstration in the observation space. In this way, analysis can be performed whether Pro-DMP is able to learn variability in the circular movement from the given demonstrations.

6.2. The Eight-Shape Dataset

The three benchmarks of the GP-DMD, the Bayesian GP-DMD, and the Pro-DMP on the *Eight-Shape Dataset (ESD)* are displayed on the right side of Figure 6.1. The ESD gives rhythm-based trajectories that follow eight-shape motions in a three-dimensional space. The red dashed lines show the respective demonstrations. In the top two right plots, the GP-DMD and the Bayesian GP-DMD are represented, and consequently, only one demonstration is considered. For the benchmark with the Pro-DMP, five demonstrations are given in the remaining right plot. The thick blue line depicts the mean estimate of each trained framework. Samples drawn from these trained frameworks are shown as thin blue lines.

The GP-DMD and the Bayesian GP-DMD both achieve satisfactory results in estimating the mean for the respective demonstration. Unlike in Section 6.1, Figure 6.1 and Table 6.1 display that the GP-DMD has a lower variance than the Bayesian GP-DMD. This issue is interesting since the Bayesian GP-DMD takes 25 inducing variables while the GP-DMD only takes four labels from the given demonstration. In comparison to the CSD (see Section 6.1), both frameworks exhibit higher variance in the samples. The major advantages and disadvantage of both frameworks are previously mentioned in Section 6.1. The Pro-DMP

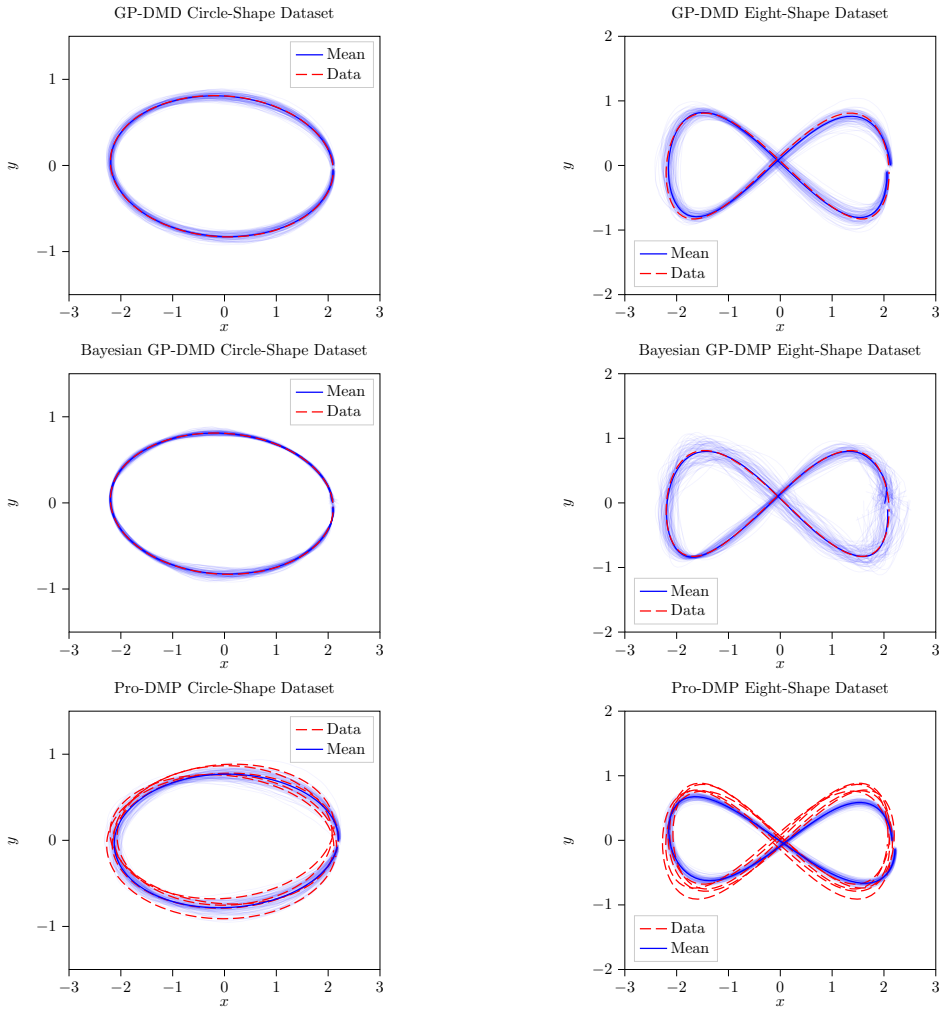


Figure 6.1.: This figure shows three benchmarks of the derived GP-DMDs, Bayesian GP-DMDs and Pro-DMPs on the Circle-Shape Dataset (CSD) and Eight-Shape Dataset (ESD). The plots on the left correspond to the CSD and those plots on the right to the ESD. The red dashed lines represent the given demonstrations. While only one demonstration is given in each of the top four plots, the last two plots exhibit five demonstrations each. The thick blue line presents the predicted mean estimate, while the thin blue lines represent samples of demonstrations. All frameworks achieve satisfactory results in reproducing the demonstrations on all plots.

in the lower right plot in Figure 6.1 achieves satisfactory results in reproducing an eight-shape based on the given demonstrations. Like the GP-DMD, the Pro-DMP requires a fixed number of labels in the observation space to reproduce the demonstrations. However, unlike for the CSD (see Section 6.1), the Pro-DMP does not match the given demonstrations' mean and hence exhibits a higher RMSE. As for CSD, this experiment considers only the mean estimates in the latent space.

6.3. The Minimum-Jerk Dataset

Figures 6.2, 6.3 and 6.4 show the benchmarks of the GP-DMD, the Bayesian GP-DMD, and the Pro-DMP on the *Minimum-Jerk Dataset (MJD)*. The MJD provides six-dimensional minimum jerk movements obtained from two joints. Indeed, it presents the most difficult benchmark for the three frameworks. The red dashed lines show the respective demonstrations. In the top plots, the temporal changes in the positions are shown. The change in the velocity over time is given in the middle plots. The lower plots correspond to the temporal change in the acceleration. Figures 6.2 and 6.3 only consider one demonstration, while in the last Figure 6.4, five demonstrations are taken into account. The thick blue lines represent the mean estimate of each trained framework. Samples drawn from these trained frameworks are shown as thin blue lines.

Figure 6.2 shows the results of the the GP-DMD benchmark on the MJD. The mean estimates achieve decent results in matching the given demonstrations. However, the variance in the samples increases significantly in the positions and the velocities compared to the accelerations. Results of the Bayesian GP-DMD are shown in Figure 6.3. For the acceleration trajectories, decent results are obtained. However, the estimated velocity trajectories deviate slightly from the given data at the edges. Regarding the temporal changes of the positions in the upper two plots, a significant deviation between the estimated mean trajectories and the given data is observed. Similarly to the GP-DMD, the position and velocity trajectories exhibit a higher variance than the acceleration trajectories. Although the Bayesian GP-DMD achieves a lower RMSE (see Table 6.1), Figure 6.3 indicates that the Bayesian GP-DMD performs worse than the GP-DMD. Finally, the performance of the the Pro-DMP is given in Figure 6.4. As for the GP-DMD and the Bayesian GP-DMD, satisfactory mean estimates of the acceleration trajectories are obtained. However, the position and velocity trajectories deviate severely from the given data and show high variance. A major drawback of the GP-DMD, the Bayesian GP-DMD, and Pro-DMP is that the resulting estimates do not exhibit consistent relationships between

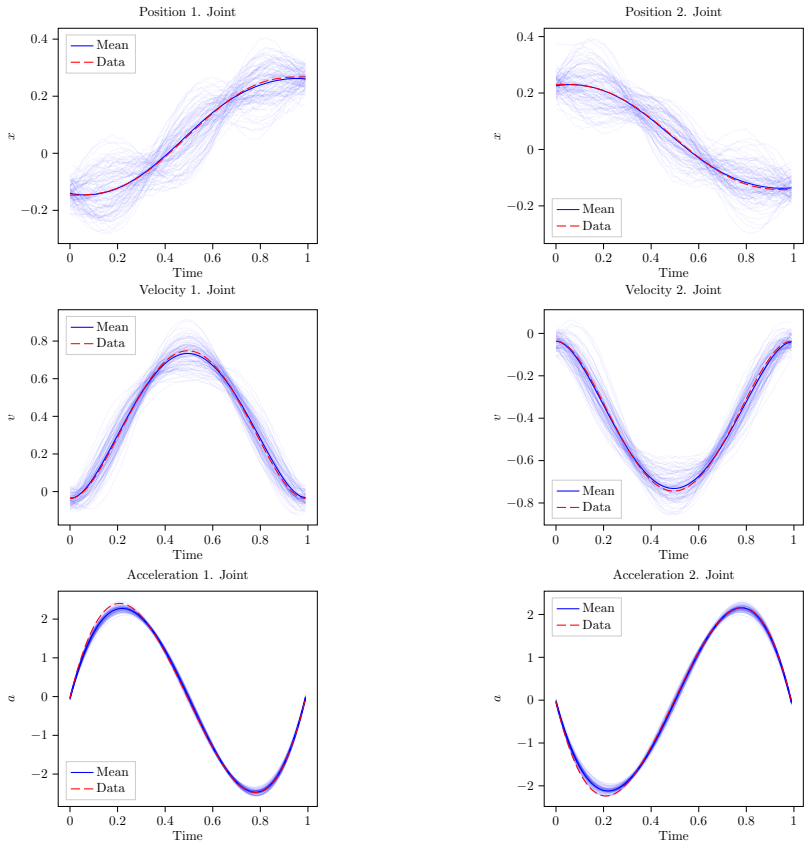


Figure 6.2.: This figure shows a benchmark of the derived GP-DMDs on the Minimum-Jerk Dataset (MJD). The MJD corresponds to the movement of two joints with their respective position, velocity, and acceleration. The movement of the first and second joint are shown on the left and right side, respectively. The red dashed lines represent the given demonstrations. The thick blue line presents the predicted mean estimate, while the thin blue lines represent samples of demonstrations. The predicted mean of the GP-DMDs achieves satisfactory results in reproducing the demonstrations on all plots. However, the GP-DMD does not guarantee that the drawn samples exhibit consistent ratios between position, velocity, and acceleration.

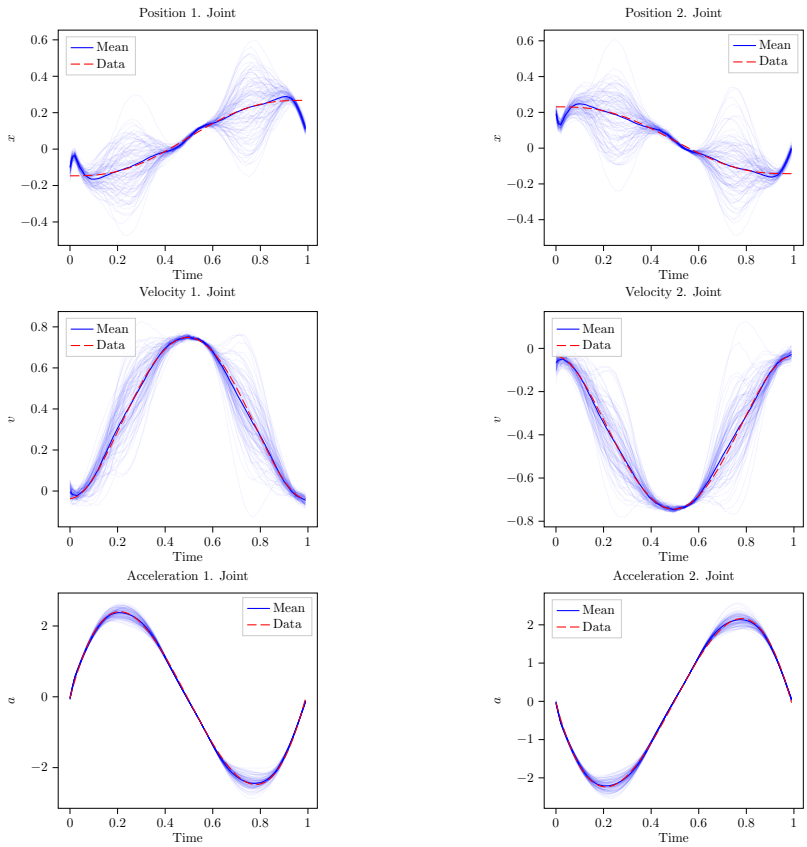


Figure 6.3.: This figure shows a benchmark of the derived Bayesian GP-DMDs on the Minimum-Jerk Dataset (MJD). The MJD corresponds to the movement of two joints with their respective position, velocity, and acceleration. The movement of the first and second joint are shown on the left and right side, respectively. The red dashed lines represent the given demonstrations. The thick blue line presents the predicted mean estimate, while the thin blue lines represent samples of demonstrations. The predicted mean value of Bayesian GP-DMDs achieves satisfactory results only for the demonstrations of the velocities and the accelerations shown in the lower four plots. In the upper two plots, strong deviations from the given positions occur. Also, the GP-DMD does not guarantee that the drawn samples exhibit consistent ratios between position, velocity, and acceleration.

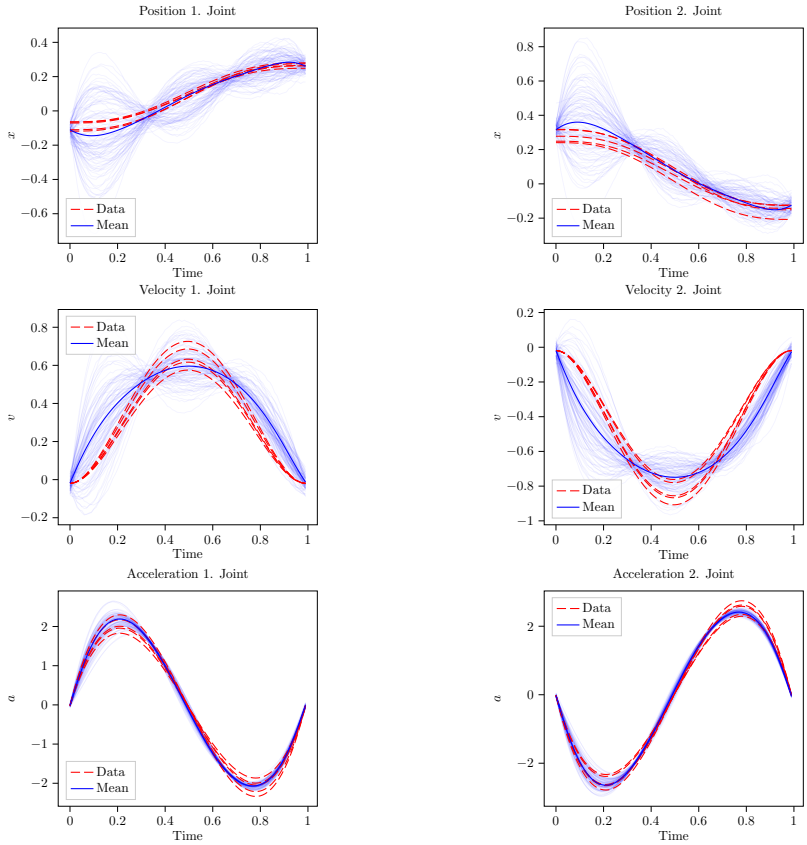


Figure 6.4.: This figure shows a benchmark of the derived Pro-DMPs on the Minimum-Jerk Dataset (MJD). The MJD corresponds to the movement of two joints with their respective position, velocity, and acceleration. The movement of the first and second joint are shown on the left and right side, respectively. The red dashed lines represent the given demonstrations. During this benchmark, five demonstrations are considered. The thick blue line presents the predicted mean estimate, while the thin blue lines represent samples of demonstrations. The predicted mean value of Bayesian GP-DMDs achieves satisfactory results only for the demonstrations of the accelerations shown in the lower two plots. In the upper four plots, strong deviations from the given positions and velocities occur. Also, the GP-DMD does not guarantee that the drawn samples exhibit consistent ratios between position, velocity, and acceleration.

Root Mean Square Error on Datasets			
Name of Dataset	CSD	ESD	MJD
GP-DMD	$\mathbf{1.98e^{-5} \pm 4.16e^{-3}}$	$3.16e^{-3} \pm 1.6e^{-2}$	$9.13e^{-3} \pm 2.06e^{-2}$
Bayesian GP-DMD	$7.59e^{-5} \pm 1.38e^{-3}$	$\mathbf{4.59e^{-4} \pm 3.49e^{-2}}$	$\mathbf{4.35e^{-3} \pm 3.79e^{-2}}$
Pro-DMP	$7.93e^{-3} \pm 1.21e^{-2}$	$3.69e^{-2} \pm 8.71e^{-3}$	$3.95e^{-2} \pm 6.37e^{-2}$

Table 6.1.: This list summarizes the performance of Gaussian Process Dynamic Mode Decomposition (GP-DMD), Bayesian Gaussian Process Dynamic Mode Decomposition (Bayesian GP-DMD) and Probabilistic Dynamic Mode Primitive (Pro-DMP) benchmarked on the Circle-Shape Dataset (CSD), the Eight-Shape Dataset (ESD), and the Minimum-Jerk Dataset (MJD). Performance is measured by the Root Mean Square Error (RMSE) between the predicted mean and the given demonstration. In the latter case of Pro-DMPs, the predicted mean is compared to the average of the five given demonstrations. In order to obtain a measure of variability in the results, the standard deviation is used w.r.t. the estimated RMSE error. For this purpose, one hundred samples are taken from each framework. Then, the RMSE between those samples and the corresponding given demonstration is calculated. Thus, the former value in each cell represents the RMSE mean, and the latter value represents the standard deviation. The thick values indicate the best RMSE mean performance on the corresponding data set.

the positions, the velocities, and the acceleration. For instance, the true first and second derivatives of a sampled position trajectory differ from the corresponding sampled velocity and acceleration trajectories.

7. Discussion and Outlook

Machine Learning techniques are attracting increasing attention to analyze and understand the intrinsic nature of given data from highly complex systems as data deluge and the technologies' ever-increasing computational power continues to grow [2, 4, 6, 25, 26, 69]. Movement primitives, a subdomain of imitation learning in robotics, leverage Machine Learning to learn movement sequences from demonstrations. [7–21]. These concepts allow the reconstruction of given demonstrations solely from data and enable the modification of those movements afterwards. Unfortunately, many of the proposed frameworks lack the extraction of comprehensible and interpretable physical information within the learned latent space. In this thesis, we focus on the Koopman Theory, which addresses highly complex dynamical systems [22–27]. This theory considers a linear evolution of selected measurement functions of the data in an infinite-dimensional Hilbert space instead of the nonlinear evolution of the collected data points themselves. The Koopman Theory and hence the DMD family provide a decomposition into spatio-temporal characteristics of the given dynamical system and consequently into interpretable physical information [24–26, 28–31].

Motivated by the strong similarity of the DMD family to continuous LVMs, such as PCA and FA, this thesis firstly proposes the GP-DMD, providing a probabilistic dual perspective based on the GP-SSM family. The GP-DMD framework embodies a MAP estimate aiming to infer a stationary linear Markov sequence in latent space, while a Gaussian Process describes the relationship to the given observations. The probabilistic formulation accounts for uncertainties and noise naturally and leads to a nonparametric Bayesian formalism due to the Gaussian Process. However, since the latent space can exhibit a higher dimensionality than the observations space, the GP-DMD framework carries a risk of overfitting. Therefore, the Bayesian GP-DMD, a fully Bayesian formalism of the GP-DMD, has been subsequently introduced in this thesis. This formalization leads to a Variational Inference procedure, which mitigates overfitting and approximates a posterior distribution over the linear operator and the linear trajectories in the latent space.

Movement primitives focus not solely on learning the underlying movement but also on learning the variability within the given demonstrations [7–10]. Both GP-DMD and the Bayesian GP-DMD fail to cope with multiple demonstrations. For this reason, the thesis proposes the Pro-DMP as a third framework. The Pro-DMP targets to express the given variability within the given demonstrations in the inferred trajectories in the latent space. Based on Probabilistic Movement Primitives, a popular movement primitive framework [10, 14, 15, 18], a hierarchical structure has been integrated into the existing GP-DMD framework. As a result, the Pro-DMP is an EM-like procedure capable of handling multiple trajectories and representing a new class of movement primitives. In order to mitigate the risk of overfitting, the Bayesian-DMP framework was introduced, which is a fully Bayesian formalization of the Pro-DMP. Similar to Bayesian GP-DMD, it expresses a Variational Inference procedure and hence provides approximations of the posterior distributions over the linear operator and the linear trajectories in the latent space. However, the development of this framework is ongoing and forms an open research topic.

The GP-DMD, the Bayesian GP-DMD, and the Pro-DMP were benchmarked on the CSD, the ESD, and the MJD. Here, the primary objective was to determine whether the frameworks are capable of learning and reproducing the given demonstrations. The GP-DMD showed the significant advantage of computational efficiency in comparison to the Bayesian GP-DMD. Even though the inducing pairs utilized in the Bayesian GP-DMD reduce the computational complexity of the kernel inversion compared to the GP-DMD, the Probabilistic Inference techniques required to infer the variational distribution over the latent states significantly slow down the Bayesian GP-DMD. However, a major drawback of the GP-DMD is the necessity of multiple labels to reproduce the given demonstration. The performance of the resulting demonstrations, however, is strongly dependent on the choice of these labels. The Bayesian GP-DMD circumvents this issue by using the inducing variables. These variables directly infer knowledge about the given demonstrations to provide labels to reproduce them indirectly. Like the GP-DMD, the Pro-DMP also requires a fixed number of labels in the observation space to reproduce the demonstrations. In this experiment, only the mean estimates in the latent space were considered. Therefore, it is interesting for future research to sample in the latent space and reproduce the corresponding demonstration in the observation space. In this way, the ability of Pro-DMP to learn the variability of the circular motion from the given demonstrations can be analyzed.

In the future development of these proposed frameworks, it is interesting to evaluate their performance on more comprehensive robotic benchmarks. The variation in potential values, e.g., the choice of kernel functions, plays a significant role in the resulting performance. In the case of fully Bayesian methods, such as Bayesian GP-DMD and Bayesian-DMP, the assumption of the specific functional form of the variational distribution over the latent

states and the choice of the Probabilistic Inference technique employed for this purpose is essential. When satisfactory performance in complex experiments is achieved, comparison with other movement primitives is an important part of future research. Also, the analysis of the underlying learned latent space constitutes an important part.

A significant challenge in the proposed frameworks arises from the bi-level optimization procedure. On the one hand, the framework aims to learn a linear transition model in the latent space. On the other hand, the linear operator is optimized based on these estimates. Here, the corresponding optimization procedure relies on first-order coordinate gradient descent methods. For parameters that have a closed-form solution, a learning rate of 1.0 is used. A vanilla gradient descent is performed on the remaining parameters for which no closed-form solution exists. Based on the stochastic VI [70, 73], the future focus lies on extending the proposed frameworks. Advantages of the natural gradient are to be used, and as such, information of the resulting Hessian matrix is to be included [70, 98]. These extended frameworks result in more natural optimization procedures considering second-order information.

The assumed prior distribution over the linear operator, expressed as a stationary matrix,is considered separately for each column in the proposed frameworks. In other words, the columns of the linear operator are independent of each other. From the perspective of Occam's Razor, this assumption is reasonable since it simplifies the optimization problem and provides decent approximations. However, it still has severe limitations since dependencies between columns are neglected. An interesting extension is the utilization of matrix priors [99]. These priors result in a richer expression that considers the entire matrix without independence assumptions. However, this consideration induces correlations and therefore entails the risk of leading to more complex optimization procedures. Similarly, future research studies can consider extending the Gaussian processes to the Student-t process [100, 101]. This consideration has the potential to result in significantly better performances of the derived frameworks. However, similar to the use of matrix priors, the induced correlations lead to more complex optimization procedures due to marginalization over precision values. Thus, from Occam's Razor perspective, these considerations are not the primary goals of future research, but they do provide interesting food for thought.

A major focus of future research is developing the Bayesian-DMP. However, the VI procedures are based on the optimization of a lower bound and, consequently, on applying the mean field assumption. On the one hand, the assumed independence between the random variables naturally results in specific functional forms of the variational distribution [4, 6]. On the other hand, however, this independence potentially induces flawed approximated posterior distribution [54]. These flawed approximations potentially lead

to a bias between the true marginal log-likelihood and the ELBO, resulting in unsatisfactory solutions. Therefore, an interesting way to circumvent the problem is to derive a sampling-based Probabilistic Inference technique for the Bayesian-DMP framework based on Particle MCMC [54, 58]. Such a technique suffers from the curse of dimensionality and becomes less suitable as the dimensionality in the demonstrations increases. However, it have the advantage that no mean field assumption has to be made.

Bibliography

[1] J. Friedman, T. Hastie, R. Tibshirani, *et al.*, *The elements of statistical learning*. Springer series in statistics New York, 2001.

[2] D. J. MacKay and D. J. Mac Kay, *Information theory, inference and learning algorithms*. Cambridge university press, 2003.

[3] M. J. Beal, *Variational algorithms for approximate Bayesian inference*. PhD thesis, UCL (University College London), 2003.

[4] C. M. Bishop, *Pattern recognition and machine learning*. springer, 2006.

[5] D. Barber, *Bayesian reasoning and machine learning*. Cambridge University Press, 2012.

[6] K. P. Murphy, *Machine learning: a probabilistic perspective*. MIT press, 2012.

[7] A. J. Ijspeert, J. Nakanishi, H. Hoffmann, P. Pastor, and S. Schaal, "Dynamical movement primitives: learning attractor models for motor behaviors," *Neural computation*, 2013.

[8] A. Billard, S. Calinon, R. Dillmann, and S. Schaal, *Robot Programming by Demonstration*. Springer Berlin Heidelberg, 2008.

[9] S. Calinon, "A tutorial on task-parameterized movement learning and retrieval," *Intelligent service robotics*, 2016.

[10] A. Paraschos, C. Daniel, J. Peters, and G. Neumann, "Using probabilistic movement primitives in robotics," *Autonomous Robots*, 2018.

[11] A. J. Ijspeert, J. Nakanishi, and S. Schaal, "Learning attractor landscapes for learning motor primitives," tech. rep., 2002.

[12] S. Schaal, "Dynamic movement primitives-a framework for motor control in humans and humanoid robotics," in *Adaptive motion of animals and machines*, Springer, 2006.

[13] S. Calinon, F. Guenter, and A. Billard, "On learning, representing, and generalizing a task in a humanoid robot," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 2007.

[14] A. Paraschos, C. Daniel, J. Peters, G. Neumann, *et al.*, "Probabilistic movement primitives," *Advances in neural information processing systems*, 2013.

[15] A. Paraschos, G. Neumann, and J. Peters, "A probabilistic approach to robot trajectory generation," in *2013 13th IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, IEEE, 2013.

[16] S. Calinon, F. D'halluin, E. L. Sauser, D. G. Caldwell, and A. G. Billard, "Learning and reproduction of gestures by imitation," *IEEE Robotics & Automation Magazine*, 2010.

[17] S. M. Khansari-Zadeh and A. Billard, "Learning stable nonlinear dynamical systems with gaussian mixture models," *IEEE Transactions on Robotics*, 2011.

[18] M. Ewerton, G. Maeda, J. Peters, and G. Neumann, "Learning motor skills from partially observed movements executed at different speeds," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2015.

[19] Y. Huang, L. Rozo, J. Silvério, and D. G. Caldwell, "Kernelized movement primitives," *The International Journal of Robotics Research*, 2019.

[20] J. Urain, M. Ginesi, D. Tateo, and J. Peters, "Imitationflow: Learning deep stable stochastic dynamic systems by normalizing flows," *arXiv preprint arXiv:2010.13129*, 2020.

[21] T. Kulak, J. Silvério, and S. Calinon, "Fourier movement primitives: an approach for learning rhythmic robot skills from demonstrations," in *Robotics: Science and Systems (RSS)*, 2020.

[22] B. O. Koopman, "Hamiltonian systems and transformation in hilbert space," *Proceedings of the national academy of sciences of the united states of america*, 1931.

[23] B. Koopman and J. v. Neumann, "Dynamical systems of continuous spectra," *Proceedings of the National Academy of Sciences of the United States of America*, 1932.

[24] I. Mezić, "Spectral properties of dynamical systems, model reduction and decompositions," *Nonlinear Dynamics*, 2005.

[25] J. N. Kutz, S. L. Brunton, B. W. Brunton, and J. L. Proctor, *Dynamic mode decomposition: data-driven modeling of complex systems*. SIAM, 2016.

[26] S. L. Brunton and J. N. Kutz, *Data-driven science and engineering: Machine learning, dynamical systems, and control*. Cambridge University Press, 2019.

[27] S. L. Brunton, B. W. Brunton, J. L. Proctor, and J. N. Kutz, "Koopman invariant subspaces and finite linear representations of nonlinear dynamical systems for control," *PloS one*, 2016.

[28] J. H. Tu, C. W. Rowley, D. M. Luchtenburg, S. L. Brunton, and J. N. Kutz, "On dynamic mode decomposition: Theory and applications," *arXiv preprint arXiv:1312.0041*, 2013.

[29] M. O. Williams, I. G. Kevrekidis, and C. W. Rowley, "A data–driven approximation of the koopman operator: Extending dynamic mode decomposition," *Journal of Nonlinear Science*, 2015.

[30] I. Kevrekidis, C. W. Rowley, and M. Williams, "A kernel-based method for data-driven koopman spectral analysis," *Journal of Computational Dynamics*, 2015.

[31] Y. Kawahara, "Dynamic mode decomposition with reproducing kernels for koopman spectral analysis," *Advances in neural information processing systems*, 2016.

[32] C. W. Rowley, I. MEZIC, S. Bagheri, P. Schlatter, D. Henningson, *et al.*, "Spectral analysis of nonlinear flows," *Journal of fluid mechanics*, 2009.

[33] P. J. Schmid, "Dynamic mode decomposition of numerical and experimental data," *Journal of fluid mechanics*, 2010.

[34] P. J. Schmid, L. Li, M. P. Juniper, and O. Pust, "Applications of the dynamic mode decomposition," *Theoretical and Computational Fluid Dynamics*, 2011.

[35] C. M. Bishop, "Latent variable models," in *Learning in graphical models*, Springer, 1998.

[36] L. E. Baum, T. Petrie, G. Soules, and N. Weiss, "A maximization technique occurring in the statistical analysis of probabilistic functions of markov chains," *The annals of mathematical statistics*, 1970.

[37] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the Royal Statistical Society: Series B (Methodological)*, 1977.

[38] C. E. Rasmussen, "Gaussian processes in machine learning," in *Summer school on machine learning*, Springer, 2003.

[39] C. E. Rasmussen and C. KI Williams, *Gaussian Processes for Machine Learning*. MIT Press, 2006.

[40] A. J. Smola and P. L. Bartlett, "Sparse greedy gaussian process regression," in *Advances in neural information processing systems*, 2001.

[41] L. Csató and M. Opper, "Sparse on-line gaussian processes," *Neural computation*, 2002.

[42] M. Seeger, C. Williams, and N. Lawrence, "Fast forward selection to speed up sparse gaussian process regression," tech. rep., 2003.

[43] J. Quinonero-Candela and C. E. Rasmussen, "A unifying view of sparse approximate gaussian process regression," *The Journal of Machine Learning Research*, 2005.

[44] N. D. Lawrence, "Gaussian process latent variable models for visualisation of high dimensional data.," in *Nips*, Citeseer, 2003.

[45] N. Lawrence and A. Hyvärinen, "Probabilistic non-linear principal component analysis with gaussian process latent variable models.," *Journal of machine learning research*, 2005.

[46] N. D. Lawrence, "Learning for larger datasets with the gaussian process latent variable model," in *Artificial intelligence and statistics*, PMLR, 2007.

[47] N. D. Lawrence and A. J. Moore, "Hierarchical gaussian process latent variable models," in *Proceedings of the 24th international conference on Machine learning*, pp. 481–488, 2007.

[48] A. C. Damianou, M. K. Titsias, and N. D. Lawrence, "Variational gaussian process dynamical systems," *arXiv preprint arXiv:1107.4985*, 2011.

[49] A. Damianou and N. D. Lawrence, "Deep gaussian processes," in *Artificial intelligence and statistics*, PMLR, 2013.

[50] M. Titsias and N. D. Lawrence, "Bayesian gaussian process latent variable model," in *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, JMLR Workshop and Conference Proceedings, 2010.

[51] N. Takeishi, Y. Kawahara, Y. Tabei, and T. Yairi, "Bayesian dynamic mode decomposition.," in *IJCAI*, 2017.

[52] R. H. Shumway, D. S. Stoffer, and D. S. Stoffer, *Time series analysis and its applications*. Springer, 2000.

[53] S. Särkkä, *Bayesian filtering and smoothing*. Cambridge University Press, 2013.

[54] R. Frigola, *Bayesian time series learning with Gaussian processes*. PhD thesis, University of Cambridge, 2015.

[55] B. Ferris, D. Fox, and N. D. Lawrence, "Wifi-slam using gaussian process latent variable models.," in *IJCAI*, 2007.

[56] J. M. Wang, D. J. Fleet, and A. Hertzmann, "Gaussian process dynamical models," in *NIPS*, Citeseer, 2005.

[57] J. M. Wang, D. J. Fleet, and A. Hertzmann, "Gaussian process dynamical models for human motion," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2008.

[58] R. Frigola, F. Lindsten, T. B. Schön, and C. E. Rasmussen, "Bayesian inference and learning in gaussian process state-space models with particle mcmc," *arXiv preprint arXiv:1306.2861*, 2013.

[59] R. Frigola, Y. Chen, and C. E. Rasmussen, "Variational gaussian process state-space models," *arXiv preprint arXiv:1406.4905*, 2014.

[60] R. Frigola, F. Lindsten, T. B. Schön, and C. E. Rasmussen, "Identification of gaussian process state-space models with particle stochastic approximation em," *IFAC Proceedings Volumes*, 2014.

[61] S. Marsland, *Machine learning: an algorithmic perspective*. CRC press, 2015.

[62] J. L. W. V. Jensen *et al.*, "Sur les fonctions convexes et les inégalités entre les valeurs moyennes," *Acta mathematica*, 1906.

[63] M. I. Jordan, Z. Ghahramani, T. S. Jaakkola, and L. K. Saul, "An introduction to variational methods for graphical models," *Machine learning*, 1999.

[64] D. G. Tzikas, A. C. Likas, and N. P. Galatsanos, "The variational approximation for bayesian inference," *IEEE Signal Processing Magazine*, 2008.

[65] W. R. Gilks, S. Richardson, and D. J. Spiegelhalter, "Introducing markov chain monte carlo," *Markov chain Monte Carlo in practice*, 1995.

[66] I. Yildirim, "Bayesian inference: Gibbs sampling," *Technical Note, University of Rochester*, 2012.

[67] D. Van Ravenzwaaij, P. Cassey, and S. D. Brown, "A simple introduction to markov chain monte–carlo sampling," *Psychonomic bulletin & review*, 2018.

[68] Z. Ghahramani and M. J. Beal, "Propagation algorithms for variational bayesian learning," *Advances in neural information processing systems*, pp. 507–513, 2001.

[69] M. J. Wainwright and M. I. Jordan, *Graphical models, exponential families, and variational inference*. Now Publishers Inc, 2008.

[70] M. D. Hoffman, D. M. Blei, C. Wang, and J. Paisley, "Stochastic variational inference," *Journal of Machine Learning Research*, 2013.

[71] R. Ranganath, S. Gerrish, and D. Blei, "Black box variational inference," in *Artificial intelligence and statistics*, PMLR, 2014.

[72] D. M. Blei, A. Kucukelbir, and J. D. McAuliffe, "Variational inference: A review for statisticians," *Journal of the American statistical Association*, 2017.

[73] C. Zhang, J. Bütepage, H. Kjellström, and S. Mandt, "Advances in variational inference," *IEEE transactions on pattern analysis and machine intelligence*, 2018.

[74] S. Boyd, S. P. Boyd, and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[75] J. Hensman, N. Fusi, and N. D. Lawrence, "Gaussian processes for big data," *arXiv preprint arXiv:1309.6835*, 2013.

[76] E. Snelson and Z. Ghahramani, "Sparse gaussian processes using pseudo-inputs," *Advances in neural information processing systems*, 2005.

[77] M. Titsias, "Variational learning of inducing variables in sparse gaussian processes," in *Artificial intelligence and statistics*, PMLR, 2009.

[78] J. Hensman, A. Matthews, and Z. Ghahramani, "Scalable variational gaussian process classification," in *Artificial Intelligence and Statistics*, PMLR, 2015.

[79] A. Wilson and H. Nickisch, "Kernel interpolation for scalable structured gaussian processes (kiss-gp)," in *International Conference on Machine Learning*, PMLR, 2015.

[80] H. Liu, Y.-S. Ong, X. Shen, and J. Cai, "When gaussian process meets big data: A review of scalable gps," *IEEE transactions on neural networks and learning systems*, 2020.

[81] C. Williams and M. Seeger, "Using the nyström method to speed up kernel machines," in *Proceedings of the 14th annual conference on neural information processing systems*, 2001.

[82] A. Gittens and M. Mahoney, "Revisiting the nystrom method for improved large-scale machine learning," in *International Conference on Machine Learning*, PMLR, 2013.

[83] C. K. Williams, C. E. Rasmussen, A. Scwaighofer, and V. Tresp, "Observations on the nyström method for gaussian process prediction," 2002.

[84] G. D. Birkhoff, *Dynamical systems*. American Mathematical Soc., 1927.

[85] N. Takeishi, Y. Kawahara, and T. Yairi, "Learning koopman invariant subspaces for dynamic mode decomposition," *arXiv preprint arXiv:1710.04340*, 2017.

[86] B. Lusch, J. N. Kutz, and S. L. Brunton, "Deep learning for universal linear embeddings of nonlinear dynamics," *Nature communications*, 2018.

[87] S. T. Dawson, M. S. Hemati, M. O. Williams, and C. W. Rowley, "Characterizing and correcting for the effect of sensor noise in the dynamic mode decomposition," *Experiments in Fluids*, 2016.

[88] M. S. Hemati, C. W. Rowley, E. A. Deem, and L. N. Cattafesta, "De-biasing the dynamic mode decomposition for applied koopman spectral analysis of noisy datasets," *Theoretical and Computational Fluid Dynamics*, 2017.

[89] D. Matsumoto and T. Indinger, "On-the-fly algorithm for dynamic mode decomposition using incremental singular value decomposition and total least squares," *arXiv preprint arXiv:1703.11004*, 2017.

[90] J. Ko and D. Fox, "Gp-bayesfilters: Bayesian filtering using gaussian process prediction and observation models," *Autonomous Robots*, 2009.

[91] M. P. Deisenroth, R. D. Turner, M. F. Huber, U. D. Hanebeck, and C. E. Rasmussen, "Robust filtering and smoothing with gaussian processes," *IEEE Transactions on Automatic Control*, 2011.

[92] R. Urtasun, D. J. Fleet, and P. Fua, "3d people tracking with gaussian process dynamical models," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, IEEE, 2006.

[93] T. Beckers and S. Hirche, "Stability of gaussian process state space models," in *2016 European Control Conference (ECC)*, IEEE, 2016.

[94] S. Eleftheriadis, T. Nicholson, M. P. Deisenroth, and J. Hensman, "Identification of gaussian process state space models," in *NIPS*, 2017.

[95] J. Bradbury, R. Frostig, P. Hawkins, M. J. Johnson, C. Leary, D. Maclaurin, G. Necula, A. Paszke, J. VanderPlas, S. Wanderman-Milne, and Q. Zhang, "JAX: composable transformations of Python+NumPy programs," 2018.

[96] D. Phan, N. Pradhan, and M. Jankowiak, "Composable effects for flexible and accelerated probabilistic programming in numpyro," *arXiv preprint arXiv:1912.11554*, 2019.

[97] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[98] S.-I. Amari, "Natural gradient works efficiently in learning," *Neural computation*, 1998.

[99] A. K. Gupta and D. K. Nagar, *Matrix variate distributions*. CRC Press, 2018.

[100] A. Shah, A. Wilson, and Z. Ghahramani, "Student-t processes as alternatives to gaussian processes," in *Artificial intelligence and statistics*, PMLR, 2014.

[101] B. D. Tracey and D. Wolpert, "Upgrading from gaussian processes to student'st processes," in *2018 AIAA Non-Deterministic Approaches Conference*, 2018.

A. Probability Distributions

The following appendix provides the probability distributions relevant to this work. The definitions and formalizations are taken from [4, 6].

A.1. Normal Distribution

The *normal distribution*, also known as *Gaussian distribution*, is the most commonly used distribution in statistics and Machine Learning [6]. In the univariate case, where the corresponding random variables are one-dimensional $x \in \mathbb{R}$, the probability density function is defined by

$$\mathcal{N}(x \mid \mu, \sigma) = \frac{1}{(2\pi\sigma^2)^{\frac{1}{2}}} \exp\left(-\frac{1}{2\sigma^2}(x-\mu)^2\right).$$

The natural parameters $\mu \in \mathbb{R}$ and $\sigma^2 > 0$ denote the mean and variance of the univariate normal distribution, respectively. The variance $\sigma^2 > 0$ is the squared standard deviation. In many cases, including this work, the precision value of the normal distribution $\lambda = 1/\sigma^2$ is used, which is simply the inverse of the variance. The first and second moments of the univariate normal distribution are denoted by

$$\begin{aligned} \text{Mean}: &\qquad \mathbb{E}(x) = \mu, \\ \text{Variance}: &\qquad \text{var}(x) = \sigma^2. \end{aligned}$$

The conjugate prior of the mean μ is again a univariate normal distribution, while for σ^2 an *inverse Gamma distribution* is assumed. If the precision value λ is used, the conjugate prior is formalized by an *Gamma distribution*.

In the multivariate case, where a normal distribution over a D-dimensional random variable $\mathbf{x} \in \mathbb{R}^D$ is considered, the probability density function equals

$$\mathcal{N}(\mathbf{x} \mid \boldsymbol{\mu}, \boldsymbol{\Sigma}) = (2\pi)^{-\frac{D}{2}} |\boldsymbol{\Sigma}|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right).$$

The natural parameters of the distribution are the mean vector $\boldsymbol{\mu} \in \mathbb{R}^D$ and the positive semidefinite covariance matrix $\boldsymbol{\Sigma} \in \mathbb{R}^{D \times D}$. As in the univariate case, the precision matrix $\boldsymbol{\Lambda} = \boldsymbol{\Sigma}^{-1}$ is simply expressed as the inverse of the covariance matrix. Moreover, in the multivariate case, the first and second moments correspond to

$$\begin{aligned} \text{Mean}: &\qquad \mathbb{E}(\mathbf{x}) = \boldsymbol{\mu}, \\ \text{Variance}: &\qquad \text{Cov}(\mathbf{x}) = \boldsymbol{\Sigma}. \end{aligned}$$

Like in the univariate case, the conjugate prior associated to the mean vector $\boldsymbol{\mu}$ is again a multivariate normal distribution. For the covariance and precision matrices, the conjugate prior is denoted as *inverse Wishart distribution* and *Wishart distribution*, respectively.

A.2. Gamma Distribution

The *Gamma distribution* is a flexible distribution over positive real-valued random variables $x \in (0, \infty)$. The probability density function is given by

$$\text{Gam}(x \mid \alpha, \beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} x^{\alpha-1} \exp(-\beta x),$$

where the natural parameters $\alpha > 0$ and $\beta > 0$ denote the shape and rate of the given distribution. It is normalized by a *gamma function* defined by

$$\Gamma(\alpha) \stackrel{\text{def}}{=} \int_0^\infty u^{\alpha-1} \exp^{-u} \, \mathrm{d}u,$$

which ensures that the probability integrates to one. The mean and variance, or the first and second moments, are expressed by

$$\begin{aligned} \mathbb{E}(x) &= \frac{\alpha}{\beta} \\ \text{var}(x) &= \frac{\alpha}{\beta^2}. \end{aligned}$$

The Gamma distribution represents the conjugate prior for the precision value λ of a normal distribution.

A.3. Wishart Distribution

In the multivariate case, the generalization of the Gamma distribution is represented by the *Wishart distribution*. This distribution models the uncertainty of positive semidefinite matrices $\mathbf{\Lambda} \in \mathbb{R}^{D \times D}$ by

$$\mathcal{W}(\mathbf{\Lambda} \mid \mathbf{W}, \nu) = B(\mathbf{W}, \nu)|\mathbf{\Lambda}|^{\frac{\nu-D-1}{2}} \exp\left(-\frac{1}{2}\mathrm{Tr}\left(\mathbf{W}^{-1}\mathbf{\Lambda}\right)\right),$$

where the natural parameters ν and $\mathbf{W}$ correspond to the *degrees of freedom* and the *scale matrix*, respectively. The normalization constant is computable by the function $B(\mathbf{W}, \nu)$ given as

$$B(\mathbf{W}, \nu) = |\mathbf{W}|^{-\frac{\nu}{2}} \left(2^{\frac{\nu D}{2}} \pi^{\frac{D(D-1)}{4}} \prod_{d=1}^{D} \Gamma\left(\frac{\nu + 1 - i}{2}\right)\right)^{-1}.$$

Note that this normalization factor exists only when $\nu > D - 1$ holds. The expected value and hence the first moment is formed by

$$\mathbb{E}(\mathbf{\Lambda}) = \nu \mathbf{W}.$$

The Wishart distribution acts as a conjugate prior for the precision matrix $\mathbf{\Lambda}$.

B. Information Theory

This appendix presents significant concepts from the field of Information Theory that have proven to be useful tools for Machine Learning and other related research areas. The definitions and formalizations are based on [4, 6].

B.1. Shannon Entropy

The *entropy* or *Shannon entropy* represents a measure of the uncertainty of a probability density function $p(\cdot)$ over a random variable $\mathbf{x}$. In other words, it provides a value that determines the information content of an event. This information content is also called *degree of surprise*. The value indicates the information content of an event under the assumption of a given distribution. Assuming the random variable has a high probability value, the event is very likely to occur and therefore has a low informative content. On the other hand, a very unlikely event has much informational content. The entropy is defined by

$$\mathrm{H}(p(\mathbf{x}) \parallel p(\mathbf{x})) = -\int p(\mathbf{x}) \log p(\mathbf{x}) \, \mathrm{d}\mathbf{x}.$$

For discrete random variables and the resulting probability mass function; a sum replaces the integral. From an information-theoretic perspective, the Shannon entropy corresponds to the expected number of bits needed to encode the data from the source distribution p.

B.2. Cross Entropy

The *cross entropy* is determined by

$$\mathrm{H}(p(\mathbf{x}) \parallel q(\mathbf{x})) = -\int p(\mathbf{x}) \log q(\mathbf{x}) \, \mathrm{d}\mathbf{x}.$$

In the case of a probability mass function, a sum replaces the integral. It quantifies the information content necessary to represent data from a true distribution $p(\cdot)$ by a modeled distribution $q(\cdot)$. For the case where the true and given distributions coincide, this becomes the Shannon entropy.

B.3. Kullback–Leibler Divergence

A combination of the previously proposed concepts provides a measure of dissimilarity between two probability distributions $p(\cdot)$ and $q(\cdot)$. This measure function is known as *Kullback–Leibler divergence (KL)* or *relative entropy*. It is defined as follows

$$\begin{aligned}\mathrm{KL}(p(\mathbf{x}) \parallel q(\mathbf{x})) &= -\mathrm{H}(p(\mathbf{x}) \parallel p(\mathbf{x})) + \mathrm{H}(p(\mathbf{x}) \parallel q(\mathbf{x})), \\ &= \int p(\mathbf{x}) \log \frac{q(\mathbf{x})}{p(\mathbf{x})} \, \mathrm{d}\mathbf{x},\end{aligned}$$

where the integral is replaceable by the sum when probability mass functions are considered. It represents the difference between cross entropy and entropy. The KL-divergence thus quantifies the additional information content needed to express the data coming from $p(\cdot)$ with the distribution $q(\cdot)$. In this context, it should be noted that the Kullback–Leibler divergence is not a distance measure mathematically, since the symmetry property $\mathrm{KL}(p(\cdot) \parallel q(\cdot)) \neq \mathrm{KL}(q(\cdot) \parallel p(\cdot))$ is violated.

C. Kernel Functions

The kernel functions listed in this appendix closely follow the definitions of [6]. In general, a kernel matrix, also known as *Gram matrix*, defined by

$$\mathbf{K}_{\mathbf{X},\mathbf{Y}}(\boldsymbol{\theta}) = \begin{bmatrix} \mathbf{k}(\mathbf{x}_0, \mathbf{y}_0, \boldsymbol{\theta}) & \cdots & \mathbf{k}(\mathbf{x}_0, \mathbf{y}_T, \boldsymbol{\theta}) \\ \vdots & \ddots & \vdots \\ \mathbf{k}(\mathbf{x}_T, \mathbf{y}_0, \boldsymbol{\theta}) & \cdots & \mathbf{k}(\mathbf{x}_T, \mathbf{y}_T, \boldsymbol{\theta}) \end{bmatrix},$$

describes the similarity between given objects without explicitly requiring the computation of the feature vector format. In each element of the kernel matrix, a kernel function $\mathbf{k}(\cdot, \cdot) \geq 0$ is applied. Given two objects $\mathbf{x}_i, \mathbf{y}_j \in \mathcal{M}$ of an abstract space $\mathcal{M}$, the kernel function $\mathbf{k}(\mathbf{x}_i, \mathbf{y}_j)$ provides a value associated with the similarity of the two objects. Typically, the resulting kernel matrix $\mathbf{K}$ is a symmetric positive semidefinite matrix [6].

C.1. Linear Kernel

The *linear kernel* represents a straightforward choice of possible kernel functions where each element of the gram matrix is expressed by an inner product

$$\mathbf{k}(\mathbf{x}, \mathbf{y}) = \theta \mathbf{x}^T \mathbf{y},$$

with θ denoting a scaling parameter. Consider the case of $\theta = 1$ and $\mathbf{x}, \mathbf{y} \in \mathbb{R}^2$, the linear kernel simply resolves into a linear combination

$$\begin{aligned} \mathbf{k}(\mathbf{x}, \mathbf{y}) &= \mathbf{x}^T \mathbf{y} \\ &= x_1 y_1 + x_2 y_2, \end{aligned}$$

of the elements of the two objects $\mathbf{x}$ and $\mathbf{y}$.

C.2. Polynomial Kernel

The *polynomial kernel* is a generalization of the linear kernel, which represents the inner product of two objects $\mathbf{x}, \mathbf{y} \in \mathcal{M}$ preprocessed by polynomial feature mappings. It is defined by

$$\mathbf{k}(\mathbf{x}, \mathbf{y}) = \left(\theta \mathbf{x}^T \mathbf{y} + r\right)^M,$$

where $r > 0$ and M denote the bias and the degree of the corresponding polynomial, respectively. For example, consider the following case of a polynomial kernel of the form

$$\begin{aligned}\mathbf{k}(\mathbf{x}, \mathbf{y}) &= \left(\mathbf{x}^T \mathbf{y} + 1\right)^2, \\ &= \left(1 + x_1 y_1 + x_2 y_2\right)^2, \\ &= 1 + 2x_1 y_1 + 2x_2 y_2 + 2x_1 x_2 y_1 y_2 + x_1^2 y_1^2 + x_2^2 y_2^2,\end{aligned}$$

where $\mathbf{x}, \mathbf{y} \in \mathbb{R}^2$, $M = 2$, $\theta = 1$, and $r = 1$. Hence, it expresses a polynomial of degree $M = 2$.

C.3. Family of RBF Kernels

The well-known *radial basis function kernel* is a special case of the generalized *squared exponential kernel* or *Gaussian kernel* defined by

$$\mathbf{k}(\mathbf{x}, \mathbf{y}) = \theta \exp\left(-\frac{1}{2}\left(\mathbf{x} - \mathbf{y}\right)^T \boldsymbol{\Sigma}^{-1} \left(\mathbf{x} - \mathbf{y}\right)\right),$$

where θ is the scaling parameter and the positive semidefinite matrix $\boldsymbol{\Sigma}$ provides a correlation parameter. In the case of a D-dimensional diagonal matrix $\boldsymbol{\Sigma}$, the squared exponential kernel resolves to

$$\mathbf{k}(\mathbf{x}_i, \mathbf{x}_j) = \theta \exp\left(-\frac{1}{2} \sum_{d=1}^{D} \frac{1}{\sigma_d^2} \left(x_i - x_j\right)^2\right),$$

which is known as *Automatic Relevance Detection (ARD) kernel*. The parameters σ^2 of the diagonal of $\boldsymbol{\Sigma}$ characterize the length or width scales of the individual dimensions,

respectively. If $\sigma^2 \to \infty$, the corresponding dimension has no relevance and is thus automatically ignored. Furthermore, if $\mathbf{\Sigma}$ is a spherical matrix, i.e. $\sigma^2 = \sigma_1^2 = \cdots = \sigma_D^2$, and a parameter $\gamma = 1/\sigma^2$ is defined, an isotropic kernel

$$\mathbf{k}(\mathbf{x}_i, \mathbf{x}_j) = \theta \exp\left(-\frac{\gamma}{2} ||\mathbf{x}_i - \mathbf{x}_j||^2\right), \tag{C.1}$$

is obtained, referring to a *radial basis function kernel*. The bandwidth is represented by γ.

D. Inference

This appendix discusses Probabilistic Inference techniques applied to State-Space Models (SSMs) (see Section 4.1). These techniques infer the probability of latent states $\mathbf{x}_0, \cdots, \mathbf{x}_T \in \mathbb{R}^M$ from given observed data $\mathbf{y}_0, \cdots, \mathbf{y}_T \in \mathbb{R}^N$. The latent and observed states are summarized in $\mathbf{X} \in \mathbb{R}^{M \times T}$ and $\mathbf{Y} \in \mathbb{R}^{N \times T}$, respectively. The posterior distribution is defined by

$$p(\mathbf{X} \mid \mathbf{Y}) = \frac{p(\mathbf{Y} \mid \mathbf{X}) p(\mathbf{X})}{p(\mathbf{Y})}.$$

The SSM frameworks assume a probabilistic Markov sequence in the latent space describing the dynamics and a measurement function representing the dependence of the observation $\mathbf{y}_t$ on the current state $\mathbf{x}_t$ [6, 53]. Thus, the posterior reformulates to

$$p(\mathbf{X} \mid \mathbf{Y}) = \frac{p(\mathbf{x}_0) p(\mathbf{y}_0 \mid \mathbf{x}_0) \prod_{t=1}^{T} p(\mathbf{y}_t \mid \mathbf{x}_t) p(\mathbf{x}_t \mid \mathbf{x}_{t-1})}{p(\mathbf{Y})}.$$

In the following, *Cubature Smoothing* and *Sequential Monte Carlo* present two established paradigms performing inference on the posterior. The definitions and formulations are based on [53].

D.1. Spherical Cubature Smoothing

The *Cubature Smoothing*, also known as *Cubature Rauch-Tung-Striebel smoother*, belongs to the family of Bayesian smoothing techniques. It combines the spherical Cubature approximation to classical additive Gaussian Rauch-Tung-Stribel smoother [53]. In general, smoothing methods compute the marginal distributions of a state $\mathbf{x}_t$ considering all measurements $\mathbf{Y}$. However, it is necessary to perform a forward procedure in advance. The

forward procedure, also called filtering, infers over a filter distribution. Filter distributions represent the marginal distributions of the current state $\mathbf{x}_t$, given by

$$p(\mathbf{x}_t \mid \mathbf{y}_0, \cdots, \mathbf{y}_t, \mathbf{x}_0, \cdots, \mathbf{x}_{t-1}) = p(\mathbf{x}_t \mid \mathbf{y}_t, \mathbf{x}_{t-1}), \tag{D.1}$$

depending on the past states $\mathbf{x}_0, \cdots, \mathbf{x}_{t-1}$ and the current and previous measurements $\mathbf{y}_0, \cdots, \mathbf{y}_t$. One appealing filtering algorithm is the well-known additive Gaussian *Cubature Kalman filter*, which is separable into a *prediction step* and an *update step*. The former predicts an estimate of the current state $\mathbf{x}_t$ based on the estimates of the mean $\mathbf{m}_{t-1} \in \mathbb{R}^M$ and the covariance matrix $\mathbf{P}_{k-1} \in \mathbb{R}^{N \times N}$ of the previous state $\mathbf{x}_{t-1}$. There are $2M$ *sigma points* defined as follows

$$\mathcal{X}_{t-1}^{(i)} = \mathbf{m}_{t-1} + \sqrt{\mathbf{P}_{k-1}}\xi^{(i)}, \quad i = 1, \cdots, 2M,$$

where the unit sigma points are given by

$$\xi^{(i)} = \begin{cases} \sqrt{M}\mathbf{e}_i, & i = 1, \cdots, M, \\ -\sqrt{M}\mathbf{e}_{i-M}, & i = M+1, \cdots, 2M, \end{cases} \tag{D.2}$$

The sigma points can be considered as deterministically selected samples representing a large amount of information of the underlying normal distribution. Then, these sigma points are propagated through the dynamical model

$$\widetilde{\mathcal{X}}_t^{(i)} = f\left(\mathcal{X}_{t-1}^{(i)}\right), \quad i = 1, \cdots, 2M,$$

where $f(\cdot)$ denotes the known linear or nonlinear dynamics of the underlying Markov sequence. Based on the resulting outcomes, the predicted mean $\mathbf{m}_t^- \in \mathbb{R}^M$ and the predicted covariance matrix $\mathbf{P}_t^- \in \mathbb{R}^{M \times M}$ are estimated by

$$\mathbf{m}_t^- = \frac{1}{2M} \sum_{i=1}^{2M} \widetilde{\mathcal{X}}_t^{(i)},$$

$$\mathbf{P}_t^- = \frac{1}{2M} \sum_{i=1}^{2M} \left(\widetilde{\mathcal{X}}_t^{(i)} - \mathbf{m}_t^-\right)\left(\widetilde{\mathcal{X}}_t^{(i)} - \mathbf{m}_t^-\right)^T + \mathbf{Q}_{t-1},$$

where $\mathbf{Q}_{t-1} \in \mathbb{R}^{M \times M}$ corresponds to the covariance matrix of the process noise.

After the prediction step, the natural parameters of the normal distribution of the current state $\mathbf{x}_t$ are estimated. To incorporate the knowledge of the seen observation $\mathbf{y}_t$, the

update step is then performed. For this purpose, the sigma points based on the current natural parameters are written as

$$\mathcal{X}_t^{-(i)} = \mathbf{m}_t^- + \sqrt{\mathbf{P}_t^-}\xi^{(i)}, \quad i = 1, \cdots, 2M,$$

with the vectors $\xi^{(i)}$ corresponding to the unit sigma points (see Equation (D.2)). Similar to the prediction step, they are propagated through the measurement model

$$\widetilde{\mathcal{Y}}_t^{(i)} = h\left(\mathcal{X}_t^{-(i)}\right), \quad i = 1, \cdots, 2M,$$

where $h(\cdot)$ describes the predefined measurement function of the SSM. Based on $\widetilde{\mathcal{Y}}_t^{(i)}$, the mean $\boldsymbol{\mu}_t$ and the covariance $\mathbf{S}_t$ are used to estimate the natural parameters of the likelihood $p(\mathbf{y}_t \mid \mathbf{x}_t)$ using

$$\boldsymbol{\mu}_t = \frac{1}{2M}\sum_{i=1}^{2M}\widetilde{\mathcal{Y}}_t^{(i)},$$

$$\mathbf{S}_t = \frac{1}{2M}\sum_{i=1}^{2M}\left(\widetilde{\mathcal{Y}}_t^{(i)} - \boldsymbol{\mu}_t\right)\left(\widetilde{\mathcal{Y}}_t^{(i)} - \boldsymbol{\mu}_t\right)^T + \mathbf{R}_t,$$

where $\mathbf{R}_t \in \mathbb{R}^{N\times N}$ represents the covariance matrix of the measurement noise. Moreover, the cross covariance of the state $\mathbf{x}_t$ and the observed state $\mathbf{y}_t$ is determined by the following equation

$$\mathbf{C}_t = \frac{1}{2M}\sum_{i=1}^{2M}\left(\widetilde{\mathcal{X}}_t^{(i)} - \mathbf{m}_t^-\right)\left(\widetilde{\mathcal{Y}}_t^{(i)} - \boldsymbol{\mu}_t\right)^T.$$

Thus, the natural parameters of the filter distribution (see Equation (D.1)) are represented by

$$\mathbf{m}_t = \mathbf{m}_t^- + \mathbf{K}_t\left(\mathbf{y}_t - \boldsymbol{\mu}_t\right),$$

$$\mathbf{P}_t = \mathbf{P}_t^- - \mathbf{K}_t\mathbf{S}_t\mathbf{K}_t^T,$$

where $\mathbf{K}_t = \mathbf{C}_t\mathbf{S}_t^{-1} \in \mathbb{R}^{M\times N}$ corresponds to the well-known *Kalman gain* or *filter gain*.

Eventually, the combination of both steps over each state $\mathbf{x}_0, \cdots, \mathbf{x}_T$ provides the forward procedure. However, to obtain an estimate of the true posterior, Bayesian smoothing

techniques are utilized, resulting in a backward path. Similar to the Cubature Kalman filter, a set of $2M$ sigma points is formed

$$\mathcal{X}_t^{(i)} = \mathbf{m}_t + \sqrt{\mathbf{P}_k}\xi^{(i)}, \quad i = 1, \cdots, 2M,$$

where $\xi^{(i)}$ represent the unit sigma points (see Equation (D.2)). During the filtering process, the natural parameters mean $\mathbf{m}_t$ and covariance $\mathbf{P}_k$ were estimated. Subsequently, the sigma points are propagated through the dynamics of the SSM using K

$$\widetilde{\mathcal{X}}_{t+1}^{(i)} = f\left(\mathcal{X}_t^{(i)}\right), \quad i = 1, \cdots, 2M.$$

Based on the resulting points $\widetilde{\mathcal{X}}_{t+1}^{(i)}$, the predicted mean $\mathbf{m}_{t+1}^-$, the predicted covariance $\mathbf{P}_{t+1}^-$, and the cross-covariance $\mathbf{D}_{t+1}^-$ are calculated as follows

$$\mathbf{m}_{t+1}^- = \frac{1}{2M}\sum_{i=1}^{2M}\widetilde{\mathcal{X}}_{t+1}^{(i)},$$

$$\mathbf{P}_{t+1}^- = \frac{1}{2M}\sum_{i=1}^{2M}\left(\widetilde{\mathcal{X}}_{t+1}^{(i)} - \mathbf{m}_{t+1}^-\right)\left(\widetilde{\mathcal{X}}_{t+1}^{(i)} - \mathbf{m}_{t+1}^-\right)^T + \mathbf{Q}_t,$$

$$\mathbf{D}_{t+1}^- = \frac{1}{2M}\sum_{i=1}^{2M}\left(\mathcal{X}_t^{(i)} - \mathbf{m}_t\right)\left(\widetilde{\mathcal{X}}_{t+1}^{(i)} - \mathbf{m}_{t+1}^-\right)^T,$$

where $\mathbf{Q}_t \in \mathbb{R}^{M\times M}$ corresponds to the process noise of SSM. The estimation of the mean and covariance of the resulting distribution is

$$\mathbf{m}_t^s = \mathbf{m}_t + \mathbf{G}_t\left(\mathbf{m}_{t+1}^s - \mathbf{m}_{t+1}^-\right),$$
$$\mathbf{P}_t^s = \mathbf{P}_t - \mathbf{G}_t\left(\mathbf{P}_{t+1}^s - \mathbf{P}_{t+1}^-\right)\mathbf{G}_t^T,$$

where $\mathbf{G}_t = \mathbf{D}_{t+1}^-\mathbf{P}_{t+1}^{-^1} \in \mathbb{R}^{M\times M}$ refers to the Kalman gain. Thus, performing the backward procedure of $T, \cdots, 0$ gives mean and covariance estimates of the posterior distribution.

Bayesian Inference techniques based on Gaussian approximations, like the Cubature Kalman filter or the Cubature Rauch-Tung-Striebel smoother, achieve satisfactory results for many SSMs [53]. However, in cases where the distributions of interest, e.g., the filtering or smoothing distributions, are multimodal Gaussian or non-Gaussian, respectively, these

techniques produce inappropriate approximations [53]. Also, applying these Bayesian Inference techniques is not possible if some random variables are discontinuous [53]. Therefore, in the next section, Monte Carlo sampling-based approximations are considered, found in the family of sequential Monte Carlo techniques.

D.2. Sequential Monte Carlo Techniques

For State-Space Models in which the distributions of interest correspond to a multimodal Gaussian distribution or a non-Gaussian distribution, or discrete random variables are encountered, Sequential Monte Carlo (SMC) techniques are applied. In general, these techniques directly approximate the distribution of interest using Monte Carlo (particle)-based techniques combined with Importance Sampling. In the case of SSMs, the interest is to track the evolution of a set of S particles $\left\{\left(w_t^{(i)}, \mathbf{x}_t^{(i)}\right) \,:\, i = 1, \cdots, S\right\}$ over time $t = 0, \cdots, T$ and derive the associated probabilities. Given the Markov property assumed in the SSMs, the posterior decomposes into the following form

$$p(\mathbf{X} \mid \mathbf{Y}) \propto p(\mathbf{y}_t \mid \mathbf{x}_t) p(\mathbf{x}_t \mid \mathbf{x}_{t-1}) p(\mathbf{x}_0, \cdots, \mathbf{x}_{t-1} \mid \mathbf{y}_0, \cdots, \mathbf{y}_{t-1}).$$

Subsequently, an importance distribution is introduced

$$\mathbf{x}_0^{(i)}, \cdots, \mathbf{x}_t^{(i)} \sim \pi(\mathbf{x}_0^{(i)}, \cdots, \mathbf{x}_t^{(i)} \mid \mathbf{y}_0, \cdots, \mathbf{y}_t),$$

enabling sampling for the current state. Indeed, this distribution is necessary since sampling from the true underlying distribution is either complicated or not possible at all. The corresponding importance weights are calculable by

$$w_t^{(i)} \propto \frac{p(\mathbf{y}_t \mid \mathbf{x}_t^{(i)}) p(\mathbf{x}_t^{(i)} \mid \mathbf{x}_{t-1}^{(i)}) p(\mathbf{x}_0^{(i)}, \cdots, \mathbf{x}_{t-1}^{(i)} \mid \mathbf{y}_0, \cdots, \mathbf{y}_{t-1})}{\pi(\mathbf{x}_0^{(i)}, \cdots, \mathbf{x}_t^{(i)} \mid \mathbf{y}_0, \cdots, \mathbf{y}_t)}.$$

Commonly, as well as throughout this thesis, the importance distribution is defined as Markovian. The calculation of the importance weights hence reformulates to

$$w_t^{(i)} \propto \frac{p(\mathbf{y}_t \mid \mathbf{x}_t^{(i)}) p(\mathbf{x}_t^{(i)} \mid \mathbf{x}_{t-1}^{(i)})}{\pi(\mathbf{x}_t^{(i)} \mid \mathbf{x}_{t-1}^{(i)}, \mathbf{y}_0, \cdots, \mathbf{y}_t)} \underbrace{\frac{p(\mathbf{x}_0^{(i)}, \cdots, \mathbf{x}_{t-1}^{(i)} \mid \mathbf{y}_0, \cdots, \mathbf{y}_{t-1})}{\pi(\mathbf{x}_0^{(i)}, \cdots, \mathbf{x}_{t-1}^{(i)} \mid \mathbf{y}_0, \cdots, \mathbf{y}_{t-1})}}_{w_{t-1}^{(i)}}.$$

This reformulation leads to a recursive expression under the assumption that the samples $\mathbf{x}_0^{(i)}, \cdots, \mathbf{x}_{t-1}^{(i)}$ from the previous steps are known.

The well-known *Sequential Importance Sampling (SIS)* framework is derived from the assumptions mentioned above. A set of S samples is first drawn from the initial prior distribution by

$$\mathbf{x}_0^{(s)} \sim p(\mathbf{x}_0), \quad i = 1, \cdots, S$$

$$w_0^{(i)} = 1/S, \quad i = 1, \cdots, S$$

with equal initialized importance weights. Normalization of the importance weights ensures a correct probability distribution. Then, the recursive procedure of SIS is applied by drawing S new samples from the importance distribution

$$\mathbf{x}_t^{(i)} \sim \pi(\mathbf{x}_t^{(i)} \mid \mathbf{x}_{t-1}^{(i)}, \mathbf{y}_0, \cdots, \mathbf{y}_t), \quad i = 1, \cdots, S.$$

Subsequently, the associated new importance weights are given by

$$w_t^{(i)} \propto w_{t-1}^{(i)} \frac{p(\mathbf{y}_t \mid \mathbf{x}_t^{(i)}) p(\mathbf{x}_t^{(i)} \mid \mathbf{x}_{t-1}^{(i)})}{\pi(\mathbf{x}_t^{(i)} \mid \mathbf{x}_{t-1}^{(i)}, \mathbf{y}_0, \cdots, \mathbf{y}_t)}.$$

In the case of a *bootstrap procedure*, where the importance distribution corresponds to the dynamic model $\pi(\mathbf{x}_t^{(i)} \mid \mathbf{x}_{t-1}^{(i)}, \mathbf{y}_0, \cdots, \mathbf{y}_t) = p(\mathbf{x}_t^{(i)} \mid \mathbf{x}_{t-1}^{(i)})$, the calculation simplifies to

$$w_t^{(i)} \propto w_{t-1}^{(i)} p(\mathbf{y}_t \mid \mathbf{x}_t^{(i)}).$$

The resulting importance weights are normalized again to ensure a correct probability distribution.

However, the SIS framework suffers from the *degeneracy problem* [53]. The degeneracy problem describes the situation where almost all particles have importance weights that are very low or equal to zero. One way to mitigate this problem is the use of resampling techniques [4, 53]. These techniques are performed after the normalization of the recalculated importance weights. The principle is to remove unlikely particles whose importance weights are consequently very small. Instead, the more likely particles with larger importance weights are duplicated. The resulting *Sequential Importance Resampling (SIR)*, also called *particle filter*, is a popular algorithm from the Sequential Monte Carlo (SMC) family. However, like all sampling-based techniques, the proposed frameworks suffer from the curse of dimensionality [4, 6, 53]. For high-dimensional state spaces, the application of

the proposed frameworks is intractable. Moreover, although SIR mitigates the effect of the degeneracy problem, it does not resolve it at all [53]. As a result, many extensions have been proposed to improve the sampling-based SMC family available in [52, 53]. However, in the context of this work, the SIR framework provided decent results.

E. Algorithms

This appendix lists the pseudocodes of the training procedures for the three implemented algorithms Gaussian Process Dynamic Mode Decomposition (GP-DMD), Bayesian Gaussian Process Dynamic Mode Decomposition (Bayesian GP-DMD), and Probabilistic Dynamic Mode Primitive (Pro-DMP).

Algorithm E.1: Training Procedure of Gaussian Process Dynamic Mode Decomposition

Input: Observations $\mathbf{Y} = [\mathbf{y}_0,\ \ldots,\ \mathbf{y}_T]$, initial latent variables $\mathbf{X} = [\mathbf{x}_0,\ \ldots,\ \mathbf{x}_T]$, kernel parameters $\boldsymbol{\theta}$, linear operator $\mathbf{A}$, hyperparameter values $\alpha_{\mathbf{y}},\ \beta_{\mathbf{y}},\ \ldots,\ \alpha_{\mathbf{a}},\ \beta_{\mathbf{a}}$, learning rates $\delta_{\boldsymbol{\theta}}, \ldots, \delta_{\mathbf{a}}$ and number of iterations I.

begin

$\lambda_{\mathbf{y}} \leftarrow \mathbb{E}(\mathrm{Gam}(\alpha_{\mathbf{y}}, \beta_{\mathbf{y}})) = \alpha_{\mathbf{y}}/\beta_{\mathbf{y}}$

$\lambda_{\mathbf{0}} \leftarrow \mathbb{E}(\mathrm{Gam}(\alpha_{\mathbf{0}}, \beta_{\mathbf{0}})) = \alpha_{\mathbf{0}}/\beta_{\mathbf{0}}$

$\lambda_{\mathbf{x}} \leftarrow \mathbb{E}(\mathrm{Gam}(\alpha_{\mathbf{x}}, \beta_{\mathbf{x}})) = \alpha_{\mathbf{x}}/\beta_{\mathbf{x}}$

$\lambda_{\mathbf{a}} \leftarrow \mathbb{E}(\mathrm{Gam}(\alpha_{\mathbf{a}}, \beta_{\mathbf{a}})) = \alpha_{\mathbf{a}}/\beta_{\mathbf{a}}$

repeat

$\mathbf{A}^*, \lambda_{\mathbf{0}}^*, \lambda_{\mathbf{x}}^*, \lambda_{\mathbf{a}}* \leftarrow$ Get Calculated Closed-Form Solutions (see Equations (4.5) to (4.8))

$\nabla_{\boldsymbol{\theta}}\mathcal{L}, \nabla_{\mathbf{X}}\mathcal{L}, \nabla_{\lambda_{\mathbf{y}}}\mathcal{L} \leftarrow$ Get Calculated Gradients (see Equation (4.9))

$\boldsymbol{\theta},\ \mathbf{X},\ \lambda_{\mathbf{y}} \leftarrow \boldsymbol{\theta} + \delta_{\boldsymbol{\theta}}\nabla_{\boldsymbol{\theta}}\mathcal{L},\ \mathbf{X} + \delta_{\mathbf{X}}\nabla_{\mathbf{X}}\mathcal{L},\ \lambda_{\mathbf{y}} + \delta_{\lambda_{\mathbf{y}}}\nabla_{\lambda_{\mathbf{y}}}\mathcal{L}$

$\mathbf{A} \leftarrow \mathbf{A} + \delta_{\mathbf{A}}\,(\mathbf{A}^* - \mathbf{A})$

$\lambda_{\mathbf{0}} \leftarrow \lambda_{\mathbf{0}} + \delta_{\lambda_{\mathbf{0}}}\,(\lambda_{\mathbf{0}}^* - \lambda_{\mathbf{0}})$

$\lambda_{\mathbf{x}} \leftarrow \lambda_{\mathbf{x}} + \delta_{\lambda_{\mathbf{x}}}\,(\lambda_{\mathbf{x}}^* - \lambda_{\mathbf{x}})$

$\lambda_{\mathbf{a}} \leftarrow \lambda_{\mathbf{a}} + \delta_{\lambda_{\mathbf{a}}}\,(\lambda_{\mathbf{a}}* - \lambda_{\mathbf{a}})$

until *I iterations are done*;

end

Algorithm E.2: Training Procedure of Bayesian Gaussian Process Dynamic Mode Decomposition

Input: Observations $\mathbf{Y} = [\mathbf{y}_0, \ldots, \mathbf{y}_T]$, initial latent variables $\mathbf{X} = [\mathbf{x}_0, \ldots, \mathbf{x}_T]$, kernel parameters $\boldsymbol{\theta}$, linear operator $\mathbf{A}$, hyperparameter values $\alpha_{\mathbf{y}}, \beta_{\mathbf{y}}, \ldots, \alpha_{\mathbf{a}}, \beta_{\mathbf{a}}$, learning rates $\delta_{\boldsymbol{\theta}}, \ldots, \delta_{\widetilde{\boldsymbol{\mu}}_{\mathbf{U}}}$, number of iterations I, inducing inputs $\mathbf{z}_0, \ldots, \mathbf{z}_D$ and the index for the corresponding inducing variables $idx_{\mathbf{z}}$.

begin

$\widetilde{\alpha}_{\mathbf{y}}, \widetilde{\beta}_{\mathbf{y}} \leftarrow \alpha_{\mathbf{y}}, \beta_{\mathbf{y}}$
$\widetilde{\alpha}_{\mathbf{0}}, \widetilde{\beta}_{\mathbf{0}} \leftarrow \alpha_{\mathbf{0}}, \beta_{\mathbf{0}}$
$\widetilde{\alpha}_{\mathbf{x}}, \widetilde{\beta}_{\mathbf{x}} \leftarrow \alpha_{\mathbf{x}}, \beta_{\mathbf{x}}$
$\widetilde{\alpha}_{\mathbf{a}}, \widetilde{\beta}_{\mathbf{a}} \leftarrow \alpha_{\mathbf{a}}, \beta_{\mathbf{a}}$
$\widetilde{\boldsymbol{\mu}}_{\mathbf{a}}, \widetilde{\boldsymbol{\Lambda}}_{\mathbf{a}} \leftarrow \mathbf{0}, \mathbf{I}$
$\widetilde{\boldsymbol{\mu}}_{\mathbf{U}}, \widetilde{\boldsymbol{\Lambda}}_{\mathbf{U}} \leftarrow \mathbf{Y}[:, idx_{\mathbf{z}}], \mathbf{I}$
repeat
$\mathbf{X} \leftarrow$ Sample Optimal Latent Trajectories (see Equation (4.24))
$\widetilde{\alpha}^*_{\mathbf{y}}, \widetilde{\beta}^*_{\mathbf{y}}, \ldots, \widetilde{\boldsymbol{\mu}}^*_{\mathbf{U}}, \widetilde{\boldsymbol{\Lambda}}^*_{\mathbf{U}} \leftarrow$ Get Calculated Closed-Form Solutions (see Equations (4.11) to (4.22))
$\nabla_{\boldsymbol{\theta}}\mathcal{L}, \nabla_{\mathbf{Z}}\mathcal{L} \leftarrow$ Get Calculated Gradients (see Equation (4.25))
$\boldsymbol{\theta}, \mathbf{Z} \leftarrow \boldsymbol{\theta} + \delta_{\boldsymbol{\theta}}\nabla_{\boldsymbol{\theta}}\mathcal{L}, \mathbf{Z} + \delta_{\mathbf{Z}}\nabla_{\mathbf{Z}}\mathcal{L}$
$\widetilde{\alpha}_{\mathbf{y}}, \widetilde{\beta}_{\mathbf{y}} \leftarrow \widetilde{\alpha}_{\mathbf{y}} + \delta_{\alpha_{\mathbf{y}}}(\widetilde{\alpha}^*_{\mathbf{y}} - \widetilde{\alpha}_{\mathbf{y}}),\ \beta_{\mathbf{y}} + \delta_{\beta_{\mathbf{y}}}(\widetilde{\beta}^*_{\mathbf{y}} - \widetilde{\beta}_{\mathbf{y}})$
$\widetilde{\alpha}_{\mathbf{0}}, \widetilde{\beta}_{\mathbf{0}} \leftarrow \widetilde{\alpha}_{\mathbf{0}} + \delta_{\alpha_{\mathbf{0}}}(\widetilde{\alpha}^*_{\mathbf{0}} - \widetilde{\alpha}_{\mathbf{0}}),\ \beta_{\mathbf{0}} + \delta_{\beta_{\mathbf{0}}}(\widetilde{\beta}^*_{\mathbf{0}} - \widetilde{\beta}_{\mathbf{0}})$
$\widetilde{\alpha}_{\mathbf{x}}, \widetilde{\beta}_{\mathbf{x}} \leftarrow \widetilde{\alpha}_{\mathbf{x}} + \delta_{\alpha_{\mathbf{x}}}(\widetilde{\alpha}^*_{\mathbf{x}} - \widetilde{\alpha}_{\mathbf{x}}),\ \beta_{\mathbf{x}} + \delta_{\beta_{\mathbf{0}}}(\widetilde{\beta}^*_{\mathbf{x}} - \widetilde{\beta}_{\mathbf{x}})$
$\widetilde{\alpha}_{\mathbf{a}}, \widetilde{\beta}_{\mathbf{a}} \leftarrow \widetilde{\alpha}_{\mathbf{a}} + \delta_{\alpha_{\mathbf{a}}}(\widetilde{\alpha}^*_{\mathbf{a}} - \widetilde{\alpha}_{\mathbf{a}}),\ \beta_{\mathbf{a}} + \delta_{\beta_{\mathbf{a}}}(\widetilde{\beta}^*_{\mathbf{a}} - \widetilde{\beta}_{\mathbf{a}})$
$\widetilde{\boldsymbol{\mu}}_{\mathbf{a}}, \widetilde{\boldsymbol{\Lambda}}_{\mathbf{a}} \leftarrow \widetilde{\boldsymbol{\mu}}_{\mathbf{a}} + \delta_{\widetilde{\boldsymbol{\mu}}_{\mathbf{a}}}(\widetilde{\boldsymbol{\mu}}^*_{\mathbf{a}} - \widetilde{\boldsymbol{\mu}}_{\mathbf{a}}),\ \widetilde{\boldsymbol{\Lambda}}_{\mathbf{a}} + \delta_{\widetilde{\boldsymbol{\Lambda}}_{\mathbf{a}}}(\widetilde{\boldsymbol{\Lambda}}^*_{\mathbf{a}} - \widetilde{\boldsymbol{\Lambda}}_{\mathbf{a}})$
$\widetilde{\boldsymbol{\mu}}_{\mathbf{U}}, \widetilde{\boldsymbol{\Lambda}}_{\mathbf{U}} \leftarrow \widetilde{\boldsymbol{\mu}}_{\mathbf{U}} + \delta_{\widetilde{\boldsymbol{\mu}}_{\mathbf{U}}}(\widetilde{\boldsymbol{\mu}}^*_{\mathbf{U}} - \widetilde{\boldsymbol{\mu}}_{\mathbf{U}}),\ \widetilde{\boldsymbol{\Lambda}}_{\mathbf{U}} + \delta_{\widetilde{\boldsymbol{\Lambda}}_{\mathbf{U}}}(\widetilde{\boldsymbol{\Lambda}}^*_{\mathbf{U}} - \widetilde{\boldsymbol{\Lambda}}_{\mathbf{U}})$
until *I iterations are done or the ELBO converges*;

end

Algorithm E.3: Training Procedure of Probabilistic Dynamic Mode Primitive

Input: S Observations Batches $\mathbf{Y}^s = [\mathbf{y}_0^s,\ \ldots,\ \mathbf{y}_T^s]$, initial S batches of latent variables $\mathbf{X}^s = [\mathbf{x}_0,\ \ldots,\ \mathbf{x}_T]^s$, kernel parameters $\boldsymbol{\theta}$, hyperparameter values $\alpha_{\mathbf{y}},\ \beta_{\mathbf{y}},\ \ldots,\ \alpha_{\mathbf{a}},\ \beta_{\mathbf{a}}$, learning rates $\delta_{\boldsymbol{\theta}}, \ldots, \delta_{\mathbf{a}}$ and number of iterations I.

begin

$\lambda_{\mathbf{y}} \leftarrow \mathbb{E}(\mathrm{Gam}(\alpha_{\mathbf{y}}, \beta_{\mathbf{y}})) = \alpha_{\mathbf{y}}/\beta_{\mathbf{y}}$

$\lambda_{\mathbf{0}} \leftarrow \mathbb{E}(\mathrm{Gam}(\alpha_{\mathbf{0}}, \beta_{\mathbf{0}})) = \alpha_{\mathbf{0}}/\beta_{\mathbf{0}}$

$\lambda_{\mathbf{x}} \leftarrow \mathbb{E}(\mathrm{Gam}(\alpha_{\mathbf{x}}, \beta_{\mathbf{x}})) = \alpha_{\mathbf{x}}/\beta_{\mathbf{x}}$

$\nu_a,\ \mathbf{W}_a \leftarrow \alpha_{\mathbf{a}}, \beta_{\mathbf{a}}\mathbf{I}$

$\boldsymbol{\mu}_{\mathbf{a}_m},\ \boldsymbol{\Lambda}_{\mathbf{a}_m} \leftarrow \mathbf{0}, \mathbf{0}$

for *all* s *in* S **do**

$\widetilde{\boldsymbol{\mu}}_{\mathbf{a}_m^s},\ \widetilde{\boldsymbol{\Lambda}}_{\mathbf{a}_m^s} \leftarrow \mathbf{0}, \mathbf{0}$

end

repeat

$\lambda_{\mathbf{0}}^*, \lambda_{\mathbf{x}}^*, \ldots, \boldsymbol{\mu}_{\mathbf{a}_m}^*, \boldsymbol{\Lambda}_{\mathbf{a}_m}^* \leftarrow$ Get Calculated Closed-Form Solutions (see Equations (5.3) to (5.8))

$\nabla_{\boldsymbol{\theta}}\mathcal{L}, \nabla_{\mathbf{X}}\mathcal{L}, \nabla_{\lambda_{\mathbf{y}}}\mathcal{L} \leftarrow$ Get Calculated Gradients (see Equation (5.9))

$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \delta_{\boldsymbol{\theta}}\nabla_{\boldsymbol{\theta}}\mathcal{L}$

$\mathbf{X} \leftarrow \mathbf{X} + \delta_{\mathbf{X}}\nabla_{\mathbf{X}}\mathcal{L}$

$\lambda_{\mathbf{y}} \leftarrow \lambda_{\mathbf{y}} + \delta_{\lambda_{\mathbf{y}}}\nabla_{\lambda_{\mathbf{y}}}\mathcal{L}$

$\boldsymbol{\mu}_{\mathbf{a}_m},\ \boldsymbol{\Lambda}_{\mathbf{a}_m} \leftarrow \boldsymbol{\mu}_{\mathbf{a}_m} + \delta_{\boldsymbol{\mu}_{\mathbf{a}_m}}\left(\boldsymbol{\mu}_{\mathbf{a}_m}^* - \boldsymbol{\mu}_{\mathbf{a}_m}\right),\ \boldsymbol{\Lambda}_{\mathbf{a}_m} + \delta_{\boldsymbol{\Lambda}_{\mathbf{a}_m}}\left(\boldsymbol{\Lambda}_{\mathbf{a}_m}^* - \boldsymbol{\Lambda}_{\mathbf{a}_m}\right)$

for *all* s *in* S **do**

$\widetilde{\boldsymbol{\mu}}_{\mathbf{a}_m^s},\ \widetilde{\boldsymbol{\Lambda}}_{\mathbf{a}_m^s} \leftarrow \widetilde{\boldsymbol{\mu}}_{\mathbf{a}_m^s} + \delta_{\widetilde{\boldsymbol{\mu}}_{\mathbf{a}_m^s}}\left(\widetilde{\boldsymbol{\mu}}_{\mathbf{a}_m^s}^* - \widetilde{\boldsymbol{\mu}}_{\mathbf{a}_m^s}\right),\ \widetilde{\boldsymbol{\Lambda}}_{\mathbf{a}_m^s} + \delta_{\widetilde{\boldsymbol{\Lambda}}_{\mathbf{a}_m^s}}\left(\widetilde{\boldsymbol{\Lambda}}_{\mathbf{a}_m^s}^* - \widetilde{\boldsymbol{\Lambda}}_{\mathbf{a}_m^s}\right)$

end

$\lambda_{\mathbf{0}} \leftarrow \lambda_{\mathbf{0}} + \delta_{\lambda_{\mathbf{0}}}\left(\lambda_{\mathbf{0}}^* - \lambda_{\mathbf{0}}\right)$

$\lambda_{\mathbf{x}} \leftarrow \lambda_{\mathbf{x}} + \delta_{\lambda_{\mathbf{x}}}\left(\lambda_{\mathbf{x}}^* - \lambda_{\mathbf{x}}\right)$

$\lambda_{\mathbf{a}} \leftarrow \lambda_{\mathbf{a}} + \delta_{\lambda_{\mathbf{a}}}\left(\lambda_{\mathbf{a}}* - \lambda_{\mathbf{a}}\right)$

until *I iterations are done*;

end

F. Parameter Settings

In this appendix we list the parameter settings fot the three implemented algorithms Gaussian Process Dynamic Mode Decomposition (GP-DMD), Bayesian Gaussian Process Dynamic Mode Decomposition (Bayesian GP-DMD), and Probabilistic Dynamic Mode Primitive (Pro-DMP) tested on Circle-Shape Dataset (CSD), Eight-Shape Dataset (ESD) and Minimum-Jerk Dataset (MJD).

Parameter Settings of GP-DMD on Datasets			
Name of Dataset	Circle-Shape	Eight-Shape	Minimum-Jerk
Seed	11	11	11
Iterations	50000	50000	50000
Learning Rate	$1\mathrm{e}^{-2}$	$1\mathrm{e}^{-3}$	$1\mathrm{e}^{-3}$
Latent Dim.	3	5	5
Number Of Labels	4	4	4
RBF Kernel: $\theta,\ \gamma$	$1/32,\ 1$	$1/32,\ 1$	$1/32,\ 1$
Prior: $\alpha_{\mathbf{y}},\ \beta_{\mathbf{y}}$	$1,\ 1\mathrm{e}^{-3}$	$1,\ 1\mathrm{e}^{-3}$	$1,\ 1\mathrm{e}^{-3}$
Prior: $\alpha_{\mathbf{0}},\ \beta_{\mathbf{0}}$	$1,\ 1\mathrm{e}^{-3}$	$1,\ 1\mathrm{e}^{-3}$	$1,\ 1\mathrm{e}^{-3}$
Prior: $\alpha_{\mathbf{x}},\ \beta_{\mathbf{x}}$	$1,\ 1\mathrm{e}^{-3}$	$1,\ 1\mathrm{e}^{-3}$	$1,\ 1\mathrm{e}^{-3}$
Prior: $\alpha_{\mathbf{a}},\ \beta_{\mathbf{a}}$	$1,\ 1\mathrm{e}^{-3}$	$1,\ 1\mathrm{e}^{-3}$	$1,\ 1\mathrm{e}^{-3}$

Table F.1.: This list summarizes the parameter setting used to train Gaussian Process Dynamic Mode Decomposition (GP-DMD) on the Circle-Shape Dataset (CSD), the Eight-Shape Dataset (ESD), and Minimum-Jerk Dataset (MJD).

Parameter Settings of Bayesian GP-DMD on Datasets			
Name of Dataset	Circle-Shape	Eight-Shape	Minimum-Jerk
Seed	11	11	11
Iterations	5000	5000	5000
Learning Rate	$1\mathrm{e}^{-3}$	$1\mathrm{e}^{-3}$	$1\mathrm{e}^{-3}$
Latent Dim.	3	5	5
Number Of Inducing Variables	25	25	25
RBF Kernel: $\theta,\ \gamma$	$1/32,\ 1$	$1/32,\ 1$	$1/32,\ 1$
Prior: $\alpha_{\mathbf{y}},\ \beta_{\mathbf{y}}$	$1\mathrm{e}^{3}, 1$	$1,\ 1\mathrm{e}^{-5}$	$1,\ 1\mathrm{e}^{-6}$
Prior: $\alpha_{\mathbf{0}},\ \beta_{\mathbf{0}}$	$1\mathrm{e}^{5},\ 1$	$1\mathrm{e}^{5},\ 1$	$1\mathrm{e}^{5},\ 1$
Prior: $\alpha_{\mathbf{x}},\ \beta_{\mathbf{x}}$	$1\mathrm{e}^{5},\ 1$	$1\mathrm{e}^{5},\ 1$	$1\mathrm{e}^{5},\ 1$
Prior: $\alpha_{\mathbf{a}},\ \beta_{\mathbf{a}}$	$1\mathrm{e}^{-3}, 1$	$1\mathrm{e}^{-3}, 1$	$1\mathrm{e}^{-3}, 1$

Table F.2.: This list summarizes the parameter setting used to train Bayesian Gaussian Process Dynamic Mode Decomposition (Bayesian GP-DMD) on the Circle-Shape Dataset (CSD), the Eight-Shape Dataset (ESD), and Minimum-Jerk Dataset (MJD).

Parameter Settings of Pro-DMP on Datasets			
Name of Dataset	Circle-Shape	Eight-Shape	Minimum-Jerk
Seed	11	11	11
Iterations	50000	50000	50000
Batch Size S	5	5	5
Learning Rate	$1\mathrm{e}^{-3}$	$1\mathrm{e}^{-3}$	$1\mathrm{e}^{-3}$
Latent Dim.	3	5	5
Number Of Labels	4	4	4
RBF Kernel: $\theta,\ \gamma$	$1/16,\ 1$	$1/16,\ 1$	$1/16,\ 1$
Prior: $\alpha_{\mathbf{y}},\ \beta_{\mathbf{y}}$	$1,\ 1\mathrm{e}^{-3}$	$1,\ 1\mathrm{e}^{-3}$	$1,\ 1\mathrm{e}^{-3}$
Prior: $\alpha_{\mathbf{0}},\ \beta_{\mathbf{0}}$	$1\mathrm{e}^{3},\ 1$	$1\mathrm{e}^{3},\ 1$	$1\mathrm{e}^{3},\ 1$
Prior: $\alpha_{\mathbf{x}},\ \beta_{\mathbf{x}}$	$1\mathrm{e}^{3},\ 1$	$1\mathrm{e}^{3},\ 1$	$1\mathrm{e}^{3},\ 1$
Prior: $\alpha_{\mathbf{a}},\ \beta_{\mathbf{a}}$	$6\mathrm{e}^{-6},\ 1\mathrm{e}^{6}$	$6\mathrm{e}^{-6},\ 1\mathrm{e}^{6}$	$6\mathrm{e}^{-6},\ 1\mathrm{e}^{6}$

Table F.3.: This list summarizes the parameter setting used to train Probabilistic Dynamic Mode Primitive (Pro-DMP) on the Circle-Shape Dataset (CSD), the Eight-Shape Dataset (ESD), and Minimum-Jerk Dataset (MJD).

G. Learning Curves

In this appendix, the learning curves of GP-DMD, Bayesian GP-DMD, and Pro-DMP on the data sets CSD, ESD and MJD are given.

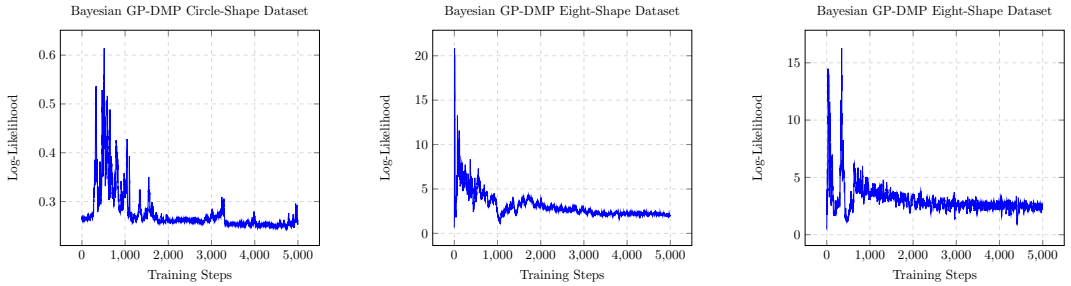


Figure G.1.: This figure shows the learning curves from the training procedures of the Bayesian GP-DMDs on the Circle-Shape Dataset (CSD), the Eight-Shape Dataset (ESD), and the Minimum-Jerk Dataset (MJD). The learning phase covers 5000 iterations. The learning curve starts at decent values due to a PCA initialization. As a result of numerical issues, these curves show strong fluctuations. However, all three plots exhibit convergence of the learning curves to the end.

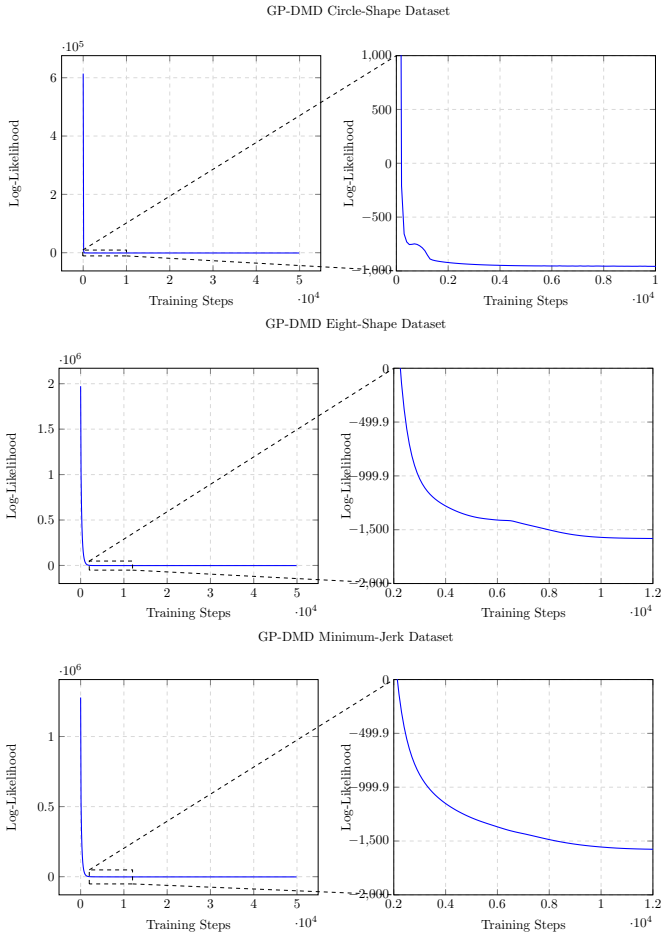


Figure G.2.: This figure shows the learning curves from the training procedures of the GP-DMDs on the Circle-Shape Dataset (CSD), the Eight-Shape Dataset (ESD), and the Minimum-Jerk Dataset (MJD). The learning phase covers 50000 iterations. On the left side, the entire learning curves are given, while on the right side, central parts are shown enlarged. All plots exhibit convergence of the learning curves to the end.

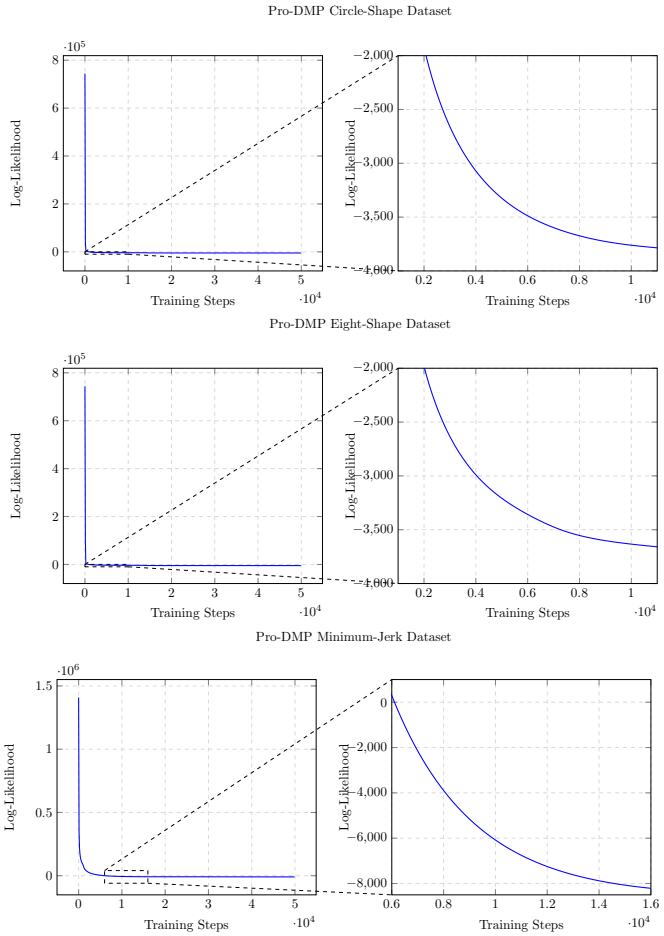


Figure G.3.: This figure shows the learning curves from the training procedures of the Pro-DMPs on the Circle-Shape Dataset (CSD), the Eight-Shape Dataset (ESD), and the Minimum-Jerk Dataset (MJD). The learning phase covers 50000 iterations. On the left side, the entire learning curves are given, while on the right side, central parts are shown enlarged. All plots exhibit convergence of the learning curves to the end.